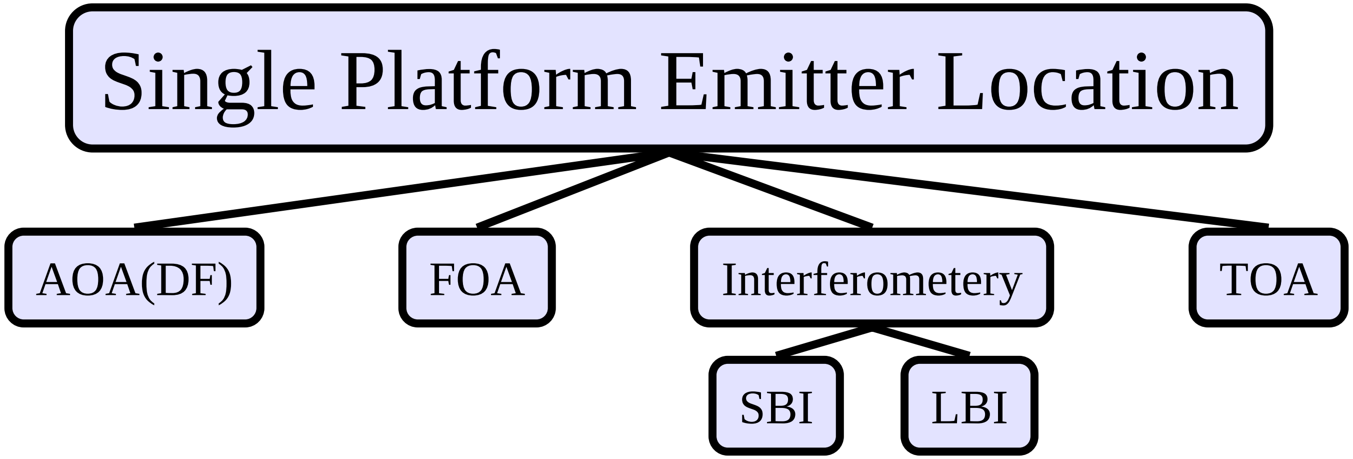


Emitter Location is Two Estimation Problems in One:

1) **Estimate Signal Parameter(s) that Depend on Emitter's Location:**

   a) Time-of-Arrival (TOA) of Pulses

   b) Phase Interferometery: Phase is measured between two different signals received at nearby antennas

      - SBI – Short Baseline Interferometery (antennas are close enough together that phase is measured without ambiguity)
      - LBI – Long Baseline Interferometery (antennas are far enough apart that phase is measured with ambiguity; ambiguity resolved either using processing or so-called self-resolved)

   c) Frequency-of-Arrival (FOA) or Doppler

   d) Angle-of-Arrival (AOA)

2) **Use Signal Parameters Measured at Several Instants to Estimate Location**

Frequency-Based Location (i.e. Doppler Location)

The Problem

- **Emitter assumed non-moving and at position (X,Y,Z)**
  - Transmitting a radar signal at unknown carrier frequency is f_o
- **Signal is intercepted by a receiver on a single aircraft**
  - A/C dynamics are considered to be perfectly known as a function of time
    - Nav Data: Position $X_p(t)$, $Y_p(t)$, $Z_p(t)$ and Velocity $V_x(t)$, $V_y(t)$, $V_z(t)$
- **Relative motion between the Tx and Rx causes Doppler shift**
  - Received carrier frequency differs from transmitted carrier frequency
  - Thus, the carrier frequency of the received signal will change with time
- **For a given set of nav data, how the frequency changes dependS on the transmitter's carrier frequency f_o and the emitter's position (X,Y,Z)**
  - Parameter Vector: $\mathbf{x} = [X\ Y\ Z\ f_o]^T$
  - f_o is a "nuisance" parameter
- **Received frequency is a function of time as well as parameter vector x**

$$f(t,\mathbf{x}) = f_o - \frac{f_o}{c}\left[\frac{V_x(t)\left(X_p(t)-X\right)+V_y(t)\left(Y_p(t)-Y\right)+V_z(t)\left(Z_p(t)-Z\right)}{\sqrt{\left(X_p(t)-X\right)^2+\left(Y_p(t)-Y\right)^2+\left(Z_p(t)-Z\right)^2}}\right] \qquad (1)$$

- **Make noisy frequency measurements at $t_1, \dots, t_N$:** $\tilde{f}(t_i,\mathbf{x}) = f(t_i,\mathbf{x}) + v(t_i)$
- **Problem: Given noisy frequency measurements and the nav data, estimate x**
- **What PDF model do we use for our data????**

In the TDOA/FDOA case… we had an ML estimator for TDOA/FDOA so we could claim that the measurements were asymptotically Gaussian. Because we then had a well-specified PDF for the TDOA/FDOA we could hope to use ML for the location processing.

However, here we have no ML estimator for the instantaneous frequency so claiming that the inst. freq. estimates are Gaussian is a bit of a stretch.

So we could:

1. Outright **ASSUME** Gaussian and then use ML approach
2. Resort to LS… which does not even require a PDF viewpoint!

Both paths get us to the exact same place:

Find the estimate that minimizes

$$J(\hat{\mathbf{x}}_e) = \sum_{i=1}^{N} [\tilde{f}(t_i, \mathbf{x}_e) - f(t_i, \hat{\mathbf{x}}_e)]^2$$

If we Assume Gaussian… we could choose:

- Newton-Raphson MLE approach: leads to double derivatives of the measurement model $f(t_i, \mathbf{x}_e)$.

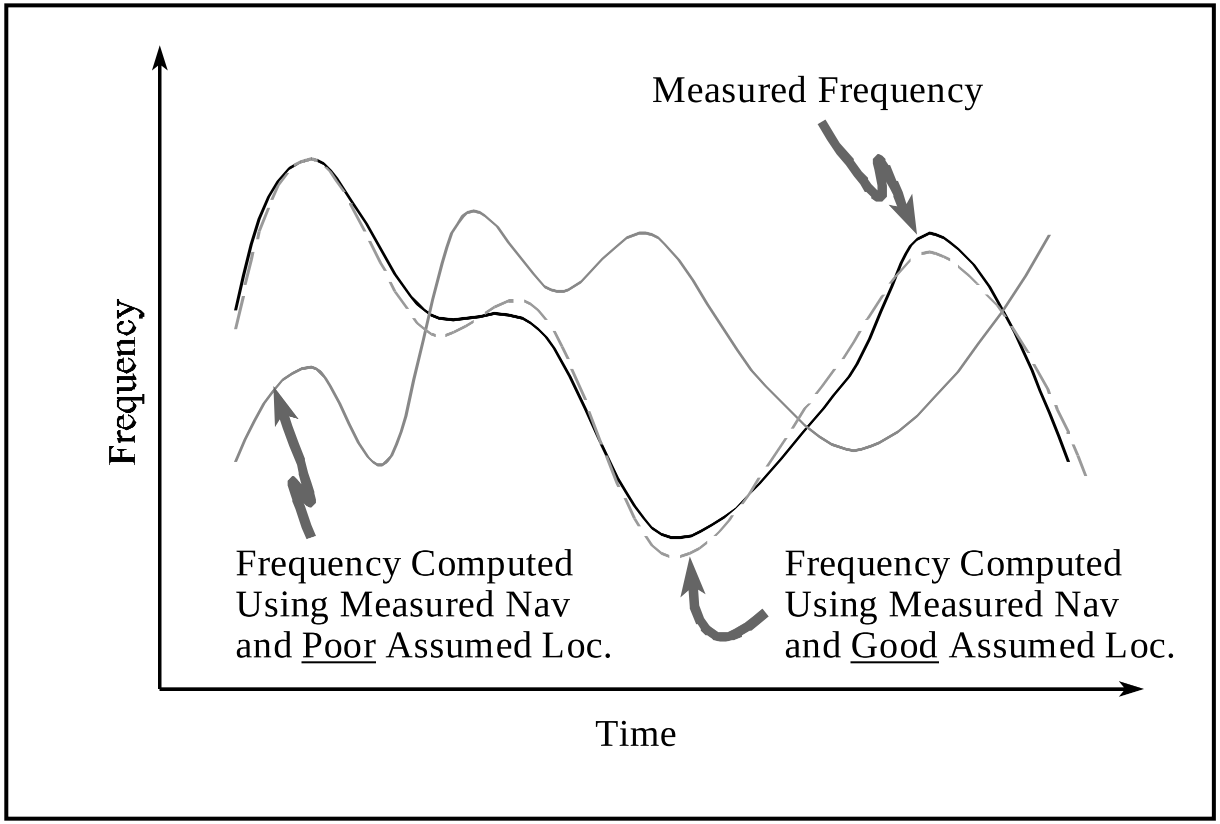
If we Resort to LS… we could choose either:

- Newton-Raphson approach, which in this case is identical to N-R under the Gaussian assumption
- Gauss-Newton approach, which needs only first derivatives of the measurement model $f(t_i, \mathbf{x}_e)$.

We'll resort to LS and use Gauss-Newton

LS Approach: Find the estimate $\hat{\mathbf{x}}$ such that the corresponding computed frequency measurements $f(t_i, \hat{\mathbf{x}})$ are "close" to the actual measurements:

- **Minimize** $J = \sum_{i=1}^{N} [\tilde{f}(t_i, \mathbf{x}) - f(t_i, \hat{\mathbf{x}})]^2$



The Solution

Measurement model in (1) is nonlinear in **x** ➔ no closed form solution

- Newton-Raphson: Linearize the derivative of the cost function
- **Gauss-Newton: Linearize the measurement model**

Thus: $\tilde{\mathbf{f}}(\mathbf{x}) \approx \mathbf{f}(\hat{\mathbf{x}}_n) + \mathbf{H}[\mathbf{x} - \hat{\mathbf{x}}_n] + \mathbf{v}$ ➔ $\Delta\mathbf{f}(\hat{\mathbf{x}}_n) \approx \mathbf{H}\,\Delta\mathbf{x} + \mathbf{v}$ (A Linear Model)

where… $\mathbf{H} = \dfrac{\partial}{\partial \mathbf{x}} f(\mathbf{t}, \mathbf{x})\Big|_{\mathbf{x}=\hat{\mathbf{x}}_n} = [\mathbf{h}_1 \,|\, \mathbf{h}_2 \,|\, \mathbf{h}_3 \,|\, \mathbf{h}_4]$

Get LS solution for update and then update current estimate:

$$\Delta\hat{\mathbf{x}} = \left(\mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}\right)^{-1} \mathbf{H}^T \mathbf{R}^{-1}\, \Delta\mathbf{f}(\hat{\mathbf{x}}_n) \qquad \hat{\mathbf{x}}_{n+1} = \hat{\mathbf{x}}_n + \Delta\hat{\mathbf{x}}$$

Under the condition that the frequency measurement errors are Gaussian, then the CRLB for the problem can be shown to be

$$\text{var}\{\mathbf{x}\} \geq \left(\mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}\right)^{-1}$$

Can use this to investigate performance under geometries of interest… even when the measurement errors aren’t truly Gaussian

The Algorithm

Initialization:

- Use the average of the measured frequencies as an initial transmitter frequency estimate.
- To get an initial estimate of the emitter's *X*,*Y*,*Z* components there are several possibilities:
  - Perform a grid search
  - Use some information from another sensor (e.g., if other on-board sensors can give a rough angle use that together with a typical range)
  - Pick several typical initial locations (e.g., one in each quadrant with some typical range)
- Let the initial estimate be

$$\boxed{\hat{\mathbf{x}}_0 = [\hat{X}_0 \; \hat{Y}_0 \; \hat{Z}_0 \;\; \hat{f}_{o,0}]}$$

Iteration:

For n = 0, 1, 2, …

1. Compute the vector of predicted frequencies at times $\{t_1, t_2, \ldots, t_N\}$ using the current nth estimate and the nav info:

$$\boxed{\hat{f}(t_j,\hat{\mathbf{x}}_n) = \hat{f}_{o,n} - \frac{\hat{f}_{o,n}}{c}\left(\frac{V_x(t_j)\left(X_p(t_j)-\hat{X}_n\right)+V_y(t_j)\left(Y_p(t_j)-\hat{Y}_n\right)+V_z(t_j)\left(Z_p(t_j)-\hat{Z}_n\right)}{\sqrt{\left(X_p(t_j)-\hat{X}_n\right)^2+\left(Y_p(t_j)-\hat{Y}_n\right)^2+\left(Z_p(t_j)-\hat{Z}_n\right)^2}}\right)}$$

$$\boxed{\mathbf{f}(\hat{\mathbf{x}}_n) = \left[\hat{f}(t_1,\hat{\mathbf{x}}_n)\ \hat{f}(t_2,\hat{\mathbf{x}}_n) \cdots \hat{f}(t_N,\hat{\mathbf{x}}_n)\right]^T}$$

2. Compute the residual vector by subtracting the predicted frequency vector from the measured frequency vector:

$$\boxed{\Delta\mathbf{f}(\hat{\mathbf{x}}_n) = \tilde{\mathbf{f}}(\mathbf{x}) - \hat{\mathbf{f}}(\hat{\mathbf{x}}_n)}$$

3. Compute Jacobian matrix **H** using the nav info and the current estimate:

$$\mathbf{H}=\frac{\partial}{\partial \mathbf{x}}f(\mathbf{t},\mathbf{x})\Big|_{\mathbf{x}=\hat{\mathbf{x}}_n}=\left[\mathbf{h}_1\,|\,\mathbf{h}_2\,|\,\mathbf{h}_3\,|\,\mathbf{h}_4\right]$$

<u>Define</u>: $\Delta\hat{X}_n(t_j)=X_p(t_j)-\hat{X}_n \qquad \Delta\hat{Y}_n(t_j)=Y_p(t_j)-\hat{Y}_n \qquad \Delta\hat{Z}_n(t_j)=Z_p(t_j)-\hat{Z}_n$

$$\hat{R}_n(t_j)=\sqrt{\Delta\hat{X}_n^2(t_j)+\Delta\hat{Y}_n^2(t_j)+\Delta\hat{Z}_n^2(t_j)},$$

$$\mathbf{h}_1(j)=\frac{\partial}{\partial X}f(t_j,\mathbf{x})\Big|_{\mathbf{x}=\hat{\mathbf{x}}_n}=-\frac{\hat{f}_o}{c}\left[\frac{-V_x(t_j)}{\hat{R}_j}+\frac{\Delta\hat{X}_n(t_j)\left[V_x(t_j)\Delta\hat{X}_n(t_j)+V_y(t_j)\Delta\hat{Y}_n(t_j)+V_z(t_j)\Delta\hat{Z}_n(t_j)\right]}{\hat{R}_j^{\,3}}\right]$$

$$\mathbf{h}_2(j)=\frac{\partial}{\partial X}f(t_j,\mathbf{x})\Big|_{\mathbf{x}=\hat{\mathbf{x}}_n}=-\frac{\hat{f}_o}{c}\left[\frac{-V_y(t_j)}{\hat{R}_j}+\frac{\Delta\hat{Y}_n(t_j)\left[V_x(t_j)\Delta\hat{X}_n(t_j)+V_y(t_j)\Delta\hat{Y}_n(t_j)+V_z(t_j)\Delta\hat{Z}_n(t_j)\right]}{\hat{R}_j^{\,3}}\right]$$

$$\mathbf{h}_3(j)=\frac{\partial}{\partial Z}f(t_j,\mathbf{x})\Big|_{\mathbf{x}=\hat{\mathbf{x}}_n}=-\frac{\hat{f}_o}{c}\left[\frac{-V_z(t_j)}{\hat{R}_j}+\frac{\Delta\hat{Z}_n(t_j)\left[V_x(t_j)\Delta\hat{X}_n(t_j)+V_y(t_j)\Delta\hat{Y}_n(t_j)+V_z(t_j)\Delta\hat{Z}_n(t_j)\right]}{\hat{R}_j^{\,3}}\right]$$

$$\mathbf{h}_4(j)=\frac{\partial}{\partial f_o}f(t_j,\mathbf{x})\Big|_{\mathbf{x}=\hat{\mathbf{x}}_n}\approx 1$$

4. Compute the estimate update:

$$\boxed{\Delta\hat{\mathbf{x}}_n = \left(\mathbf{H}^T\mathbf{C}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{C}^{-1}\,\Delta\mathbf{f}(\hat{\mathbf{x}}_n)}$$

C is the covariance of the frequency measurements; usually assumed to be diagonal with measurement variances on the diagonal

In practice you would implement this inverse using Singular Value Decomposition (SVD) due to numerical issues of **H** being near to singular (MATLAB will give you a warning when this is a problem)
See pp. 676-677 of the book "Numerical Recipes …"

5. Update the estimate using

$$\boxed{\hat{\mathbf{x}}_{n+1} = \hat{\mathbf{x}}_n + \Delta\hat{\mathbf{x}}_n}$$

6. Check for convergence of solution: look to see if update is small in some specified sense.

If “Not Converged”… go to Step 1.

If Converged or Maximum number of iterations… quit loop & Set $\hat{\mathbf{x}} = \hat{\mathbf{x}}_{n+1}$

7. Compute Least-Squares Cost of Converged solution

$$C(\hat{\mathbf{x}}) = \sum_{n=1}^{N} \frac{\left(\tilde{f}(t_n, \mathbf{x}) - \hat{f}(t_n, \hat{\mathbf{x}})\right)}{\sigma_n^2}$$

This last step is often done to allow assessment of how much confidence you have in the solution. There are other ways to assess confidence – see discussion in Ch. 15 of “Numerical Recipes …”

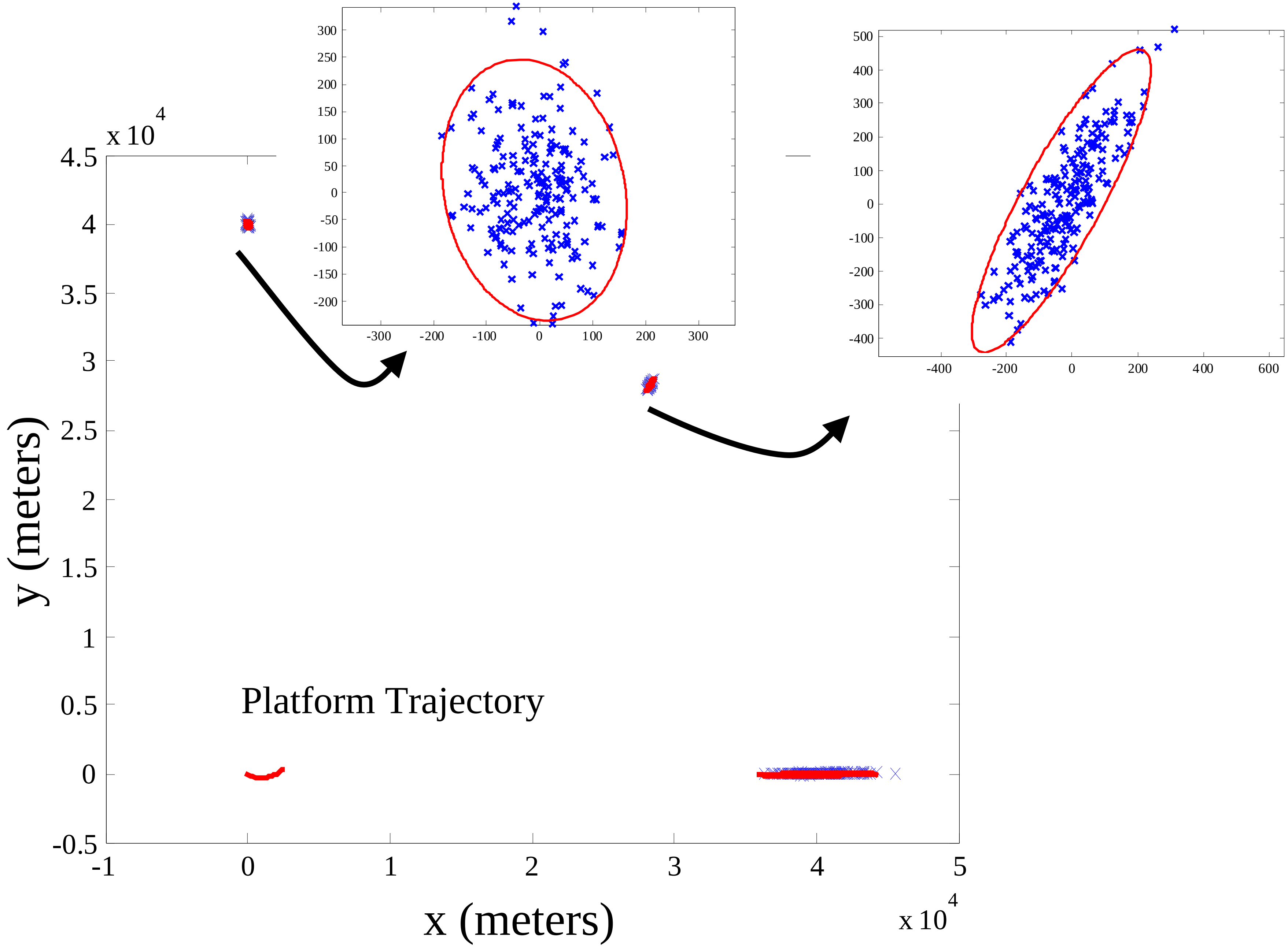
<u>Note</u>: There is no guarantee that this algorithm will converge… it might not converge at all… it might:
(i) simply wander around aimlessly,

(ii) oscillate back and forth along some path, or

(iii) wander off in complete divergence.

In practical algorithms it is a good idea to put tests into the code to check for such occurrences

Simulation Results with 95% CRLB Error Ellipses


x 10^4
y (meters)
Platform Trajectory
x 10^4
x (meters)