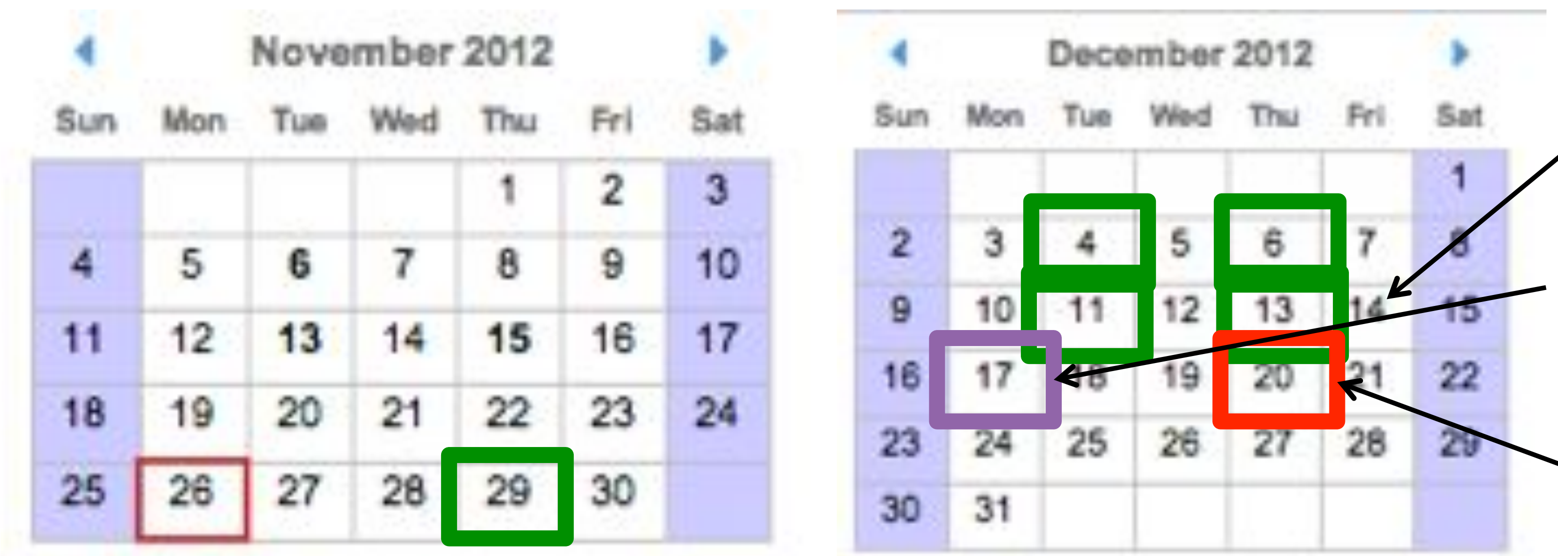
ASEN 5070
Statistical Orbit Determination I
Fall 2012

Professor Jeffrey S. Parker
Professor George H. Born

Lecture 24: Process Noise

Announcements

- HW 10 due Today (solutions).
- HW 11 due next week.



Last Day of Classes

Take-Home Exam Due

Final Project Due
All HW Due

Quiz 20 Review

Question 1 (1 point)

Let's say "T" is a matrix that corresponds to a rotation. Is T an orthogonal matrix?

- ○ Yes
- ○ No

Quiz 20 Review

Question 1 (1 point)

Let's say "T" is a matrix that corresponds to a rotation. Is T an orthogonal matrix?

- Yes
- No

Quiz 20 Review

CCAR
Colorado Center for
Astrodynamics Research

Question 2 (1 point)

Let's say that you are a satellite navigator and you know that you're not modeling everything perfectly. So you include process noise in your filter (atta boy/girl!). Further, let's say that you know downright that you have a very poor model for solar radiation pressure.

You decide to add State Noise Compensation as we discussed in class. Please rank the following strategies in order of how effective they will be in your filter (from least-effective to most-effective). "Effective" means that you end up with a good, accurate estimate of your state with the smallest covariance possible.

The "Q_k" term in the following answers reflects the process noise covariance matrix in the update to the P matrix: Pbar_k = Phi*P_k*Phi^T + Q_k.

- [ ] Don't use a Q_k matrix (Q_k = 0)
- [ ] Derive Q in a coordinate frame tied to the Sun-Earth system such that the largest element of th
- [ ] Use a constant symmetric Q_k matrix - like the one we developed in class.

Quiz 20 Review

CCAR
Colorado Center for
Astrodynamics Research

Question 2 (1 point)

Let's say that you are a satellite navigator and you know that you're not modeling everything perfectly. So you include process noise in your filter (atta boy/girl!). Further, let's say that you know downright that you have a very poor model for solar radiation pressure.

You decide to add State Noise Compensation as we discussed in class. Please rank the following strategies in order of how effective they will be in your filter (from least-effective to most-effective). "Effective" means that you end up with a good, accurate estimate of your state with the smallest covariance possible.

The "Q_k" term in the following answers reflects the process noise covariance matrix in the update to the P matrix: Pbar_k = Phi*P_k*Phi^T + Q_k.

1	Don't use a Q_k matrix (Q_k = 0)
3	Derive Q in a coordinate frame tied to the Sun-Earth system such that the largest element of th
2	Use a constant symmetric Q_k matrix - like the one we developed in class.

Contents

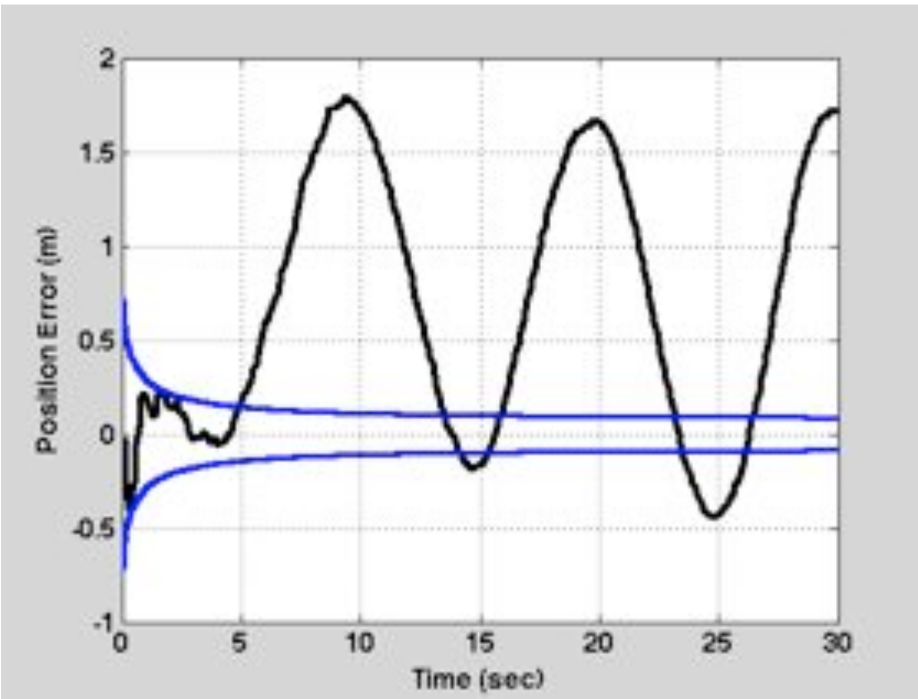
- Process Noise

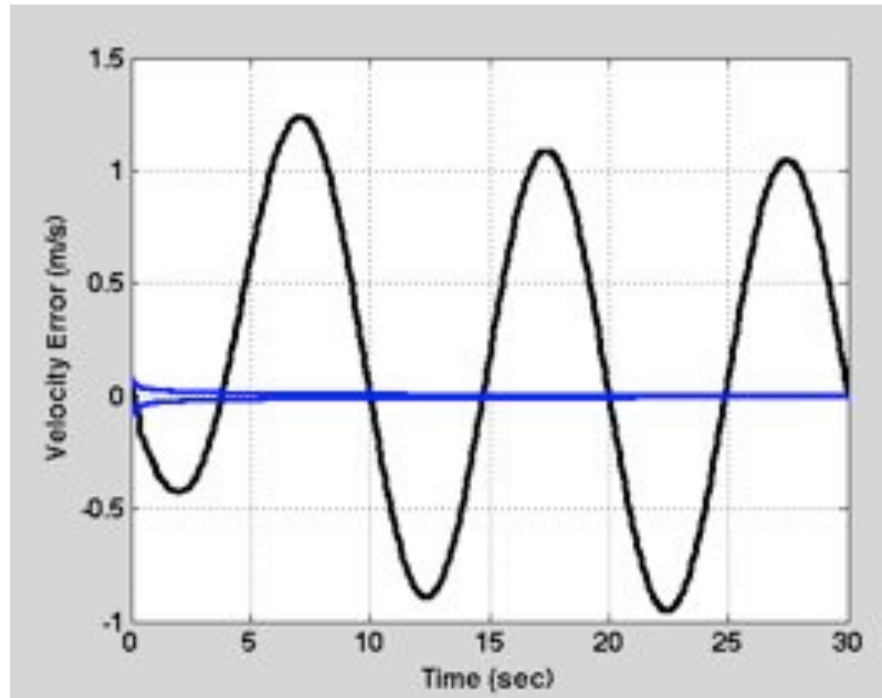
- Last time:
  - Illustrated process noise via an example
    - State Noise Compensation
    - Dynamic Model Compensation
  - Began deriving State Noise Compensation

- Today:
  - Quick review of SNC to the point where we left off
  - Finish SNC
  - Apply to our problem
  - Derive DMC, Gauss-Markov process

Example of SNC

- Filtering in the presence of an unknown acceleration without any process noise compensation

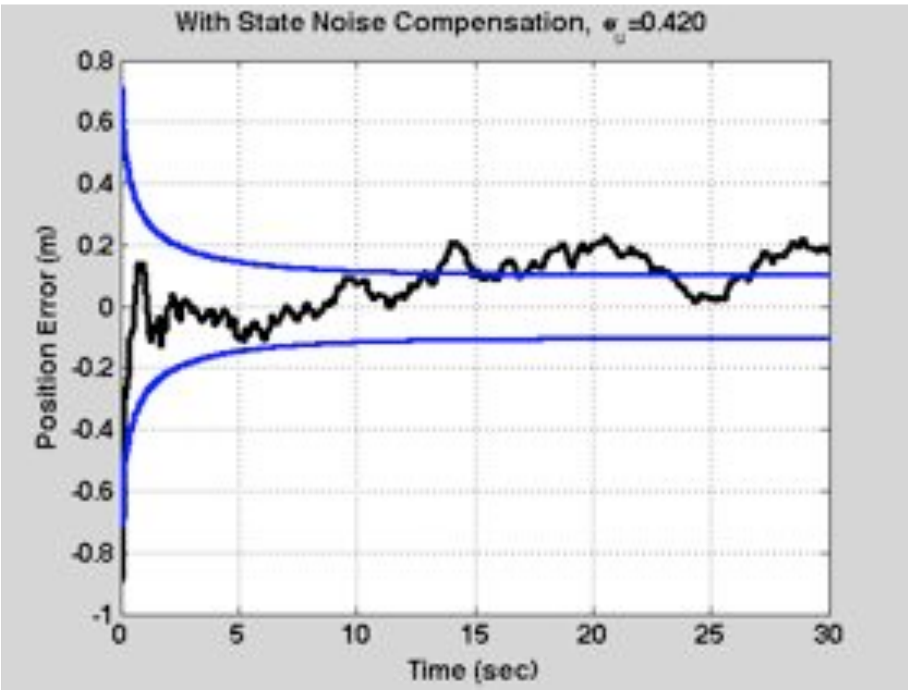


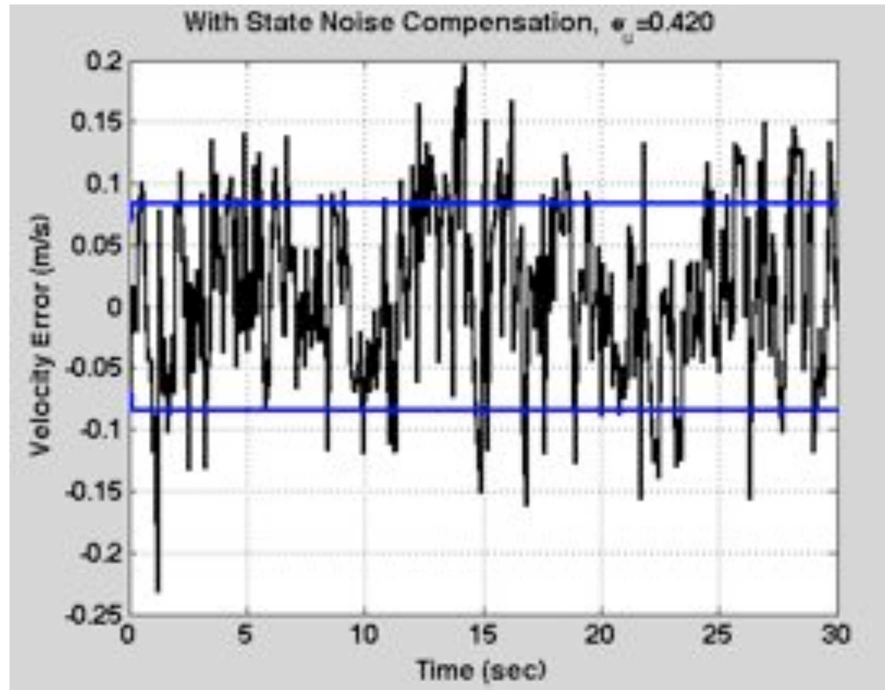


Example of SNC

CCAR
Colorado Center for
Astrodynamics Research

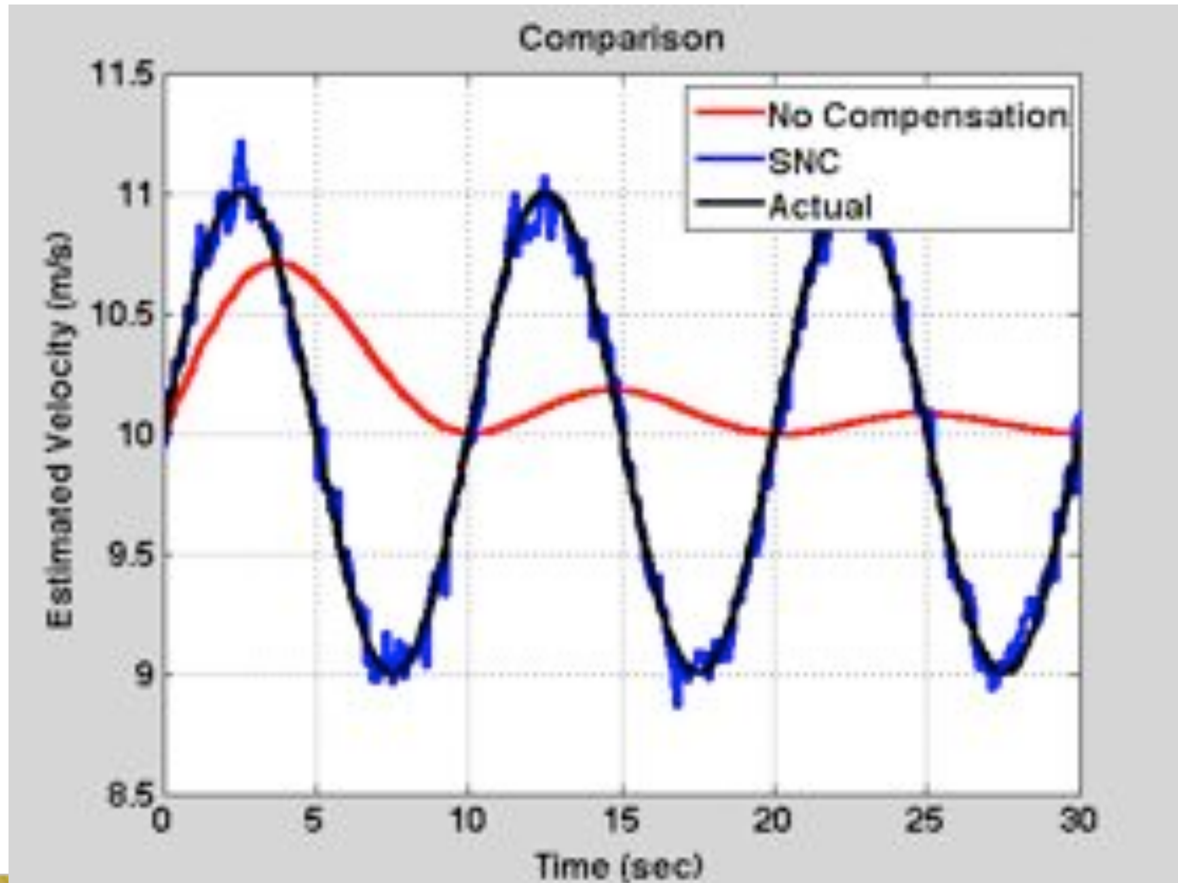
- Result that includes SNC





Example of SNC

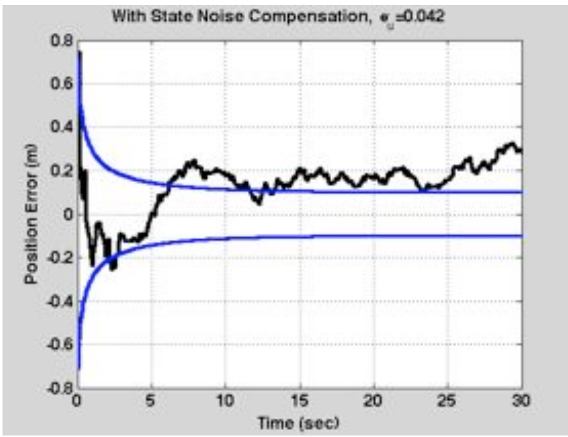
- Comparison of estimated velocity

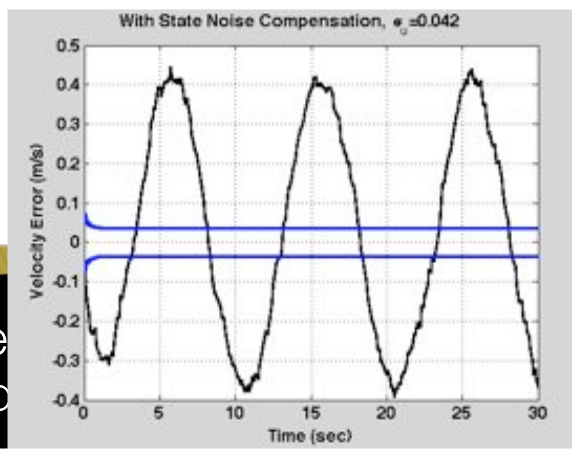


Example of SNC

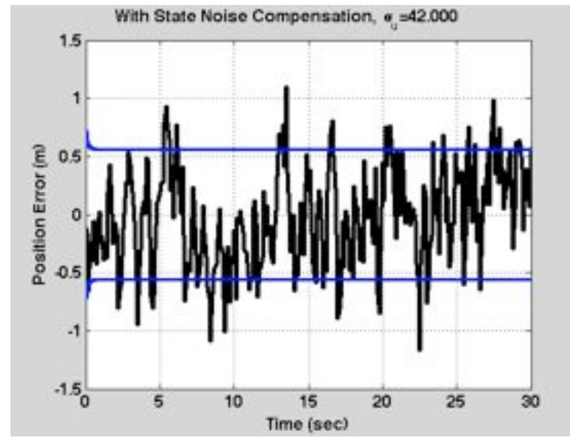
- This does depend on the value of σ_u^2

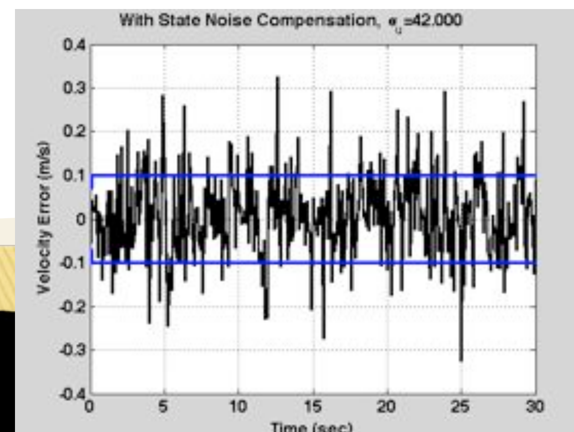
Too low: not enough help





Too high: no carry-over information





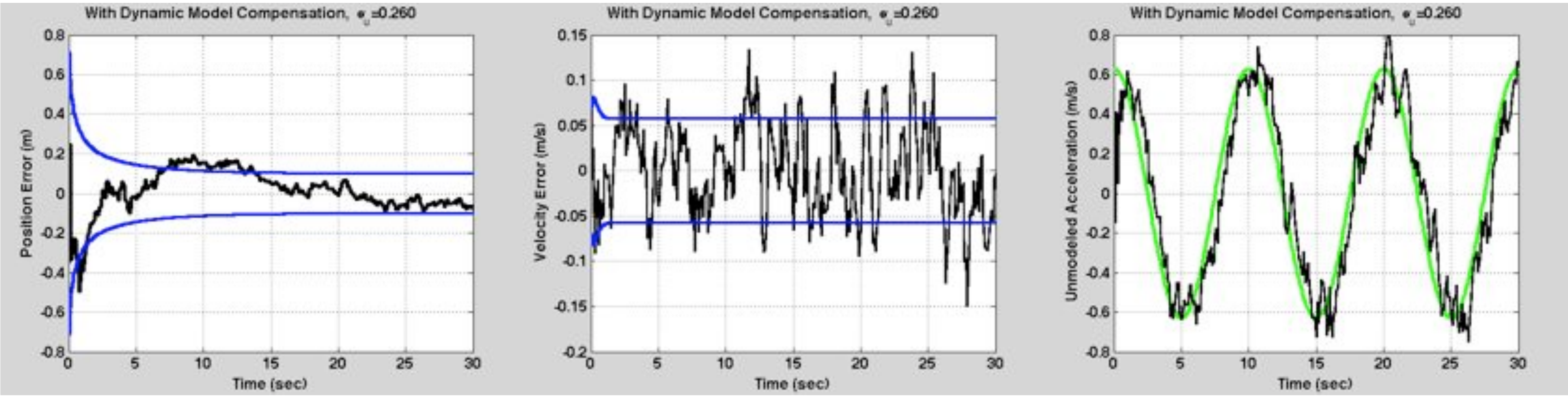
Example DMC

- Estimated State

$$\eta(t) = \eta_0 e^{-\beta(t-t_0)} + \int_{t_0}^{t} e^{-\beta(t-\tau)} u(\tau) d\tau$$

$$\tau = 200 \text{ sec}$$

$$\sigma_u = 0.26 \text{ m/sec}^{5/2}$$



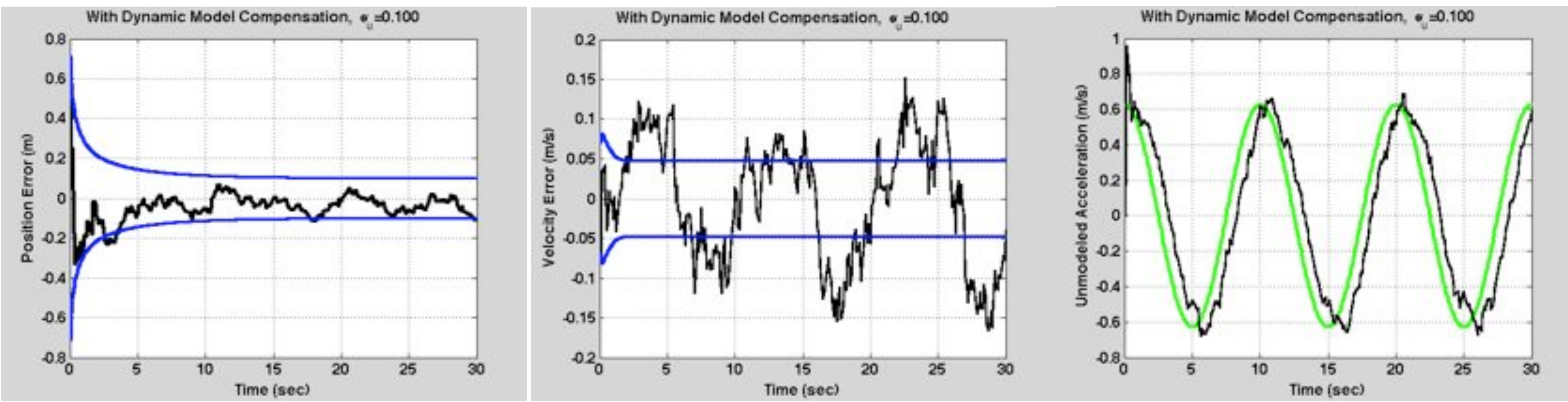
Example DMC

CCAR
Colorado Center for
Astrodynamics Research

- Estimated State

$$\tau = 200 \text{ sec}$$

$$\sigma_u = 0.1 \text{ m/sec}^{5/2}$$



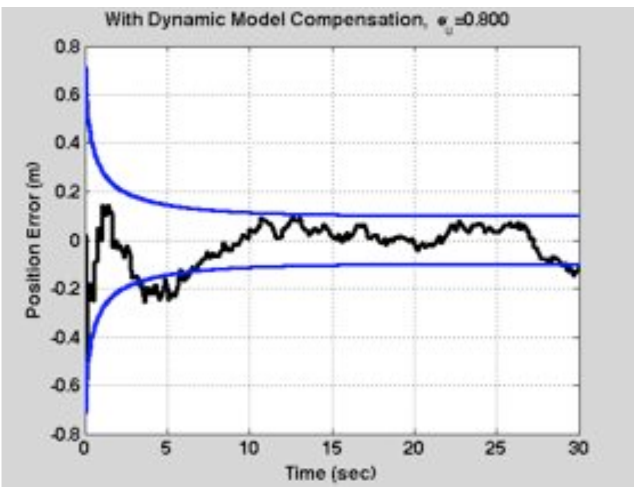
Example DMC

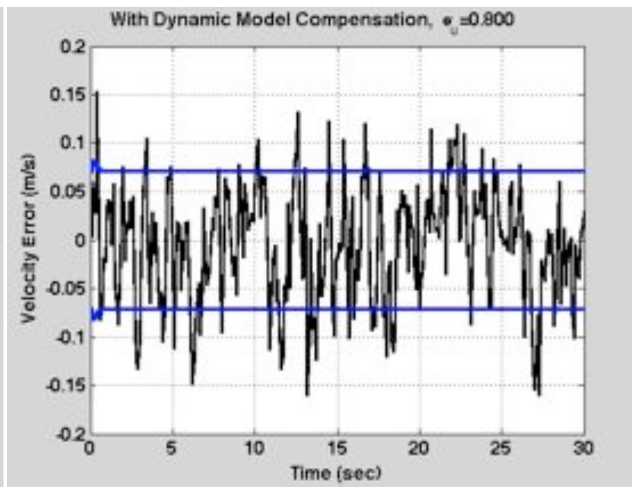
CCAR
Colorado Center for
Astrodynamics Research

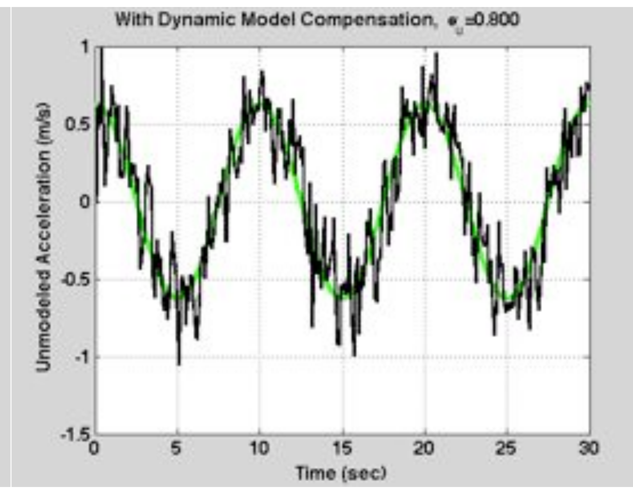
- Estimated State

$$\tau = 200 \text{ sec}$$

$$\sigma_u = 0.8 \text{ m/sec}^{5/2}$$







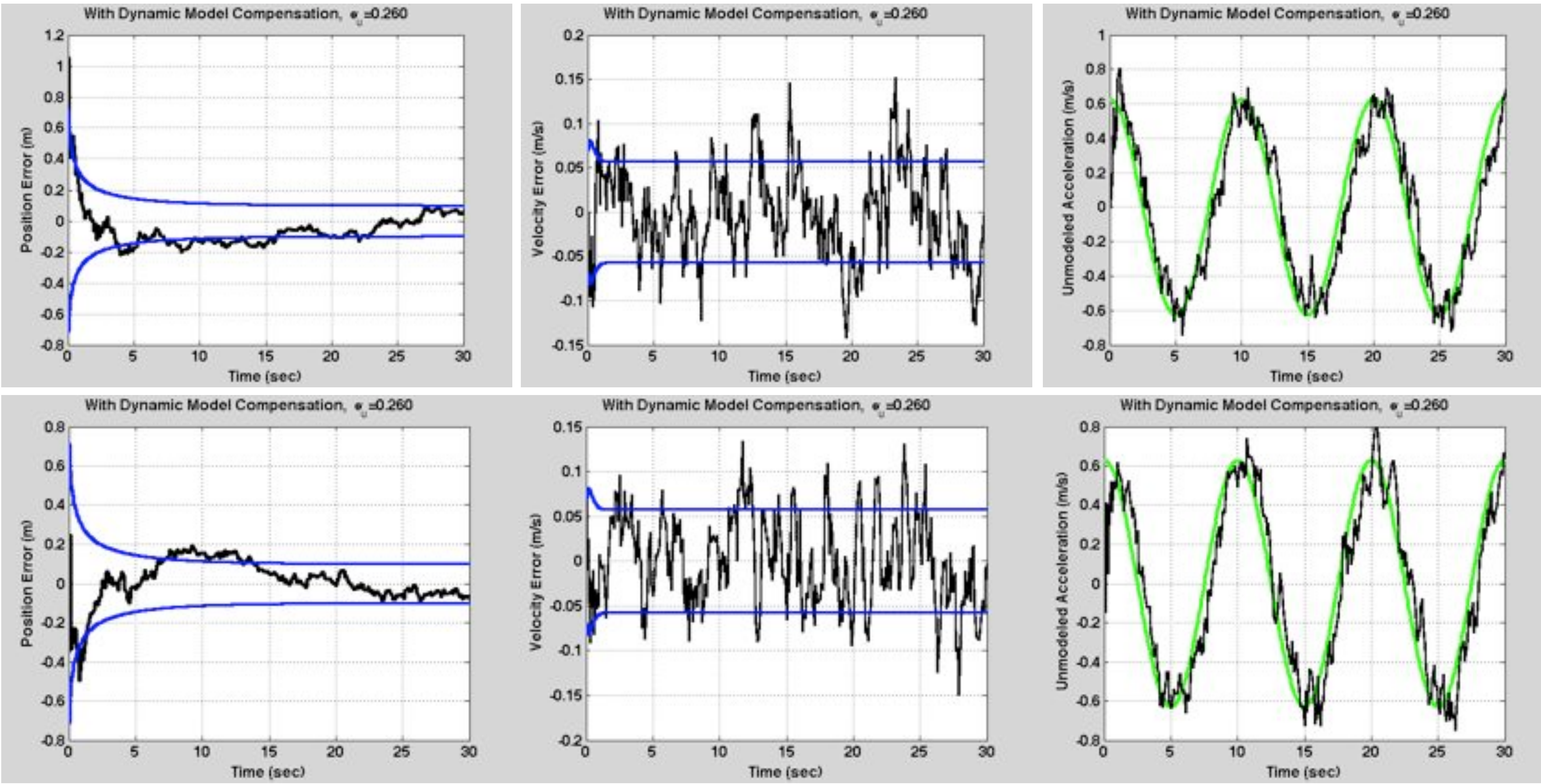
Example DMC

CCAR
Colorado Center for
Astrodynamics Research

- Estimated State

$$\tau = 10 \text{ sec}$$

$$\sigma_u = 0.26 \text{ m/sec}^{5/2}$$



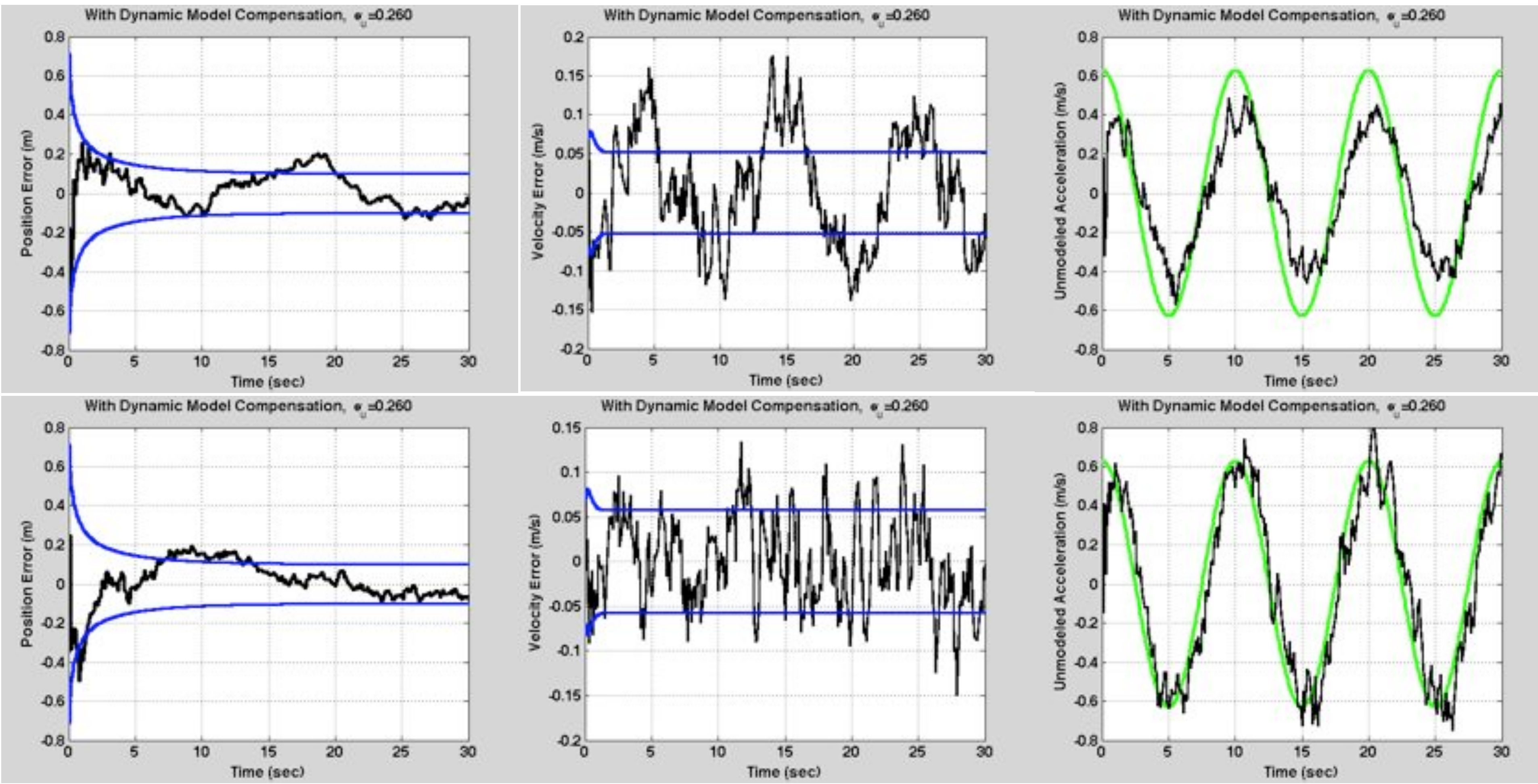
Example DMC

CCAR
Colorado Center for Astrodynamics Research

- Estimated State

$$\tau = 1 \text{ sec}$$

$$\sigma_u = 0.26 \text{ m/sec}^{5/2}$$



Example DMC

CCAR
Colorado Center for Astrodynamics Research

- DMC is less sensitive to tuning than SNC

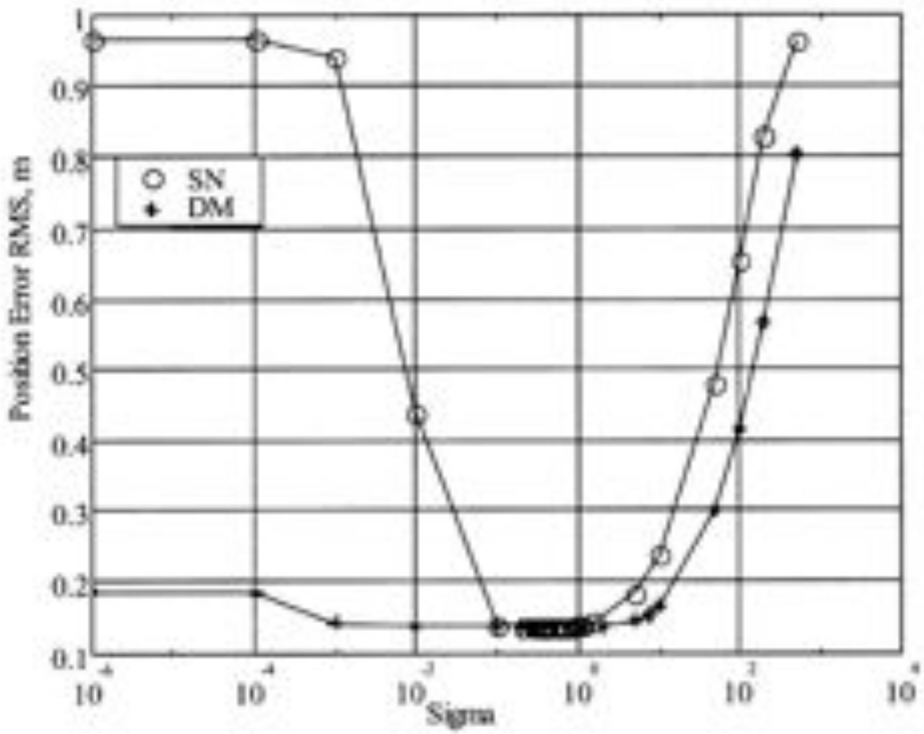


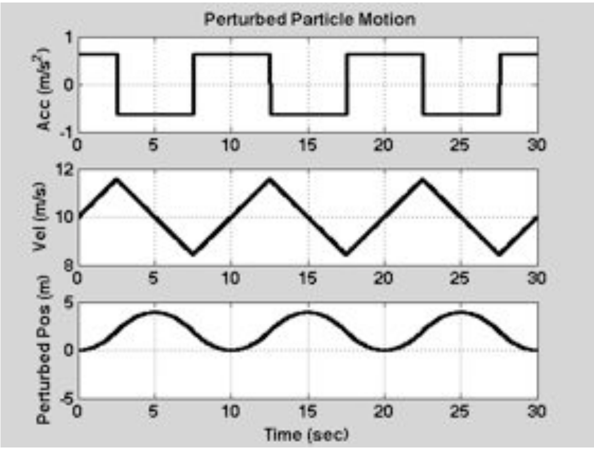
Figure F.3.2: RMS Position error as a function of σ_u for the two-state SNC and three-state DMC filters.

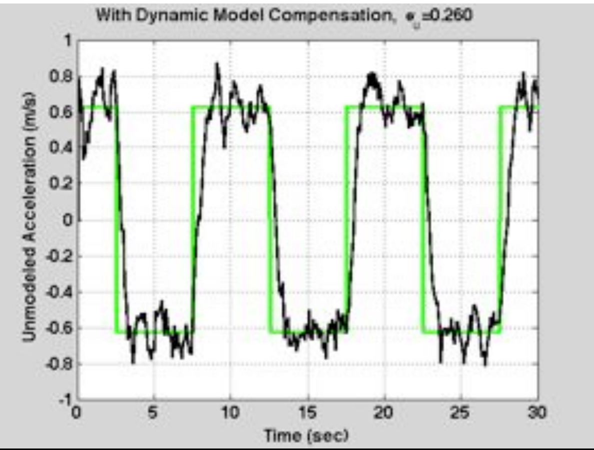
Example SNC / DMC

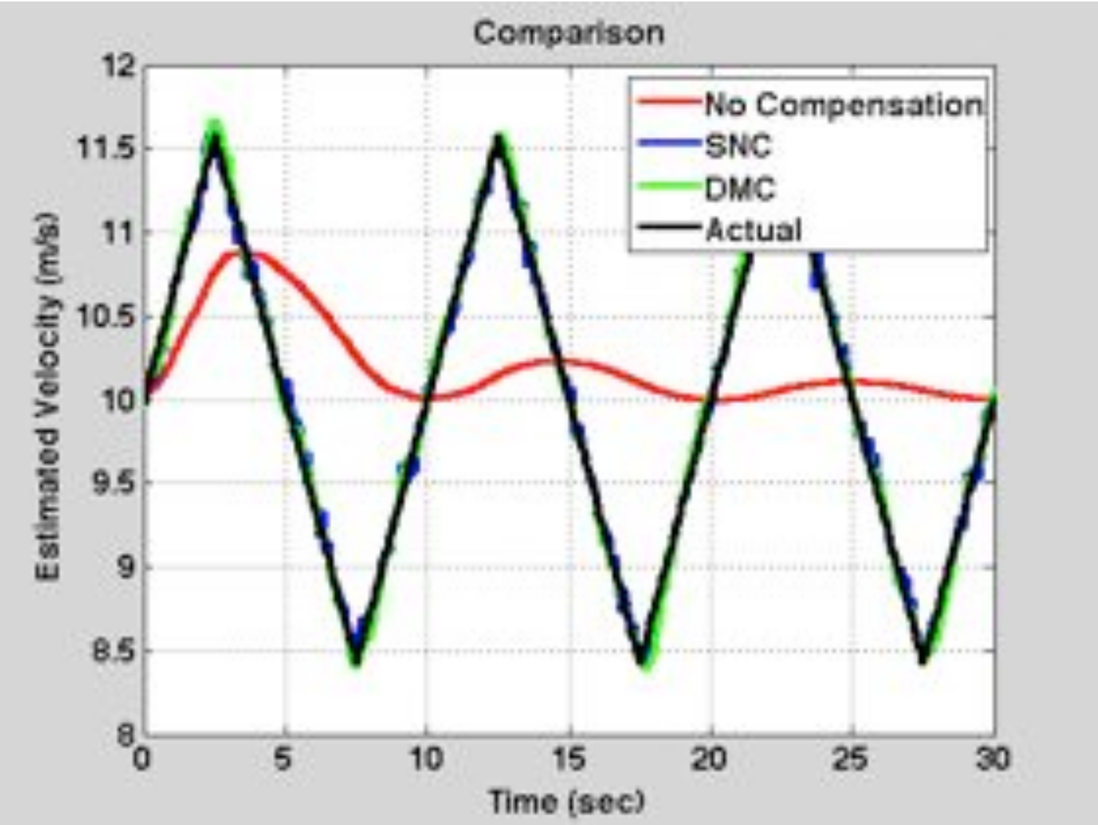
- What if we changed the perturbing acceleration?
  - Triangle wave
  - Square wave
  - Constant acceleration
  - White noise

Example SNC / DMC

- Square Wave

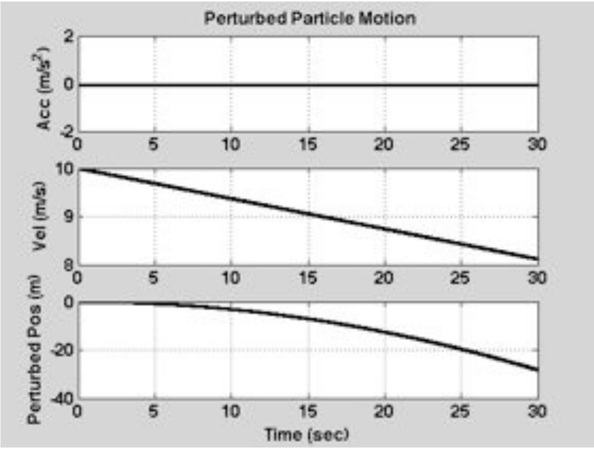


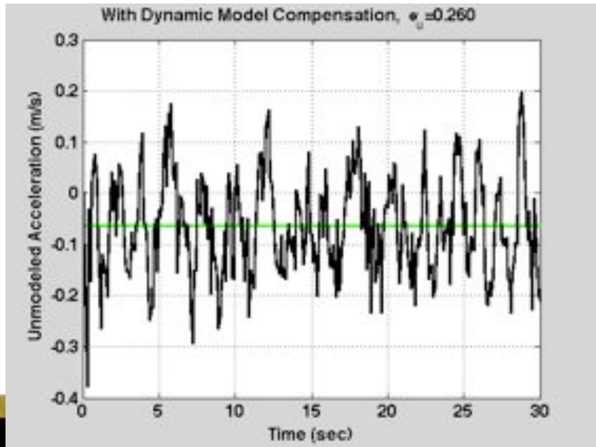


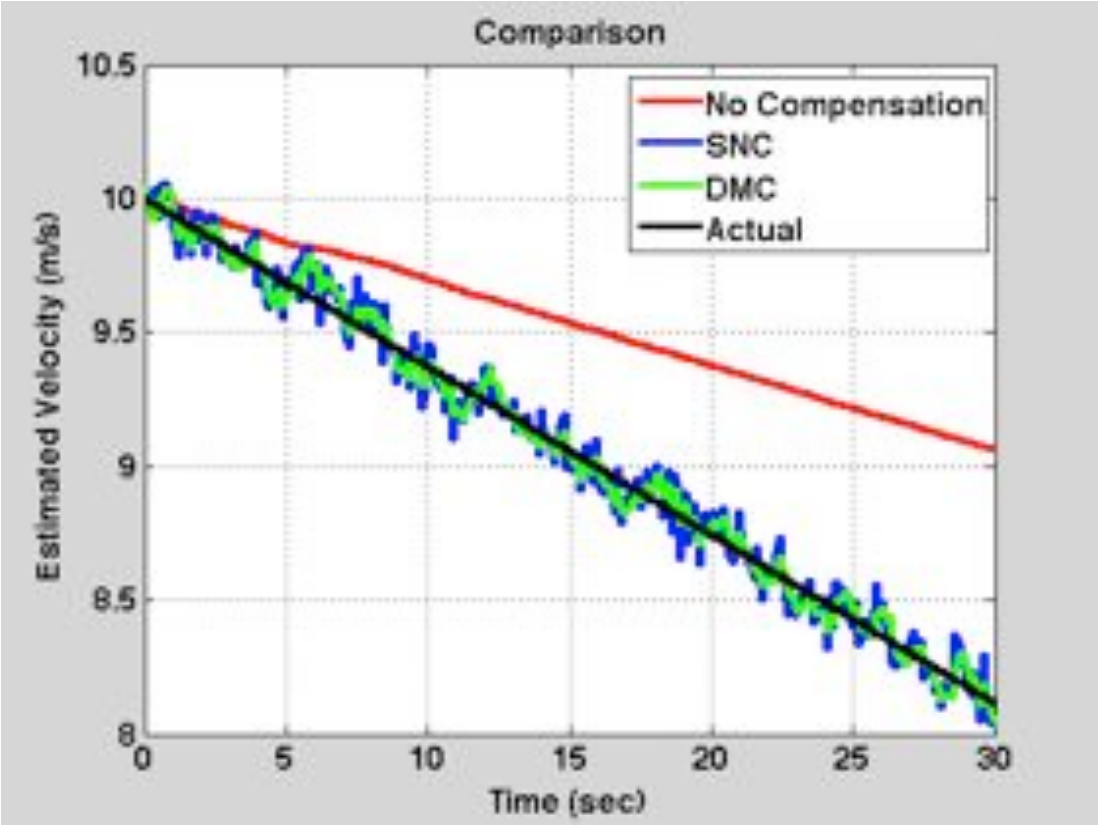


Example SNC / DMC

- Constant Acceleration



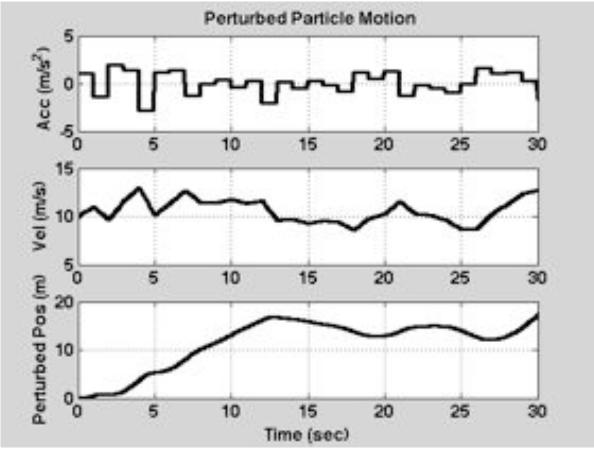


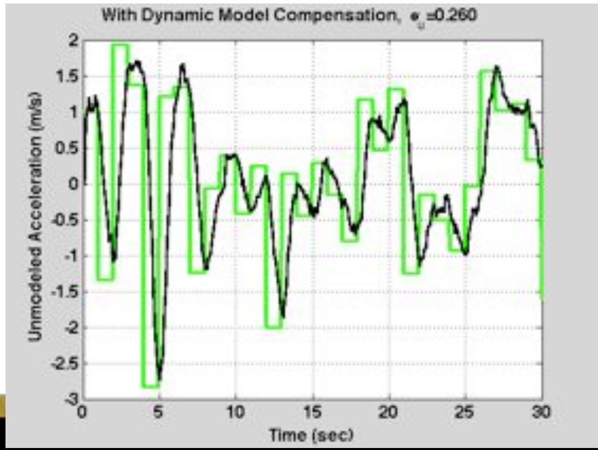


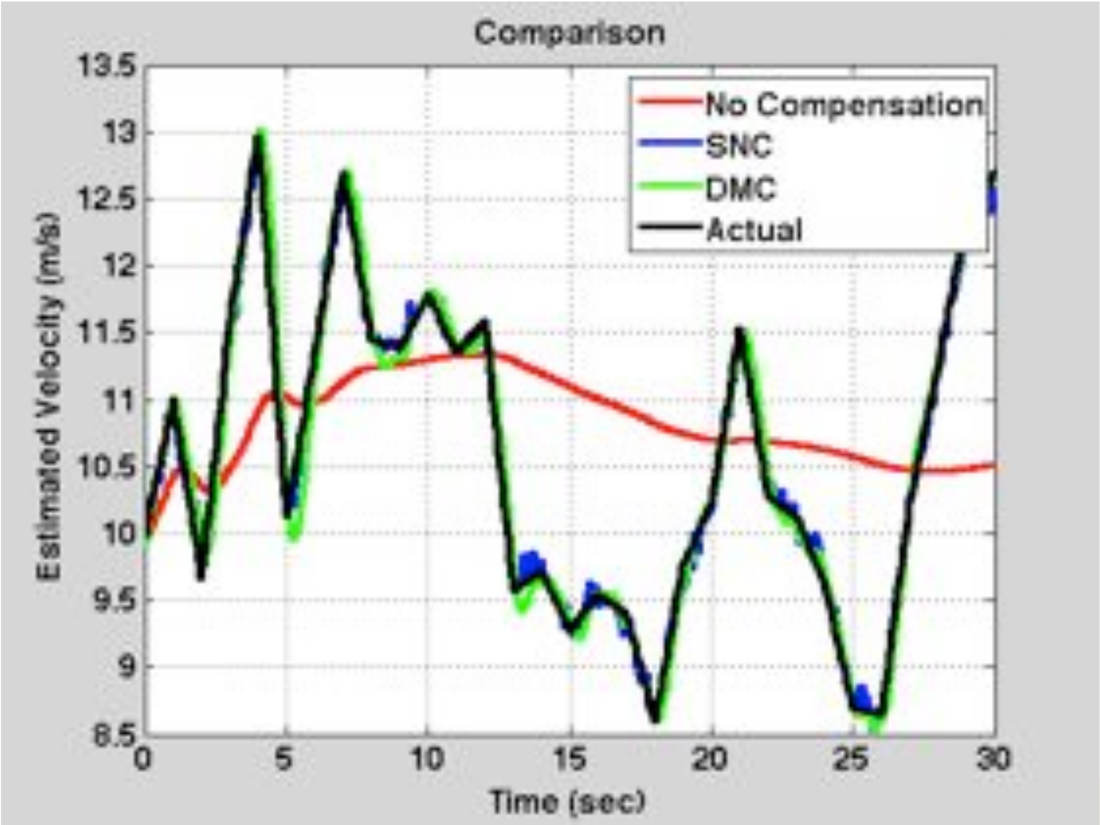
Example SNC / DMC

CCAR
Colorado Center for Astrodynamics Research

- ~White Noise



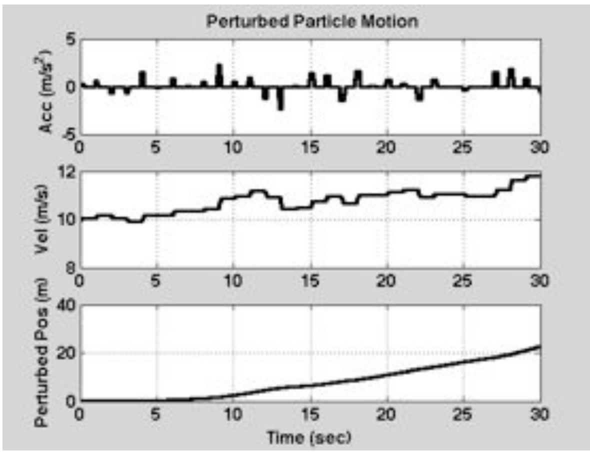


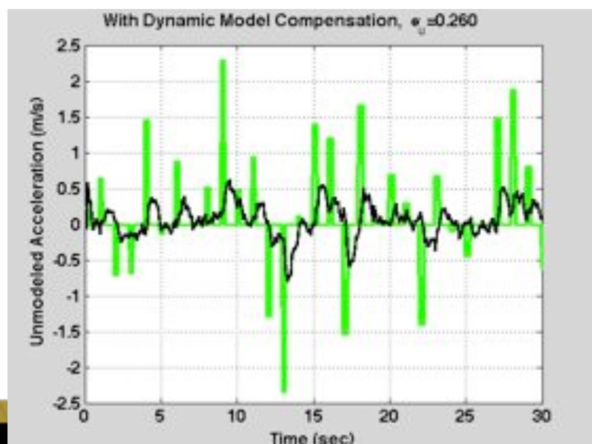


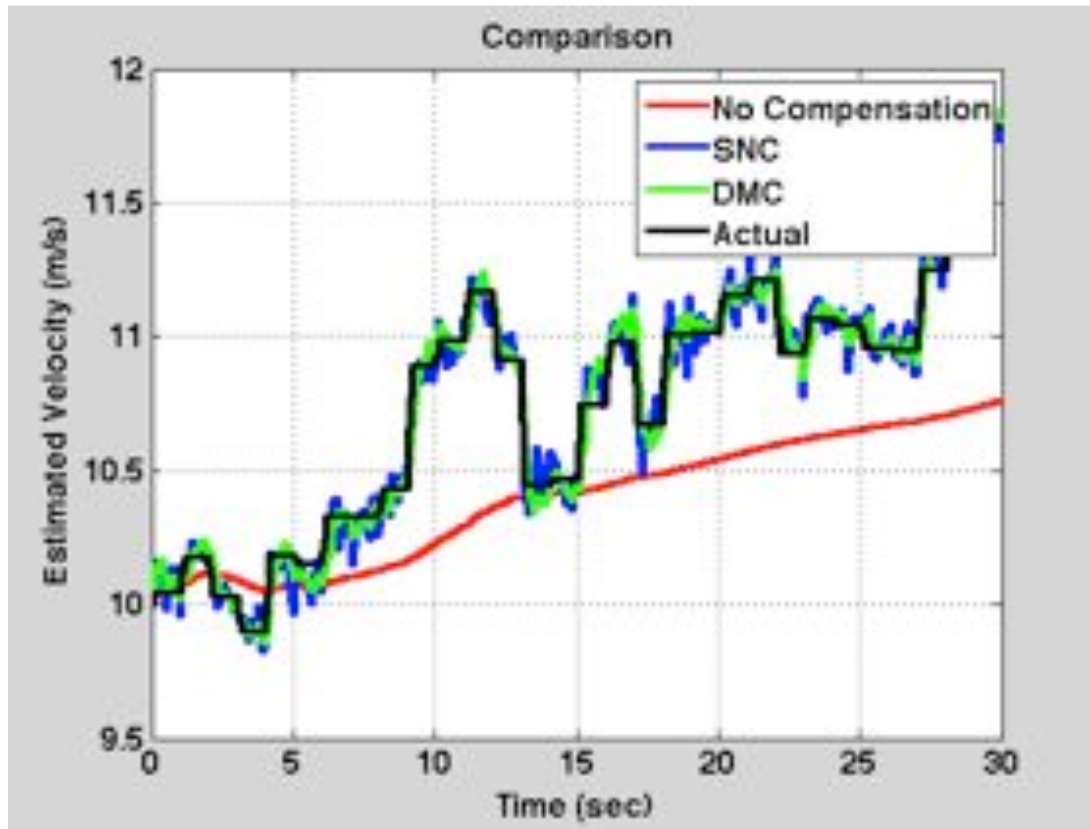
Example SNC / DMC

CCAR
Colorado Center for Astrodynamics Research

- ~Impulses







Status

- Shown an example
- Now we'll derive the SNC and DMC formulations
- Then we'll apply them to our satellite state estimation problem.

Derivation of State Noise Compensation

- Our setting:
  - We are not modeling everything perfectly – and we either suspect it or know it.

- State dynamics of a linear system under the influence of process noise:

$$\dot{\mathbf{x}}(t) = A(t)\mathbf{x}(t) + B(t)\mathbf{u}(t)$$

  - A(t) and B(t) are known.
  - u(t) is mx1
  - B(t) is nxm

Derivation of State Noise Compensation

- State dynamics of a linear system under the influence of process noise:

$$\dot{\mathbf{x}}(t) = A(t)\mathbf{x}(t) + B(t)\mathbf{u}(t)$$

  - u(t) can include all kinds of processes.
    - White noise, constant noise, piecewise constant, correlated, etc.

  - The standard *State Noise Compensation* (SNC) algorithm assumes that u(t) is white noise
  - Dynamic Model Compensation assumes that u(t) is more complex, as we saw earlier.

  - SNC: $$E[\mathbf{u}(t)] = 0$$

$$E[\mathbf{u}(t)\mathbf{u}^T(t)] = Q(t)\delta(t-\tau)$$

Derivation of State Noise Compensation

$$\mathbf{x}(t) = \Phi(t,t_0)\mathbf{x}_0 + \int_{t_0}^{t} \Phi(t,\tau)B(\tau)\mathbf{u}(\tau)d\tau$$

- This equation is the general solution for the inhomogeneous equation

$$\dot{\mathbf{x}}(t) = A(t)\mathbf{x}(t) + B(t)\mathbf{u}(t)$$

- It indicates how the true state propagates under the influence of process noise.

Derivation of State Noise Compensation

Next, we wish to understand how the estimate of the state $\mathbf{x}(t)$ propagates in the presence of process noise.

$$\mathbf{x}(t) = \Phi(t, t_0)\mathbf{x}_0 + \int_{t_0}^{t} \Phi(t, \tau)B(\tau)\mathbf{u}(\tau)d\tau$$

is a stochastic integral and cannot be evaluated in a deterministic sense.

$$\bar{\mathbf{x}}(t) = E\left[\mathbf{x}(t) \middle| \mathbf{y}_{k-1}\right] \qquad \text{for} \qquad t \geq t_{k-1}$$

$$\dot{\bar{\mathbf{x}}}(t) = E\left[\dot{\mathbf{x}}(t) \middle| \mathbf{y}_{k-1}\right] = E\left[\left\{A(t)\mathbf{x}(t) + B(t)\mathbf{u}(t)\right\} \middle| \mathbf{y}_{k-1}\right]$$

but $\qquad E\left[\mathbf{u}(t) \middle| \mathbf{y}_{k-1}\right] = 0$

Derivation of State Noise Compensation

so
$$\dot{\bar{\mathbf{x}}}(t) = A(t)E\left[\mathbf{x}(t)\middle|\mathbf{y}_{k-1}\right] = A(t)\bar{\mathbf{x}}(t)$$

$$\dot{\bar{\mathbf{x}}}(t) = A(t)\bar{\mathbf{x}}(t)$$

We have our familiar solution

$$\bar{\mathbf{x}}(t) = \Phi(t, t_{k-1})\hat{\mathbf{x}}_{k-1}$$

This is good news! It means that we don't have to change our methods to update x-bar in our sequential algorithm.

Derivation of State Noise Compensation

- Since $E[\mathbf{u}(t)] = 0$

  we have shown that the time update of the state is unchanged

$$\bar{\mathbf{x}}(t_k) = \Phi(t_k, t_{k-1})\hat{\mathbf{x}}(t_{k-1})$$

- If the mean of the process noise is nonzero

$$E[\mathbf{u}(t)] = \bar{\mathbf{u}}$$

$$\bar{\mathbf{x}}(t_k) = \Phi(t_k, t_{k-1})\hat{\mathbf{x}}(t_{k-1}) + \Gamma(t_k, t_{k-1})\bar{\mathbf{u}}$$

(defined later)

- The derivation of the time update of the covariance matrix is more in-depth.
- It begins with

$$\bar{P}(t) = E\left[(\bar{\mathbf{x}}(t) - \mathbf{x}(t))(\bar{\mathbf{x}}(t) - \mathbf{x}(t))^T \;\middle|\; \mathbf{y}_{k-1}\right] \quad t \geq t_{k-1}$$

$$\mathbf{x}(t) = \Phi(t, t_0)\mathbf{x}_0 + \int_{t_0}^{t} \Phi(t, \tau)B(\tau)\mathbf{u}(\tau)d\tau$$

$$\bar{\mathbf{x}}(t_k) = \Phi(t_k, t_{k-1})\hat{\mathbf{x}}(t_{k-1})$$

Derivation of State Noise Compensation

CCAR
Colorado Center for Astrodynamics Research

- After some work, we find ourselves aiming to solve the following expression:

$$\dot{\bar{P}}(t) = A(t)\bar{P}(t) + \bar{P}(t)A^T(t) + B(t)Q(t)B^T(t)$$

where A(t) and B(t) are the same as previously defined $\dot{\mathbf{x}}(t) = A(t)\mathbf{x}(t) + B(t)\mathbf{u}(t)$

and Q(t) comes from the nature of u(t)

$$E[\mathbf{u}(t)] = 0$$

$$E[\mathbf{u}(t)\mathbf{u}^T(t)] = Q(t)\delta(t-\tau)$$

Derivation of State Noise Compensation

$$\dot{\bar{P}}(t) = A(t)\bar{P}(t) + \bar{P}(t)A^T(t) + B(t)Q(t)B^T(t)$$

- Solve this using a similar method of variation of parameters.

- Homogenous differential equation:

$$\dot{\bar{P}}(t) = A(t)\bar{P}(t) + \bar{P}(t)A^T(t)$$

- has a solution of the form

$$\bar{P}(t) = \Phi_{t,t_0} P_0 \Phi_{t,t_0}^T$$

Derivation of State Noise Compensation

Colorado Center for Astrodynamics Research

$$\bar{P}(t) = \Phi_{t,t_0} P_0 \Phi_{t,t_0}^T$$

- Letting P_0 become a function of time, we find the derivative of P:

$$\dot{\bar{P}}(t) = \dot{\Phi}_{t,t_0} P_0 \Phi_{t,t_0}^T + \Phi_{t,t_0} P_0 \dot{\Phi}_{t,t_0}^T + \Phi_{t,t_0} \dot{P}_0 \Phi_{t,t_0}^T$$

Derivation of State Noise Compensation

Colorado Center for Astrodynamics Research

- Setting $\dot{\bar{P}}(t) = \dot{\Phi}_{t,t_0} P_0 \Phi^T_{t,t_0} + \Phi_{t,t_0} P_0 \dot{\Phi}^T_{t,t_0} + \Phi_{t,t_0} \dot{P}_0 \Phi^T_{t,t_0}$
- Equal to $\dot{\bar{P}}(t) = A(t)\bar{P}(t) + \bar{P}(t)A^T(t) + B(t)Q(t)B^T(t)$
- Yields

$$A(t)\bar{P}(t) + \bar{P}(t)A^T(t) + B(t)Q(t)B^T(t) = \dot{\Phi}_{t,t_0} P_0 \Phi^T_{t,t_0} + \Phi_{t,t_0} P_0 \dot{\Phi}^T_{t,t_0} + \Phi_{t,t_0} \dot{P}_0 \Phi^T_{t,t_0}$$

$$A(t)\bar{P}(t) + \bar{P}(t)A^T(t) + B(t)Q(t)B^T(t) =$$
$$\dot{\Phi}_{t,t_0}P_0\Phi^T_{t,t_0} + \Phi_{t,t_0}P_0\dot{\Phi}^T_{t,t_0} + \Phi_{t,t_0}\dot{P}_0\Phi^T_{t,t_0}$$

- Once again, recall our old friend:

$$\dot{\Phi}(t,t_0) = A(t)\Phi(t,t_0)$$

$$A(t)\bar{P}(t) + \bar{P}(t)A^T(t) + B(t)Q(t)B^T(t) =$$
$$A(t)\Phi_{t,t_0}P_0\Phi^T_{t,t_0} + \Phi_{t,t_0}P_0\Phi^T_{t,t_0}A^T(t) + \Phi_{t,t_0}\dot{P}_0\Phi^T_{t,t_0}$$

Derivation of State Noise Compensation

$$A(t)\bar{P}(t) + \bar{P}(t)A^T(t) + B(t)Q(t)B^T(t) =$$

$$A(t)\Phi_{t,t_0}P_0\Phi_{t,t_0}^T + \Phi_{t,t_0}P_0\Phi_{t,t_0}^T A^T(t) + \Phi_{t,t_0}\dot{P}_0\Phi_{t,t_0}^T$$

- Also recall that $\bar{P}(t) = \Phi_{t,t_0}P_0\Phi_{t,t_0}^T$

$$\cancel{A(t)\bar{P}(t) + \bar{P}(t)A^T(t)} + B(t)Q(t)B^T(t) =$$

$$\cancel{A(t)\bar{P}(t) + \bar{P}(t)A^T(t)} + \Phi_{t,t_0}\dot{P}_0\Phi_{t,t_0}^T$$

$$B(t)Q(t)B^T(t) = \Phi_{t,t_0}\dot{P}_0\Phi_{t,t_0}^T$$

Derivation of State Noise Compensation

$$B(t)Q(t)B^T(t) = \Phi_{t,t_0}\dot{P}_0\Phi_{t,t_0}^T$$

- or

$$\dot{P}_0 = \Phi_{t,t_0}^{-1}B(t)Q(t)B^T(t)\Phi_{t,t_0}^{-T}$$

- Integrating yields

$$P_0(t) = P_0 + \int_{t_0}^{t}\Phi_{\tau,t_0}^{-1}B(\tau)Q(\tau)B^T(\tau)\Phi_{\tau,t_0}^{-T}d\tau$$

Derivation of State Noise Compensation

CCAR
Colorado Center for Astrodynamics Research

- Substituting $P_0(t) = P_0 + \int_{t_0}^{t} \Phi_{\tau,t_0}^{-1} B(\tau) Q(\tau) B^T(\tau) \Phi_{\tau,t_0}^{-T} d\tau$

  into $\bar{P}(t) = \Phi_{t,t_0} P_0 \Phi_{t,t_0}^T$

  yields

$$\bar{P}(t) = \Phi_{t,t_0} P_0 \Phi_{t,t_0}^T + \int_{t_0}^{t} \underbrace{\Phi_{t,t_0} \Phi_{\tau,t_0}^{-1}}_{\Phi(t,\tau)} B(\tau) Q(\tau) B^T(\tau) \underbrace{\Phi_{\tau,t_0}^{-T} \Phi_{t,t_0}^T}_{\Phi^T(t,\tau)} d\tau$$

$$\bar{P}(t) = \Phi_{t,t_0} P_0 \Phi_{t,t_0}^T + \int_{t_0}^{t} \Phi_{t,\tau} B(\tau) Q(\tau) B^T(\tau) \Phi_{t,\tau}^T d\tau$$

Derivation of State Noise Compensation

CCAR
Colorado Center for Astrodynamics Research

$$\bar{\mathbf{x}}(t_k) = \Phi(t_k, t_{k-1})\hat{\mathbf{x}}(t_{k-1})$$

$$\bar{P}(t_k) = \Phi_{k,k-1} P_{k-1} \Phi_{k,k-1}^T + \int_{t_{k-1}}^{t_k} \Phi_{t_{k-1},\tau} B(\tau) Q(\tau) B^T(\tau) \Phi_{t_{k-1},\tau}^T d\tau$$

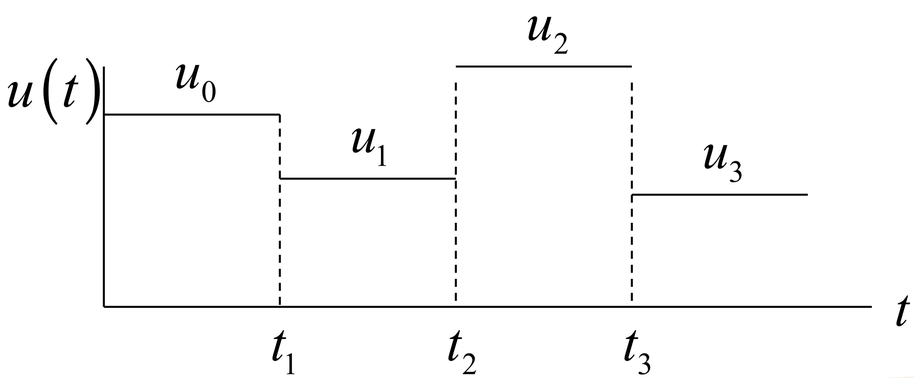
- This is a great start.
- These are the equations for propagating the estimate of the state and the associated covariance for a *continuous* system.
- The stat OD process uses discrete observations.

Derivation of State Noise Compensation

- We'd like to discretize the time update equations.
- Replace the continuous white-noise process u(t) with a white *random sequence*

$$E[\mathbf{u}(t_i)] = 0$$

$$E[\mathbf{u}(t_i)\mathbf{u}^T(t_j)] = Q_i \delta_{ij} \qquad \delta_{ij} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}$$



Derivation of State Noise Compensation

CCAR
Colorado Center for Astrodynamics Research

- Now we have

$$\mathbf{x}_k = \Phi(t_k, t_{k-1})\mathbf{x}_{k-1} + \Gamma(t_k, t_{k-1})\mathbf{u}_{k-1}$$

where

$$\Gamma(t_k, t_{k-1}) = \int_{t_{k-1}}^{t_k} \Phi(t_k, \tau)B(\tau)d\tau$$

Gamma is the *process noise transition matrix*

Derivation of State Noise Compensation

- Gamma is the *process noise transition matrix*

$$\mathbf{x}_k = \Phi(t_k, t_{k-1})\mathbf{x}_{k-1} + \Gamma(t_k, t_{k-1})\mathbf{u}_{k-1}$$

$$\Gamma(t_k, t_{k-1}) = \int_{t_{k-1}}^{t_k} \Phi(t_k, \tau)B(\tau)d\tau$$

$$\Gamma(t_k, t_{k-1}) = \frac{\partial \mathbf{X}(t_k)}{\partial \mathbf{u}(t_{k-1})}$$

Quiz: What is the size of the Gamma matrix?

Derivation of State Noise Compensation

- Now we can return to the definition of the estimation error covariance matrix

$$\bar{P}_k = E\left[(\bar{\mathbf{x}}_k - \mathbf{x}_k)(\bar{\mathbf{x}}_k - \mathbf{x}_k)^T\right]$$

$$\bar{\mathbf{x}}(t_k) = \Phi(t_k, t_{k-1})\hat{\mathbf{x}}(t_{k-1})$$

$$\mathbf{x}_k = \Phi(t_k, t_{k-1})\mathbf{x}_{k-1} + \Gamma(t_k, t_{k-1})\mathbf{u}_{k-1}$$

$$\bar{P}_k = E\left[\Big(\Phi(t_k, t_{k-1})\hat{\mathbf{x}}(t_{k-1}) - \Phi(t_k, t_{k-1})\mathbf{x}_{k-1} - \Gamma(t_k, t_{k-1})\mathbf{u}_{k-1}\Big)\cdots \right.$$
$$\left. \times \Big(\Phi(t_k, t_{k-1})\hat{\mathbf{x}}(t_{k-1}) - \Phi(t_k, t_{k-1})\mathbf{x}_{k-1} - \Gamma(t_k, t_{k-1})\mathbf{u}_{k-1}\Big)^T\right]$$

Derivation of State Noise Compensation

$$\bar{P}_k = E\Big[\Big(\Phi(t_k, t_{k-1})\hat{\mathbf{x}}(t_{k-1}) - \Phi(t_k, t_{k-1})\mathbf{x}_{k-1} - \Gamma(t_k, t_{k-1})\mathbf{u}_{k-1}\Big)\cdots$$

$$\times \Big(\Phi(t_k, t_{k-1})\hat{\mathbf{x}}(t_{k-1}) - \Phi(t_k, t_{k-1})\mathbf{x}_{k-1} - \Gamma(t_k, t_{k-1})\mathbf{u}_{k-1}\Big)^T\Big]$$

- We can simplify this knowing that

$$\boxed{\begin{aligned} &E[(\hat{\mathbf{x}}_k - \mathbf{x}_k)\mathbf{u}_k^T] = 0 \\ &E[(\hat{\mathbf{x}}_k - \mathbf{x}_k)(\hat{\mathbf{x}}_k - \mathbf{x}_k)^T] = P_k \\ &E[\mathbf{u}_k\mathbf{u}_k^T] = Q_k \end{aligned}}$$

$$\bar{P}_k = \Phi_{k,k-1}P_{k-1}\Phi_{k,k-1}^T + \Gamma_{k,k-1}Q_{k-1}\Gamma_{k,k-1}^T$$

Quiz: What is the size of the Q matrix?

- FINALLY, we have our SNC algorithm

$$\bar{\mathbf{x}}(t_k) = \Phi(t_k, t_{k-1})\hat{\mathbf{x}}(t_{k-1})$$

$$\bar{P}_k = \Phi_{k,k-1} P_{k-1} \Phi^T_{k,k-1} + \Gamma_{k,k-1} Q_{k-1} \Gamma^T_{k,k-1}$$

$$\dot{\mathbf{x}}(t) = A(t)\mathbf{x}(t) + B(t)\mathbf{u}(t)$$

$$\Gamma(t_k, t_{k-1}) = \int_{t_{k-1}}^{t_k} \Phi(t_k, \tau) B(\tau) d\tau$$

$$E[\mathbf{u}(t_i)] = 0$$

$$E[\mathbf{u}(t_i)\mathbf{u}^T(t_j)] = Q_i \delta_{ij} \qquad \delta_{ij} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}$$

Actual Process

- Determine what your B matrix is.
- Determine Q via documentation, *a priori* experience, or plain trial and error!
- Compute Lambda
- Tune it.

Tuning the Filter

- To tune the filter:
- Practice on many simulated runs.
- Aim to tune the filter to add the least amount of inflation to the P-matrix while reducing the residuals to near-noise.
- Check the trace of the P-matrix. Is it too large?
- Do you know anything about the uncertainty in the model? Perhaps you can orient the Q matrix to accommodate for expected uncertainties (drag, SRP, maneuver errors, etc)

Consider our scenario

$$\mathbf{X}(t) = \begin{bmatrix} \mathbf{R}(t) \\ \mathbf{V}(t) \\ \mathbf{C}(t) \end{bmatrix} \quad \dot{\mathbf{X}}(t) = \begin{bmatrix} \mathbf{V}(t) \\ \mathbf{A}(t) \\ \mathbf{0}(t) \end{bmatrix}$$

$$\boxed{\begin{aligned} &\dot{\mathbf{x}}(t) = A(t)\mathbf{x}(t) + B(t)\mathbf{u}(t) \\ &\Gamma(t_k, t_{k-1}) = \int_{t_{k-1}}^{t_k} \Phi(t_k, \tau)B(\tau)d\tau \\ &E[\mathbf{u}(t_i)] = 0 \\ &E[\mathbf{u}(t_i)\mathbf{u}^T(t_j)] = Q_i\delta_{ij} \qquad \delta_{ij} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} \end{aligned}}$$

Consider our scenario

Colorado Center for Astrodynamics Research

$$\mathbf{X}(t) = \begin{bmatrix} \mathbf{R}(t) \\ \mathbf{V}(t) \\ \mathbf{C}(t) \end{bmatrix} \quad \dot{\mathbf{X}}(t) = \begin{bmatrix} \mathbf{V}(t) \\ \mathbf{A}(t) \\ \mathbf{0}(t) \end{bmatrix}$$

$$\left[\dot{\mathbf{x}}(t)\right]_{n\times 1} = \left[A(t)\right]_{n\times n}\left[\mathbf{x}(t)\right]_{n\times 1} + \left[B(t)\right]_{n\times m}\left[\mathbf{u}(t)\right]_{m\times 1}$$

$$\begin{bmatrix} \delta\dot{x}(t) \\ \delta\dot{y}(t) \\ \delta\dot{z}(t) \\ \delta\ddot{x}(t) \\ \delta\ddot{y}(t) \\ \delta\ddot{z}(t) \\ \delta\dot{\mu}(t) \\ \delta\dot{J}_2(t) \\ \delta\dot{C}_D(t) \\ \delta\dot{x}_{S1}(t) \\ \vdots \\ \delta\dot{z}_{S3}(t) \end{bmatrix} = A(t)\mathbf{x}(t) + \quad \text{B*u}$$

What is B?

Where does the white noise appear in the state?

Consider our scenario

CCAR
Colorado Center for Astrodynamics Research

$$\mathbf{X}(t) = \begin{bmatrix} \mathbf{R}(t) \\ \mathbf{V}(t) \\ \mathbf{C}(t) \end{bmatrix} \quad \dot{\mathbf{X}}(t) = \begin{bmatrix} \mathbf{V}(t) \\ \mathbf{A}(t) \\ \mathbf{0}(t) \end{bmatrix}$$

$$\left[\dot{\mathbf{x}}(t)\right]_{n\times 1} = \left[A(t)\right]_{n\times n}\left[\mathbf{x}(t)\right]_{n\times 1} + \left[B(t)\right]_{n\times m}\left[\mathbf{u}(t)\right]_{m\times 1}$$

$$\begin{bmatrix} \delta\dot{x}(t) \\ \delta\dot{y}(t) \\ \delta\dot{z}(t) \\ \delta\ddot{x}(t) \\ \delta\ddot{y}(t) \\ \delta\ddot{z}(t) \\ \delta\dot{\mu}(t) \\ \delta\dot{J_2}(t) \\ \delta\dot{C_D}(t) \\ \delta\dot{x_{S1}}(t) \\ \vdots \\ \delta\dot{z_{S3}}(t) \end{bmatrix} = A(t)\mathbf{x}(t) + \begin{bmatrix} \mathbf{0}_{3\times 3} \\ \mathbf{I}_{3\times 3} \\ \mathbf{0}_{3\times 3} \\ \vdots \\ \mathbf{0}_{3\times 3} \end{bmatrix} \begin{bmatrix} u_x(t) \\ u_y(t) \\ u_z(t) \end{bmatrix}$$

This B(t) matrix is constant and represents a spherically uncertain acceleration.

It could vary with time according to some geometry, say, to reflect uncertainties with SRP.

Consider our scenario

CCAR
Colorado Center for Astrodynamics Research

- Given B, we compute Gamma

$$\Gamma(t_k, t_{k-1}) = \int_{t_{k-1}}^{t_k} \Phi(t_k, \tau) B(\tau) d\tau$$

$$B(t) = \begin{bmatrix} \mathbf{0}_{3\times 3} \\ \mathbf{I}_{3\times 3} \\ \mathbf{0}_{3\times 3} \\ \vdots \\ \mathbf{0}_{3\times 3} \end{bmatrix}$$

$$\Gamma_{k,k-1} \rightarrow \begin{bmatrix} \frac{(t_k - t_{k-1})^2}{2} \times \mathbf{I} \\ (t_k - t_{k-1}) \times \mathbf{I} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}$$

(See handout on web for why, but it applies to dense data s.t. the spacecraft's velocity doesn't change between observations.)

Consider our scenario

Colorado Center for Astrodynamics Research

- Construct Q to capture the expected uncertainty in the dynamics.
- Or, set it via trial and error to assist the filter.
- It is usually a diagonal matrix with small numbers on the diagonal. Given our scenario it will be a 3x3 that is constant over time.

$$\bar{P}_k = \Phi_{k,k-1} P_{k-1} \Phi^T_{k,k-1} + \Gamma_{k,k-1} Q_{k-1} \Gamma^T_{k,k-1}$$

$$Q = \begin{bmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_z^2 \end{bmatrix}$$

i.e., Variance of noise in ddx

Up next:

- Dynamic Model Compensation
- The Gauss-Markov Process

Dynamic Model Compensation

- DMC is an advanced form of SNC.
- Rather than assuming that the unmodeled accelerations are purely white-noise, we assume that there is some sort of structure.
- Estimate that acceleration.
- Use the estimates of unmodeled acceleration as a clue to improve your dynamical model.
  - This can provide a good hint that an acceleration is missing from your model.
  - This can provide a good hint that your attitude (or other parameter) is wrong.
  - Other hints.

4.9.1 The Gauss-Markov Process

A first-order Gauss-Markov process is often used for dynamic model compensation in orbit determination problems to account for unmodeled or inaccurately modeled accelerations acting on a spacecraft.

Gauss-Markov DMC

A Gauss-Markov process obeys a differential equation (often referred to as a Langevin equation) of the form

$$\dot{\eta}(t) = -\beta\eta(t) + u(t) \tag{4.9.51}$$

where $u(t)$ is white Gaussian noise with

$$\begin{aligned} E(u) &= 0 \\ E\left[u(t)u(\tau)\right] &= \sigma^2\delta(t-\tau) \end{aligned} \tag{4.9.52}$$

and

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

where τ is the time constant or correlation time (not the same as τ in Eq. (4.9.52)).

Gauss-Markov DMC

CCAR
Colorado Center for Astrodynamics Research

$$\dot{\eta}(t) = -\beta\eta(t) + u(t) \qquad (4.9.51)$$

Equation (4.9.51) can be solved by the method of variation of parameters to yield

$$\eta(t) = \eta(t_0)e^{-\beta(t-t_0)} + \int_{t_0}^{t} e^{-\beta(t-\tau)}u(\tau)d\tau. \qquad (4.9.53)$$

Deterministic

Random

Gauss-Markov DMC

CCAR
Colorado Center for Astrodynamics Research

- Consider the autocorrelation function

$$E\left[\eta(t_j)\eta(t_i)\right] = e^{-\beta(t_j - t_i)} E\left[\eta(t_i)\eta(t_i)\right]$$

- Indicates that the DMC process is correlated over time via a fading exponential, at the rate of the time constant $\tau = 1/\beta$

Gauss-Markov DMC

- The acceleration function cannot be deterministically evaluated, on account of the stochastic integral:

$$\eta(t) = \eta(t_0)e^{-\beta(t-t_0)} + \int_{t_0}^{t} e^{-\beta(t-\tau)}u(\tau)d\tau$$

- Statistically, the stochastic integral has:
  - Mean: 0 (since u has mean zero)
  - Variance: $\sigma_\eta^2 = \dfrac{\sigma^2}{2\beta}(1 - e^{-2\beta(t-t_0)})$

Gauss-Markov DMC

Colorado Center for Astrodynamics Research

- The stochastic integral is a Gaussian process.
  - It is uniquely defined by its mean and variance.
- If another function can be found with the same mean and variance, it may be used as an equivalent process. Such a discrete process is:

What?

Mean: 0

Variance: $\sigma_\eta^2 = \dfrac{\sigma^2}{2\beta}(1 - e^{-2\beta(t-t_0)})$

University of Colorado Boulder

Gauss-Markov DMC

- The stochastic integral is a Gaussian process.
  - It is uniquely defined by its mean and variance.
- If another function can be found with the same mean and variance, it may be used as an equivalent process. Such a discrete process is:

$$L_k \equiv u_k \sqrt{\frac{\sigma^2}{2\beta}(1 - e^{-2\beta(t-t_0)})}$$

Gauss-Markov DMC

CCAR
Colorado Center for Astrodynamics Research

The solution for $\eta(t)$ is given by

$$\eta(t_j) = e^{-\beta(t_j - t_i)}\eta(t_i) + u_k(t_i)\sqrt{\frac{\sigma^2}{2\beta}\left(1 - e^{-2\beta(t_j - t_i)}\right)} \qquad (4.9.60)$$

where $u_k(t_i)$ is a random number chosen by sampling from a Gaussian density function with a mean of zero and variance of 1.

Gauss-Markov DMC

$$\eta(t) = e^{-\beta(t-t_0)}\eta(t_0) + u_k(t_0)\sqrt{\frac{\sigma^2}{2\beta}(1 - e^{-2\beta(t-t_0)})}$$

- This process behaves differently for different values of sigma and beta.

- What happens if beta → 0?

$$\eta(t) = \eta(t_0) + u_k(t_0)\sigma\sqrt{t - t_0}$$

Gauss-Markov DMC

CCAR
Colorado Center for
Astrodynamics Research

$$\eta(t) = e^{-\beta(t-t_0)}\eta(t_0) + u_k(t_0)\sqrt{\frac{\sigma^2}{2\beta}(1 - e^{-2\beta(t-t_0)})}$$

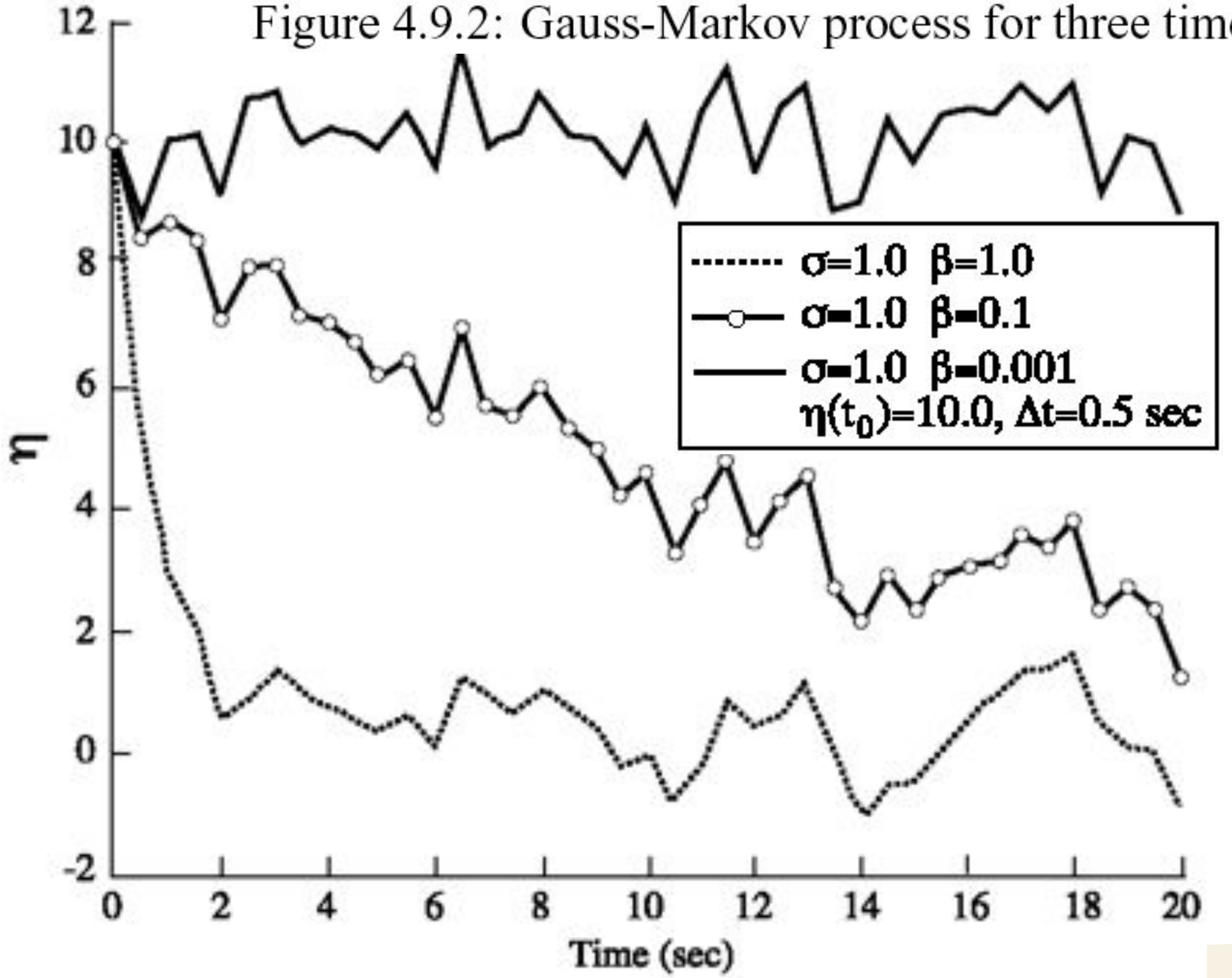
- This process behaves differently for different values of sigma and beta.

- What happens if sigma → 0?

$$\eta(t) = e^{-\beta(t-t_0)}\eta(t_0)$$

Gauss-Markov DMC

Figure 4.9.2: Gauss-Markov process for three time constants.



Gauss-Markov DMC

CCAR
Colorado Center for Astrodynamics Research

To use the Gauss-Markov process to account for unmodeled accelerations in the orbit determination procedure we may formulate the problem as follows. The equations of motion are

$$\begin{aligned}\dot{\mathbf{r}} &= \mathbf{v} \\ \dot{\mathbf{v}} &= -\frac{\mu\mathbf{r}}{r^3} + \mathbf{f}(t) + \boldsymbol{\eta}(t) \\ \dot{\boldsymbol{\eta}}(t) &= -\beta\boldsymbol{\eta}(t) + \mathbf{u}(t),\end{aligned} \tag{4.9.63}$$

where $-\frac{\mu\mathbf{r}}{r^3} + \mathbf{f}(t)$ represents the known accelerations and $\boldsymbol{\eta}(t)$ is a 3×1 vector of unknown accelerations. The procedure is to estimate the deterministic portion of $\boldsymbol{\eta}(t)$ and perhaps the associated time constants as part of the state vector. The random portion of $\boldsymbol{\eta}(t)$ contributes to the process noise covariance matrix Q.

Applications

- Model the accelerations in different ways
  - Symmetric?
  - Directional?
- Any need to use other DMC formulations?
  - That Gauss-Markov formulation works really well!
  - How to adapt your dynamical model

Applications

- I encourage you to try out SNC and DMC in these forms for the final project.
- There *are* unmodeled accelerations in the final project. Perhaps you can make a guess as to what they are.
- Compare the covariances with and without process noise. If the SNC/DMC is tuned properly, the covariance trace should be smaller with compensation.

The End

- HW 10 due Today (solutions).
- HW 11 due next week.