ASEN 5070

Statistical Orbit Determination I

Fall 2012

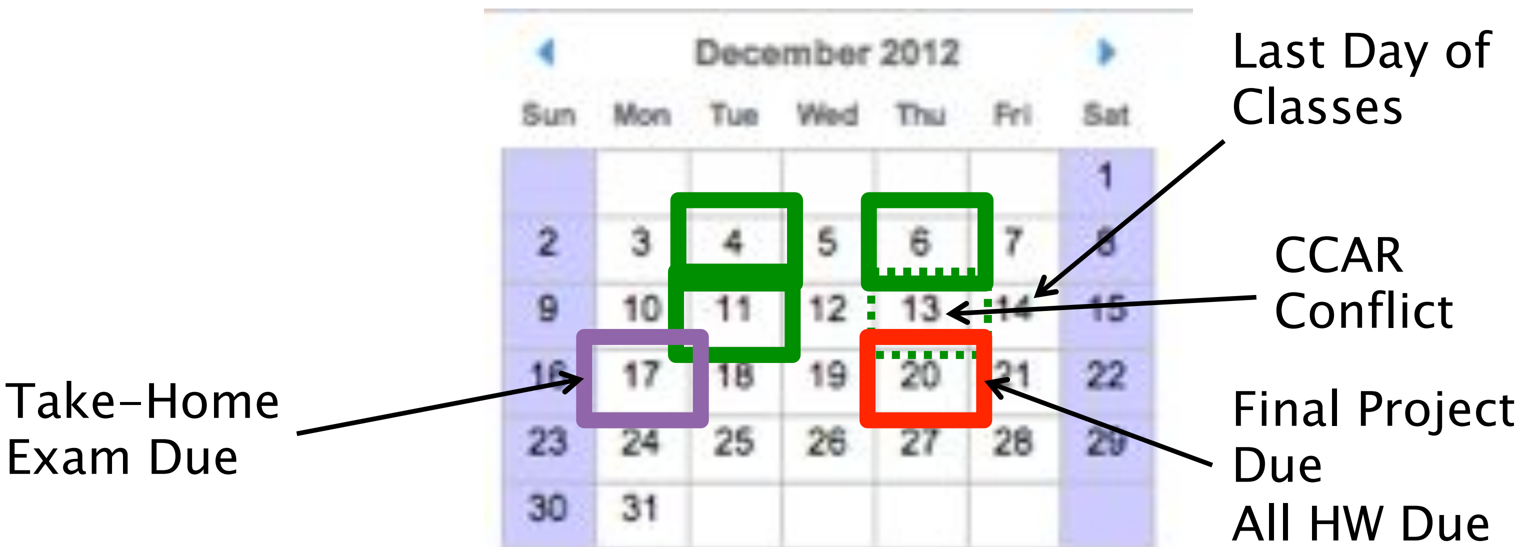
Professor Jeffrey S. Parker

Professor George H. Born

Lecture 25: Error Ellipsoids and Smoothing

Announcements

- HW 11 due this week (solutions soon).



Quiz 21 Review

CCAR
Colorado Center for
Astrodynamics Research

Question 1 (1 point)

Let's say you have a normal situation: you have an a priori state estimate, an a priori covariance, and a bunch of observations spanning 10 minutes of time. You process the observations using the Conventional Kalman Filter (no process noise or anything else). You want to know what the best estimate of the spacecraft's state is at EACH time using ALL observations. True/False: You can do this by taking the final estimate and mapping it back through time using the state transition matrix.

- True
- False

Quiz 21 Review

Question 1 (1 point)

Let's say you have a normal situation: you have an a priori state estimate, an a priori covariance, and a bunch of observations spanning 10 minutes of time. You process the observations using the Conventional Kalman Filter (no process noise or anything else). You want to know what the best estimate of the spacecraft's state is at EACH time using ALL observations. True/False: You can do this by taking the final estimate and mapping it back through time using the state transition matrix.

- True
- False

Just like the Batch

Quiz 21 Review

CCAR
Colorado Center for Astrodynamics Research

Question 2 (1 point)

Let's say you have the same situation as in Problem 1, but in this scenario you've processed all of your observations using a CKF with state noise compensation.

Again, you want to know what the best estimate of the spacecraft's state is at EACH time using ALL observations. True/False: You can do this by taking the final estimate and mapping it back through time using the state transition matrix.

- True
- False

Quiz 21 Review

Question 2 (1 point)

Let's say you have the same situation as in Problem 1, but in this scenario you've processed all of your observations using a CKF with state noise compensation.

Again, you want to know what the best estimate of the spacecraft's state is at EACH time using ALL observations. True/False: You can do this by taking the final estimate and mapping it back through time using the state transition matrix.

- True
- False

The introduction of process noise artificially raises the covariance, meaning that you can't just propagate a state estimate or its covariance through time using the state transition matrix.

You have to pay attention to the estimated covariance through time as well.

This leads into our discussion on smoothing!

Quiz 21 Review

CCAR
Colorado Center for Astrodynamics Research

Question 3 (1 point)

Let's say that three countries each tracked a satellite using their own tracking systems. They each come up with their own estimate of the state of the satellite at the same epoch with their own associated covariance matrix.

You want the BEST estimate of the satellite. What is the best practice to compute the best estimate of the satellite given these solutions?

- Trick question: it can't be done.
- Of the three solutions, use the solution that has the smallest covariance matrix.
- Take each of the three solutions and average them to find the best solution.
- Take each of the three solutions and compute a weighted average, weighed appropriately to each solution's covariance.

Quiz 21 Review

Question 3 (1 point)

Let's say that three countries each tracked a satellite using their own tracking systems. They each come up with their own estimate of the state of the satellite at the same epoch with their own associated covariance matrix.

You want the BEST estimate of the satellite. What is the best practice to compute the best estimate of the satellite given these solutions?

- Trick question: it can't be done.
- Of the three solutions, use the solution that has the smallest covariance matrix.
- Take each of the three solutions and average them to find the best solution.
- Take each of the three solutions and compute a weighted average, weighed appropriately to each solution's covariance.

Quiz 21 Review

- Combining estimates

- You can combine *unbiased*, *uncorrelated* estimates using their covariances as weights
  - If they don't have associated covariances, perhaps they have a specified weight.
    - If they don't have anything else, perhaps just average them.

$$\hat{\mathbf{x}} = \left(\sum_{i=1}^{n} P_i^{-1} \right)^{-1} \sum_{i=1}^{n} P_i^{-1} \hat{\mathbf{x}}_i$$

Combining Estimates

- Optimal estimate of x:

$$\mathbf{x} = \hat{\mathbf{x}} + \eta$$

$$E[\eta] = 0$$

$$f(\eta_1, \eta_2, \ldots, \eta_n) = f(\eta_1) \times f(\eta_2) \times \cdots \times f(\eta_n)$$

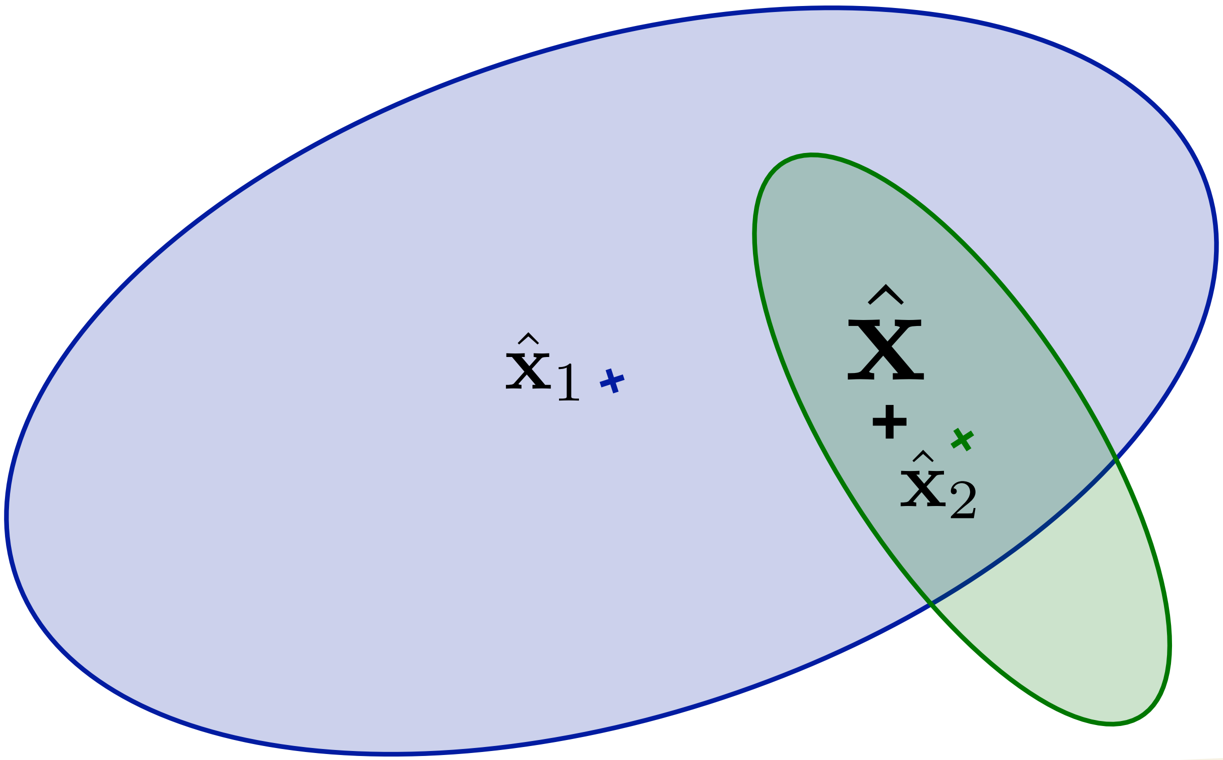
$$E[\hat{\mathbf{x}}] = \mathbf{x}$$

$$\hat{\mathbf{x}} = \left(\sum_{i=1}^{n} P_i^{-1} \right)^{-1} \sum_{i=1}^{n} P_i^{-1} \hat{\mathbf{x}}_i$$

Combining Estimates

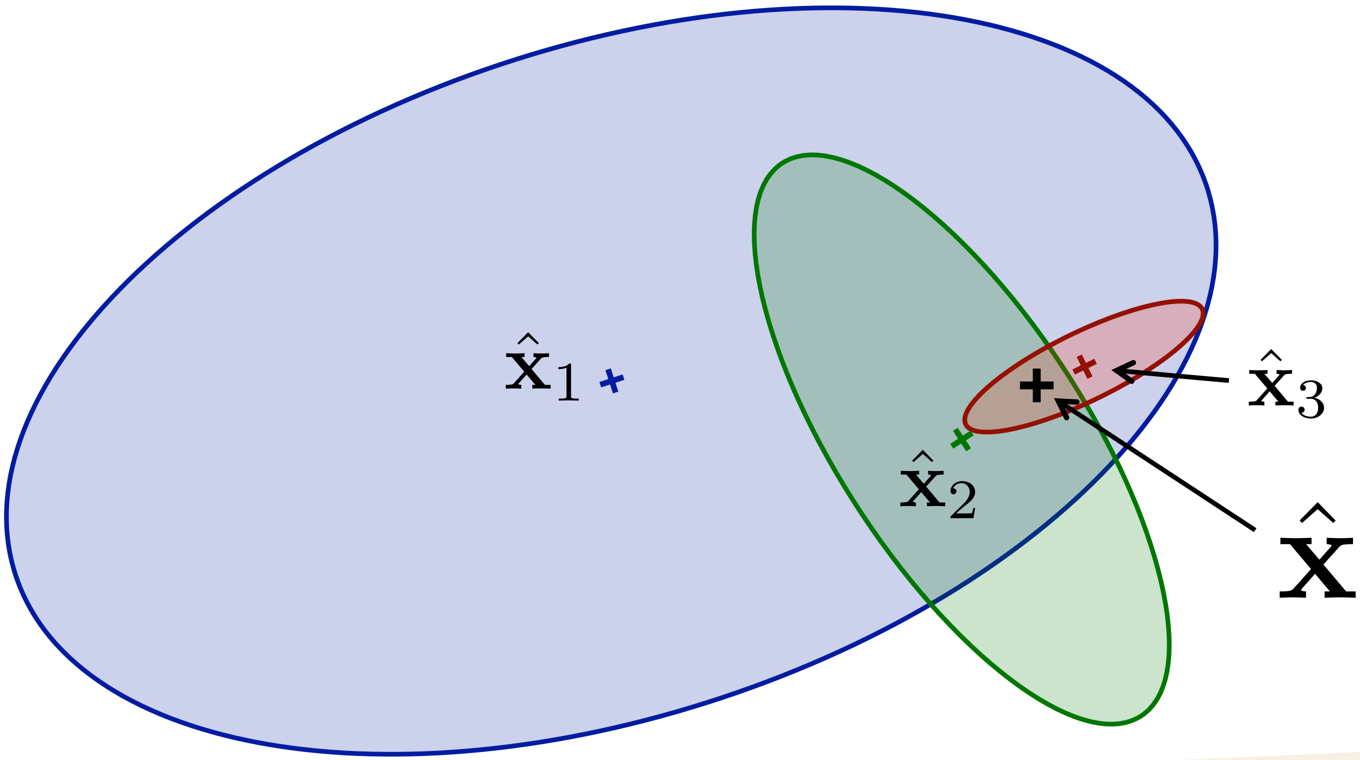
CCAR
Colorado Center for Astrodynamics Research

- Where would the optimal estimate lie given these estimates and associated errors covariances?



Combining Estimates

- Where would the optimal estimate lie given these estimates and associated errors covariances?



Combining Estimates

- Where would the optimal estimate lie given these estimates and associated errors covariances?

A little suspicious…
A good navigator would understand what caused this difference!

Contents – aiming for the future

- Combining Estimates (check)
- EKF Implementation Issues
- Final Project Checklist
- Probability Ellipsoids
- Smoothing
- Monte Carlo
- Overlap Studies, etc.

EKF Implementation Issues

CCAR
Colorado Center for Astrodynamics Research

- Let's consider the Extended Kalman Filter applied to the final project
- A priori covariance matrix:
  - Very large
- Observation error covariances:
  - Very small
- Effect
  - The filter will adjust the best estimate of the state deviation vector to perfectly fit the data.
  - This can cause divergence if the data is noisy!

- Divergence is bad. What should we do?

EKF Implementation Issues

- Two solutions:
  - Solution 1.
  - Adjust the *a priori* covariance matrix.
    - Good? Bad? Why?
      - In general you shouldn't change anything that's been given to you. But in our case, the *a priori* covariance represents a very large uncertainty (except for the boat).
    - What would be a reasonable adjustment?
      - Reasonable to tighten up J2 and CD! Perhaps even tighten the velocities a little.
    - Will this work?
      - Perhaps, but for our case the *a priori* matrix should always represent a very loose initial condition! Which means that almost by definition it shouldn't work!

EKF Implementation Issues

CCAR
Colorado Center for Astrodynamics Research

- Two solutions:
  - Solution 2.
  - Use the CKF until the *a priori* covariance matrix at some time update has come down sufficiently to avoid filter divergence.
    - Good? Bad? Why?
      - This shouldn't be a bad thing at all.
      - Recall that the EKF is postulated to improve the filter performance, not guarantee it!
    - How long should we wait before switching to EKF?
      - Until you're confident that the covariance is small enough to avoid filter divergence.
      - Likely to a point when the *a priori* covariance is still larger than the observation error covariance, but not too much larger.
    - Will this work?
      - Sure! It will avoid filter divergence. SNC, DMC, etc all work with CKF and EKF.
      - This may make smoothing challenging; and you wouldn't want to iterate the solution.

Final Project

CCAR
Colorado Center for Astrodynamics Research

- Requirements
  - These will get you a B at best
- Extra Credit Items
  - These will push you into an A (and make up for some lost points from above)
- Here's what we're expecting in general:

Final Project: Required Elements

- 1. General description of the OD problem and the batch and sequential algorithms.
- 2. Discussion of the results – contrasting the batch processor and sequential filter. Discuss the relative advantages, shortcomings, applications, etc. of the algorithms.
- 3. Show plots of residuals for all algorithms. Plot the trace of the covariance for position and velocity for the sequential filter for the first iteration. You may want to use a log scale.
- 4. When plotting the trace of P for the position and velocity, do any numerical problems show up? If so discuss briefly how they may be avoided.

Final Project: Required Elements

CCAR
Colorado Center for Astrodynamics Research

- 5. Contrast the relative strengths of the range and range rate data. Generate solutions with both data types alone for the batch and discuss the solutions. How do the final covariances differ? You could plot the two error ellipsoids for position. What does this tell you about the solutions and the relative data strength?

- 6. Why did you fix one of the stations? Would the same result be obtained by not solving for one of the stations i.e., leaving it out of the solution list? Does it matter which station is fixed?

- 7. A discussion of what you learned from the term project and suggestions for improving it.

- Clear and complete. This should be a complete report with full sentences, clear descriptions, and clean graphics.
- Note: it doesn't have to follow this order: use what makes sense to you!

Final Project: Extra Credit Elements

- 8. Include the Extended Kalman Filter; compare, contrast, etc.
- 9. How does varying the *a priori* covariance and data noise covariance affect the solution? What would happen if we used an *a priori* more compatible with the actual errors in the initial conditions, i.e., a few meters in position etc.
- 10. Do an overlap study (I'll describe this soon).
- 11. Compare the CKF with the Joseph formulation.
- 12. Compare the Potter algorithm's results to the conventional Kalman filter.

Final Project: Extra Credit Elements

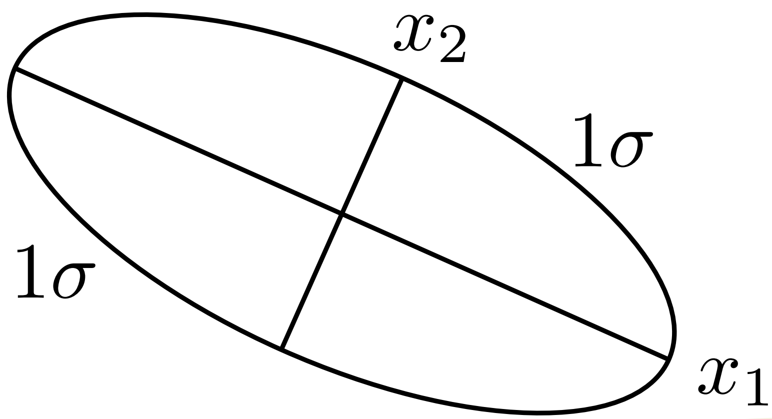
- 13. Solve for the state deviation vector using the Givens square root free algorithm. Compare solution and RMS residuals for range and range rate from Givens solution with results from conventional batch processor (Cholesky and/or Matlab inversion).
- 14. Add in SNC / DMC / other process noise compensation techniques. Compare the results with the CKF.
- Other additions are of course welcome!

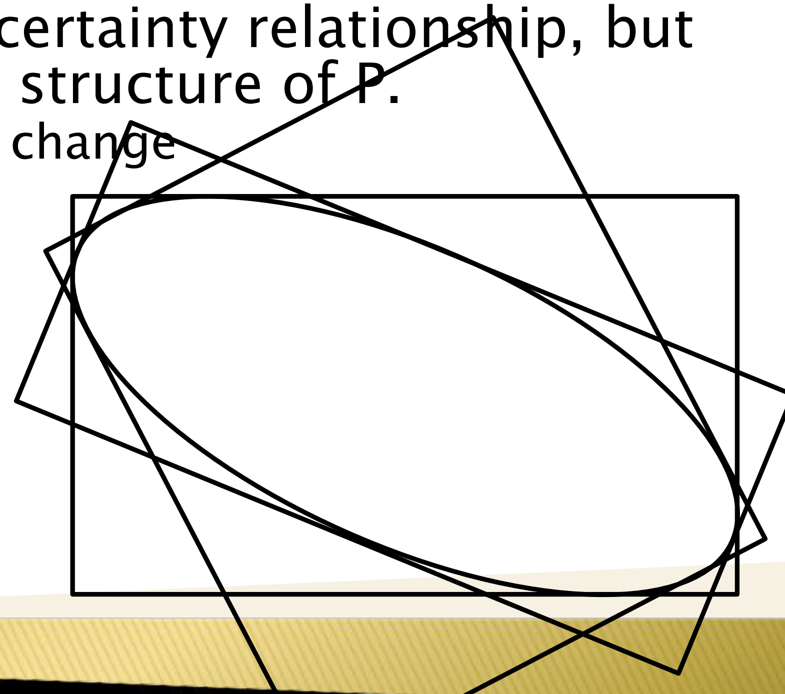
Probability Ellipsoids

- First off, an "ellipsoid" is an n-dimensional ellipse or more generally a hyperellipsoid.

- P, the variance-covariance matrix, represents the uncertainty in the state estimate.

- The truth is that there is an uncertainty relationship, but how you sample it changes the structure of P.
  - As we rotate P, the trace of P may change

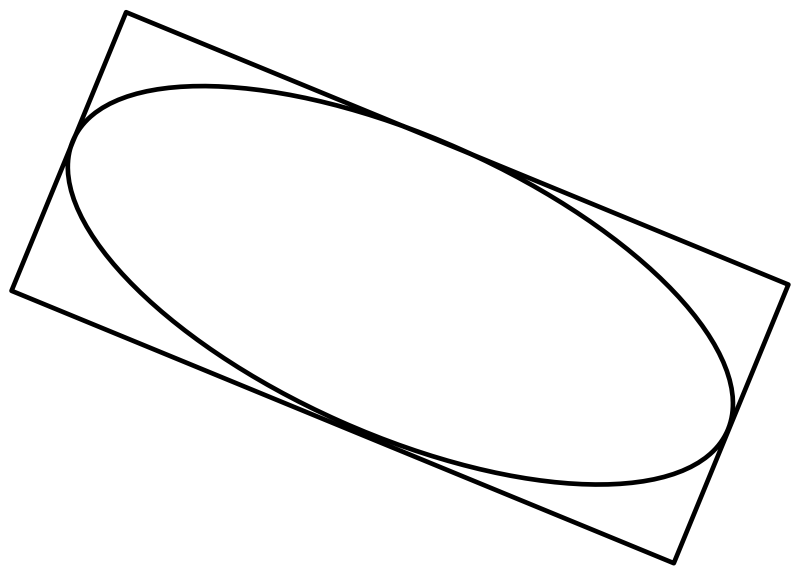




Probability Ellipsoids

CCAR
Colorado Center for
Astrodynamics Research

- Generally the best to represent the probability ellipsoid using the covariance matrix's principal axes.



The Probability Ellipsoid

Given a normally distributed random vector, $\mathbf{x}$, with mean $\overline{\mathbf{x}}$, and variance-covariance P, the function

$$(\mathbf{x}-\overline{\mathbf{x}})^T P^{-1}(\mathbf{x}-\overline{\mathbf{x}}) = \ell^2 \tag{4.16.1}$$

is a positive definite quadratic form representing a family of hyperellipsoids of constant probability density.

- 3-D case is important because we are often interested in the 3-D ellipsoids associated with the position uncertainty of a satellite.

The Probability Ellipsoid

CCAR
Colorado Center for Astrodynamics Research

- Probability ellipse is most easily constructed relative to principal axes
- *Theorem*: If $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n$ is an orthonormal system of Eigenvectors associated, respectively, with the Eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$ of an $n \times n$ symmetric positive definite matrix, P, and if

$$U = [\mathbf{u}_1,\ \mathbf{u}_2, \ldots,\ \mathbf{u}_n]_{n\times n},$$

then

$$U^T P U = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix} = D[\lambda_1,\ \lambda_2, \ldots,\ \lambda_n]; \tag{4.16.2}$$

- $U^T PU$ is a diagonal matrix containing the Eigenvalues of P.

The Probability Ellipsoid

Colorado Center for Astrodynamics Research

$$U = [\mathbf{u}_1,\ \mathbf{u}_2, \ldots,\ \mathbf{u}_n]_{n \times n},$$

then

$$U^T P U = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix} = D[\lambda_1,\ \lambda_2, \ldots,\ \lambda_n]; \tag{4.16.2}$$

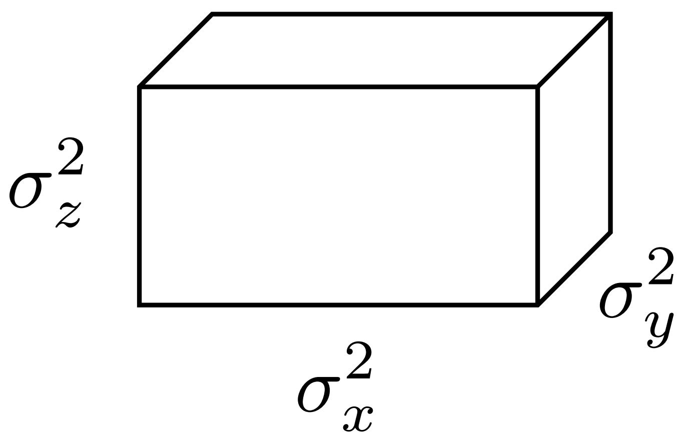
- The transformation matrix U^T used to diagonalize P is called a *principal axes transformation*.
- For the random vector $\mathbf{x}$ with mean $\bar{\mathbf{x}}$ and variance-covariance, P, the principal axes, $\mathbf{x}'$, are given by

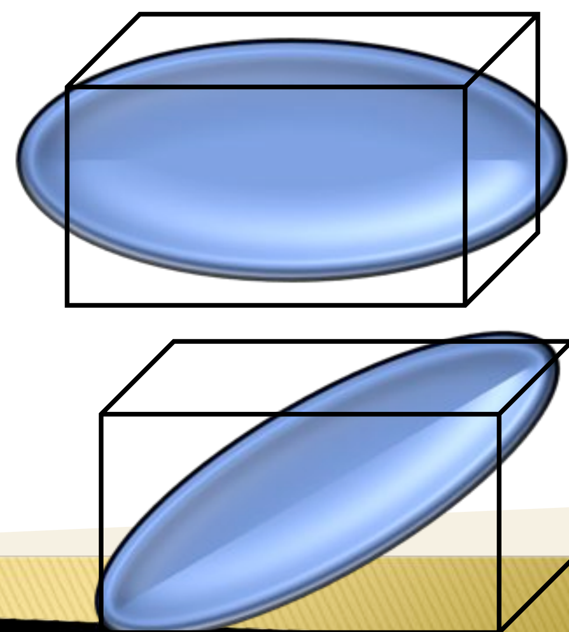
$$\mathbf{x}' = U^T \mathbf{x}. \tag{4.16.3}$$

The Probability Ellipsoid

CCAR
Colorado Center for
Astrodynamics Research

- This is really useful, because if P is oriented in Cartesian coordinates, we don't really know what the size of a probability ellipsoid is.





The Probability Ellipsoid

- The variance-covariance matrix, P', associated with the principal axes is given by

$$\begin{aligned} P' &\equiv E[(\mathbf{x}' - \overline{\mathbf{x}}')(\mathbf{x}' - \overline{\mathbf{x}}')^T] \\ &= U^T E[(\mathbf{x} - \overline{\mathbf{x}})(\mathbf{x} - \overline{\mathbf{x}})^T]\, U \\ &= U^T P U = D\,[\lambda_1 \ldots \lambda_n]. \end{aligned} \tag{4.16.4}$$

- Let $\Delta\mathbf{x}$ represent the estimation error vector defined by

$$\Delta\mathbf{x} \equiv \hat{\mathbf{x}} - \mathbf{x} \equiv [\tilde{x}\ \tilde{y}\ \tilde{z}]^T,$$

with zero mean and the estimation error variance-covariance matrix is given by

$$P = E[\Delta\mathbf{x}\Delta\mathbf{x}^T]. \tag{4.16.5}$$

The Probability Ellipsoid

Colorado Center for Astrodynamics Research

- The equation for the probability ellipsoid is

$$[\tilde{x}\ \tilde{y}\ \tilde{z}]P^{-1}\begin{bmatrix}\tilde{x}\\ \tilde{y}\\ \tilde{z}\end{bmatrix} = \ell^2. \tag{4.16.6}$$

- The ellipsoids for $\ell = 1, 2,$ and 3 are called the 1σ, 2σ, and 3σ error ellipsoids
- In the principal axis system the probability ellipsoids are given by

$$[\tilde{x}'\ \tilde{y}'\ \tilde{z}']\begin{bmatrix}1/\lambda_1 & & \\ & 1/\lambda_2 & \\ & & 1/\lambda_3\end{bmatrix}\begin{bmatrix}\tilde{x}'\\ \tilde{y}'\\ \tilde{z}'\end{bmatrix} = \ell^2, \tag{4.16.8}$$

- or

$$\frac{\tilde{x}'^2}{\lambda_1} + \frac{\tilde{y}'^2}{\lambda_2} + \frac{\tilde{z}'^2}{\lambda_3} = \ell^2. \tag{4.16.9}$$

The Probability Ellipsoid

The axes of the 1σ ellipsoid are given by solving Eq (4.16.9) for $\ell = 1$ and sequentially setting two of the coordinate values to zero, i.e., to obtain the semimajor axis, a, set $\tilde{x}' = a$ and $\tilde{y}' = \tilde{z}' = 0$.

- The axis of the $\ell\sigma$ ellipsoid are given by

$$a^2 = \ell^2\lambda_1, \quad b^2 = \ell^2\lambda_2, \quad c^2 = \ell^2\lambda_3. \qquad \ell = 1, 2, 3$$

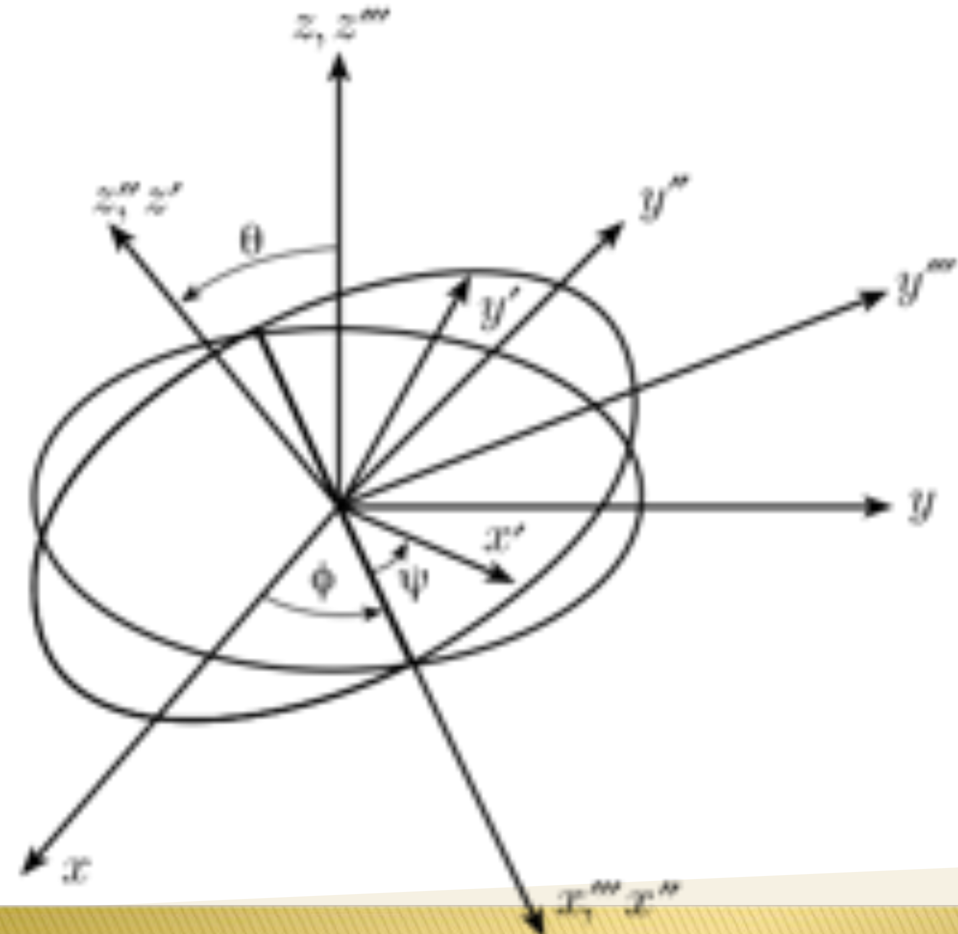
- The orientation of the ellipse relative to the original axis system is obtained by solving for the three Euler angles given by the transformation matrix, U.

$$U = \begin{bmatrix} C_\phi C_\psi - C_\theta S_\phi S_\psi & -C_\phi S_\psi - C_\theta S_\phi C_\psi & S_\theta S_\phi \\ S_\phi C_\psi + C_\theta C_\phi S_\psi & -S_\phi S_\psi + C_\theta C_\phi C_\psi & -S_\theta C_\phi \\ S_\theta S_\psi & S_\theta C_\psi & C_\theta \end{bmatrix}, \qquad (4.16.11)$$

The Probability Ellipsoid

CCAR
Colorado Center for
Astrodynamics Research

- The Euler angles, ϕ, θ and ψ , are defined by



The Probability Ellipsoid

- and are computed from

$$\phi = \text{atan2}\left[\frac{U_{13}}{-U_{23}}\right], \quad 0 \leq \phi \leq 2\pi \tag{4.16.12}$$

$$\theta = \text{acos}\left[U_{33}\right], \quad 0 \leq \theta \leq \pi \tag{4.16.13}$$

$$\psi = \text{atan2}\left[\frac{U_{31}}{U_{32}}\right], \quad 0 \leq \psi \leq 2\pi. \tag{4.16.14}$$

The Probability Ellipsoid

Example:

Consider a simple case where the Eigenvectors

$$\mathbf{U} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$U^T P \mathbf{U} = \begin{bmatrix} 10 & 0 & 0 \\ 0 & 100 & 0 \\ 0 & 0 & 30 \end{bmatrix} = D\left(\lambda_1, \lambda_2, \lambda_3\right)$$

- Plot this error ellipsoid using Matlab

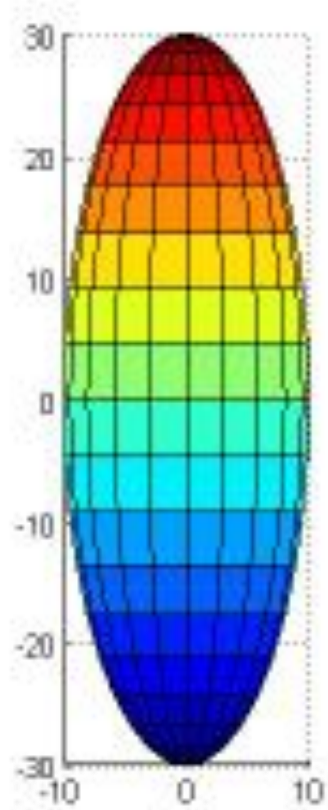
$$\text{Semi} = \left(\text{sqrt}\,(\lambda), \ \text{sqrt}\,(\lambda_2), \ \text{sqrt}\,(\lambda_3)\right)$$

$$\text{PlotEllipsoid}\ \left(\mathbf{U}, \ \text{Semi}\right)$$

The Probability Ellipsoid

CCAR
Colorado Center for
Astrodynamics Research

Views of Error Ellipsoid



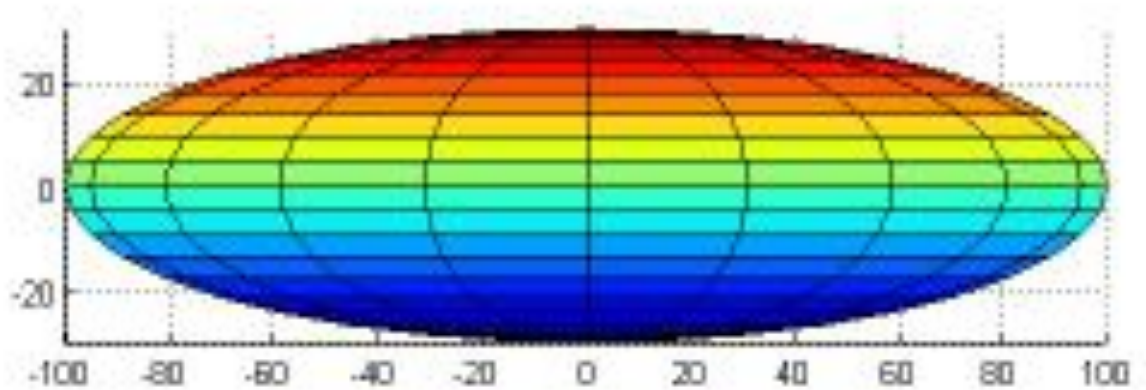
view (0,0)*
azimuth =0, elevation =0
view down the negative *y*-axis

*view(azimuth, elevation), azimuth is a clockwise rotation about the positive *z*-axis

The Probability Ellipsoid

CCAR
Colorado Center for
Astrodynamics Research

Views of Error Ellipsoid



view (90°,0)
view down the positive x-axis

The Probability Ellipsoid

CCAR
Colorado Center for Astrodynamics Research

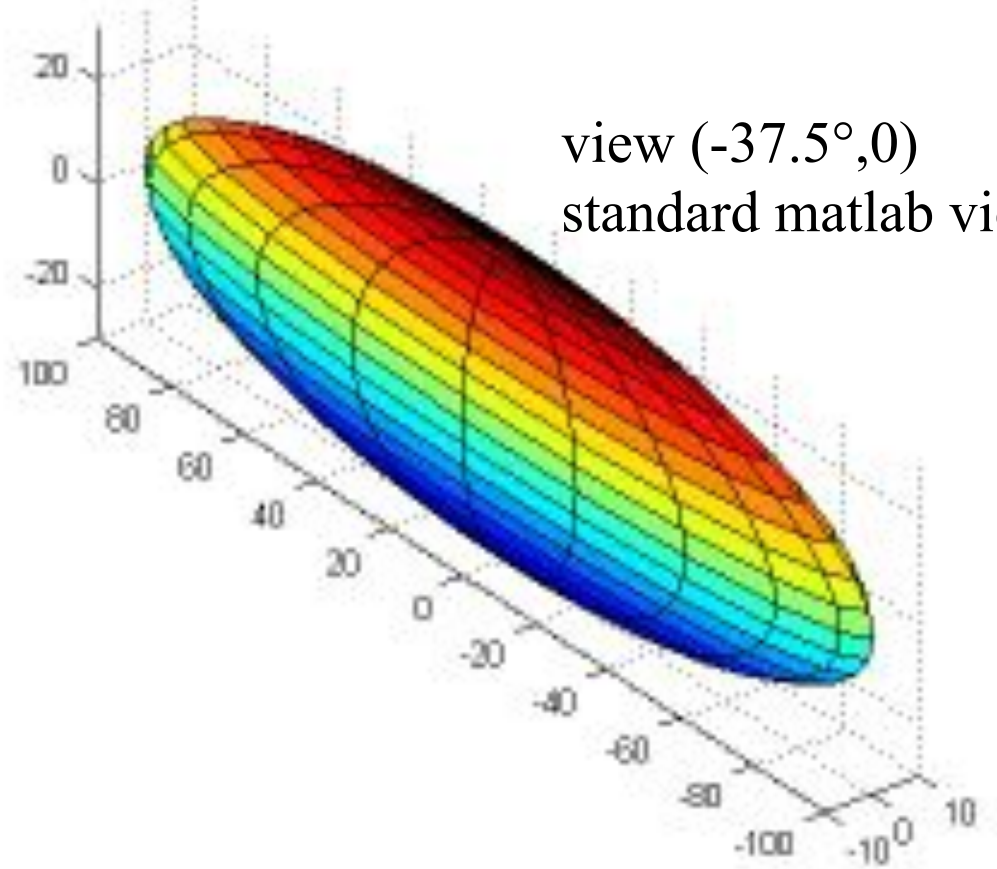
Views of Error Ellipsoid

view (0,90°)
view down the positive *z*-axis

The Probability Ellipsoid

CCAR
Colorado Center for
Astrodynamics Research

Views of Error Ellipsoid



view (-37.5°,0)
standard matlab view

Smoothing

- Quick Break
- Then, smoothing

Smoothing

- What is the best way to determine the best estimate of a state given ALL observations (those before and after in time)?
  - Batch
  - CKF
  - CKF w/process noise
  - EKF
  - EKF w/process noise

Smoothing

CCAR
Colorado Center for
Astrodynamics Research

- Background

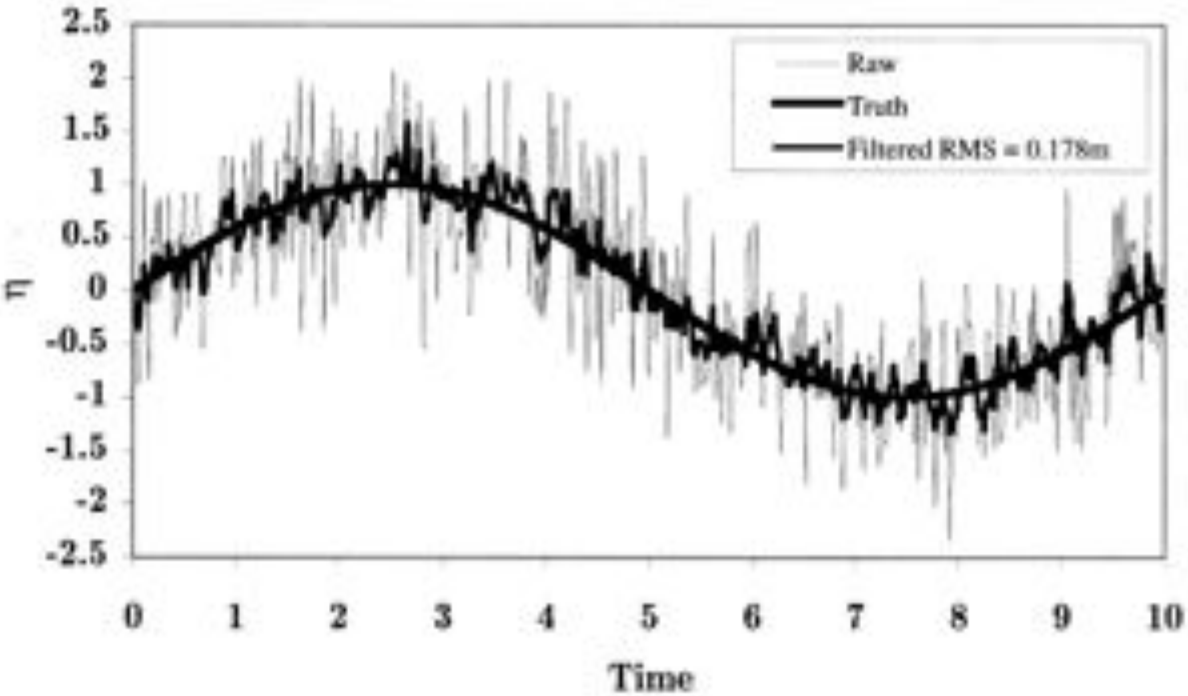


Figure 4.19.1: Process noise/sine wave recovery showing truth, raw data (truth plus noise) and the filtered solution. $\eta_0 = 0$, $\sigma = 2.49$, $\beta = .045$.

Smoothing

- We'd prefer:

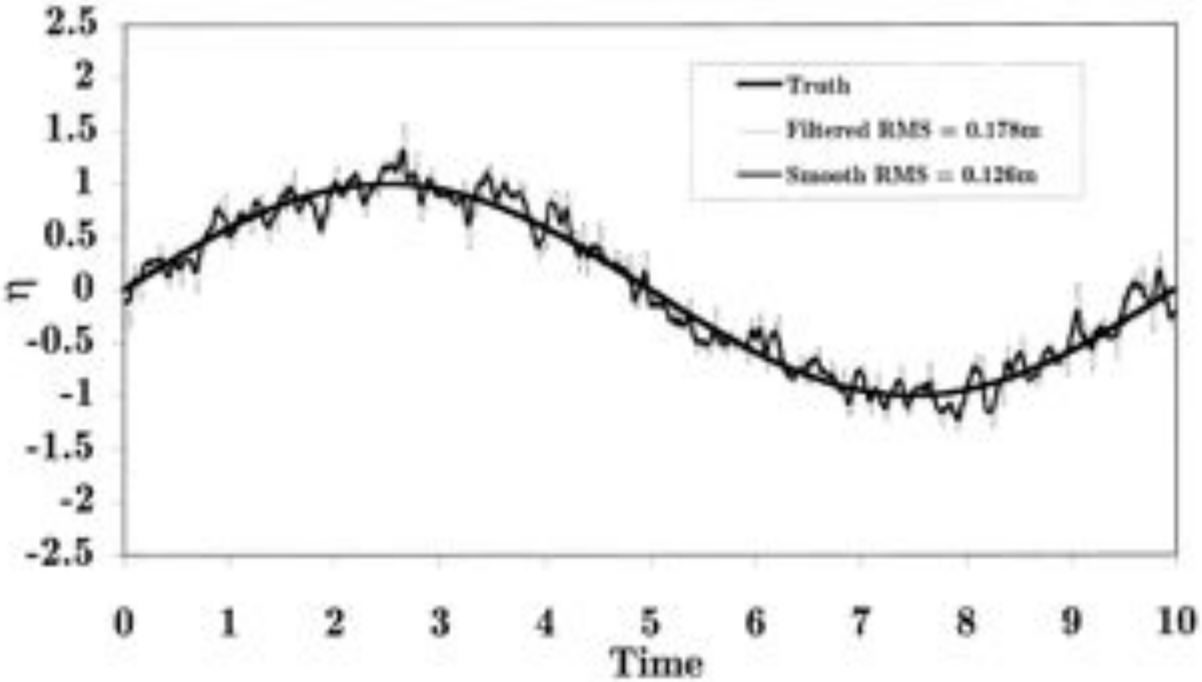


Figure 4.19.2: Process noise/sine wave recovery showing the truth, the filtered, and the smoothed solution. $\bar{\eta}_0 = 0$, $\sigma = 2.49$, $\beta = .045$.

Smoothing

4.15 SMOOTHING

It is often desirable to perform a smoothing operation when using a sequential filter. In this case, we are searching for the best estimate of the state at some time t_k based on all observations through time t_ℓ where $\ell > k$. For the case where there is no random component to the dynamical equation of state—for example, the no-process noise case—the batch estimation algorithm along with the prediction equation, Eqs. (4.4.19) and (4.4.22), will give the smoothed solution. However, as noted, the batch estimation approach has difficulty including the effects of process noise. The smoothing algorithms have been developed to overcome this difficulty. Following Jazwinski (1970), the smoothing algorithm can be derived using a Bayesian approach of maximizing the density function of the state conditioned on knowledge of the observations through time, t_ℓ. Our system is described in Section 4.9 (see Eq. (4.9.46)).

$$\mathbf{x}_{k+1} = \Phi(t_{k+1}, t_k)\mathbf{x}_k + \Gamma(t_{k+1}, t_k)\mathbf{u}_k$$
$$\mathbf{y}_k = \widetilde{H}_k\mathbf{x}_k + \epsilon_k.$$

Smoothing

CCAR
Colorado Center for Astrodynamics Research

We will use the notation $\hat{\mathbf{x}}_k^\ell$ to indicate the best estimate of $\mathbf{x}$ at t_k based on observations through t_ℓ, where in general $\ell > k$. Following the Maximum Likelihood philosophy, we wish to find a recursive expression for $\hat{\mathbf{x}}_k^\ell$ in terms of $\hat{\mathbf{x}}_{k+1}^\ell$, which maximizes the conditional density function

$$p(\mathbf{x}_k, \mathbf{x}_{k+1}/\mathbf{Y}_\ell), \quad \text{where} \quad \mathbf{Y}_\ell = \mathbf{y}_1, \mathbf{y}_2 \cdots \mathbf{y}_k \cdots \mathbf{y}_\ell. \tag{4.15.1}$$

After some algebra

$$p\left(\mathbf{x}_k, \mathbf{x}_{k+1} / \mathbf{Y}_\ell\right) = \frac{p(\mathbf{Y}_k)}{p(\mathbf{Y}_\ell)} p\left(\mathbf{y}_{k+1} \ldots \mathbf{y}_\ell / \mathbf{x}_{k+1}\right) \times p\left(\mathbf{x}_{k+1} / \mathbf{x}_k\right) p\left(\mathbf{x}_k / \mathbf{Y}_k\right). \tag{4.15.5}$$

Smoothing

- Finally

$$\hat{\mathbf{x}}_k^\ell = \hat{\mathbf{x}}_k^k + S_k\left(\hat{\mathbf{x}}_{k+1}^\ell - \Phi(t_{k+1}, t_k)\hat{\mathbf{x}}_k^k\right) \tag{4.15.9}$$

where

$$\begin{aligned} S_k &= P_k^k \Phi^T(t_{k+1}, t_k)[\Phi(t_{k+1}, t_k) P_k^k \Phi^T(t_{k+1}, t_k) \\ &\quad + \Gamma(t_{k+1}, t_k) Q_k \Gamma^T(t_{k+1}, t_k)]^{-1} \\ &= P_k^k \Phi^T(t_{k+1}, t_k)(P_{k+1}^k)^{-1}. \end{aligned} \tag{4.15.10}$$

Eq. (4.15.9) is the smoothing algorithm. Computation goes backward in index k, with $\hat{\mathbf{x}}_\ell^\ell$, the filter solution, as initial conditions. Note that the filter solutions for $\hat{\mathbf{x}}_k^k$, P_k^k, $\Phi(t_{k+1}, t_k)$, and $\Gamma(t_{k+1}, t_k)$ are required and should be stored in the filtering process. The time update of the covariance matrix, P_{k+1}^k, may be stored or recomputed.

Smoothing

The equation for propagating the smoothed covariance is derived next (Jazwinski, 1970; Rausch *et al.*, 1965) . It can easily be shown from Eq. (4.15.9) that $\hat{\mathbf{x}}_k^\ell$ is unbiased; hence, the smoothed covariance is defined by

$$P_k^\ell = E\left[(\hat{\mathbf{x}}_k^\ell - \mathbf{x}_k)(\hat{\mathbf{x}}_k^\ell - \mathbf{x}_k)^T\right]. \tag{4.15.11}$$

The equation for the smoothed covariance is given by

$$P_k^\ell = P_k^k + S_k\,(P_{k+1}^\ell - P_{k+1}^k)S_k^T. \tag{4.15.24}$$

Smoothing Computational Algorithm

CCAR
Colorado Center for
Astrodynamics Research

Given (from the filtering algorithm)

$$\hat{\mathbf{x}}_{\ell}^{\ell},\ \hat{\mathbf{x}}_{\ell-1}^{\ell-1},\ P_{\ell}^{\ell-1},\ P_{\ell-1}^{\ell-1},\ \Phi(t_{\ell}, t_{\ell-1});$$

set $k = \ell - 1$

$$S_{\ell-1} = P_{\ell-1}^{\ell-1}\, \Phi^T(t_{\ell}, t_{\ell-1})\, (P_{\ell}^{\ell-1})^{-1} \tag{4.15.29}$$

$$\hat{\mathbf{x}}_{\ell-1}^{\ell} = \hat{\mathbf{x}}_{\ell-1}^{\ell-1} + S_{\ell-1}(\hat{\mathbf{x}}_{\ell}^{\ell} - \Phi(t_{\ell}, t_{\ell-1})\, \hat{\mathbf{x}}_{\ell-1}^{\ell-1}).$$

Smoothing Computational Algorithm

Given (from the filtering algorithm and the previous step of the smoothing algorithm)

$$\hat{\mathbf{x}}_{\ell-2}^{\ell-2},\;\; P_{\ell-1}^{\ell-2},\;\; P_{\ell-2}^{\ell-2},\;\; \hat{\mathbf{x}}_{\ell-1}^{\ell},\;\; \Phi(t_{\ell-1}, t_{\ell-2});$$

set $k = \ell - 2$, and compute

$$S_{\ell-2} = P_{\ell-2}^{\ell-2}\,\Phi^T\,(t_{\ell-1}, t_{\ell-2})\,(P_{\ell-1}^{\ell-2})^{-1} \tag{4.15.30}$$

$$\hat{\mathbf{x}}_{\ell-2}^{\ell} = \hat{\mathbf{x}}_{\ell-2}^{\ell-2} + S_{\ell-2}\,(\hat{\mathbf{x}}_{\ell-1}^{\ell} - \Phi\,(t_{\ell-1}, t_{\ell-2})\,\hat{\mathbf{x}}_{\ell-2}^{\ell-2})$$

$$\vdots$$

and so on.

Smoothing

- If we suppose that there is no process noise (Q=0), then the smoothing algorithm reduces to the CKF mapping relationships:

$$S_k = \Phi^{-1}(t_{k+1}, t_k)$$

$$\hat{\mathbf{x}}_k^l = \Phi(t_k, t_l)\hat{\mathbf{x}}_l^l$$

$$P_k^l = \Phi(t_k, t_l) P_l^l \Phi^T(t_k, t_l)$$

(41) Generate 1000 equally spaced observations of one cycle of a sine wave with amplitude 1 and period 10. Add Gaussian random noise with zero mean and variance $= 0.25$. Set up a sequential estimation procedure to estimate the amplitude of the sine wave as a function of time using the noisy raw data. Model the sine wave as a Gauss-Markov process as given by Eq. (4.9.60),

$$\eta_{i+1} = m_{i+1}\eta_i + \Gamma_{i+1}u_i$$

where

$$u_i = N(0, 1)$$

$$m_{i+1} = e^{-\beta(t_{i+1}-t_i)}$$

$$\Gamma_{i+1} = \sqrt{\frac{\sigma^2}{2\beta}(1 - m_{i+1}^2)}$$

$$\beta = \frac{1}{\tau}$$

and τ is the time constant. The sequential algorithm is given by

1. $\bar{\eta}_i = \Phi(t_i, t_{i-1})\hat{\eta}_{i-1} \quad (i = 1, 2 \ldots 1000)$

   $\Phi(t_i, t_{i-1}) = m_i = e^{-\beta(t_i - t_{i-1})}$

   $\overline{P}_i = \Phi(t_i, t_{i-1})P_{i-1}\Phi^T(t_i, t_{i-1}) + \Gamma_i Q_{i-1}\Gamma_i^T$

   Note that P, Φ, Q, and Γ are scalars

   $Y_i = \eta_i$, thus $\widetilde{H}_i = 1$, assume $R_i = 1$, $Q_i = 1$, $\bar{\eta}_0 = 0$, $\overline{P}_0 = 1$

   $$K_i = \overline{P}_i \widetilde{H}_i^T (\widetilde{H}_i \overline{P}_i \widetilde{H}_i^T + R_i)^{-1} = \frac{\overline{P}_i}{\overline{P}_i + 1}$$

   $\hat{\eta}_i = \bar{\eta}_i + K_i(Y_i - \widetilde{H}_i\bar{\eta}_i) = \bar{\eta}_i + K_i(Y_i - \bar{\eta}_i)$, ($Y_i$ is the observation data)

   $P_i = (I - K_i\widetilde{H}_i)\overline{P}_i = K_i$

   Next i

Plot your observations, the truth data, and $\hat{\eta}$ versus time. You will need to guess initial values for σ and β.

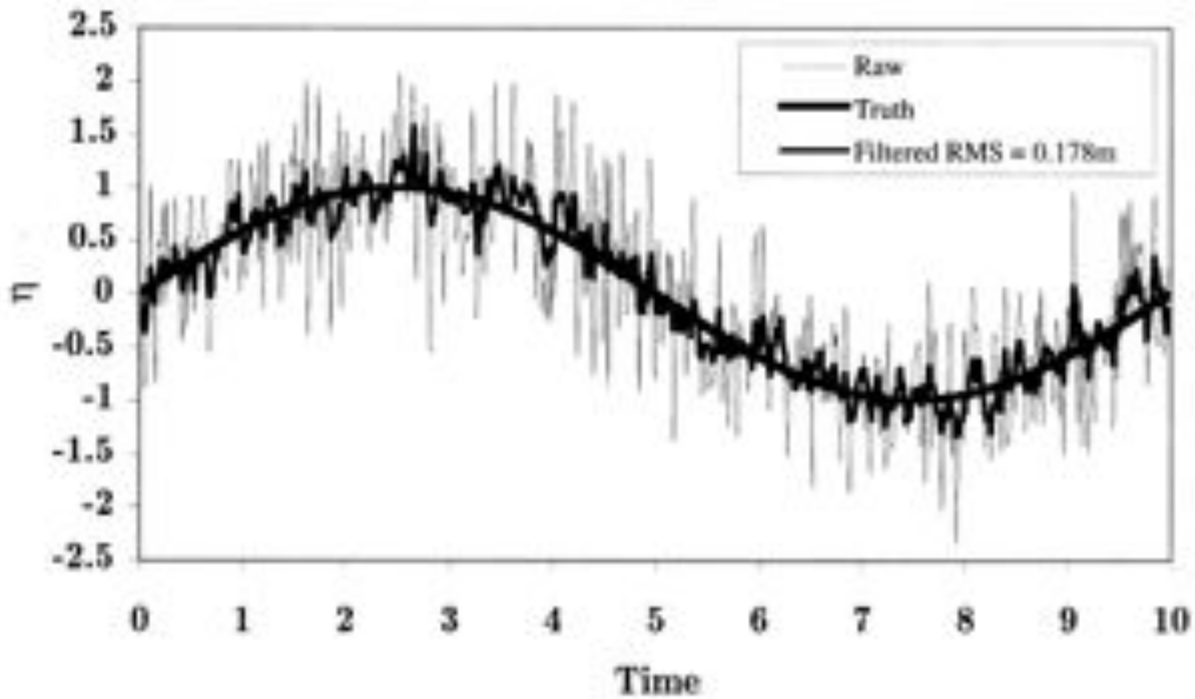


Figure 4.19.1: Process noise/sine wave recovery showing truth, raw data (truth plus noise) and the filtered solution. $\eta_0 = 0$, $\sigma = 2.49$, $\beta = .045$.

where

$$\mathrm{RMS} = \left\{ \sum_{i=1}^{N} \frac{(T_i - \hat{\eta}_i)^2}{N} \right\}^{1/2}$$

Given (from the filtering algorithm)

$$\hat{\mathbf{x}}_\ell^\ell, \;\; \hat{\mathbf{x}}_{\ell-1}^{\ell-1}, \;\; P_\ell^{\ell-1}, \;\; P_{\ell-1}^{\ell-1}, \;\; \Phi\left(t_\ell, t_{\ell-1}\right);$$

set $k = \ell - 1$

$$S_{\ell-1} = P_{\ell-1}^{\ell-1}\, \Phi^T\left(t_\ell, t_{\ell-1}\right) \left(P_\ell^{\ell-1}\right)^{-1} \tag{4.15.29}$$

$$\hat{\mathbf{x}}_{\ell-1}^\ell = \hat{\mathbf{x}}_{\ell-1}^{\ell-1} + S_{\ell-1}\left(\hat{\mathbf{x}}_\ell^\ell - \Phi\left(t_\ell, t_{\ell-1}\right) \hat{\mathbf{x}}_{\ell-1}^{\ell-1}\right).$$

Given (from the filtering algorithm and the previous step of the smoothing algorithm)

$$\hat{\mathbf{x}}_{\ell-2}^{\ell-2}, \;\; P_{\ell-1}^{\ell-2}, \;\; P_{\ell-2}^{\ell-2}, \;\; \hat{\mathbf{x}}_{\ell-1}^{\ell}, \;\; \Phi(t_{\ell-1}, t_{\ell-2});$$

set $k = \ell - 2$, and compute

$$\begin{aligned} S_{\ell-2} &= P_{\ell-2}^{\ell-2}\, \Phi^T\, (t_{\ell-1}, t_{\ell-2})\, (P_{\ell-1}^{\ell-2})^{-1} \\ \hat{\mathbf{x}}_{\ell-2}^{\ell} &= \hat{\mathbf{x}}_{\ell-2}^{\ell-2} + S_{\ell-2}\, (\hat{\mathbf{x}}_{\ell-1}^{\ell} - \Phi\, (t_{\ell-1}, t_{\ell-2})\, \hat{\mathbf{x}}_{\ell-2}^{\ell-2}) \end{aligned} \tag{4.15.30}$$

$$\vdots$$

and so on.

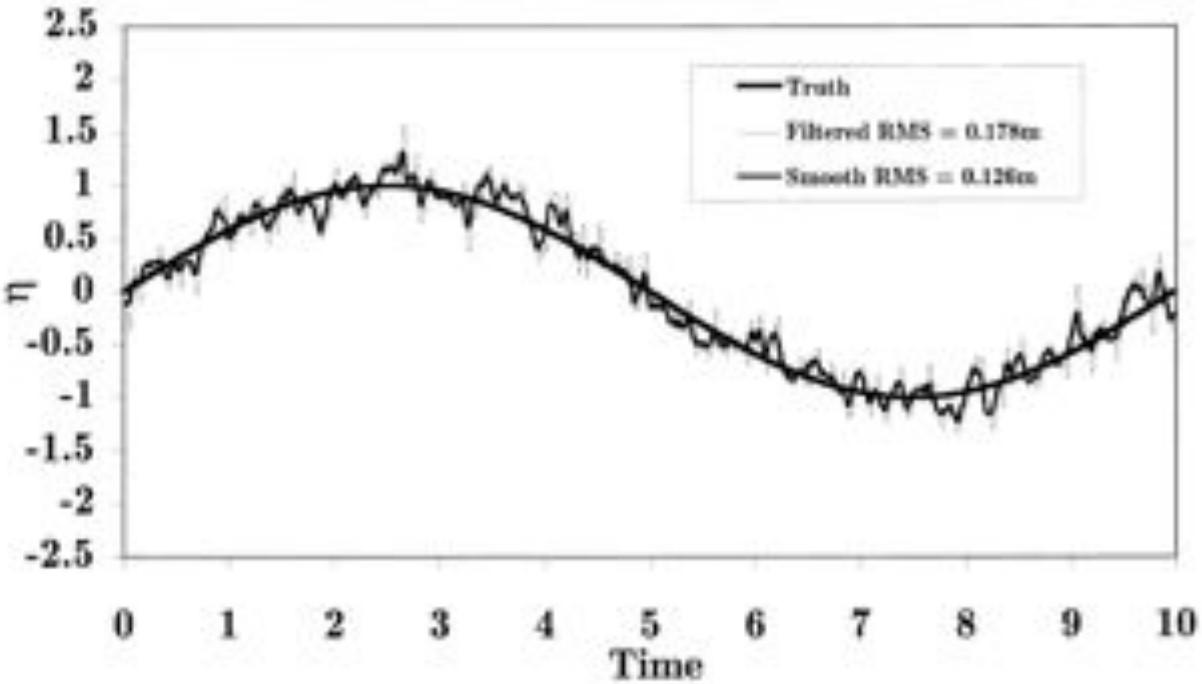


Figure 4.19.2: Process noise/sine wave recovery showing the truth, the filtered, and the smoothed solution. $\bar{\eta}_0 = 0$, $\sigma = 2.49$, $\beta = .045$.

The End

- HW 11 due this week (solutions soon).