ENGO 620 Estimation for Navigation

Fall 2013

Lab1 - Analysis

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

DEPARTMENT OF GEOMATICS ENGINEERING
UNIVERSITY OF CALGARY, ALBERTA, CANADA

Contents

List of Figures

List of Tables

1 Task 1 – Cartesian Co-ordinate Solution

The pseudo-range measurement (ρ_k) obtained from *k*th satellite at a given time epoch (GPS time), after correcting the raw measurement for known errors (satellite clock offset, relativity effect, and ionospheric delay), is given as follows.

$$\rho_k = r_k + c.\delta t_u + \varepsilon_k \quad (1.1)$$

r_k : Actual distance between user (receiver) and satellite *k*

δt_u : User clock offset from GPS time.

c : Speed of light

ε_k : Total residual errors

The sign of the bias term $c.\delta t_u$ is set arbitrarily. It can either be added to the true range, or subtracted, depending on whether the receiver clock is in negative bias or positive bias with respect to GPS time. In this analysis, clock bias due to clock offset is considered positive and hence added to true range term r_k. Let the vectorss **rx** = (x_r,y_r,z_r) and $\mathbf{tx_k}$ = (x_{sk},y_{sk},z_{sk}) , for k=1,2,3,…K, represent the position of the user at the given measurement epoch and the position of *k*th satellite at the time of signal transmission in Cartesian co-ordinate system.

Table 1 Parametric model for estimating user position and clock bias

Model parameter	Equivalent expression
z_k	ρ_k,
$h_k(x)$	$\sqrt{(x_{sk}-x_r)^2+(y_{sk}-y_r)^2+(z_{sk}-z_r)^2}$ + b
v_k	ε_k

Since the model expressed in Table 1 is non-linear, it is linearized and expressed as follows, neglecting the higher order terms [ENGO 620 Lecture notes, Fall 2013].

$$\delta z_k = H_k \delta x + v_k \quad (1.4)$$

Where, δz_k is called the measurement misclosure for *k*th satellite. It is expressed as follows.

$$\delta z_k = z_k - h_k(\hat{x}) \quad (1.5)$$

In the present context, it is simply the different between actual pseudo-range observation z_k and estimated pseudo-range observation $\hat{z}_k = h_k(\hat{x})$.

H_k is called the design matrix which relates error in the state vector $\delta x = (x - \hat{x})$ to observation misclosure δz_k

$$H_k = \frac{dh_k(x)}{dx}\Big|_{x=\hat{x}} \quad (1.6)$$

For K pseudorange observations from K satellites, misclosure vector is a Kx1 column matrix containing δz_k in *k*th row representing misclosure for *k*th satellite. Design matrix H is a Kx4 matrix containing H_k elements in *k*th row. The observation misclosure is used to observe the error is state vector δx. Least-square solution expression used in this analysis is given below [ENGO 620 Lecture notes, Fall 2013]

$$\delta\hat{x} = (H^T R^{-1} H)^{-1} H^T R^{-1} \delta z \quad (1.7)$$

R is the co-variance matrix of measurement errors, with their standard deviations denoted σ_k. Its general form of *R* is given as,

$$R = \begin{bmatrix} \sigma_1^2 & 0 & \cdots & 0 \\ 0 & \sigma_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_K^2 \end{bmatrix}_{KxK}$$

Elements of *R* matrix need to be set carefully, based on the expected level of uncertainty in each measurement. Usually, in case of a GNSS receiver, C/No could be a direct indicator to set values for σ_k. However, in this lab, elevation angle (true elevation angle which is given as input) is used to set σ_k as suggested in the handout. The estimated error $\delta\hat{x}_i$, say in *i*th iteration, is added to initial estimate, $\hat{x}_i$ to generate $\hat{x}_{i+1} = \hat{x}_i + \delta\hat{x}_i$. $\hat{x}_{i+1}$ is then used as a point of expansion to calculate δz_{i+1}, H_{i+1}, and $\delta\hat{x}_{i+1}$ in *(i+1)*th iteration. The process is explained in detail in subsequent sections.

For a system of K satellites, design matrix *H*, misclosure vector δz are given in Table 2.

Table 2 Misclosure vector and design matrix – Cartesian Co-ordinate approach

δz	$\begin{bmatrix} z_1 - (\sqrt{(x_{s1} - \hat{x}_r)^2 + (y_{s1} - \hat{y}_r)^2 + (z_{s1} - \hat{z}_r)^2} + \hat{b}) \\ z_2 - (\sqrt{(x_{s2} - \hat{x}_r)^2 + (y_{s2} - \hat{y}_r)^2 + (z_{s2} - \hat{z}_r)^2} + \hat{b}) \\ \vdots \\ z_K - (\sqrt{(x_{sK} - \hat{x}_r)^2 + (y_{sK} - \hat{y}_r)^2 + (z_{sK} - \hat{z}_r)^2} + \hat{b}) \end{bmatrix}_{Kx1}$
H	$\begin{bmatrix} \frac{-(x_{s1} - \hat{x}_r)}{\hat{r}_1} & \frac{-(y_{s1} - \hat{y}_r)}{\hat{r}_1} & \frac{-(z_{s1} - \hat{z}_r)}{\hat{r}_1} & 1 \\ \frac{-(x_{s2} - \hat{x}_r)}{\hat{r}_1} & \frac{-(y_{s2} - \hat{y}_r)}{\hat{r}_1} & \frac{-(z_{s2} - \hat{z}_r)}{\hat{r}_1} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ \frac{-(x_{sK} - \hat{x}_r)}{\hat{r}_1} & \frac{-(y_{sK} - \hat{y}_r)}{\hat{r}_1} & \frac{-(z_{sK} - \hat{z}_r)}{\hat{r}_1} & 1 \end{bmatrix}_{KxK}$

2 Task 2 – Curvilinear Co-ordinate Solution

In this case, the least square solution is directly represented in curvilinear co-ordinate system. Unlike in the previous case, where the state vector ***δx*** was represented in Cartesian co-ordinate system, state vector will be solved in local frame to result $\delta x_{ENU=}(\delta x_e, \delta x_n, \delta x_u)$, i.e the position errors in east, north, and up directions. These are then converted to errors in curvilinear system, $\delta x_{PLH} = (\delta x_P, \delta x_L, \delta x_H)$, i.e the errors in phi (lattitude), lambda (longitude), and height (altitude). Also, the design matrix, H, is derived as a function of azimuth and elevation angles, which are computed by the east, north, and up components (x_e, x_n, x_u) of the position vector between *k*th satellite and estimated user position. The H matrix is derived as follows.

Consider the relative position vector P_k between *k*th satellite and the user, i.e,

$$\vec{P}_k = t\vec{x}_k - r\vec{x} \tag{2.1}$$

Components of this relative position vector along east, north, and up direction in local co-ordinate system can be derived with the help of Figure 1. Observing that the position vector is from satellite to user, components the *magnitude* of the position vector along ENU axes can be written as,

$$\begin{bmatrix} x_e \\ x_n \\ x_u \end{bmatrix} = \begin{bmatrix} -\cos(el_k)\sin(az_k) \\ -\cos(el_k)\cos(az_k) \\ -\sin(el_k) \end{bmatrix} [|\vec{P}_k|] = B.\ [|\vec{P}_k|] \tag{2.2}$$

Minus sign is due to the fact the position vector is from satellite to receiver.

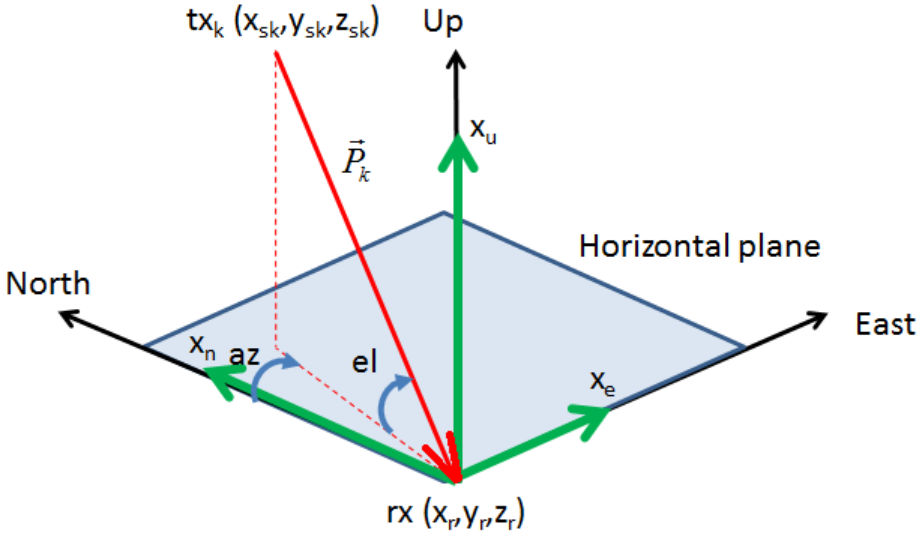

Figure 1 Resolving position vector in ENU frame

For a system of K satellites, design matrix *H*, misclosure vector *δz* are given in Table 3.

Table 3 Misclosure vector and design matrix – Curvilinear approach

δz	$\begin{bmatrix} z_1-(\sqrt{\hat{x}_{e1}{}^2+\hat{x}_{n1}{}^2+\hat{x}_{u1}{}^2}+\hat{b}) \\ z_2-(\sqrt{\hat{x}_{e2}{}^2+\hat{x}_{n2}{}^2+\hat{x}_{u2}{}^2}+\hat{b}) \\ \vdots \\ z_K-(\sqrt{\hat{x}_{eK}{}^2+\hat{x}_{nK}{}^2+\hat{x}_{uK}{}^2}+\hat{b}) \end{bmatrix}_{Kx1}$
H	$\begin{bmatrix} -\cos(el_1)\sin(az_1) & -\cos(el_1)\cos(az_1) & -\sin(el_1) & 1 \\ -\cos(el_2)\sin(az_2) & -\cos(el_2)\cos(az_2) & -\sin(el_2) & 1 \\ & \vdots & & \\ -\cos(el_K)\sin(az_K) & -\cos(el_K)\cos(az_K) & -\sin(el_K) & 1 \end{bmatrix}_{KxK}$

2.1 Steps to obtain least square solution in curvilinear co-ordinate

Following points discuss the steps required to obtain LS solution from the pseudo-range measurements in a *given epoch*. In a given epoch, batch processing carried out considering all the measurement at once. Process is initiated with an initial guess of states. Number of iteration is set arbitrary which depends on the confidence level of initial guess. Same number of iterations is set for each epoch, with same initial guess of the state.

1. Start the process with an initial guess/rough estimate of the states.
   $(\hat{\phi}_r \quad \hat{\lambda}_r \quad \hat{h}_r \quad \hat{b})=(\varphi_0,\lambda_0,h_0,b_0)$.
   Assumption is that the initial guess is within the linear region of convergence. Best way to start with is to assume receiver is at center of the earth i.e, (0,0,-R), where, R is the radius of earth. Initial bias value set as zero.
2. Iteration begins. For every iteration *i*, convert the estimated position in curvilinear co-ordinate to Cartesian co-ordinate.
   $$\begin{bmatrix} \hat{\phi}_{ri} & \hat{\lambda}_{ri} & \hat{h}_{ri} \end{bmatrix} \xrightarrow{\substack{Convert \\ toCartesian}} \begin{bmatrix} \hat{x}_{ri} & \hat{y}_{ri} & \hat{z}_{ri} \end{bmatrix}$$
3. Obtain satellite position (x_{sk} y_{sk} z_{sk}) from *k*th satellite to compute relative position vector P_k.
   $$\vec{P}_{ki}=\begin{bmatrix}(x_{sk}-\hat{x}_{ri}) & (y_{sk}-\hat{y}_{ri}) & (z_{sk}-\hat{z}_{ri})\end{bmatrix}^T$$
4. Since the position vector is in ECEF frame, transform this vector from ECEF frame to ENU frame to obtain a vector represented in ENU frame,
   $$\hat{x}_{ki}^{ENU}=\begin{bmatrix}\hat{x}_{ei} & \hat{x}_{ni} & \hat{x}_{ui}\end{bmatrix}^T=R_{ECEFi}^{ENU}.P_{ki}$$
   $$R_{ECEFi}^{ENU}=\begin{bmatrix} -\sin\hat{\lambda}_{ri} & \cos\hat{\lambda}_{ri} & 0 \\ -\sin\hat{\varphi}_{ri}\cos\hat{\lambda}_{ri} & -\sin\hat{\varphi}_{ri}\sin\hat{\lambda}_{ri} & \cos\hat{\varphi}_{ri} \\ \cos\hat{\varphi}_{ri}\cos\hat{\lambda}_{ri} & \cos\hat{\varphi}_{ri}\sin\hat{\lambda}_{ri} & \sin\hat{\varphi}_{ri} \end{bmatrix}$$
5. Obtain the azimuth and elevation angle as given in Misra (2006)

$$\hat{az}_i = \tan^{-1}\left(\frac{\hat{x}_{ei}}{\hat{x}_{ni}}\right)$$

$$\hat{el}_i = \sin^{-1}\left(\frac{\hat{x}_{ui}}{\sqrt{\hat{x}_{ei}+\hat{x}_{ni}+\hat{x}_{ui}}}\right)$$

6. Compute elements of design matrix H_i and misclosure vector δz_i using the computed azimuth and elevation angles, as per the Table 3.
7. Execute least square equation (1.7) to obtain estimated delta change in receiver position along east, north, and up directions.

$$\delta\hat{x}_{ki}^{ENU} = \begin{bmatrix} \delta\hat{x}_{ei} & \delta\hat{x}_{ni} & \delta\hat{x}_{ui} & \delta\hat{b}_i \end{bmatrix}^T$$

8. Compute the delta change receiver position in latitude, longitude, and altitude using the expressions,

$$\delta\hat{\phi}_{ri} = \left(\frac{\delta\hat{x}_{ni}}{\hat{M}_i + \hat{h}_{ri}}\right)$$

$$\delta\hat{\lambda}_{ri} = \left(\frac{\delta\hat{x}_{ei}}{(\hat{N}_i + \hat{h}_{ri})\cos\hat{\varphi}_{ri}}\right)$$

$$\delta\hat{h}_{ri} = \delta\hat{x}_{ui}$$

Where, $\hat{M}_i$ and $\hat{N}_i$ are meridian and prime vertical radii of curvature for the ellipsoid at a given latitude, computed using current estimate of the latitude, $\hat{\varphi}_{ri}$. MATLAB™ functions *calcm.m* and *calcn.m* are used

9. Once corrections for estimated values of latitude, longitude, and height are obtained, they are used to generate the update/point of expansion of next iteration.

$$\hat{\phi}_{r(i+1)} = \hat{\phi}_{ri} + \delta\hat{\phi}_{ri}$$

$$\hat{\lambda}_{r(i+1)} = \hat{\lambda}_{ri} + \delta\hat{\lambda}_{ri}$$

$$\hat{h}_{r(i+1)} = \hat{h}_{ri} + \delta\hat{h}_{ri}$$

10. Repeat from step 2 until convergence is achieved.

2.2 Declaring convergence

There is no specific equation to set the threshold for convergence in least square solution. In a given epoch, iterations are carried out until the elements in the correction vector $\delta\hat{x}_i$ are sufficiently small when compared to the expected accuracy. As a rule of thumb, they are usually set to $1/10^{th}$ of expected accuracy. For e.g, for an application which demands 10m accuracy in 3D, the iteration is repeated until all four elements in $\delta\hat{x}_i$ are below 1 m. However, it is observed that if the initial guess of the state is very well within the region of linearity, elements in correction vector will converge to milli-meter level in just 4 to 5 iterations. Hence convergence depends on how best the problem is approximated by linear model, and also how best the initial guess is selected.

2.3 Statistical tests

Statistical tests are necessary to assess the reliability of the solution, performed using the residual obtained after the final solution, in a given measurement epoch. These tests are based on an assumption that the residuals are normally distributed.

Test 1: Test on Variance factor

We assume an *a priori* value for variance factor, σ_0^2, which relates observation co-factor matrix Q_R with observation co-variance matrix R with the following relation.

$$R = \sigma_0^2 Q_R \tag{2.7}$$

However, the assumed value for σ_0^2 may not be reliable or could not be guessed at all. Then, *a posteriori* variance factor $\hat{\sigma}_0^2$ is computed as,

$$\hat{\sigma}_0^2 = \frac{r^T Q_R^{-1} r}{n-m} \tag{2.8}$$

Where, r is the residual obtained after the final solution in a given measurement epoch, (n-m) is the degrees of freedom. With the assumption that σ_0^2 is known, test statistic for the null hypothesis test H_0 is given as, [ENGO 620 Lecture notes, Fall 2013]

$$\xi = r^T R_R^{-1} r \tag{2.9}$$

As assumed earlier, since measurement errors are normally distributed, and thus the residual r, ξ is a chi-square distribution with (n-m) degrees of freedom, χ^2(n-m). Two tailed test is performed on ξ, with probability of type-1 error (incorrectly rejecting null hypothesis) α, to check whether ξ lies between two thresholds given as below [ENGO 620 Lecture notes, Fall 2013],

$$\chi^2_{1-\alpha/2} < \xi < \chi^2_{\alpha/2} \tag{2.10}$$

Since the residuals are being tested together, above test is also called as *the global test of residuals*. Methodology to derive the thresholds is discussed below.

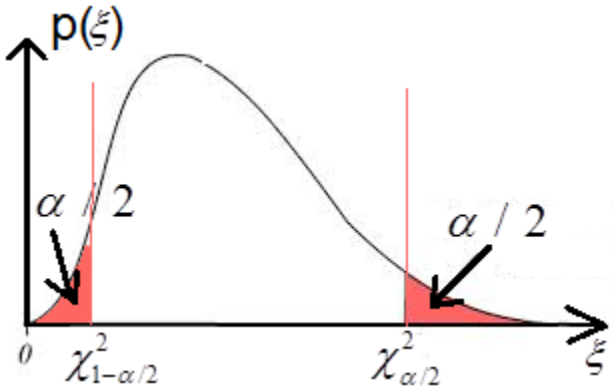

Figure 2 Chi-Square distribution of ξ – global test

From Figure 2, the probability of type-1 error in lower tail of the distribution is given as,

$$\frac{\alpha}{2} = p(\xi < \chi^2_{1-\alpha/2}) = CDF(\chi^2_{1-\alpha/2})$$

Hence,

$$\chi^2_{1-\alpha/2} = CDF^{-1}(\alpha/2)$$

Similarly, probability of type-1 error in upper tail of the distribution is given as,

$$\frac{\alpha}{2} = p(\xi > \chi^2_{\alpha/2}) = 1 - p(\xi < \chi^2_{\alpha/2}) = 1 - CDF(\chi^2_{\alpha/2})$$

Hence, $$\chi^2_{\alpha/2} = CDF^{-1}(1-\alpha/2)\,.$$

These are obtained by taking inverse of the cumulative distribution function of chi-square distribution, for a given degree of freedom (n-m). MATLAB™ function *chi2inv* will be used. For e.g, if confidence level of the test is 95% (α = 5%), values of the thresholds are given in following table for different degrees of freedom.

Table 4 Thresholds for two-tailed global test

Degrees of freedom (n-m)	$\chi^2_{1-\alpha/2}$	$\chi^2_{\alpha/2}$
(9-4) = 5	0.831	12.832
(10-4) = 6	1.237	14.449
(11-4) = 7	1.689	16.012

Test 2: Blunder detection

In case of failure in global test, it is necessary to detect the blunders in the measurement in order to reject it and recomputed the least square solution. We continue the test assuming $\sigma_0^2 = \hat{\sigma}_0^2$. Each residual ($r_i$) is tested individually (also called *the local test*). Each residual is assumed to be normally distributed with zero mean and variance given by the corresponding diagonal element $(Cr)_{ii}$ in the co-variance matrix of the residuals. Test statistic ξ is obtained by normalizing each residual by its standard deviation. *No blunder* is declared if ξ is below a threshold as given below.

$$|\xi| = \left| \frac{r_i}{\sqrt{(C_r)_{ii}}} \right| < N_{1-\alpha/2}$$

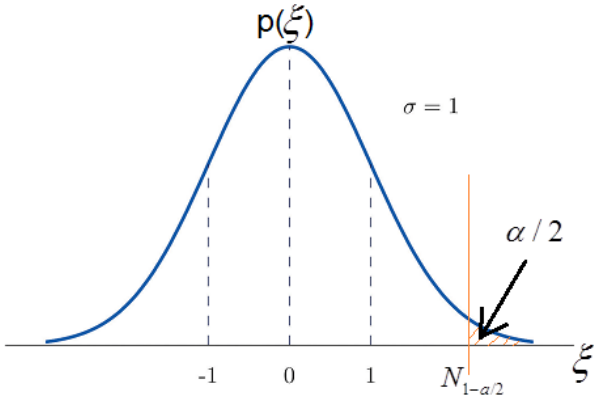

Figure 3 standard normal distribution of ξ - local test

From Figure 3, value of the threshold for local test can be derived.

$$1 - \frac{\alpha}{2} = p(\xi < N_{1-\alpha/2}) = CDF(N_{1-\alpha/2})$$

$$N_{1-\alpha/2} = CDF^{-1}(1-\alpha/2)$$

CDF inverse is computed by *norminv* function in MATLAB™. For α = 5%, $N_{1-\alpha/2}$ = 1.959.

If any of the observations fail the local test, it is rejected and the LS solution is recomputed using remaining observations. If multiple observations fail, the one with largest absolute standardized residual is rejected, and computation is repeated. This process could be repeated until we have obtained *no-blunder* stage. However it is possible that the degree of freedom would reach down to *one* by rejecting all the blunder observations, in which case, the LS solution obtained with 5 satellites (for 4 states) itself would be the best possible solution in a given epoch.

2.4 Statistical reliability

Once the statistical tests confirm the presence of blunder and the solution is recomputed, we need to assess the reliability of the solution in terms of the level of the blunder that can be detected on an observation (internal reliability), as well as the effect of this blunder on our estimated states (external reliability). With the assumption that only one blunder observation is present in a given epoch, and that the measurement errors (and residuals) are Gaussian, the minimum detectable blunder on the *i*th observation with detection probability (1-β)%, with (1-α)% confidence, is given by [ENGO 620 Lecture notes, Fall 2013]

$$\nabla_{MDB_i} = \frac{(R)_{ii}}{\sqrt{(C_r)_{ii}}}\delta_0$$

The non-centrality parameter δ_0 is given by,

$$\delta_0 \approx N_{1-\alpha/2} + N_{1-\beta}$$

$N_{1-\alpha/2}$ and $N_{1-\beta}$ are obtained from the CDF of standardized residual of *i*th observation.

$N_{1-\alpha/2} = CDF^{-1}(1-\alpha/2)$ and $N_{1-\beta/2} = CDF^{-1}(1-\beta/2)$. CDF inverse is computed by *norminv* function in MATLAB™

For α=5% and β=10%, value of δ_0 is given by,

$$\delta_0 = norminv(1-0.05/2,0,1) + norminv(1-0.1,0,1) = 3.24$$

The error is state due to the minimum detectable blunder in *i*th observation, also known as the protection level to estimated state with 95% confidence with power of the test 90% is computed as,

$$\Delta\hat{x}_i = (H^T R^{-1} H)^{-1} H R^{-1} \frac{(R)_{ii}}{\sqrt{(C_r)_{ii}}} 3.24$$

2.5 Other outputs from Least square algorithm

Co-variance matrix of states P

P matrix accounts for precision of estimated states.

$$P=\sigma^2_0 Q_P$$

Where, σ^2_0 is the a priori variance factor, which interprets the measurement precision. This is the best possible value of the term factored out from the observation co-variance matrix *R*, to rewrite *R* as,

$$R=\sigma^2_0 Q_R.$$

While setting the values of *R* (as discussed before), diagonal element with largest measurement uncertainty magnitude will be factored out to set it as σ^2_0. This process leaves behind an observation co-factor matrix Q_R, which accounts for relative weighting between the observation errors. The matrix Q_P, which is the state co-factor matrix, is computed as,

$$Q_P = (H^T Q_R^{-1} H)^{-1}$$

Dilution of Precision (DOP) values

From a user point of view, it is convenient to express DOP values in local co-ordinate frame, i.e, as north DOP (NDOP), east DOP (EDOP), vertical DOP (VDOP), horizontal DOP, position DOP (PDOP, in 3D), and geometric DOP (GDOP). These values are all derived from the diagonal elements $Q_{P,}$ state co-factor matrix. If the states are not solved in ENU frame, Q_P matrix needs to be transformed from solution frame to ENU frame. For e.g, in task 1, since the states are solved in ECEF frame, Q_{P_ECEF} will be transformed to Q_{P_ENU} as (for $\mathbf{x}=[x,y,z,b]^T$),

$$Q_{P\_ENU(4X4)} = T\, Q_{P\_ECEF(4X4)}\, T^T$$

$$T = \begin{bmatrix} -\sin(\phi)\sin(\lambda) & -\sin(\phi)\cos(\lambda) & \cos(\phi) & 0 \\ \cos(\lambda) & -\sin(\lambda) & 0 & 0 \\ \cos(\phi)\sin(\lambda) & \cos(\phi)\cos(\lambda) & \sin(\phi) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Various DOP values are then computed from the diagonal elements of $Q_{P\_ENU(4X4)}$ as follows,

Table 5 Various DOP values in ENU frame

EDOP	$\sqrt{Q^2_{P\_ENU(1,1)}}$
NDOP	$\sqrt{Q^2_{P\_ENU(2,2)}}$
VDOP	$\sqrt{Q^2_{P\_ENU(3,3)}}$
HDOP	$\sqrt{EDOP^2 + NDOP^2}$
PDOP	$\sqrt{HDOP^2 + VDOP^2}$
GDOP, which includes effect of bias as well.	$\sqrt{PDOP^2 + Q^2_{P\_ENU(4,4)}}$

Residuals

Residual vector, which is the difference between the actual measurement, z, and the predicted measurement $h(\hat{x})$, is computed at the end of convergence in a given epoch as

$$r = z - h(\hat{x})$$

Corresponding co-variance matrix of residual, which is an important tool in statistical tests, is computed as,

$$Cr = R - H(H^T R^{-1} H)^{-1} H^T$$

3 Task 3 to 8 - Analysis

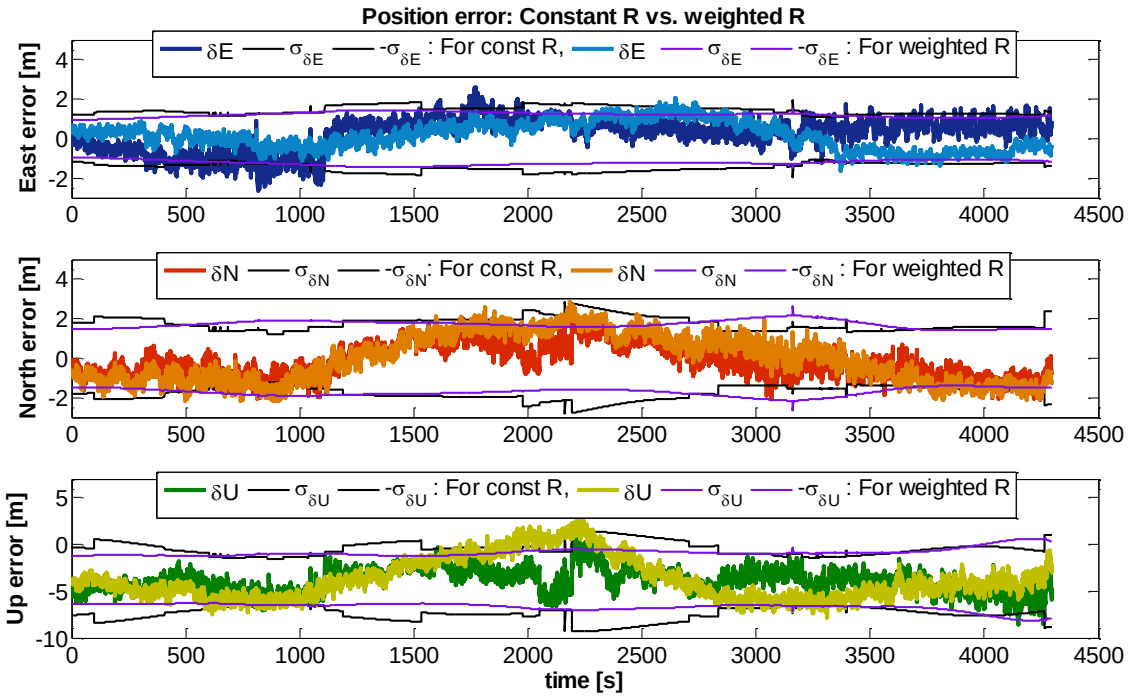

Figure 4 Position error for R matrix weighing scheme

Least square solution with states estimated in curvi-linear co-ordinate system is implemented. Two iterations are carried out for each epoch, once by setting R matrix elements with constant weight. In all relevant plots, it is referred as *constant R (or Pure Geometry DOP)*. In the 2nd iteration, elements of R matrix are weighted based on the elevation angle of the satellite from receiver. True elevation angle is used for the analysis. Figure 4 shows the plot of position error in East (E), North (N), and Vertical (U) axes. It also shows the estimated standard deviation of states for each case. Ideally we should expect zero mean error for each state. However, due to the presence of residual systematic error it will not be the case.

Table 6 lists the observations obtained from above figure. We can see that weighted the elements of R matrix relatively gives us better result than setting all diagonal elements constant. Each observation is trusted differently than giving equal importance, as their measurement uncertainties are all different. In the context of GNSS, satellites which we think as less reliable are given more uncertainty, probably due to the presence of blunders. Generally, satellites with low elevation angle are prone to contain blunders due to systematic errors. Less importance will be given for measurements from such satellites within least square algorithm. Hence we get optimum result by weighting R matrix.

Table 6 Position error: Const R vs. Weighted R

Direction	Constant R		Weighted R	
	Mean [m]	Std [m]	Mean [m]	Std [m]
E	0.21	0.85	0.14	0.69
N	-0.17	0.85	-0.04	0.92
U	-3.92	1.32	-3.75	1.41

More clear idea is obtained by checking the histogram plot of the state errors. Figure 5 clearly shows the importance of weighted R matrix. With weighted R scheme, we see that the mean of each state error would tend more close to zero than un-weighted scheme. Overall bias in error states, if any, due to the presence of blunders is improved by differentially weighting the measurements. It is also observed that changing the a-priori variance factor (σ^2_0) and repeating these tests did not change the distribution of error states, or the time series of the error states. This is an expected result, as the states do not depend on σ^2_0. Only the estimated standard deviation changed as per the given σ^2_0. In all the following analysis, weighted R scheme is used.

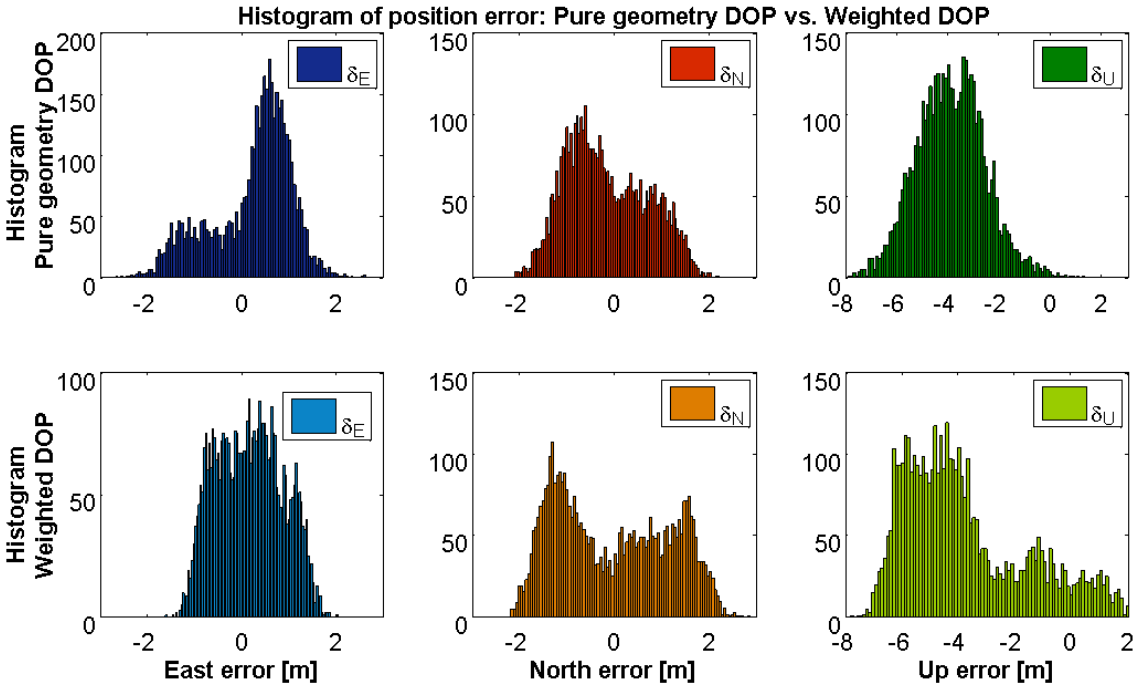

Figure 5 Distribution of ENU errors: Pure geometry DOP vs. weighted DOP

3.1 Tuning σ^2_0

σ^2_0 is tuned with *Global Test* approach. We know that the global test would fail if σ^2_0 is wrong, or, if the observation model is wrong. In order to tune σ^2_0, we assume that there is no systematic error present (i.e, model for atmospheric error, orbital error, multipath, etc., are ignored), and observation model in Table 2 or Table 3 is valid. Also, it is given that the confidence level 95% (α = 5%). With this, we have to select such a σ^2_0 value, that would fail the global test 5% times (with the assumption no systematic error). Although this statistical tuning needs infinite number of samples, only about 4000 given epochs are considered. The thresholds used for the global test are as discussed earlier. σ^2_0 is tuned to a value equal to *1*, with number of fails about 120 out 4000 epochs, resulting in 97% confidence level. Again, this tuning has assumed that systematic errors are completely zero throughout all epochs. Hence, σ^2_0 is still not an optimum value as the global test would have failed at some epochs due to blunders (systematic errors). A detailed analysis of effect of σ^2_0 on global and local tests is presented in following discussions.

Note: σ^2_0 is also tuned for a scheme employing constant R matrix (un-weighted) in a similar way. Its value is tuned as 6, with about 200 fails out of 4000 epochs, resulting in 95% confidence level.

Figure 6 shows the 2D error ellipse in local co-ordinate system for first and last epoch. The solution is from blunder-less measurements. Relevant data of the two ellipses is also shown in the same plot. Ellipse corresponding to the first epoch has error mostly along north. Also, the correlation between east and north error is less as the orientation of semi-major axis towards north axis is very small (1.9 degree). The ellipse corresponding to the last epoch has larger area, inferring that the estimated horizontal position is more uncertain than the position computed in first epoch. Also, magnitude and orientation of semi-major axis suggests that the correlation of east and north errors has increased. This effect is due to the changing geometry of satellites. Correlation-coefficients between the estimated states are discussed next.

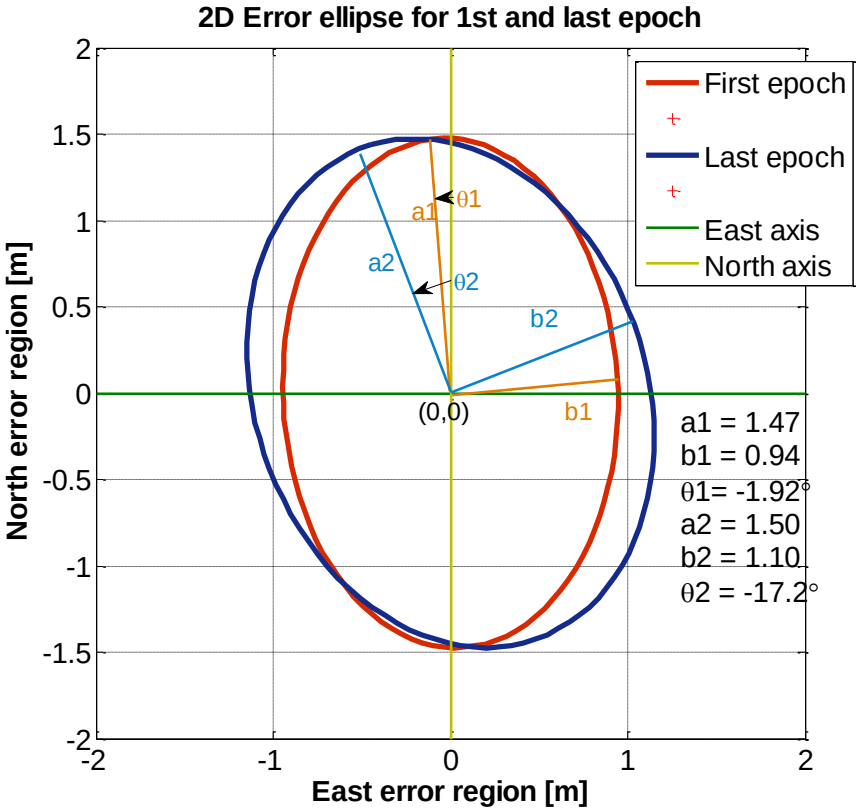

Figure 6 2D error ellipse in local co-ordinate system

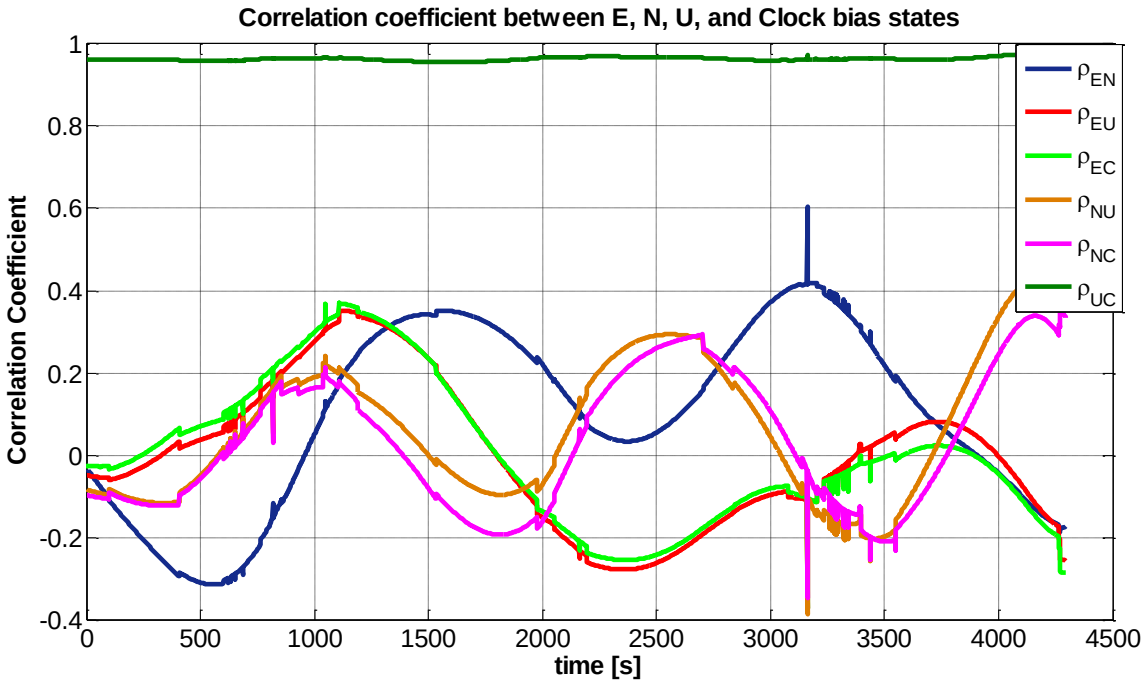

Figure 7 Correlation coefficients between E,N,U, and Clock bias (C) states

Figure 7 shows the time series of correlation coefficient between estimated states. These are computed before rejecting any blunder. The Up position state and Clock bias state always have a strong correlation. This is because the clock bias error is directly proportional to pseudo-range error; any error in clock bias will be directly added as an error in misclosure. Also, measurements (GPS satellites) are not always available in zenith, and no measurements are available from bottom region of the receiver to form a good geometry to estimate position in vertical direction. Hence, we see that the errors in vertical position are generally worse than in other directions. It can also be seen in Figure 4, where we see large negative bias in vertical error. Continuing the discussion on Figure 7, high correlation between clock bias and vertical error leads to equal correlation between East-Vertical error and East-Clock bias error, and likewise for North-Vertical error and North-Clock bias error.

We see variations of state errors with respect to time as in Figure 4. The least square algorithm, initial estimate of states, and $\sigma^2_{0,}$ are same in all epochs. The observations, observation accuracy, and geometry keep changing over time. As the geometry matrix is computed from true position of satellite, we may suspect on measurements for variations in state errors along time. First, the initial assumption of no systematic error in measurements is not valid. Second, observations do contain erroneous measurements termed as blunders. A single or multiple errors in a measurement vector leads to variations/ large errors in estimated states.

Figure 8 shows time series of DOP and number of satellites considered for position computation, before rejecting (BR) any observation and after rejecting (AR) an observation. All DOP values are jittery when the system experiences a new satellite/s joining in or when it loses some. This effect can be predominantly seen when a blunder observation is rejected from the set altogether.

Comparing the below figure with Figure 4, we observe increase in error along all three directions when the observable satellites are reduced, for example, between epochs 1000 and 3500, number of satellites is about 7, east error is about 1 m from mean, north error is about 2 m from mean, and height error is about 3 m from mean. Also, along the time, we can observe that geometry of the observations gets worse as indicated in DOP plot.

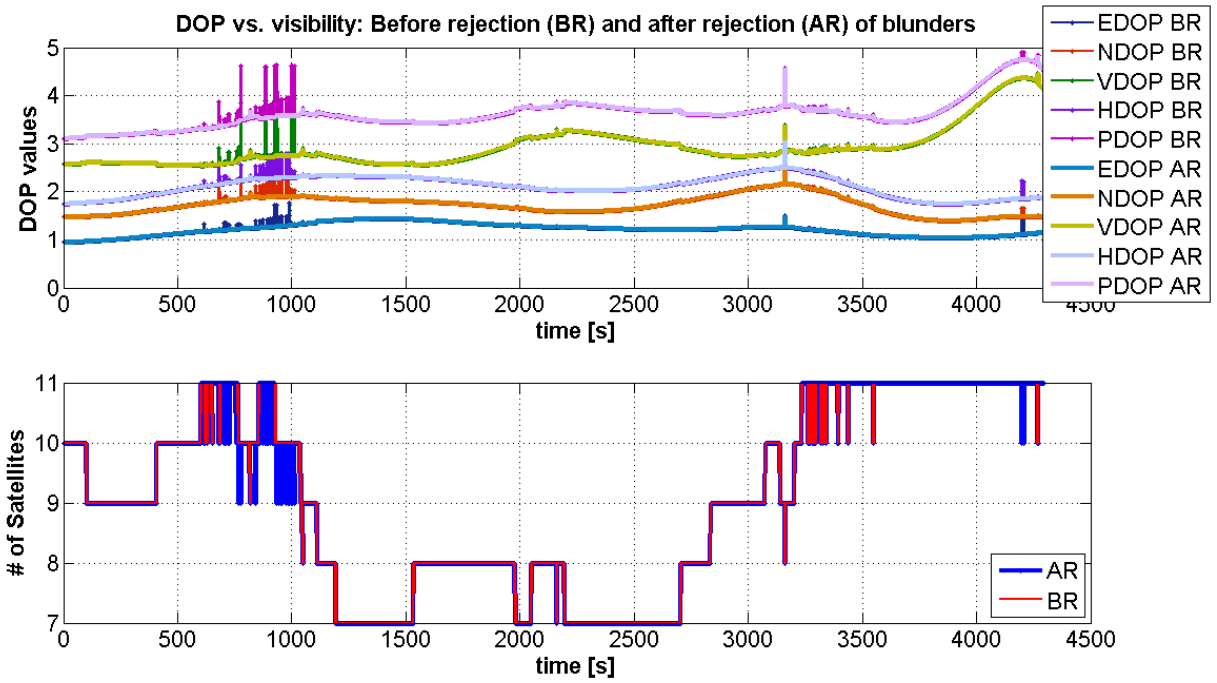

Figure 8 DOP vs. number of satellites considered

The DOP values are derived from elements in $Q_{P,}$ which is a function of H, which itself is a function of measurement geometry. Hence, DOP quantifies the effect of geometry of the measurements (location of satellites under consideration) on accuracy of estimated states. As given in Table 5, each axis of the co-ordinate frame under consideration has a DOP associated with it. Intuitively, the dilution along these axes is inversely proportional to the volume enclosed by the receiver and the satellites. This means, as the number of satellites (number of observations) decrease, we expect DOP to grow relatively.

3.2 Blunder detection and rejection process

Blunder detection and rejection procedure is discussed here. Thresholds used for the test are as discussed before.

- First, a global test on observations is carried out (after σ^2_0 is tuned). If global test is failed, local test is conducted with a-posteriori variance factor $\hat{\sigma}_0^2$
- If local test fails, corresponding measurement is flagged as a blunder. Array of all measurements who fail in local test in a given epoch is formed.

- An observation with maximum test statistics ξ is noted and corresponding measurement is declared to be rejected.
    - **Justification**: A non-blunder observation may also fail the local test. Residues of all measurements are affected by the blunder/blunders in other observations. Ideally, from the local test, we cannot pin-point to residue from one observation and declare it as a blunder, or for that matter, not as a blunder. However, in case of GNSS, it is highly likely that the failure is due to blunder in observation under consideration. Consider a scenario of positioning with four satellites. Say, 3 of the satellites are blunder-less and one has a blunder. Then 3 satellites will range a user to position it at (x,y) with some uncertainty. The fourth satellite with blunder will pull/ push the position towards/ away along the vector between 4^{th} satellite and the receiver. Hence it is highly likely that resulting residual for that satellite will be large enough to declare it as blunder.
    - Nevetheless, this may not be true always. In some cases, even if the Global test fails, local test for all satellites may pass. This is explained as follows. Say, error is present 2 observations. These errors get distributed across all (say 10) residuals of LSQ fit. Ideally, the expectation would be to get 2 residuals, with only 2 observations failing in local-test. This may not happen all the time because the residuals of these 2 observations are distributed into other residuals as well. Because of this distribution (increase/decrease-depending on the number of observations with blunders), these 2 residuals may just fall below the local-test threshold (local test passes). However, in the Global test where all the residuals are combined then the cumulated error is accounted and it might cross the Global test threshold leading to failure of Global test. This type of scenario leads to failed Global test and passed local test. In such situations, we are helpless and cannot reject any observation since local test says everything is okay, and we cannot say which observation is making the Global test to fail. We could do subset testing, with (n-1) observations considered at a time out of (n) observations. When a subset makes the global test pass, we can say the left out observation should be rejected. However, all combination of the set needs to be checked.
    - In the present analysis an observation with maximum ξ is rejected as justified above.
    - If the global test is passed, regardless of the local test failures, no observations are rejected. As long as global requirements are met, need to evaluate individual blunders is not necessary. Also, rejecting a blunder unnecessarily may result in poor performance due to loss of degree of freedom.

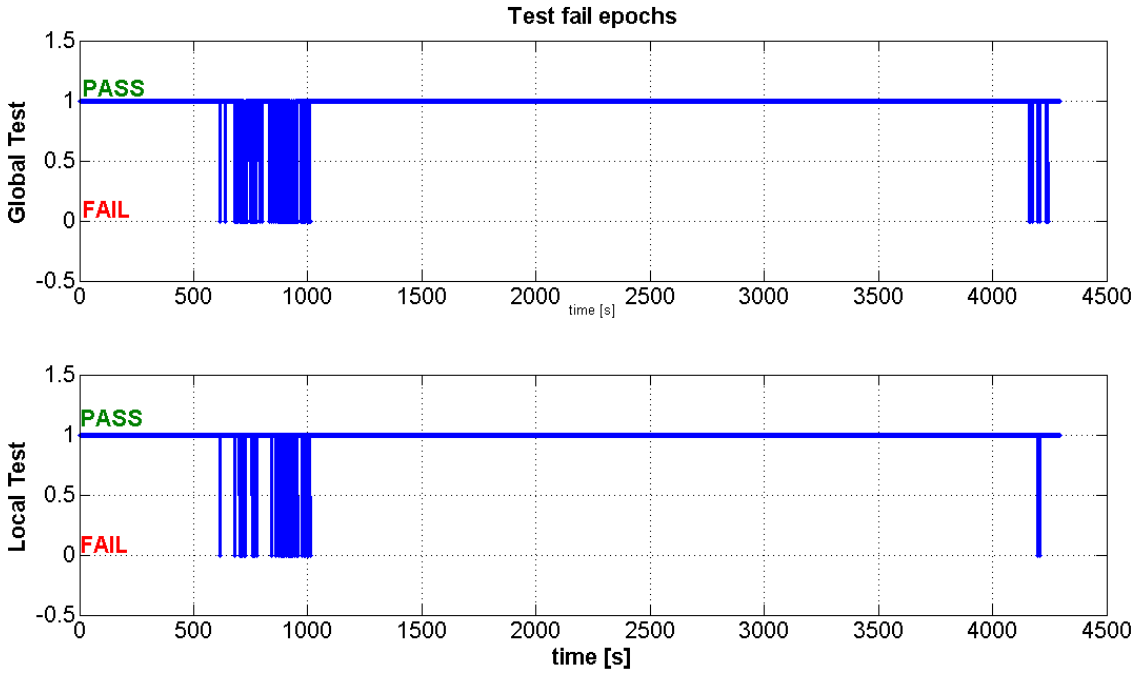

Figure 9 Test fail epochs

Figure 9 shows the epochs at which the global and local tests have failed.

Residue is computed at the end of iterations in an epoch. Residues of all satellites are computed. It is observed that 23 different PRNs are available in the observation period at different epochs. Most of the PRNs were observed over short period of time. Hence, residue plot is shown only for four satellites which were consistent. Figure 10 and Figure 11 show residue plot for PRNs 08, 11, 17, and 28. These are shown after rejecting blunder, if any. Clearly, residues have zero mean. Sharp jumps in residues can be observed when a satellite appears or vanishes. For example, at about 1200th epoch residue varies sharply in all four. For comparison, residue computed from LS with constant R scheme is overlapped on residue computed from LS with weighted R scheme. Variance of latter is lesser than the former.

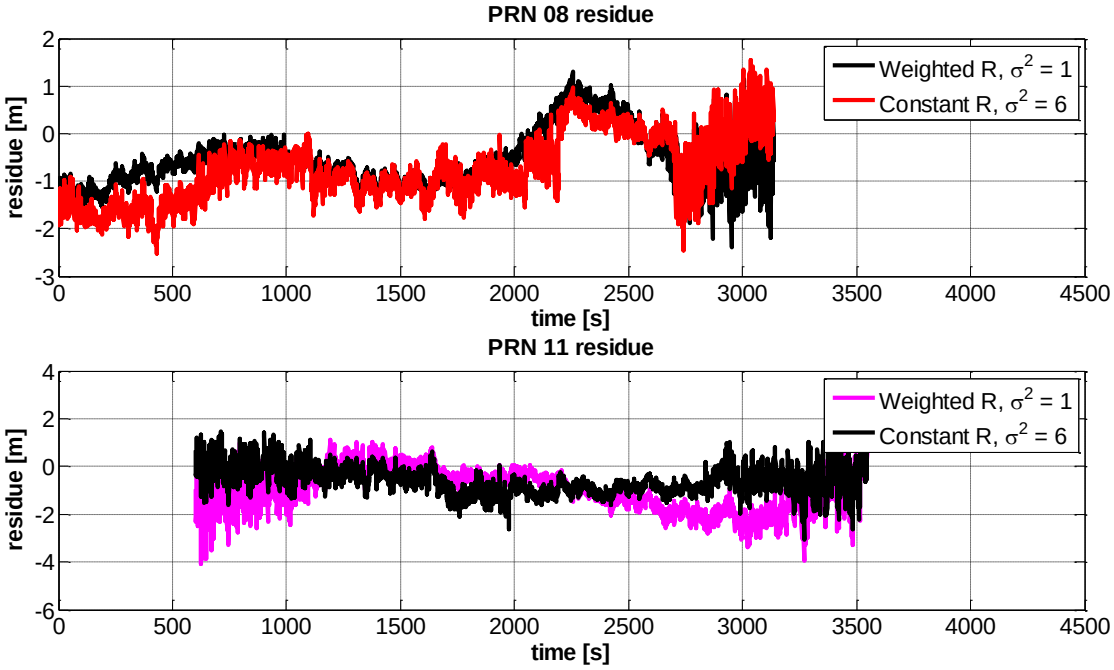

Figure 10 Residue: PRN 08 and PRN11

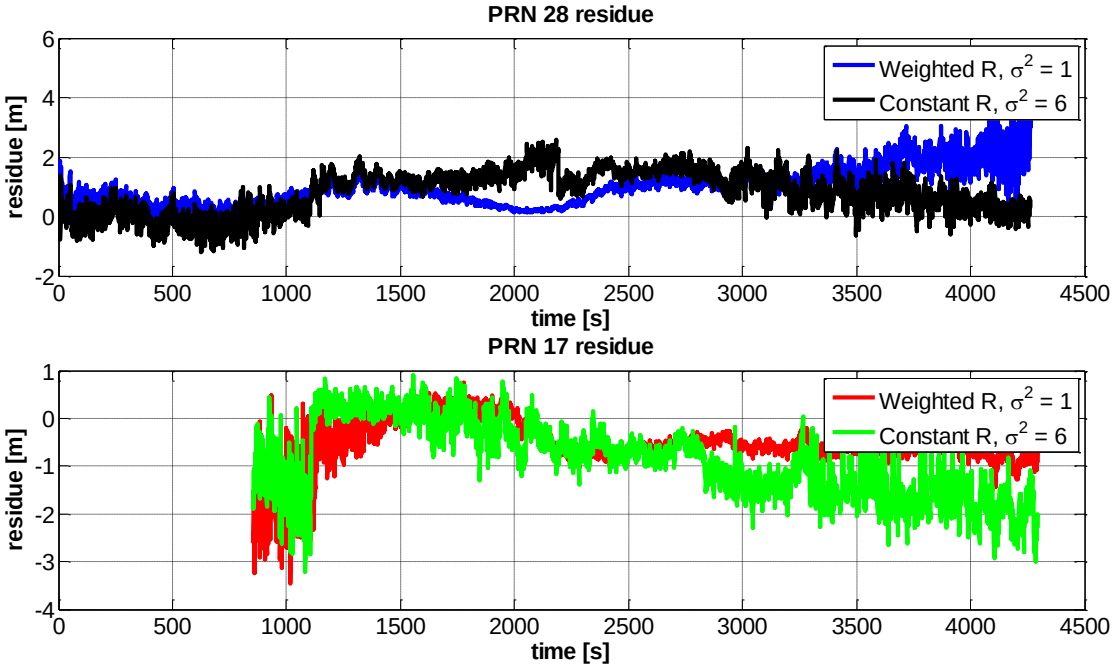

Figure 11 Residue: PRN 28 and PRN 17

To check the distribution of residues, histogram of sample PRN 28 is shown in Figure 12. Clearly, it is not a Gaussian distribution. Since the residuals do not follow a strict normal distribution, so do the measurements errors. Estimated solution does not depend on distribution of measurement errors. Hence, there is no impact on estimated state. However, all test statistics and reliability analysis are affected as the tests assume measurement errors are normal distribution.

Changing σ_0^2 did not affect residuals. However, it did affect the number of global and local test failures. This is due to the fact that global and local test statistics are dependent on measurement accuracy R.

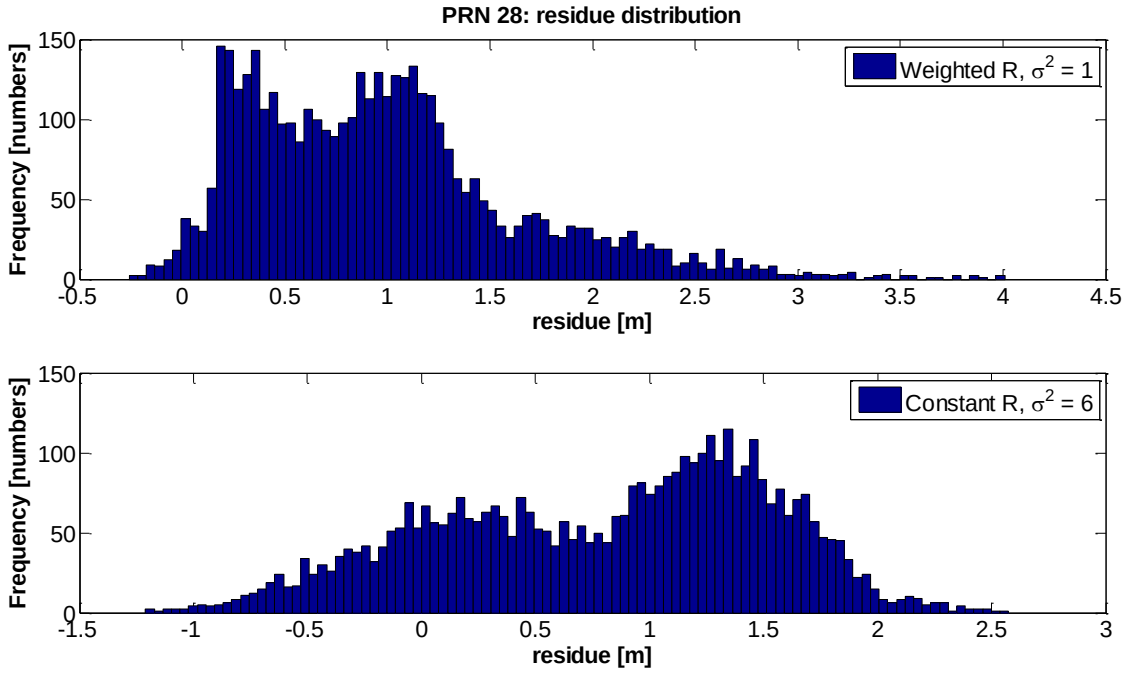

Figure 12 Histogram of residue: PRN 28

3.3 Effect of rejecting a blunder

Residuals are the level of agreement of the estimated states; we expect residuals to get reduced when a blunder in an epoch is rejected. Since it is tedious to show this effect in plots, following table is prepared. A blunder is found in 615th second. PRN 19 is found to have blunder. Table 7 shows the residues of all satellites before and after rejecting PRN 19. It can be seen that residues of eight PRNs have reduced in magnitude when PRN19 is rejected and solution is re-computed. Similar observation in found in other epochs with blunder observation.

Table 7 Reside before and after rejection: epoch 615th second

PRN ID	Residue before rejecting PRN19	Residue after rejection
13	-1.06	-0.82
06	-1.70	-0.70
15	-0.70	-0.37
03	-1.02	-0.08
28	0.06	-0.13
07	-0.02	0.36
19	1.10	NA
08	-0.26	-0.20
11	-1.93	-1.21
05	1.37	1.20
10	1.79	1.67

In Figure 13, Figure 14, and Figure 15 internal reliability, the minimum detectable error is shown for all 23 available satellites in the log file. Comparing these figures with Figure 8, MDB is inversely proportional to the available number of satellites. Lesser the number of satellites it is difficult to separate out an observation if it is a blunder. More measurements will bring out the level of the least blunder that could be detected in every observation.

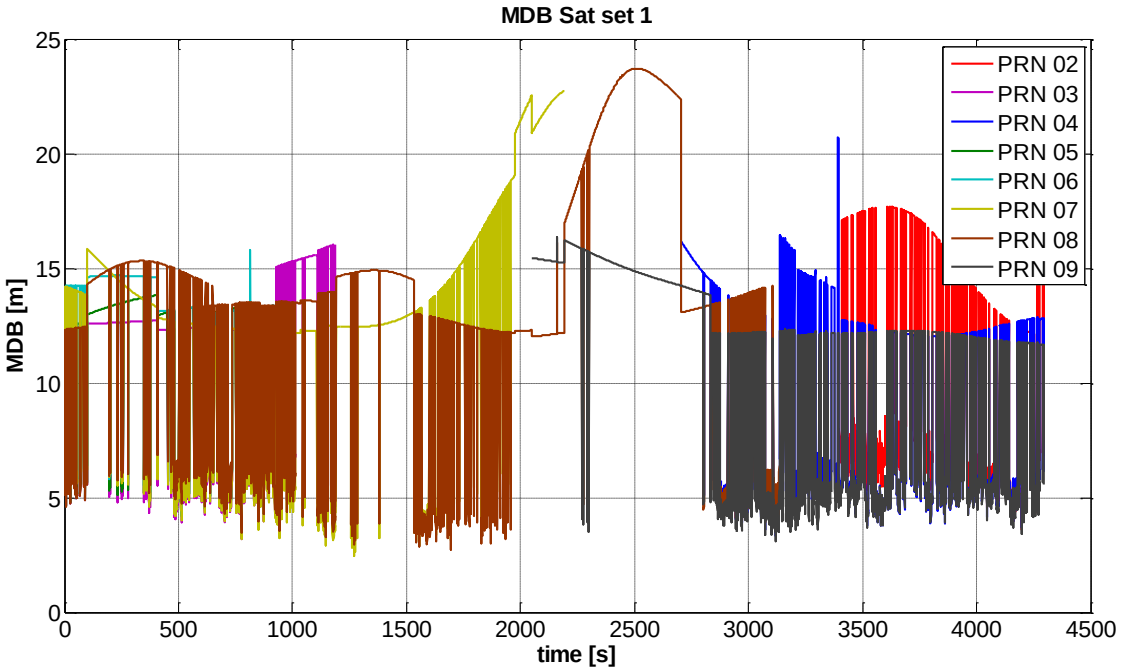

Figure 13 MDB Sat set 1

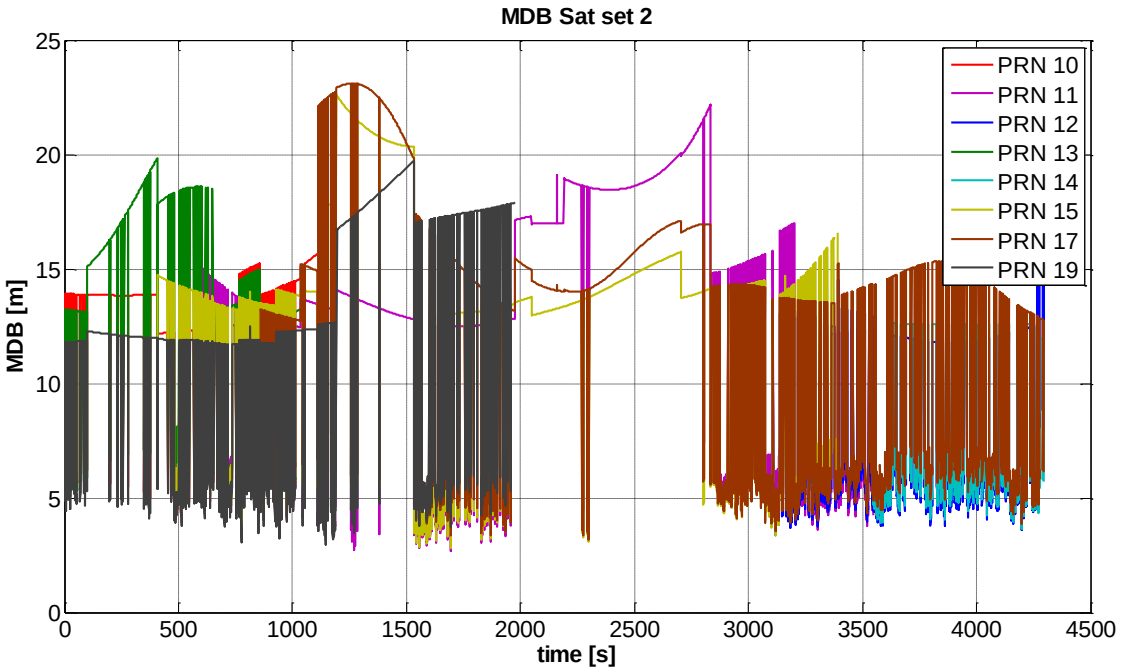

Figure 14 MDB Sat set 2

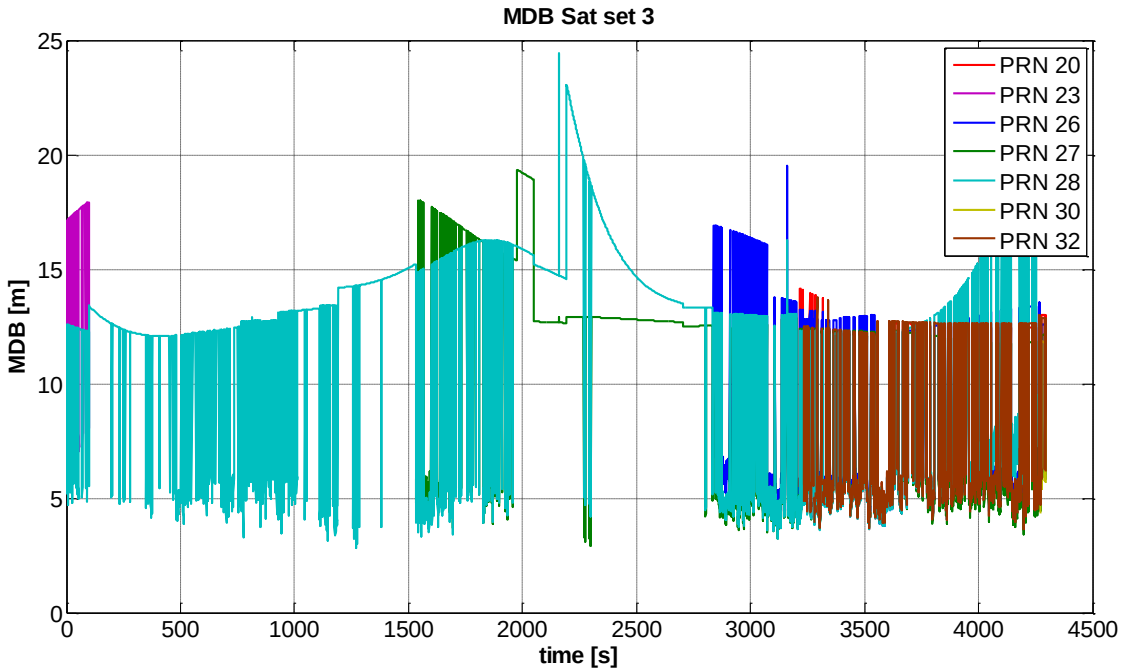

Figure 15 MDB Sat set 3

Figure 16, Figure 17, and Figure 18 show the horizontal protection level external reliability parameter shown in 3 sets for all 23 PRNs. Figure 19, Figure 20, and Figure 21 show the vertical protection level. Comparing these figures with Figure 8, reveals the inverse relation of external reliability (protection level) with satellite geometry. Since more uncertainty is associated with poor geometry measurement, it is difficult to filter out a blunder. Minimum detectable error would increase; likewise protection level associated with this MDB grows. Protection corresponding to increased horizontal or vertical position level is guaranteed only if MDB is detected even with poor geometry.

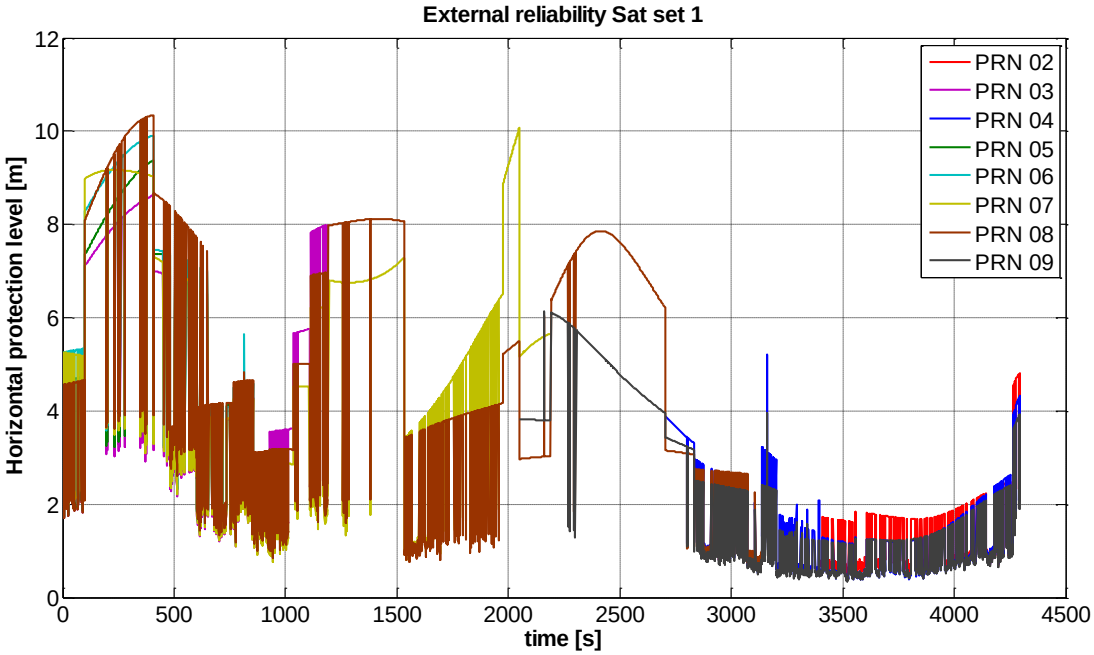

Figure 16 Horizontal protection level Sat set 1

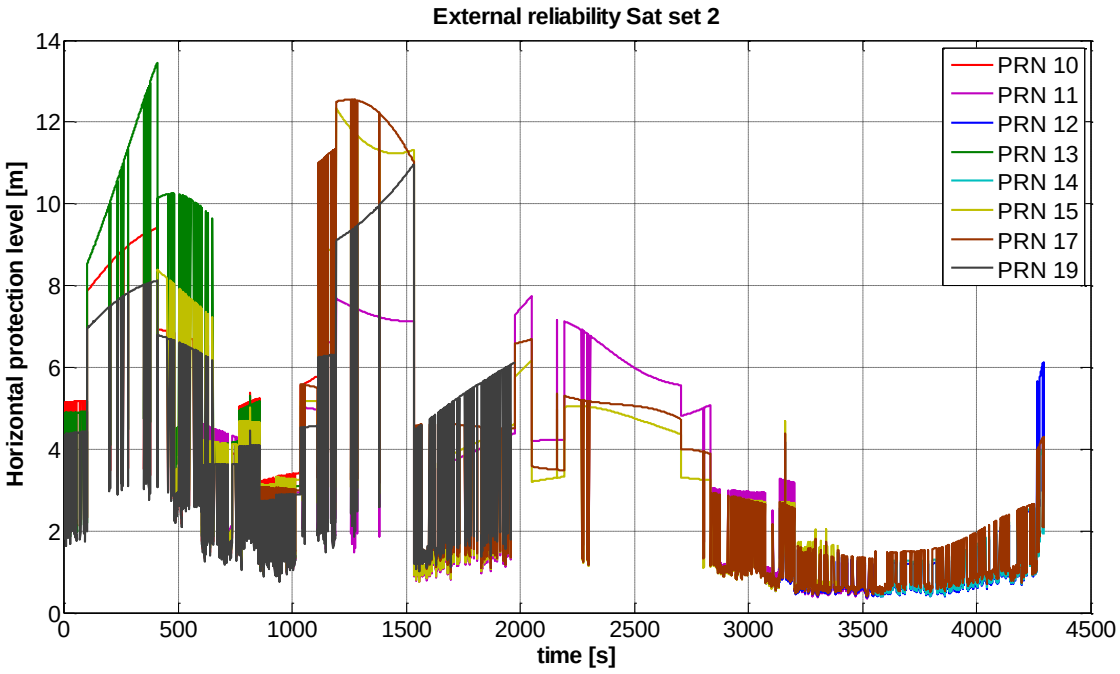

Figure 17 Horizontal protection level Sat set 2

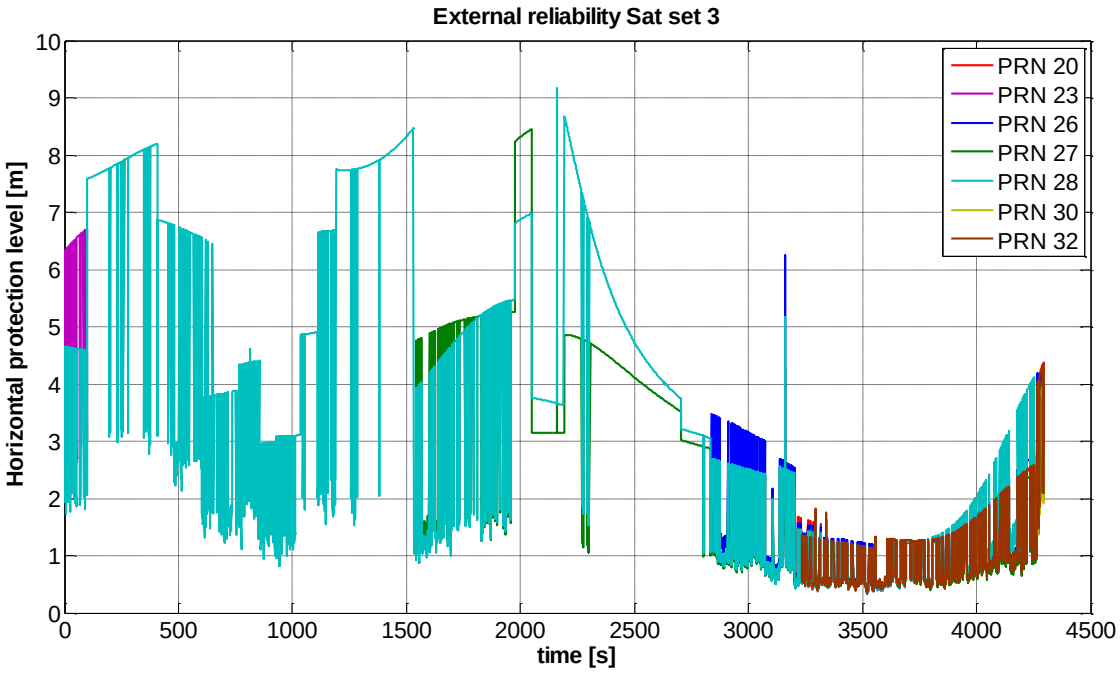

Figure 18 Horizontal protection level Sat set 3

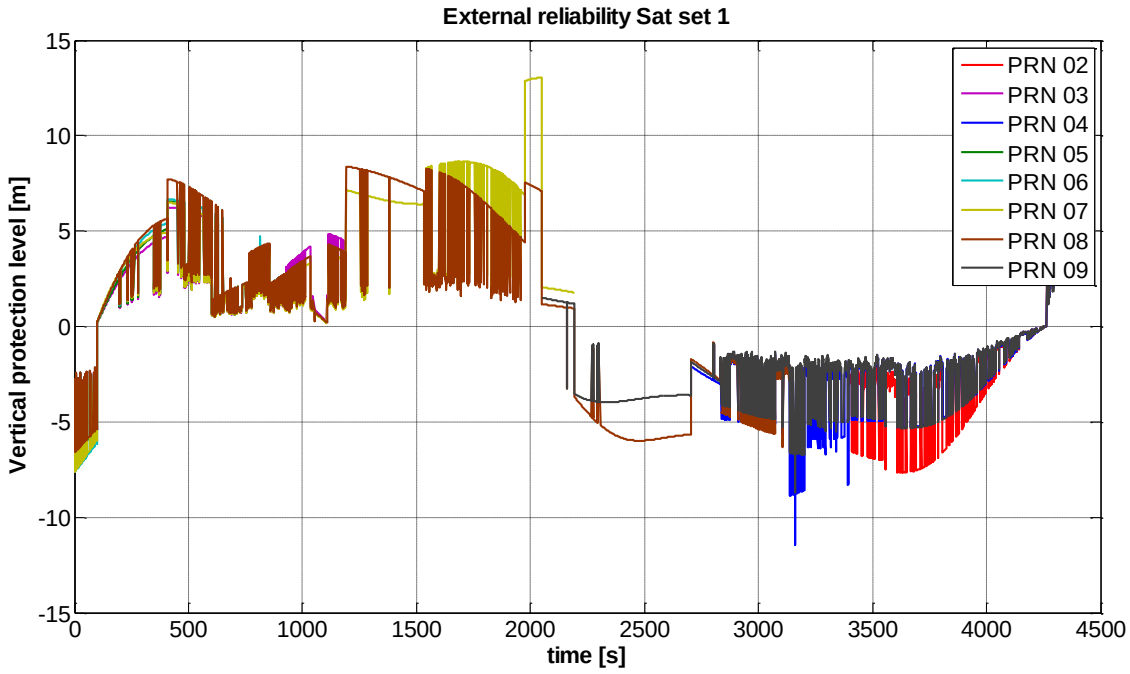

Figure 19 Vertical protection level - Sat set 1

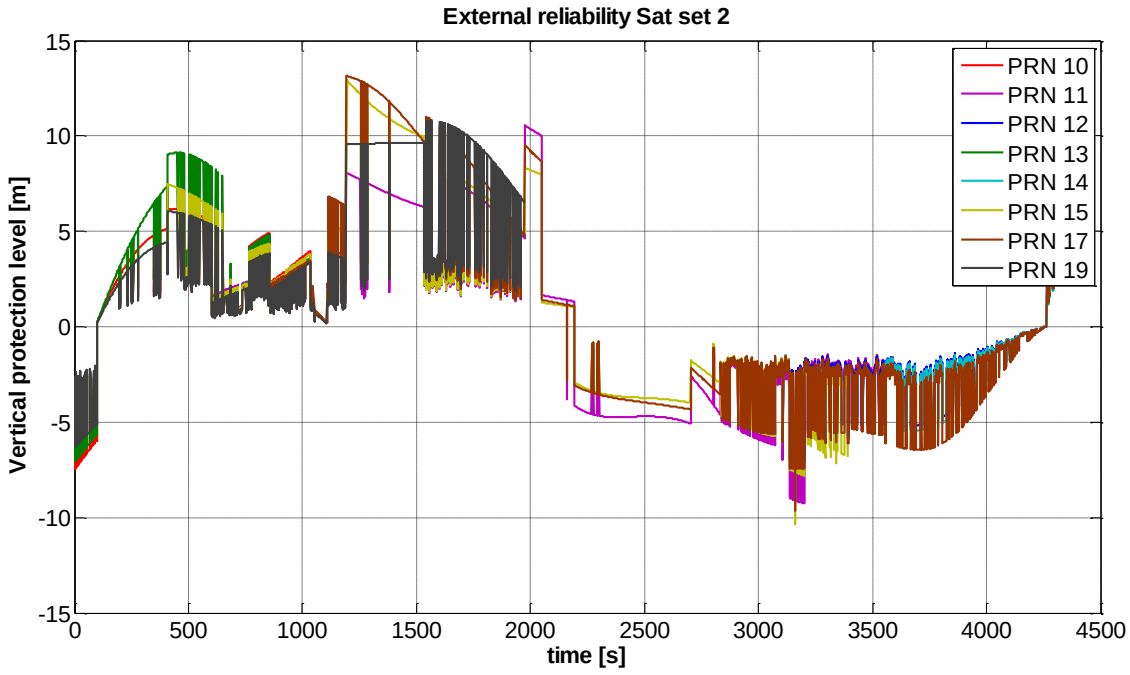

Figure 20 Vertical protection level - Sat set 2

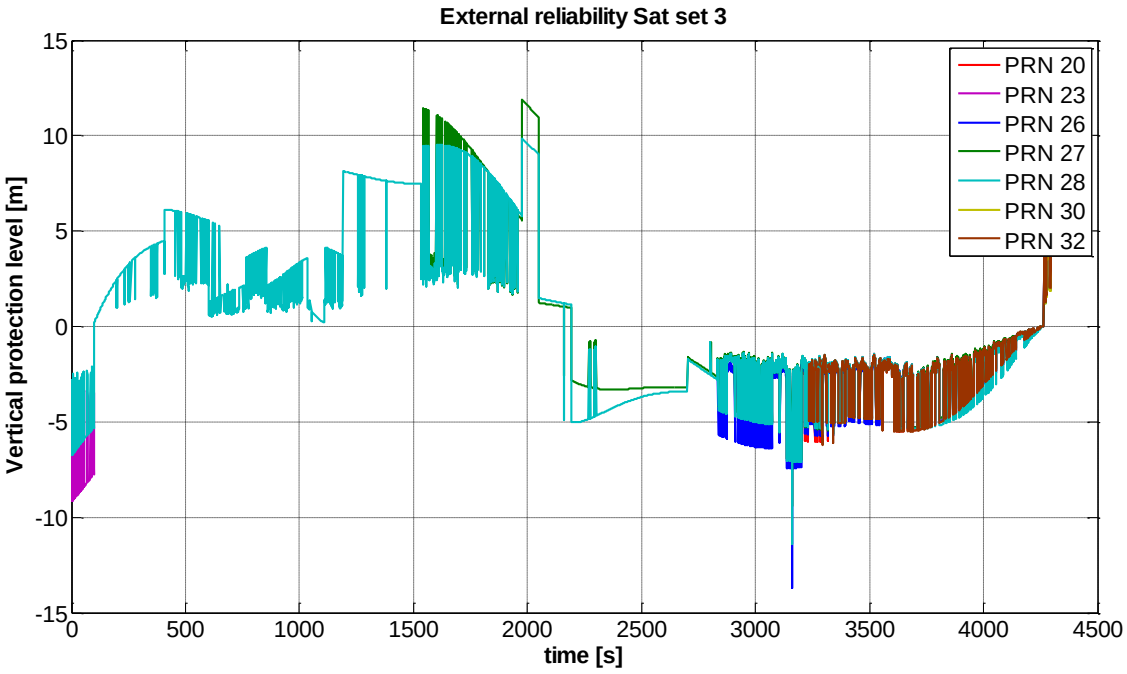

Figure 21 Vertical protection level - Sat set 3

3.4 Height Constraint

LS solution with height constraint in curvi-linear co-ordinate frame is computed with known height as a pseudo-observation. Figure 22 shows the state errors. Comparing this figure with Figure 4 and Table 6, following observations can be drawn. Errors along all three axes have reduced relatively since states are inter-dependent in curvi-linear solution. A known height with a little variance acts as an additional input, thus increasing the number of measurement.

Table 8 Position error with height constraint

Direction	Weighted R	
	Mean [m]	Std [m]
E	0.12	0.62
N	0.05	0.95
U	0.04	0.02

Profile of DOP is also improved not because of improvement in geometry, but due to the presence of additional input even when some measurements with blunders are rejected. DOP plot is shown in Figure 23.

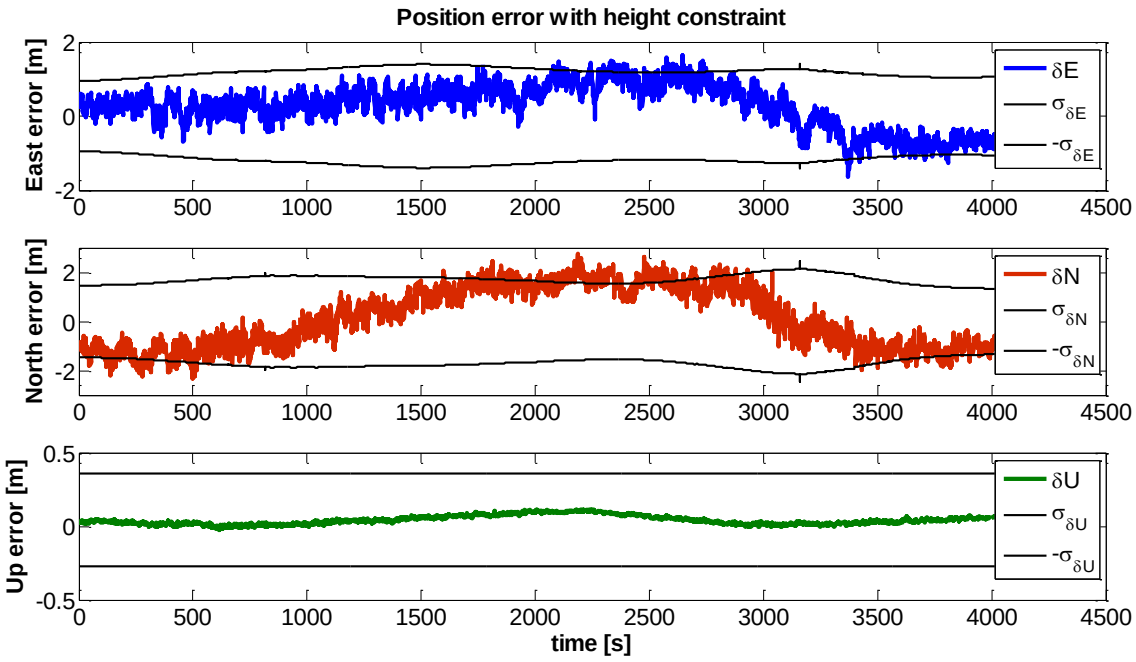

Figure 22 Position error with height constraint

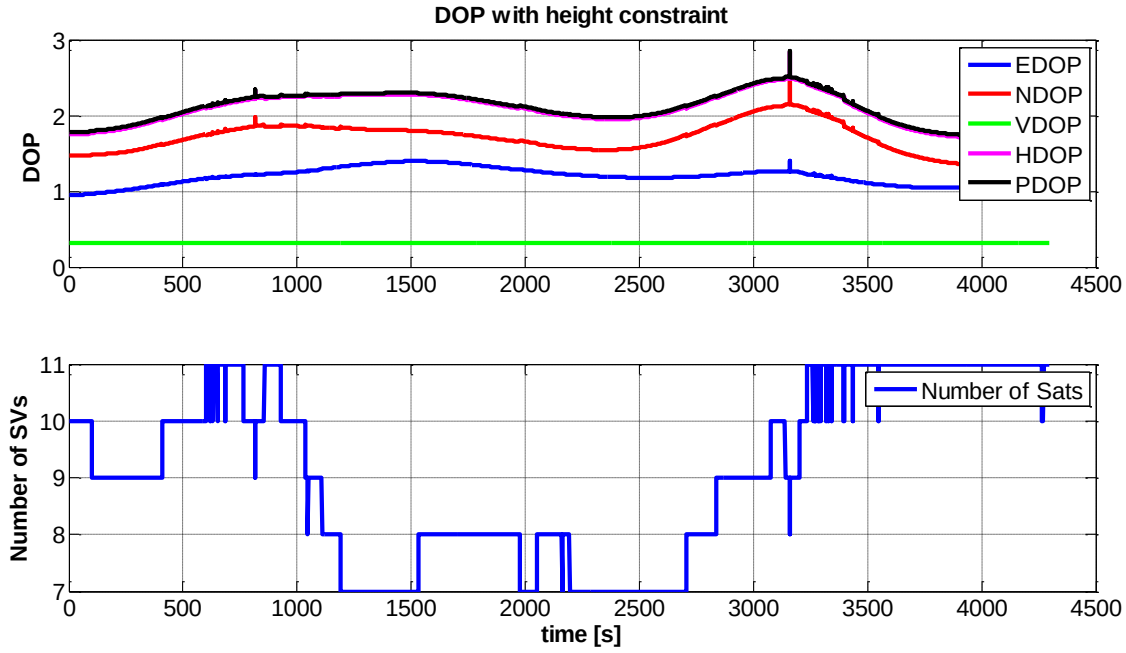

Figure 23 DOP with height constraint

Figure 24 shows 2D error ellipse of position computed with height constraint. There is no much improvement in the position from first epoch. Size of error ellipse from the last epoch has reduced when compared to Figure 6. Reduction is size of error ellipse improves probability of confidence.

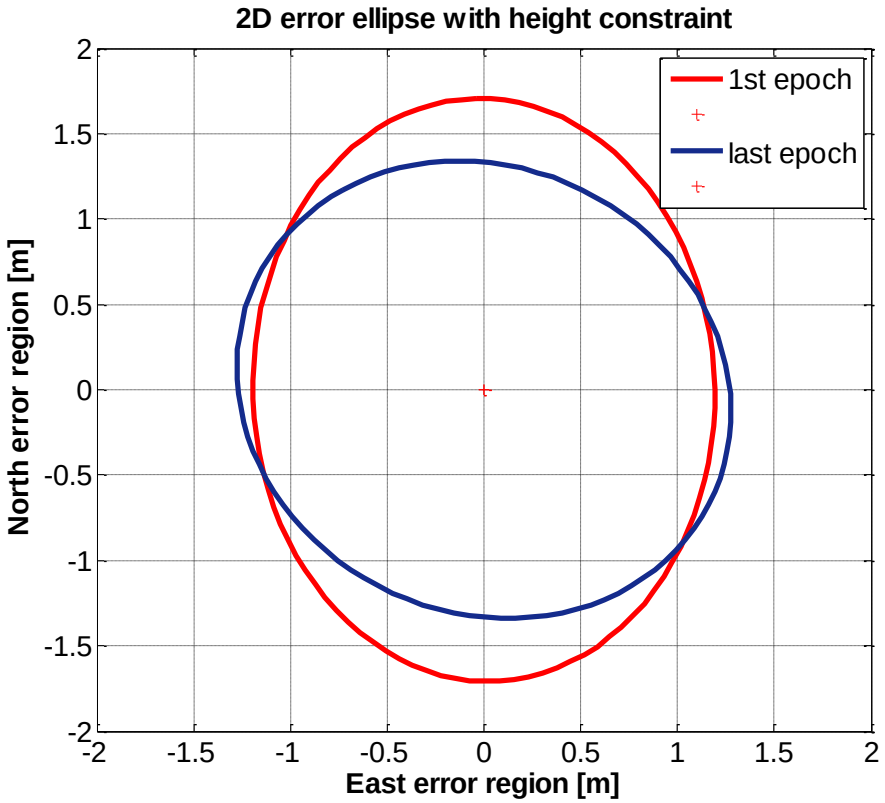

Figure 24 2D error ellipse with height constraint