Estimating Pseudorange Noise from Residuals

Thursday, October 29, 2015 3:47 PM

PART I: Characterizing GPS Measurement Noise when Ground Truth is known

Ordinarily we would estimate sensor noise using ground truth. For GPS model pseudorange as

$$\delta\rho = \rho_{meas} - \rho_{model}(\vec{x})$$

x is the state vector

Model of pseduorange

Measured pseudorange

Innovation

Note $\rho_{meas} = \bar{r} + \bar{b} + \nu$

Random error vector

True clock shift

True range

$$\delta\rho = \bar{r} + \bar{b} + \nu - \rho_{model}$$

$$= \bar{\delta\rho} + \nu \qquad \text{where} \quad \bar{\delta\rho} = \bar{r} + \bar{b} - \rho_{model}$$

Use overbar to denote true value of innovation

If we knew the true value of clock and position (e.g. for a well surveyed installation), then we could measure error directly:

$$\nu = \rho_{meas} - \bar{r} - \bar{b}$$

In turn, if we could had many samples of the random error nu, then we could compute metrics to describe the sample population. For instance, we could compute variance. I believe the correct formula is:

$$R = E[\nu\nu^T] \approx \frac{1}{N}\sum_{i=1}^{N} \nu\nu^T$$

Note this assumes mean is known to be zero

Part II: Characterizing Code Measurement Noise with Carrier Phase Data

Sometimes carrier phase is used to serve as ground truth for a moving platform. The code-minus-carrier signal can be used to estimate multipath and thermal noise; however, tropo errors are not observable (because they are common to code and carrier) and static iono errors are not observable (because the carrier integer is unknown). Note that ionosphere divergence effects (changes in time) are doubled.

$$\text{code:} \quad \rho = r + b + I + T + \nu_{therm,mult}$$

$$\text{carrier} \quad \phi = r + b + \lambda N - I + T + \gamma_{therm,mult}$$

$$CMC = \rho - \phi = \nu_{therm,mult} - \lambda N + 2I + \gamma_{therm,mult}$$

Small: carrier thermal noise and multipath is much smaller than code

$$\langle CMC \rangle = \frac{1}{N}\sum_{i=1}^{N} CMC_i \approx 2I(t_0) - \lambda N$$

$$\delta CMC = CMC - \langle CMC \rangle \cong \nu_{therm,mult} + 2(I - I(t_0))$$

If ionosphere changes little over the samples considered then

$$\rightarrow I \approx I(t_0)$$

$$\therefore$$

$$\delta CMC \cong \nu_{therm,mult}$$

In other words, the code-minus-carrier signal can be used to directly observe the thermal noise and multipath error over short periods of time when the ionosphere delay is approximately constant. This noise is NOT the same thing as the noise from Part I (ground-truth based measurement characterization) because the noise vector there include multipath, thermal, troposphere, ionosphere, and ephemeris errors.

Part III: Characterizing Measurement Noise from Measurement Residuals

Now let's consider another case. What if we don't have ground truth, but we want to characterize measurement noise? (And what if we don't have carrier data?) Can we still estimate measurement noise?

We do still have some characterization of error through measurement residuals (so long as there are more than 4 measurements). However, we do not have full observability of the measurement errors, so we cannot easily compile a set of samples. That makes it hard to characterize the covariance (as in the last equation of Part I). So how can we make up for the lack of observability?

One possibility is to use a model. For instance, a very simple model might define variance to be a function of satellite elevation.

variance to be a function of satellite elevation.

$\hat{\sigma}^2 = f(E)$ ← Model of σ^2

↑ Elevation

If we assume that all satellite errors are independent then the covariance model is :

$$\hat{R} = \begin{bmatrix} f(E_1) & 0 & 0 & \cdots \\ 0 & f(E_2) & 0 & \cdots \\ 0 & 0 & f(E_3) & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

This model isn't necessarily very good (because it ignores correlation between troposphere and ionosphere errors) and it ignores multipath effects that depend on the proximity of buildings. That said, this model is a place to start.

Using a model like this we could regularize the statistics of the residual vector to remove the effects of geometry and changing number of satellites in view. We could then look across different models and identify the best model that describes our data. By this process, we have an indirect means of computing the covariance R!

How might this work if we were to define an algorithm?

Let's start by looking at the residual of the position solution. We will define the residual as equal to the innovation when the iterative solution is converged.

$r \equiv \delta\rho = G\,\delta x$

↑ final iteration

Note $\delta\rho = \bar{\delta\rho} + \nu$

$\delta x = \bar{\delta x} + \varepsilon$ ← position error

↑ true position correction

When del-x is zero then the true correction is exactly opposite to the error. (But no further correction can happen because del-x is zero, so the error remains!)

$\therefore \quad \bar{\delta\rho} + \nu = G(\bar{\delta x} + \varepsilon)$

but $\bar{\delta\rho} = G\bar{\delta x} \approx 0 \qquad \nu = G\varepsilon$

Now let's augment the state vector to include the notion of a parity vector **p**. In this case we can rewrite the above equations:

$$\delta\rho = \begin{bmatrix} G & N \end{bmatrix} \begin{bmatrix} \delta x \\ p \end{bmatrix}$$

Null space matrix defined such that $N^T G = 0$

In other words, the basis for the Null space is orthogonal to the basis for the geometry matrix G

If the pseudorange error is zero mean, then the parity vector is a zero-mean random vector. The value of the parity vector can be related directly to the noise.

Removing $\bar{\delta\rho} = G\,\bar{\delta x}$

$$\nu = \begin{bmatrix} G & N \end{bmatrix} \begin{bmatrix} \varepsilon \\ p \end{bmatrix}$$

Multiply by N transpose:

$$N^T\nu = \underbrace{N^T G}_{N^TG=0}\,\varepsilon + \underbrace{N^T N}_{N^TN=I}\,p$$

Note here that N is orthogonal matrix, such that all its columns are orthonormal and unit length

$$\boxed{p = N^T\nu}$$

Note that by the same argument, we can also relate the parity vector to the residual.

$$r = \delta\rho = \begin{bmatrix} G & N \end{bmatrix} \begin{bmatrix} \delta x \\ p \end{bmatrix}$$

$$N^T r = \cancel{N^T G}^{\,0}\,\delta x + \cancel{N^T N}^{\,I}\,p$$

$$\boxed{p = N^T r}$$

These equations are subtlely different. The error vector (**nu**) has a lot more information than the residual (**r**). This information is all orthogonal to the null space basis (it is geometry information), though, so the difference between **nu** and **r** does not factor into **p**.

The covariance for the parity vector is:

$$Q = E[pp^T] = E[N^T \nu \nu^T N]$$
$$= N^T E[\nu\nu^T] N$$
$$= N^T R N$$

$$\boxed{Q = N^T R N}$$

Now let's do something a little unusual. Let's regularize the parity so that the resulting regularized vector has an identity covariance matrix. An identity covariance matrix means that each element of the random vector had an identical and independent distribution (IID).

Define $y = L^{-1} p$

where $Q = LL^T$

Note L is sometimes called $Q^{1/2}$

L can be computed with a Cholesky decomposition

Check the covariance of **y**

$$E[yy^T] = E[L^{-1} pp^T L^{-T}]$$
$$= L^{-1} E[pp^T] L^{-T}$$
$$= L^{-1} Q L^{-T} = L^{-1} L L^T L^{-T} = I$$

By construction we have defined **y** to have a covariance matrix equal to identity.

Another way to write **y** is

$y = L^{-1} p$ → $Q = N^T R N = LL^T = N^T R^{1/2} (R^{1/2})^T N$

$\therefore L = N^T R^{1/2}$

$$\rightarrow \quad p = N^T r$$

$$\therefore \quad \boxed{y = \left(N^T R^{1/2}\right)^{-1} N^T r}$$

This equation is important, because it lets us convert a residual vector (**r**) into an approximately IID set of samples using a model covariance matrix R. If the model is good, then the samples should look more like an IID distribution with unity variance. So we have a way to test if our model is good!

For instance, we could look for a matrix R that makes the standard deviation of all samples as close to one as possible. The cost function might be J:

$$J(\hat{R}) = \left(\hat{\sigma}^2 - 1\right)^2$$

where

$$\hat{\sigma}^2 = \frac{1}{M}\sum_{i=1}^{M} \frac{1}{N_i} y_i^T y_i$$

Here N_i is the number of vector elements in each vector $\mathbf{y}_i$; M is the number of sample vectors $\mathbf{y}_i$.

Recall that y is a function of the model covariance Rhat.

The condition that the variance of y should be one is a necessary (but not necessarily sufficient) condition for a good model Rhat.

By minimizing J over a set of models Rhat, we can identify the best candidate model. For CERIM, this would mean that we could compute Q from the best Rhat, evaluated at each time step.