Fundamentals of Kalman Filtering and Applications to GNSS

Mohinder S. Grewal, Ph.D., P.E.
California State University, Fullerton

May 1, 2011

Outline

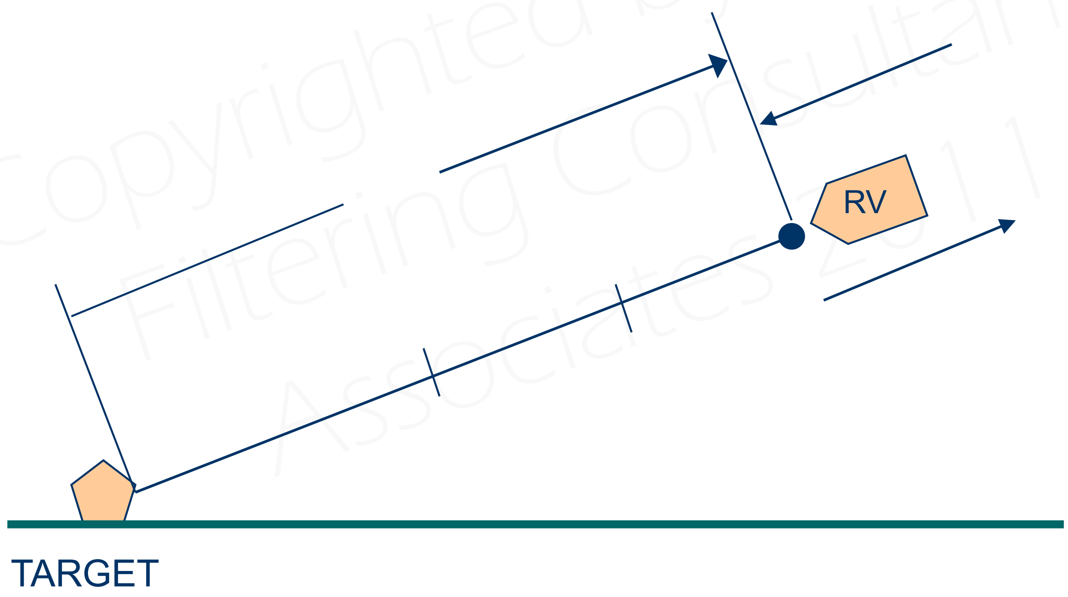
What Is A Kalman Filter?

Problem

- Develop an algorithm to process the data

$$Z(t_0 + i\Delta t),\ \ i = 1, 2, 3, \ldots 30$$

and form an estimate of position $P(t_0 + 6)$. Use all relevant information.



Problem (cont.)

- Kalman filter based upon given model(s)
  - Measurement process model

$$Z(t_0 + i\Delta t) = P(t_0 + \Delta t) + v(t_0 + \Delta t), \qquad i = 1, 2, 3, \ldots 30$$

$\Delta t = 0.1$ SEC

Data — RV Position — Measurement Error +/- 50 ft

  - System dynamics model

$$\dot{P} = V$$

$$\dot{V} = A$$

$$\dot{A} = 0$$

  - Initial condition model

$$P(t_0) = 100 \pm 10 \text{ KFT}$$

$$V(t_0) = -15 \pm 1 \text{ KPS}$$

$$A(t_0) = 20 \pm 1 \text{ Gs}$$

Alternative Problems, Issues

- What is the required radar accuracy to achieve a prescribed accuracy in the final estimate of $P(t_0 + 6)$.
- What are the implications of having range rate information available in addition to range data?
- How about samples every 0.05 sec, 0.01 sec., etc.?
- How sensitive is the algorithm to deviations from assumed model?
  - Suppose $\dot{A}$ is not constant
  - Suppose measurement errors not “independent”
- Given the initial conditions, how accurately could one estimate $P(t_0 + 6)$ without any data?

What is a Kalman Filter?

- Algorithm for generating estimates of the state of a system based upon
  - A mathematical model
    - System states are governed by linear differential or difference equations driven by white noise
    - Measurements are linear functions of the states + white measurement noise
  - An initial estimate
    - Initial estimates (can be very poor) are required. Also level of uncertainty needs specification
  - A set of measurements
    - Data from real hardware (sensors) such as GNSS
      - Pseudoranges
      - Delta Pseudoranges

RV Example

- Mathematical MODEL

$$\dot{P} = V;\ \dot{V} = A;\ \dot{A} = 0$$ (no noise in this example)

$$Z(t_0 + i\Delta t) = P(t_0 + i\Delta t) + v(t_0 + i\Delta t)$$

$$v(t_0 + i\Delta t) = \pm 50 \text{ FT}$$

- Initial estimate

$$P(t_0) = 100 \pm 10 \text{ KFT}$$

$$V(t_0) = -15 \pm 1 \text{ KPS}$$

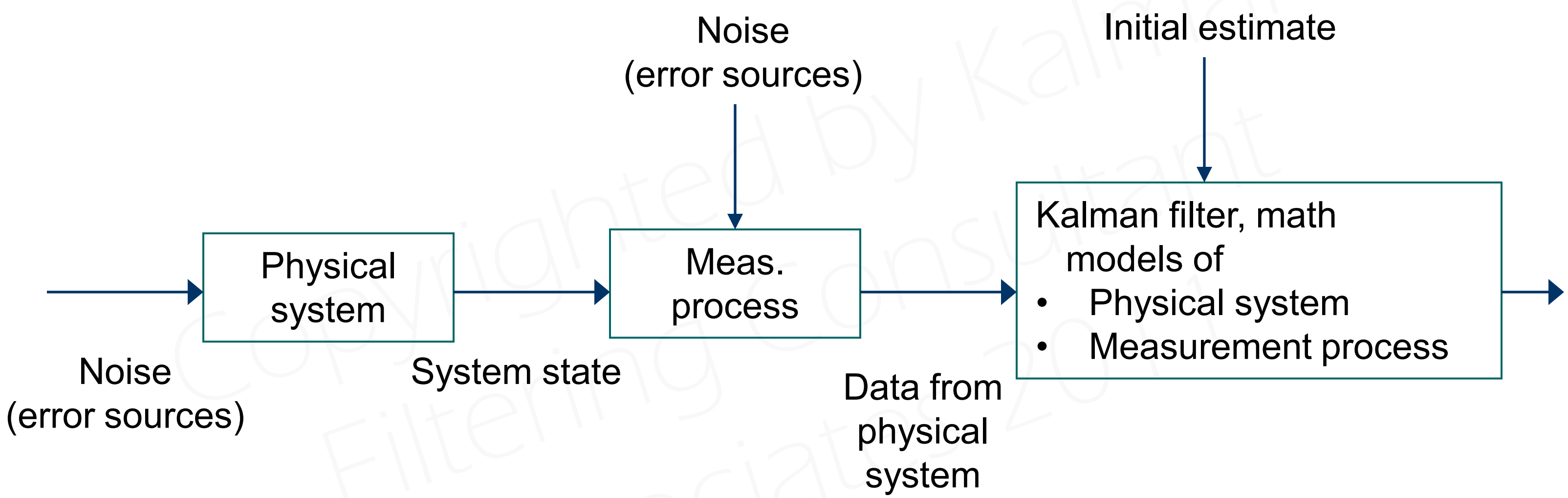
$$A(t_0) = 20 \pm 1 \text{ Gs}$$

- Measurements

100752, 99243, 97732, …

Kalman Filtering Problem

- Top level sketch

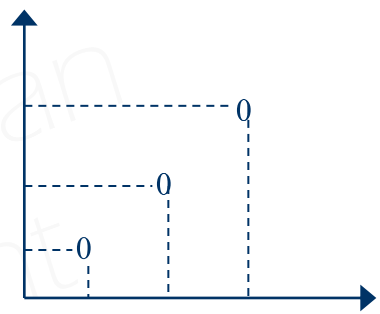


Attributes of Kalman Filter

- ± Kalman filter is "optimal"— if physical world and mathematical model coincide. This is never the case. Dealing with model disparities is major task of Kalman filter designer.
- \+ Kalman filter is "recursive"— estimates are updated upon receipt of each measurement. No need to save past data.
- ± Kalman filter creates its own error analysis— the "1σ" estimation errors are generated as part of the algorithm. But the values are only as good as the model.
- — Kalman filters can create numerical difficulties— precision, memory, speed.
- \+ Kalman filters are easily reconfigured to handle wild data points and model changes.
- ± Kalman filter model is linear— real world is always non-linear, to some extent.
- ± Kalman filter operates in vector/matrix format— concepts and operations are independent of number of states.

Example: Curve Fitting

- Given
  - Three data points $(t_1, z_1), (t_2, z_2), (t_3, z_3)$
  - A relationship between t and Z of the form $Z = \alpha + \beta t + \gamma t^2$ (1)
  - Find α, β, γ in terms of the given data.



- Solution

$$z_1 = \alpha + t_1 \beta + t_1^2 \gamma$$
$$z_2 = \alpha + t_2 \beta + t_2^2 \gamma$$
$$z_3 = \alpha + t_3 \beta + t_3^2 \gamma$$

$$\begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} = \begin{bmatrix} 1 & t_1 & t_1^2 \\ 1 & t_2 & t_2^2 \\ 1 & t_3 & t_3^2 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} \qquad (2)$$

$$Z = HX$$

$$H^{-1}Z = H^{-1}HX = X \qquad \boxed{X = H^{-1}Z \quad (3)}$$

- Solution amounts to solving 3 equations for 3 unknowns α, β, γ

  Since Eq. (2) is satisfied, the Z vs t curve determined by Eqs (1) and (3) is guaranteed to pass through the three data points $(t_1, z_1), (t_2, z_2), (t_3, z_3)$

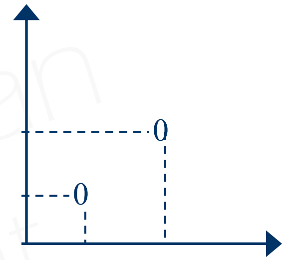
z_3

z_2

Z t_1 t_2 t_3

Example: Curve Fitting (cont.)

- Given
  - Two data points $(t_1, z_1), (t_2, z_2)$
  - A relationship between t and Z of the form $Z = \alpha + \beta t + \gamma t^2$
  - Find $\alpha, \ \beta, \ \gamma$ in terms of the given data.



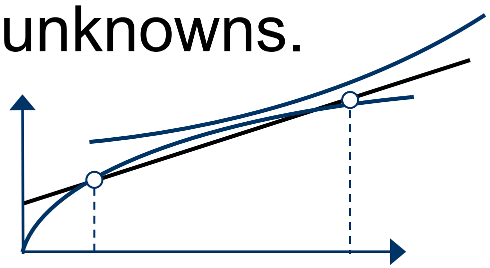
- Solution

$$z_1 = \alpha + t_1 \beta + t_1^2 \gamma$$
$$z_2 = \alpha + t_2 \beta + t_2^2 \gamma$$

two equations and 3 unknowns ⇨ **UNDERDETERMINED**

$$\begin{bmatrix} z_1 - \alpha \\ z_2 - \alpha \end{bmatrix} = \begin{bmatrix} t_1 & t_1^2 \\ t_2 & t_2^2 \end{bmatrix} \begin{bmatrix} \beta \\ \gamma \end{bmatrix} \quad (1) \qquad \begin{bmatrix} z_1 - t_1\beta \\ z_2 - t_2\beta \end{bmatrix} = \begin{bmatrix} 1 & t_1^2 \\ 1 & t_2^2 \end{bmatrix} \begin{bmatrix} \alpha \\ \gamma \end{bmatrix} \quad (2) \qquad \begin{bmatrix} z_1 - t_1^2\gamma \\ z_2 - t_2^2\gamma \end{bmatrix} = \begin{bmatrix} 1 & t_1 \\ 1 & t_2 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \quad (3)$$

- Procedure: choose an $\alpha, \ \beta,$ or γ arbitrary and solve Eqs (1), (2), or (3) for remaining two unknowns.
  - Result: An infinite number of solutions exist. Some possibilities are

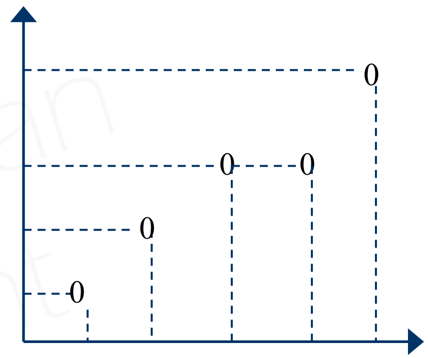


Example: Curve Fitting (cont.)

- Given
  - Five data points

$$(t_1, z_1), (t_2, z_2), (t_3, z_3), (t_4, z_4), (t_5, z_5)$$

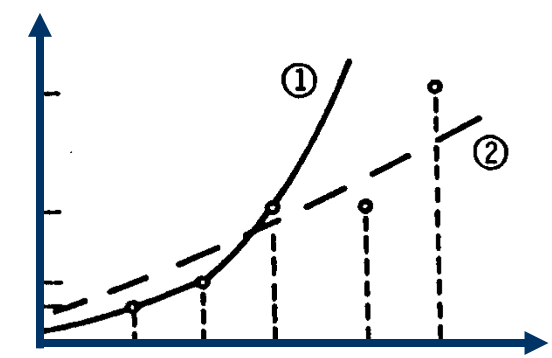
  - A relationship between t and Z of the form $Z = \alpha + \beta t + \gamma t^2$
  - Find α, β, γ in terms of the given data



- Solution

$$\begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \end{bmatrix} = \begin{bmatrix} 1 & t_1 & t_1^2 \\ 1 & t_2 & t_2^2 \\ 1 & t_3 & t_3^2 \\ 1 & t_4 & t_4^2 \\ 1 & t_5 & t_5^2 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} \qquad Z_{5\times1} = H_{5\times3} X_{3\times1}$$

5 equations and 3 unknowns → **OVERDETERMINED**



Approach 1: Use only the first 3 equations and solve for α, β, γ as in first part of this example; ignore the last 2 equations..
Result: curve will pass through first 3 data points. May not even come close to curve 1

Approach 2: Try drawing a "best fit" to the data. See curve 2.

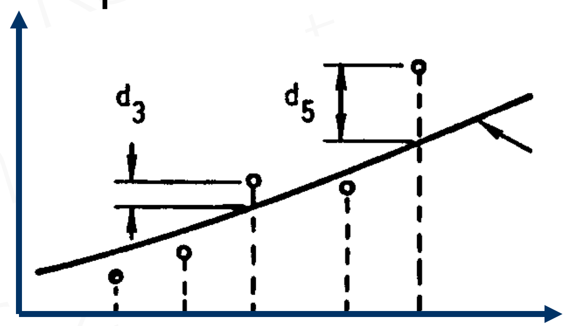
Example: Curve Fitting (cont.)

- Denote a chosen set of $\alpha,\ \beta,\ \gamma$ values by $\hat{\alpha},\ \hat{\beta},\ \hat{\gamma}$. Also let $\hat{Z}_i = \hat{\alpha} + \hat{\beta} t_i + \hat{\gamma} t_i^2 \qquad i = 1, 2, 3, \ldots 5$
  - We propose to choose the $\hat{\alpha},\ \hat{\beta},\ \hat{\gamma}$ according to the criteria that the fit determined is best in the "least squares" sense.

Specifically, we minimize J where

$$J = \sum_{1}^{n} \left(Z_i - \hat{Z}_i\right)^2 = \sum_{1}^{n} \left(d_i\right)^2 \quad ; \quad n = 5$$

For this example



- Procedure

$$J = \sum_{1}^{n} \left(Z_i - \hat{Z}_i\right)^2 = \left(Z - \hat{Z}\right)^T \left(Z - \hat{Z}\right) = \left(Z - H\hat{X}\right)^T \left(Z - H\hat{X}\right)$$

$$\frac{\partial}{\partial \hat{X}} \left[\left(Z - H\hat{X}\right)^T \left(Z - H\hat{X}\right) \right] = 0 \qquad \text{(NECESSARY CONDITION)}$$

  - The above leads to $\boxed{\hat{X} = \left(H^T H\right)^{-1} H^T Z \qquad \left|H^T H\right| \neq 0}$

$t_1 \quad t_2 \quad t_3 \quad t_4 \quad t_5$

Example: Curve Fitting Summary

- Given
  - Observation vector $Z_{m\times 1}$
  - Vector of parameters to be determined $X_{n\times 1}$
  - Assumed model of the form $Z = HX \quad ; \quad H_{m\times n}$
  - Find X
    - Case 1: Underdetermined $m < n$
      - Infinite number of solutions exist
      - That is, not enough information (equations) to uniquely specify all elements of X
    - Case 2: Exactly determined $n = m$
      - Unique solution for X exists provided $|H| \neq 0$

        $X = H^{-1}Z$
    - Case 3: Over determined $m > n$
      - Cannot guarantee a perfect fit to the data
      - A "best fit" in the least squares sense can be determined to be

        $$\hat{X} = \left(H^T H\right)^{-1} H^T Z \quad ; \quad \left|H^T H\right| \neq 0$$
      - This solution guarantees $J = \left(Z - H\hat{X}\right)^T \left(z - H\hat{X}\right) =$ minimum
      - **This case is of the most interest.**

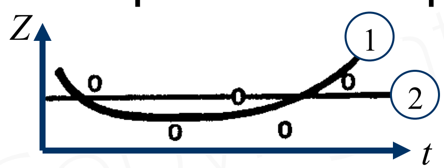
Example: Curve Fitting Summary

- Curve-fitting least squares fit

$$Z = HX$$

$$\hat{X} = \left(H^T H\right)^{-1} H^T Z \qquad \left|H^T H\right| \neq 0$$

1. Results predicated upon assumed model $Z = HX$



MODEL 1

2

Model validity is always a problem

2. All residuals $\left(Z_i - \hat{Z}_i\right)$ are weighted equally. No provisions for "de-weighting" some points.
3. No information regarding "a priori" knowledge of parameters used.
4. Batch processing is implied by $\hat{X} = \left(H^T H\right)^{-1} H^T Z$
5. Criteria is one of fitting data; not minimizing estimation error, $X - \hat{X}$

$$Z = \alpha + \beta t + \gamma t^2$$

$$Z = \alpha + \beta t$$

Example—The Kalman Filter

Curve-fitting least squares fit

- The Kalman filter
  - Kalman filtering brings into consideration 2, 3, 4, 5.
  - Modeling remains a problem.
  - The least squares curve fit and the Kalman filter yield the same estimates when
    - Initial uncertainty in X is large.
    - All observations are of equal quality.
    - System is overdetermined or exactly determined.
  - The Kalman filter is recursive.
  - The Kalman filter accommodates a dynamical model for X.

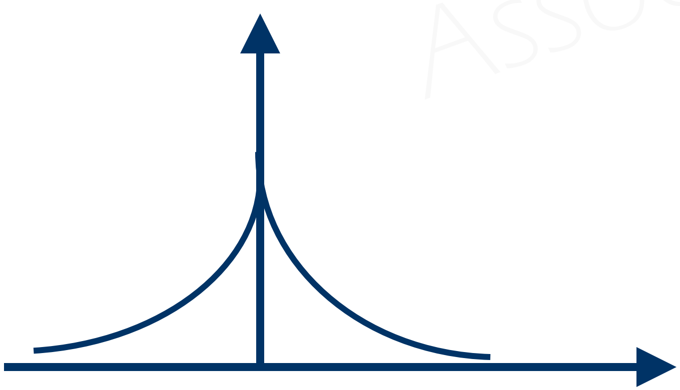
Error Models for Random Processes and Sequences

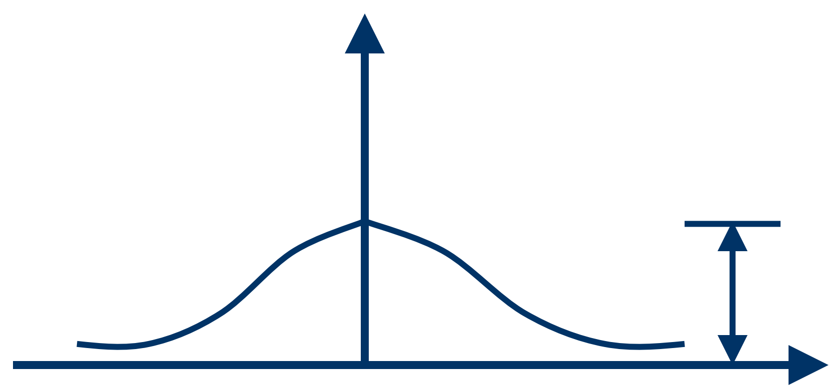
Exponentially Correlated Noise

- A process *x(t)* is called exponentially correlated if it has zero mean and an autocorrelation function of the form

$$E\left\{x(t_1)x(t_2)\right\} = \sigma_x^2\, e^{-\alpha\,|t_2 - t_1|} \;=\; \sigma_x^2\, e^{-\alpha|\tau|} = \Psi_{xx}(\tau);\;\; \tau = |t_2 - t_1|$$

  - where $\frac{1}{\alpha} =$ "correlation time."
  - This process is stationary and has a power spectral density given by $\Psi_{xx}(\omega) = \int_{-\infty}^{+\infty} \Psi_{xx}(\tau)\, e^{-j\omega\tau}\, d\tau = \dfrac{2\,\sigma_x^2\,\alpha}{\omega^2 + \alpha^2}$
  - $\Psi_{xx}(\tau)$ and $\Psi_{xx}(\omega)$ are sketched below

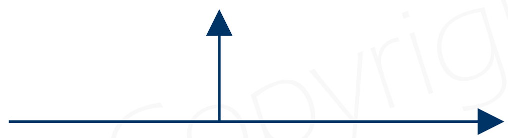


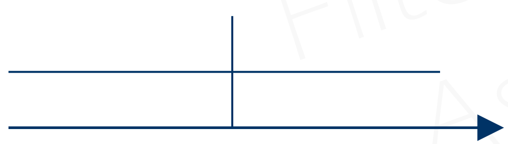


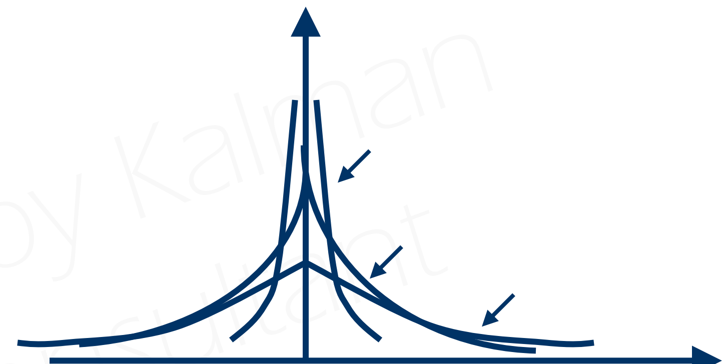
$\dfrac{2\,\sigma_x^2}{\alpha}$ σ_x^2 $\Psi_{xx}(\omega)$

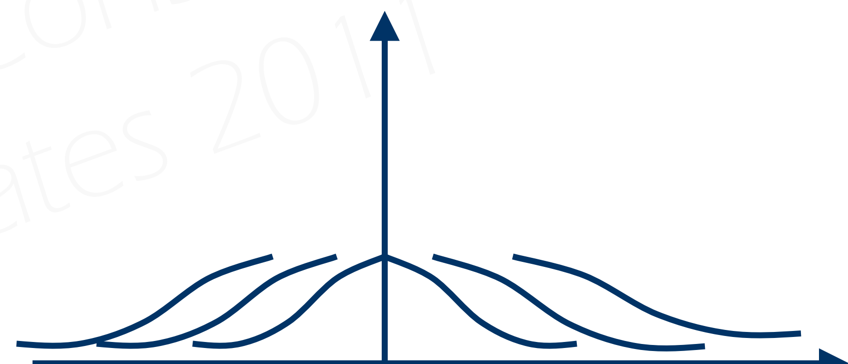
White Noise

- Define white noise the limit of an exponential process as
  - Correlation time $= \dfrac{1}{\alpha} \rightarrow 0$
  - $\Psi(0) = \dfrac{2\sigma^2}{\alpha} =$ constant
  - In the limit









  - Comments
    - White noise variance $\Psi(0) = \infty$
    - Parameter characterizing white noise $= \dfrac{2\sigma^2}{\alpha}$ is not dimensionally the same as σ^2
    - True white noise does not exist in nature due to infinite variance

White Noise

- Question: If white noise does not exist in nature, then what good is it ?
- Answer:
  1. In many cases, it can be used to approximate the real world.
  2. Numerous additional processes can be generated as

     "Solutions to linear differential equations
     driven by white noise"

     "Responses to linear shaping filters
     driven by white noise"

     Two ways of saying the same thing.

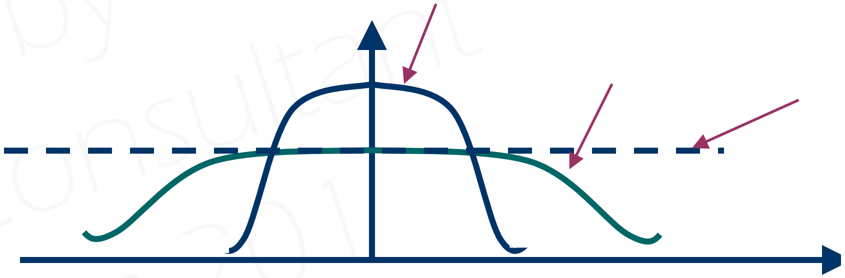
- We will make extensive use of item (2) throughout the remainder of this section.
- Brief examples of (1) and (2) follow.

White Noise

- Example: Use of white noise as a simplifying approximation

$$w \longrightarrow \boxed{G(S)} \overset{x}{\longrightarrow}$$

  - Given: $G(S), \ \Psi_w(\omega) = \dfrac{2\sigma^2\alpha}{\alpha^2 + \omega^2}$
  - Find: $\Psi_x(\omega)$



$$\Psi_x(\omega) = |G(j\omega)|^2 \ \Psi_w(\omega)$$

$$= |G(j\omega)|^2 \ \frac{2\sigma^2\alpha}{\alpha^2 + \omega^2}$$

approximating $\Psi_w(\omega)$ with

$$= |G(j\omega)|^2 \ \frac{2\sigma^2}{\alpha}$$

$$= |G(j\omega)|^2 \ \Psi_w$$

Approximation works because

$\Psi_w(\omega) \cong \Psi_w$ for low ω

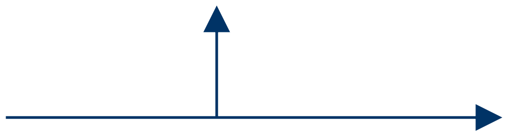
$|G(j\omega)| \cong 0$ for high ω

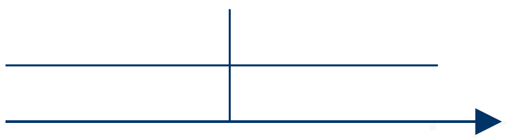
where $\Psi_w(\omega) \neq \Psi_w$

$\underbrace{|G(j\omega)|^2}$ $\underbrace{\text{high } \omega}$

Example

- Generating a random walk as integral of white noise





$$x(t) = \int_0^t w(\tau)\, d\tau$$

$$E\{x(t)\} = E\left\{\int_0^t w(\tau)\, d\tau\right\} = \int_0^t E\{w(\tau)\, d\tau\} = 0$$

$$E\{x(t_1)\, x(t_2)\} = E\left\{\int_0^{t_1}\int_0^{t_2} [w(\tau_1)\, w(\tau_2)]\, d\tau_1\, d\tau_2\right\}$$

$$= \int_0^{t_1}\int_0^{t_2} E\{w(\tau_1)\, w(\tau_2)\, d\tau_1\, d\tau_2\}$$

$$= \int_0^{t_1}\int_0^{t_2} \sigma^2 \delta(\tau_2 - \tau_1)\, d\tau_2\, d\tau_1$$

$= \sigma^2$ minimum $\{t_1, t_2\}$

If $w(t)$ is a white noise process with **parameter** σ_x^2

$$x(t) = \int_0^t w(\tau)\, d\tau$$

Then $x(t)$ is a random walk σ^2 process with **variance growth rate,** i.e., $\sigma_{x(t)}^2 = \sigma^2 t$

- We have **generated** a random walk process using white noise.
- **We will make extensive use of this concept.**

Random Walk

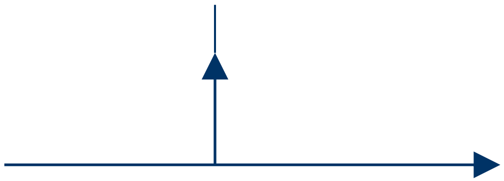
- A process $x(t)$ is a random walk if it has a zero mean and an autocorrelation function of the form

$$E\{x(t_1)x(t_2)\} = \sigma^2 \quad \text{minimum} \quad \{t_1, \; t_2\}$$

  - For $t_1 = t_2 = t$, we have $E\{x^2(t)\} = \sigma_x^2(t) = \sigma^2 t$.
  - Note
    - The variance of $x(t)$ grows linearly with time
    - $x(t)$ is non-stationary
  - Caution
    - The parameter σ^2 has units of x^2 per unit time.
    - σ^2 is not a variance, but a **variance growth rate**.

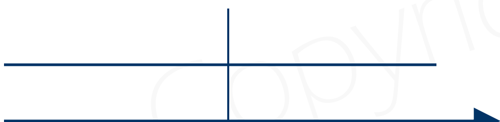
Example

- Generating an exponentially correlated process as the solution to a differential equation driven by white noise



generated as solution
to differential equation
driven by white noise

Shaping Filter
Representation



- PSD for x

Remember for exponentially correlated noise

Compare

Hence, $x(t)$ is exponentially correlated with

$$\Psi_x(\omega) = |G(j\omega)|^2 \; \Psi_w(\omega)$$

$$\Psi_e(\tau) = \sigma_e^2 \, e^{-\beta|\tau|}$$

$$\Psi_e(\omega) = \frac{2\,\sigma_e^2\,\beta}{\omega^2+\beta^2}$$

corr. time

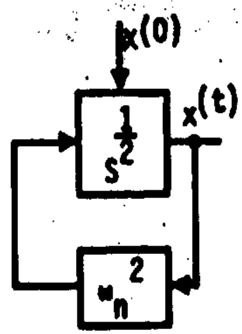
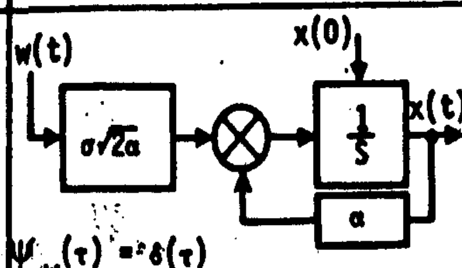
$$\sigma_w^2 = \frac{2\,\frac{\sigma_w^2}{2\alpha}\,\alpha}{\alpha^2}$$

Previous Result

- Reworded slightly:
  - If we wish to be an exponentially correlated process with **given** variance σ_x^2 and correlation time, then will have these properties.
  - A steady state solution to
  - where $w(t)$ is a white noise process with

$$\Psi_w(\tau) = 2\,\sigma_x^2\alpha\ \ \delta(\tau) \quad ; \quad \Psi_w(\omega) = 2\,\sigma_x^2\alpha$$

Error Modeling

	PSD AUTOCORRELATION FUNCTION	SHAPING FILTER	STATE SPACE FORMULATION
WHITE NOISE	$\Psi_w(\tau)=\sigma^2\delta(\tau)$ $\Psi_w(\omega)=\sigma^2$		
RANDOM WALK			$\dot{x}=w(t)$ $\sigma_x^2(0)=0$
RANDOM CONSTANT	$\Psi_x(\tau)=\sigma^2$ $\Psi_x(\omega)=2\pi\sigma^2\delta(\omega)$		$\dot{x}=0$ $\sigma_x^2(0)=\sigma^2$
SINUSOID	$\Psi_x(\tau)=\sigma^2\cos\omega_0\tau$ $\Psi_x(\omega)=\pi\sigma^2\left[\delta(\omega-\omega_0)+\delta(\omega+\omega_0)\right]$		$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix}=\begin{bmatrix}0 & 1\\ -\omega_0^2 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}$ (1) $P(0)=\begin{bmatrix}\sigma^2 & 0\\ 0 & 0\end{bmatrix}$ (2) $x_1(t)$ GENERATED IN ACCORDANCE WITH (1) AND (2) WILL HAVE PSD AND AUTOCORRELATION FUNCTION IN LEFT HAND COLUMN
EXPONENTIALLY CORRELATED NOISE (1ST ORDER MARKOV)	$\Psi_x(\tau)=\sigma^2 e^{-\alpha\lvert\tau\rvert}$ $\Psi_x(\omega)=\frac{2\sigma^2\alpha}{\omega^2+\alpha^2}$ $\frac{1}{\alpha}$ = CORRELATION TIME		$\dot{x}=-\alpha x+\sigma\sqrt{2\alpha}\,w$ $\sigma_x^2(0)=\sigma^2$

GNSS Clock Modeling and Corrections

- On board clock errors
  - Space vehicle (SV) time
    - Timing of the signal transmission from each satellite
    - Directly controlled by its own atomic clock with NO corrections applied
  - GPS time
    - Highly accurate
    - However, errors can be large enough to require corrections
    - Difficult to directly synchronize clocks in all the satellites
    - Instead, clocks allowed some degree of relative drift estimated by ground station observations and used to generate clock correction data in GPS navigation message
    - SV time is corrected using this data, and result is called “GPS time”

Time Calculations (GPS book, p. 64)

(computed)

The user should correct the time received from the space vehicle in seconds with the equation below :

(From Subframe 1)

$$t = t_{SV} - \Delta\left(t_{SV}\right)_{L_1} \qquad (3.8)$$

$$\left(\Delta t_{SV}\right)_{L_1} = \Delta t_{SV} - \tau_{GD}$$

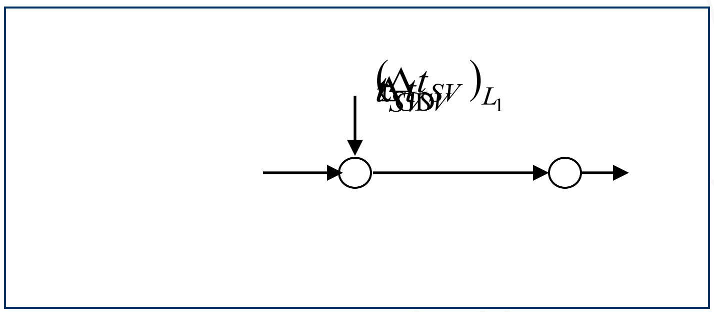
where

t = GPS system time (seconds) , τ_{GD} = Differential bias provided on subframe 1

t_{SV} = effective SV PRN code phase time at message transmission time (seconds)

(Time from pseudorange time tagged)

Δt_{SV} = SV PRN code phase time offset (seconds)



The SV PRN code phase offset is given by

$$\Delta t_{SV} = a_0 + a_1\left(t - t_{oc}\right) + a_2\left(t - t_{oc}\right)^2 + \Delta t_r \qquad (3.9)$$

$$= a_{f0} + a_{f1}\left(t - t_{OC}\right) + a_{f2_{L_1}}\left(t - t_{OC}\right)$$

where

a_{f0}, a_{f1}, and a_{f2} = the polynomial coefficients given in the ephemeris data file

t_{oc} = the clock data reference time (seconds)

Δt_r = the relativistic correction term (seconds) given by

$$\Delta t_r = F\, e \sqrt{A}\, \sin E_k \qquad (3.10)$$

In equation (A-3), F is a constant whose value is

$$F = \frac{-2\sqrt{\mu}}{c^2} = -4.442807633 \times 10^{-10} \quad \mathrm{sec}/\mathit{meter}^{1/2} \qquad (3.11)$$

where

$$c = 2.99792458 \times 10^8 \quad \mathit{meters}/\mathrm{sec} = \textit{speed of light}$$

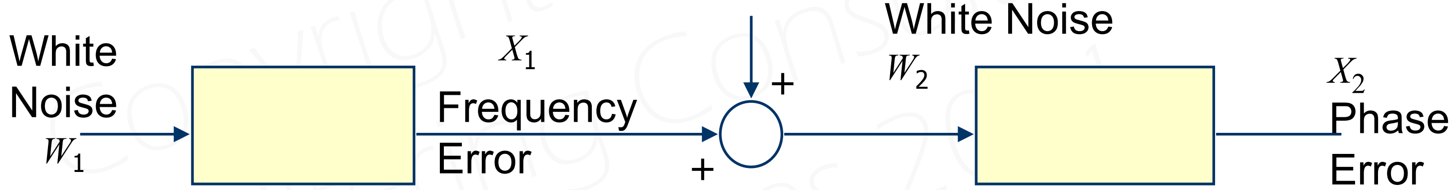
GNSS Receiver Clock Modeling

- Receiver clocks
  - Relatively inexpensive, stable over periods of time (0-10S)
  - Works well for GNSS receiver applications IF receiver can use the timing information from hyperaccurate GNSS satellite clocks to maintain required long term stability and accuracy of receiver clocks
- Clock phase and frequency tracking
  - Most common implementation
    - Uses 2-state random process model
    - Keeps the receiver clock synchronized to GNSS satellite clocks
    - Kalman filter with two state variables

Receiver Crystal Clock Modeling

- Two errors due to clock
  - Clock bias
  - Clock drift

- Continuous domain



(Drift)

$$\frac{X_2}{W_2 + X_1} = \frac{1}{S}$$

RANDOM WALK

$$\frac{X_1}{W_1} = \frac{1}{S}$$

$$\dot{X}_1 = W_1$$

$$\dot{X}_2 = X_1 + W_2$$

(Bias)

$$\frac{1}{S}$$

$$SX_1 = W_1$$

Clock Model (cont.)

$$\dot{X}_2 = X_1 + W_2$$

$$\dot{X}_1 = W_1$$

$$\begin{bmatrix} \dot{X}_2 \\ \dot{X}_1 \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}}_{} \begin{bmatrix} X_2 \\ X_1 \end{bmatrix} + \begin{bmatrix} W_2 \\ W_1 \end{bmatrix}$$

– Need PSD (power spectral density of W_1, W_2)

$$\phi = e^{F\Delta t} = I + F\,\Delta t = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}$$

– Discretize with back difference

$$\begin{bmatrix} X_2^K \\ X_1^K \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_2^{K-1} \\ X_2^{K-1} \end{bmatrix} + \begin{bmatrix} W_2^{K-1} \\ W_1^{K-1} \end{bmatrix}$$

sampling time Δt sec.

Process Noise Covariance for Receiver Clocks

$$Qt = \begin{bmatrix} qb & 0 \\ 0 & qd \end{bmatrix} \quad , \quad Qt = E\left[W(t)W^{\mathrm{T}}(t)\right] \quad , \quad W(t) = \begin{bmatrix} W_2(t) \\ W_1(t) \end{bmatrix}$$

$$Q_{k-1} = \begin{bmatrix} qb\Delta t + qd\,\frac{\Delta t^3}{3} & qd\,\frac{\Delta t^2}{2} \\ qd\,\frac{\Delta t^2}{2} & qd\Delta t \end{bmatrix} \quad , \quad Q_t = \begin{bmatrix} qb(c)^2 & 0 \\ 0 & qd(c)^2 \end{bmatrix} = \begin{bmatrix} .036 & 0 \\ 0 & .09 \end{bmatrix}$$

- Clock process noise covariances (bias and drift)

$$Q_{k-1} = \begin{bmatrix} qb(c)^2\,\Delta t + qd(c)^2\,\frac{\Delta t^3}{3} & qd(c)^2\,\frac{\Delta t^2}{2} \\ qd(c)^2\,\frac{\Delta t^2}{2} & qd(c)^2\,\Delta t \end{bmatrix} = \begin{bmatrix} .085 & .07 \\ .07 & .014 \end{bmatrix}$$

$$qb = \text{spectral amplitude} = 0.4\left(10^{-18}\right)\text{sec} \sim \frac{h_0}{2}$$

$$qd = \text{spectral amplitude} = 1.58\left(10^{-18}\right)\text{sec}^{-1} \sim 2\pi^2 h_{-2}$$

$$h_0 = 1.8 \times 10^{-19}, \quad h_{-2} = 3.8 \times 10^{-21}$$

- From Allan variance plot with asymptotes for a typical crystal oscillator
- Reference page 472, KF book

Process Noise Covariance for Receiver Clocks (cont.)

- Frequency drift variance

$$qd(c)^2 = .09m^2/s^3$$

  - Value depends primarily on
    - Quality of quartz crystals
    - Its temperature control
    - Stability of its associated control electronics
- Phase noise variance

$$qb(c)^2 = .036m^2/s$$

  - Value depends more on the electronics
    - c = speed of light = 3×10^8 m/s

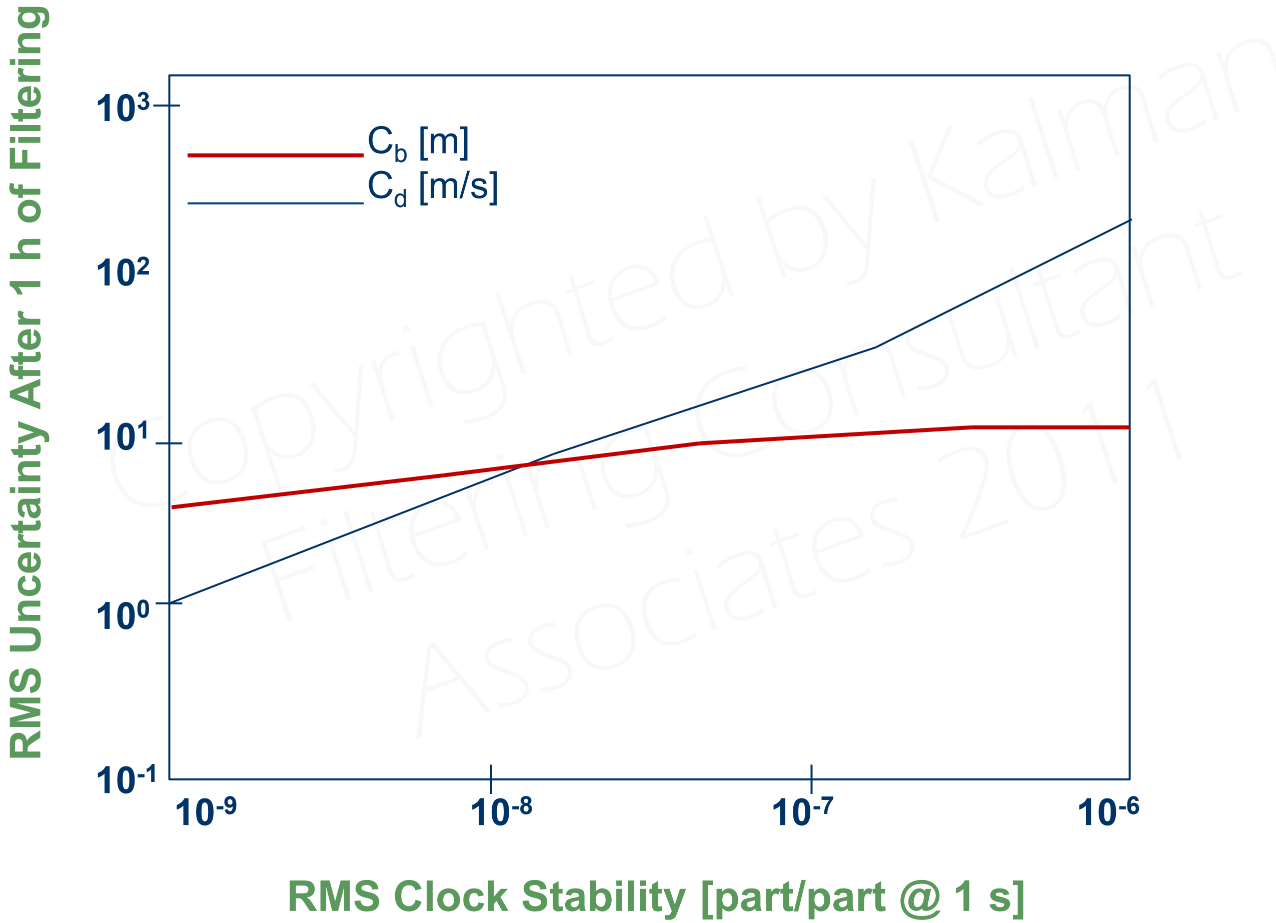
GNSS Receiver Clock Modeling

- Notes
  - Zero mean white noise processes $W_1(t)$ and $W_2(t)$ are uncorrelated
  - FLICKER noise
    - This model is a short term approximation of what is called "flicker" noise in clocks
    - Power Spectral Density (PSD) of clicker noise as a function of frequency falls off at 1/*f* .
    - Behavior cannot be modeled exactly by linear stochastic differential equations
  - Real clock drift characteristics
    - Studied extensively
    - ALLAN variance plots depict the amount of RMS drift that occurs over specified period Δt
    - Reference: J. A. Barnes, "Models for the Interpretation of Frequency Stability Measurement," NBS Tech. Note 683, Boulder CO, August 1976

Clock Estimation

- Clock estimation uncertainties vs. clock stability
  - For stationary receiver with good satellite geometries
  - Under such ideal conditions, clock stability does not severely compromise location uncertainty
  - But it does compromise clock synchronization (frequency tracking)
  - Tends to corrupt the navigation solution
  - See plot

Estimation Uncertainties vs. Stability



GNSS Measurement Models

General Measurement Models

- Let measurement model be non-linear

$$Z_k = h(x_k \, , k)$$

- Expand this L.H.S. in Taylor series about some X_k^{NOM}

$$X_k = \left[x_k^1 \; x_k^2 \; x_k^3 x_k^4 x_k^5 \right]$$

$$Z_k = h(x_k, k) = h(x_k^{\text{NOM}}, k) + \left. \frac{\partial h(x,k)}{\partial x} \right|_{x = x_k^{\text{NOM}}} \delta x_k + H.O.T.$$

$$\delta x_k = X_k - X_k^{\text{NOM}}$$

$$\delta z_k = h(x_k, k) - h(X_k^{\text{NOM}}, k)$$

General Measurement Models

- Equation becomes

$$\delta z_k = \left. \frac{\partial h(x,k)}{\partial x} \right|_{X = X_k^{\text{NOM}}} \delta x_k$$

$$\delta z_k = H_k^{[1]} \delta x_k$$

GNSS Measurement Models

- Measured pseudorange

$$\rho = \rho_r + \beta_\rho + v_\rho + Cb$$

where

$$\rho_r = \sqrt{(x - X)^2 + (y - Y)^2 (z - Z)^2}$$

X, Y, Z user position (unknown)

x, y, z satellite position (known)

$\beta_\rho =$ time correlated errors

$v_\rho =$ measurement noise

Expand $\rho(X, Y, Z)$ about $\underbrace{X_0, Y_0, Z_0}_{\text{approximate position of user}}$

in Taylor series

$$\rho(X, Y, Z) = \rho_r(X_0, Y_0, Z_0) + \left.\frac{\partial \rho_r}{\partial \mathrm{X}}\right|_{\mathrm{X} = X_0, Y_0, Z_0} \delta x + \beta_\rho + v_\rho$$

GNSS Observation Equation

- Linearization

$$\delta x = X - X_0$$

$$\delta y = Y - Y_0$$

$$\delta z = Z - Z_0$$

where

$$\frac{\partial \rho_r^i}{\partial X} = \left. \frac{-(x_i - X)}{\sqrt{(x_i - X)^2 + (y_i - Y)^2 + (z_i - Z)^2}} \right|_{X_0, Y_0, Z_0}$$

$$\delta z_\rho = \rho(X, Y, Z) - \rho_r(X_0, Y_0, Z_0) = \left. \frac{\partial \rho_r}{\partial X} \right| \delta x + \nu_\rho$$

Linearization (cont.)

$i = 1$

$$\frac{\partial \rho_r^1}{\partial X} = \frac{-(x_i - X_0)}{\sqrt{(x_i - X_0)^2 + (y_i - Y_0)^2 + (z_i - Z_0)^2}}$$

$$\frac{\partial \rho_r^1}{\partial Y} = \frac{-(y_i - Y_0)}{\sqrt{(x_i - X_0)^2 + (y_i - Y_0)^2 + (z_i - Z_0)^2}}$$

$$\frac{\partial \rho_r^1}{\partial Z} = \frac{-(z_i - Z_0)}{\sqrt{(x_i - X_0)^2 + (y_i - Y_0)^2 + (z_i - Z_0)^2}}$$

$i = 2 \quad \vdots$

$i = 3 \quad \vdots$

$i = 4 \quad \vdots$

Linearization (cont.)

$$\underbrace{\begin{bmatrix} \delta z_\rho^1 \\ \delta z_\rho^2 \\ \delta z_\rho^3 \\ \delta z_\rho^4 \end{bmatrix}}_{4\times1} = \underbrace{\begin{bmatrix} \frac{\partial \rho_r^1}{\partial x} & \frac{\partial \rho_r^1}{\partial y} & \frac{\partial \rho_r^1}{\partial z} & 1 & 0 \\ \frac{\partial \rho_r^2}{\partial x} & \frac{\partial \rho_r^2}{\partial y} & \frac{\partial \rho_r^2}{\partial z} & 1 & 0 \\ \frac{\partial \rho_r^3}{\partial x} & \frac{\partial \rho_r^3}{\partial y} & \frac{\partial \rho_r^3}{\partial z} & 1 & 0 \\ \frac{\partial \rho_r^4}{\partial x} & \frac{\partial \rho_r^4}{\partial y} & \frac{\partial \rho_r^4}{\partial z} & 1 & 0 \end{bmatrix}}_{4\times5} \underbrace{\begin{bmatrix} \delta x \\ \delta y \\ \delta z \\ Cb \\ Cd \end{bmatrix}}_{5\times1} + \underbrace{\begin{bmatrix} v_\rho^1 \\ v_\rho^2 \\ v_\rho^3 \\ v_\rho^4 \end{bmatrix}}_{4\times1}$$

$\delta x, \delta y, \delta z$ position errors, Cb=clock bias

Cd= clock drift

$$\overset{4\times1}{Z_k} = \overset{4\times5}{H}\,\overset{5\times1}{X_k} + \overset{4\times1}{v_k}$$

Linearized Observation Model

- 8 States

$$\underbrace{\begin{bmatrix} \delta z_\rho^1 \\ \delta z_\rho^2 \\ \delta z_\rho^3 \\ \delta z_\rho^4 \end{bmatrix}}_{4\times 1} = \underbrace{\begin{bmatrix} \frac{\partial \rho_r^1}{\partial x} & 0 & \frac{\partial \rho_r^1}{\partial y} & 0 & \frac{\partial \rho_r^1}{\partial z} & 0 & 1 & 0 \\ \frac{\partial \rho_r^2}{\partial x} & 0 & \frac{\partial \rho_r^2}{\partial y} & 0 & \frac{\partial \rho_r^2}{\partial z} & 0 & 1 & 0 \\ \frac{\partial \rho_r^3}{\partial x} & 0 & \frac{\partial \rho_r^3}{\partial y} & 0 & \frac{\partial \rho_r^3}{\partial z} & 0 & 1 & 0 \\ \frac{\partial \rho_r^4}{\partial x} & 0 & \frac{\partial \rho_r^4}{\partial y} & 0 & \frac{\partial \rho_r^4}{\partial z} & 0 & 1 & 0 \end{bmatrix}}_{4\times 8} \underbrace{\begin{bmatrix} \delta x_1 \\ \delta x_2 \\ \delta x_3 \\ \delta x_4 \\ \delta x_5 \\ \delta x_6 \\ \delta x_7 \\ \delta x_8 \end{bmatrix}}_{8\times 1} + \underbrace{\begin{bmatrix} v_\rho^1 \\ v_\rho^2 \\ v_\rho^3 \\ v_\rho^4 \end{bmatrix}}_{4\times 1}$$

$\delta x_1 =$ x Position error
$\delta x_2 =$ x Velocity error
$\delta x_3 =$ y Position error
$\delta x_4 =$ y Velocity error
$\delta x_5 =$ z Position error
$\delta x_6 =$ z Velocity error
$\delta x_7 =$ Clock bias error
$\delta x_8 =$ Clock drift error

System Dynamics Models

Modeling (cont.)

4 STATES	5 STATES	8 STATES	11 STATES
X	X	X	X
Y	Y	Y	Y
Z	Z	Z	Z
Cb	Cb	Cb	Cb
	$\dot{C}b$	$\dot{C}b$	$\dot{C}b$
		$\dot{X}$	$\dot{X}$
		$\dot{Y}$	$\dot{Y}$
		$\dot{Z}$	$\dot{Z}$
			$\ddot{X}$
			$\ddot{Y}$
			$\ddot{Z}$
STATIONARY	STATIONARY	SLOW DYNAMICS	FAST DYNAMICS

8 States - Slow Dynamics

- Discrete Process Model--Car or boat

$$\overset{8\times1}{X_k} = \overset{8\times8}{\Phi}\,\overset{8\times1}{X}_{k-1} + \overset{8\times1}{W}_{k-1}$$

$$\overset{8\times1}{X_k} = \begin{bmatrix} X & \dot{X} & Y & \dot{Y} & Z & \dot{Z} & Cb & Cd \\ X_k^1 & X_k^2 & X_k^3 & X_k^4 & X_k^5 & X_k^6 & X_k^7 & X_k^8 \end{bmatrix}^T$$

$$X_k = \begin{bmatrix} 1 & \Delta t & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & \Delta t & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \Delta t & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \overset{8\times1}{X}_{k-1} + \overset{8\times1}{W}_{k-1}$$

Linearized Observation Model

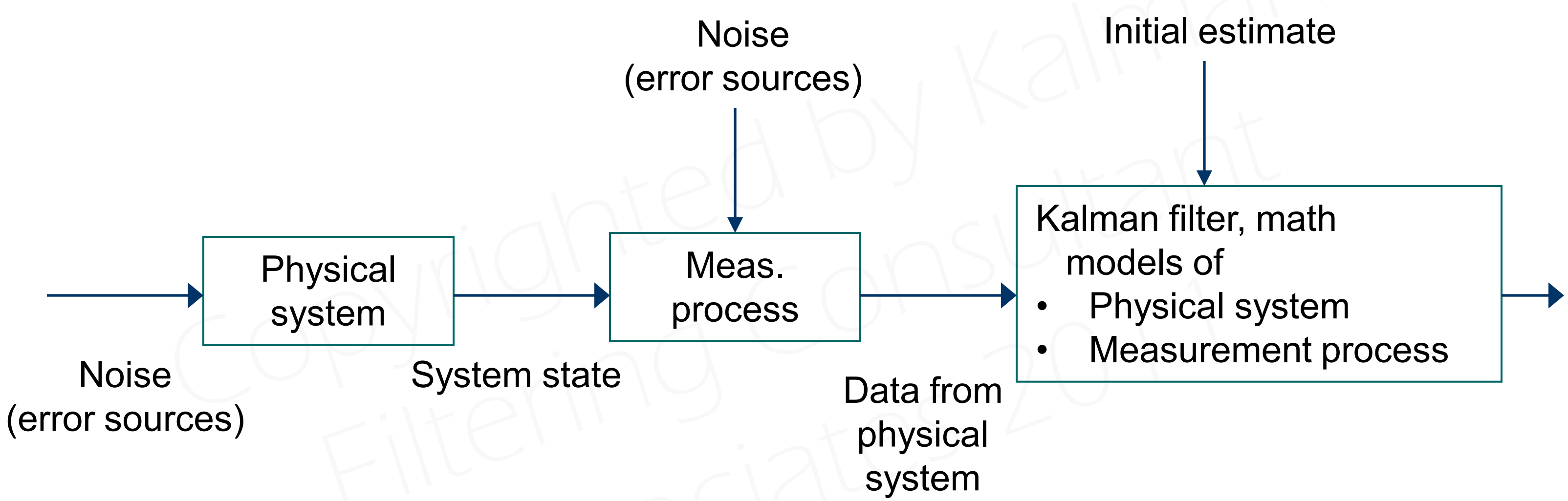
- For 8 states

$$\underbrace{\begin{bmatrix} \delta z_\rho^1 \\ \delta z_\rho^2 \\ \delta z_\rho^3 \\ \delta z_\rho^4 \end{bmatrix}}_{4\times1} = \underbrace{\begin{bmatrix} \frac{\partial \rho_r^1}{\partial x} & 0 & \frac{\partial \rho_r^1}{\partial y} & 0 & \frac{\partial \rho_r^1}{\partial z} & 0 & 1 & 0 \\ \frac{\partial \rho_r^2}{\partial x} & 0 & \frac{\partial \rho_r^2}{\partial y} & 0 & \frac{\partial \rho_r^2}{\partial z} & 0 & 1 & 0 \\ \frac{\partial \rho_r^3}{\partial x} & 0 & \frac{\partial \rho_r^3}{\partial y} & 0 & \frac{\partial \rho_r^3}{\partial z} & 0 & 1 & 0 \\ \frac{\partial \rho_r^4}{\partial x} & 0 & \frac{\partial \rho_r^4}{\partial y} & 0 & \frac{\partial \rho_r^4}{\partial z} & 0 & 1 & 0 \end{bmatrix}}_{4\times8} \underbrace{\begin{bmatrix} \delta x_1 \\ \delta x_2 \\ \delta x_3 \\ \delta x_4 \\ \delta x_5 \\ \delta x_6 \\ \delta x_7 \\ \delta x_8 \end{bmatrix}}_{8\times1} + \underbrace{\begin{bmatrix} v_\rho^1 \\ v_\rho^2 \\ v_\rho^3 \\ v_\rho^4 \end{bmatrix}}_{4\times1}$$

$$\overset{4\times1}{Z_k} = \overset{4\times8}{H}\ \overset{8\times1}{X_k} + \overset{4\times1}{\nu_k}$$

Kalman Filtering Problem

- Top level sketch



Discrete Linear Kalman Estimator

System model: $\overset{n\times1}{x_k} = \overset{n\times n}{\Phi_{k-1}}\ \overset{n\times1}{x_{k-1}} + \overset{n\times1}{w_{k-1}}$, $w_k \sim \overset{n\times n}{N\left(0, Q_k\right)}$

Measurement model: $\overset{\ell\times1}{z_k} = \overset{\ell\times n}{H_k}\ \overset{n\times1}{x_k} + \overset{\ell\times1}{v_k}$, $v_k \sim \overset{\ell\times\ell}{N\left(0, R_k\right)}$ (white)

Initial conditions: $E\left\langle x_0 \right\rangle = \hat{x}_0$, $E\left\langle \tilde{x}_0\ \tilde{x}_0^T \right\rangle = \overset{n\times n}{P_0}(+)$

Other assumptions: $E\left\langle w_k\ v_j^T \right\rangle = 0$ for all K, j

State estimate extrapolation: $\hat{x}_k(-) = \Phi_{k-1}\ \hat{x}_{k-1}(+)$

Error covariance extrapolation: $P_k(-) = \Phi_{k-1}\ P_{k-1}(+)\,\Phi_{k-1}^T + Q_{k-1}$

State estimate update: $\hat{x}_k(+) = \hat{x}_k(-) + \overline{K}_k\left[z_k - H_k\hat{x}_k(-)\right]$

Error covariance update: $P_k(+) = \left[I - \overline{K}_k\ H_k\ \right]\ P_k(-)$

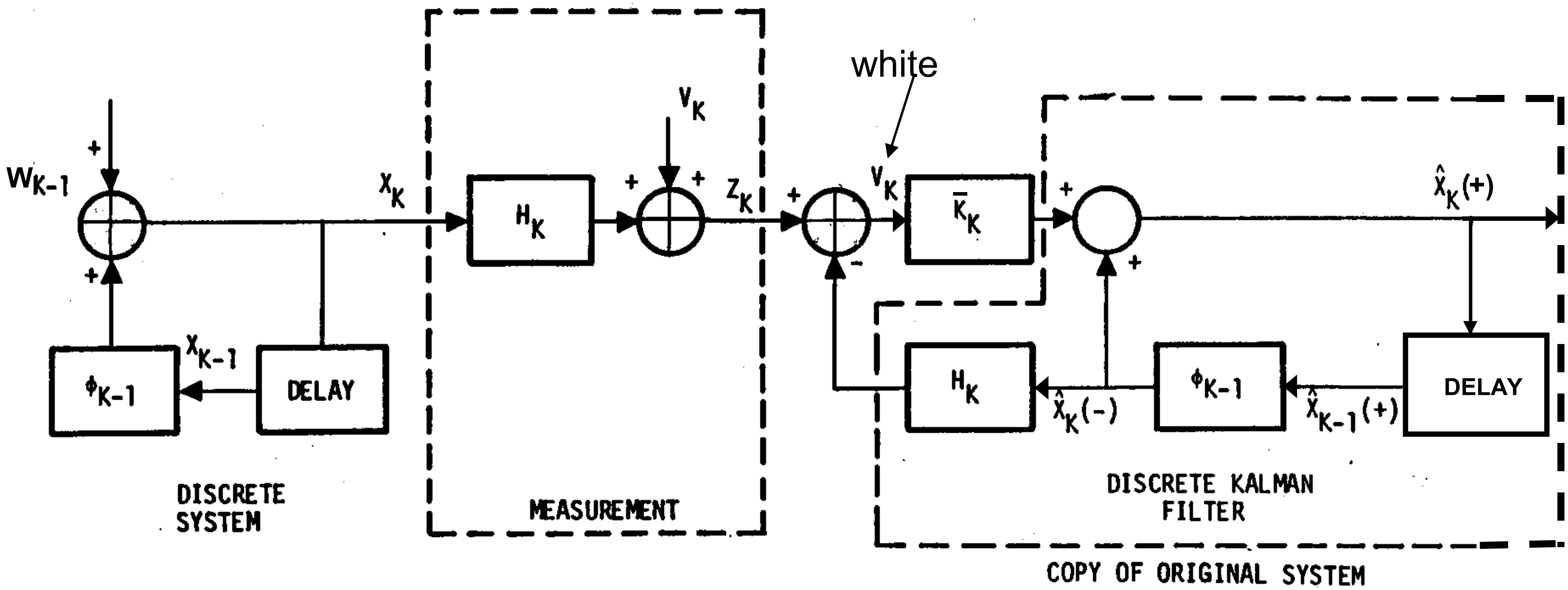
Kalman gain matrix: $\overline{K}_k = P_k(-)\,H_k^T\left[H_k\ P_k(-)H_k^T + R_k\right]^{-1}$

Q_k = cov. of process noise w_k $\quad$ $P_k(+)$ = error cov. of states (*a posteriori*)

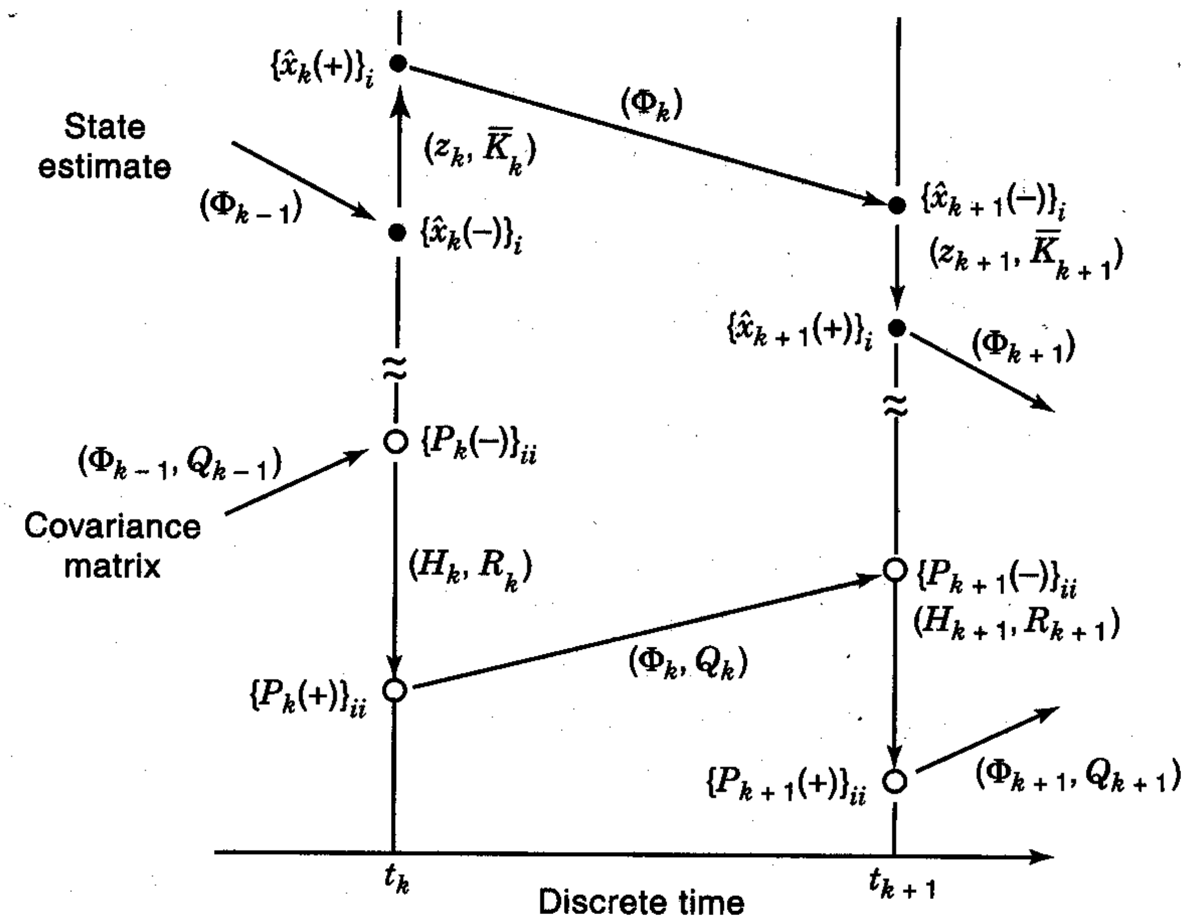
R_k = cov. of process noise v_k $\quad$ $P_k(-)$ = error cov. of states (*a priori*)

Block Diagram

- System, measurement models & discrete Kalman filter (one step prediction)



Representative Sequence of Values of Filter Variables in Discrete Time



Representative sequence of values of filter variables in discrete time.

Block Diagram (cont.)

- Comments
  - It is important to notice that $\bar{K}_k$ and $P_k(-), P_k(+)$ are independent of observations (measurements)
  - There are simpler forms of $\bar{K}_k$ and $P_k(+)$
  - Procedure of computations

1) Compute $P_k(-)$ with $P_{k-1}(+), \Phi_{k-1}, Q_{k-1}$ (given)

2) Compute $\bar{K}_k$ with $P_k(-), H_k, R_k$ (known)

3) Compute $P_k(+)$ with $\bar{K}_k(-), H_k, P_k(-)$ (known)

4) Compute $\hat{x}_k(+)$ with $\hat{x}_k(-), \bar{K}_k, z_k(-)$ (known)

Continuous Kalman Filter

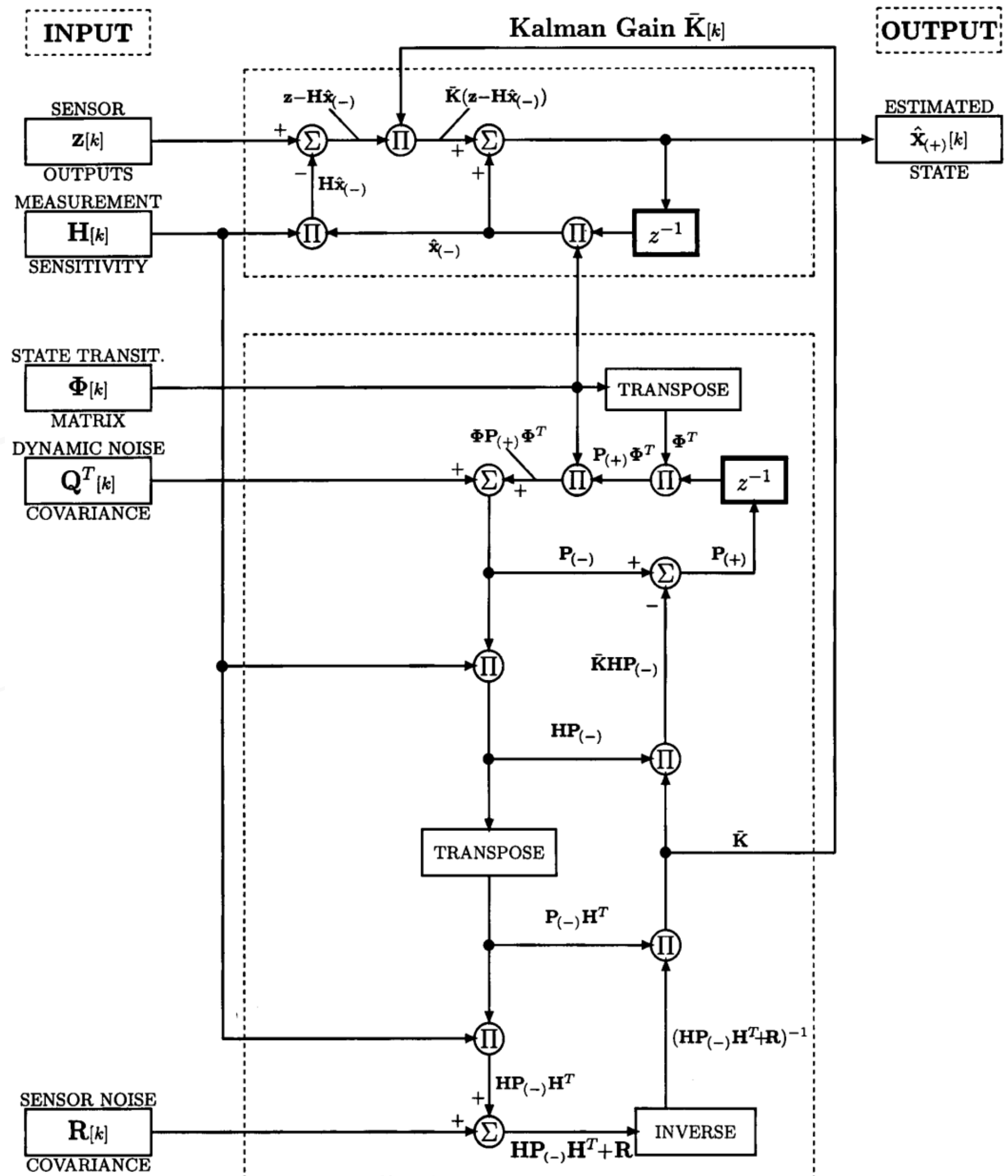
- Can be derived by using the orthogonality principal
  - System model: $\dot{x}(t) = F(t)\,x(t) + G(t)\,w(t) \quad , \quad w(t) \sim N(0, Q(t)\,)$
  - Measurement model: $z(t) = H(t)\,x(t) + v(t) \quad , \quad v(t) \sim N(0, R(t)\,)$
  - Initial conditions: $E\left\langle x_0 \right\rangle = \hat{x}_0 \quad , \quad E\left\langle \tilde{x}_0\, \tilde{x}_0^T \right\rangle = P_0$
  - Other assumptions: $R^{-1}(t)$ exists , $E\left[w(t)\,v^T(\eta)\right] = 0$
  - State Estimate: $\dot{\hat{x}}(t) = F(t)\,\hat{x}(t) + \overline{K}(t)\left[z(t) - H(t)\,\hat{x}(t)\right] \quad , \quad \hat{x}(0) = \hat{x}_0$
- Error covariance propagation

$$\dot{p}(t) = F(t)\,P(t) + P(t)\,F^T(t) + G(t)\,Q(t)\,G^T(t) - \overline{K}(t)\,R(t)\,\overline{K}^T(t) \quad , \quad P(0) = P_0$$

- Kalman gain matrix

$$\overline{K}(t) = P(t)\; H^T(t) R^{-1}(t)$$

Kalman Filter Data Flow



Kalman Filter Examples

- Let the system dynamics and observations be given by the following equations:

$$
\begin{aligned}
x_k &= x_{k-1} + w_{k-1} \\
z_k &= x_k + v_k \\
\mathrm{E}\langle v_k \rangle &= \mathrm{E}\, w_k = 0 \\
\mathrm{E}\langle v_{k_1} v_{k_2} \rangle &= 2\Delta(k_2 - k_1) \\
\mathrm{E}\langle w_{k_1} w_{k_2} \rangle &= \Delta(k_2 - k_1) \\
z_1 &= 2 \\
z_2 &= 3 \\
\mathrm{E}\langle x(0) \rangle &= \hat{x}_0 = 1 \\
\mathrm{E}\left\langle [x(0) - \hat{x}_0]\,[x(0) - \hat{x}_0]^T \right\rangle &= P_0 = 10.
\end{aligned}
$$

- The objective is to find $\hat{x}_3$ and the steady state covariance matrix P_∞ . One can use the equations on page 23 with

$$\Phi = 1 = H, \quad Q = 1, \quad R = 2$$

Example (cont.)

- For which

$$\boxed{P_k(-) = P_{k-1}^{(+)} + 1}$$

$$\bar{K}_k = \frac{P_k(-)}{P_k(-) + 2} = \frac{P_{k-1}^{(+)} + 1}{P_{k-1}^{(+)} + 3}$$

$$P_k(+) = \left[1 - \frac{P_{(k-1)}^{(+)} + 1}{P_{(k-1)}^{(+)} + 3}\right]\left(P_{k-1}^{(+)} + 1\right)$$

$$\boxed{P_k(+) = \frac{2(P_{k-1}^{(+)}+1)}{P_{k-1}^{(+)}+3}}$$

$$\boxed{\hat{x}_k(+) = \hat{x}_{k-1}^{(+)} + \bar{K}_k(z_k - \hat{x}_{k-1}^{(+)})}$$

Example (conc.)

- Let

$$
\begin{aligned}
P_k(+) = P_{k-1}(+) &= P \quad \text{(steady state cov.)} \\
P &= \frac{2(P+1)}{P+3} \\
P^2 + P - 2 &= 0
\end{aligned}
$$

$P = 1$, Positive definite solution

For $k = 1$

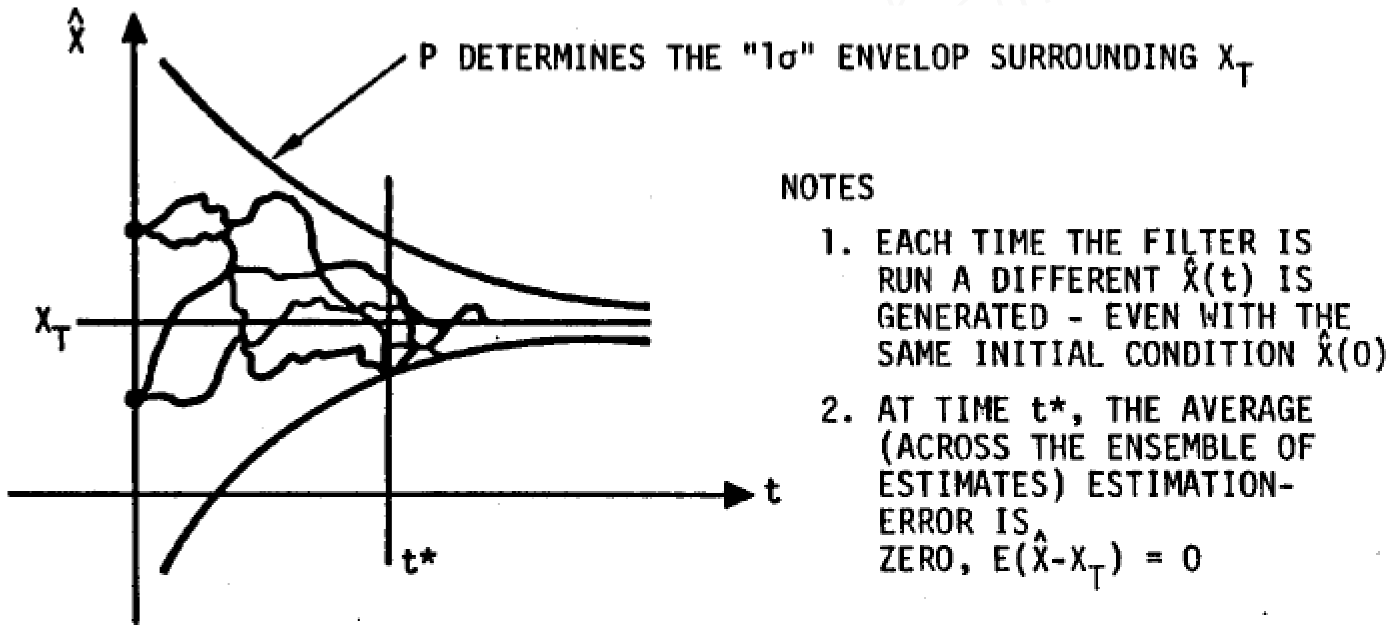
$$
\hat{x}_1(+) = \hat{x}_0 + \frac{P_0 + 1}{P_0 + 3}(2 - \hat{x}_0) = 1 + \frac{11}{13}(2 - 1) = \frac{24}{13}
$$

- Following is a table for the various values of the Kalman filter

k	$P_k(-)$	$P_k(+)$	$\overline{K}_k$	$\hat{x}_k(+)$
1	11	$\frac{22}{13}$	$\frac{11}{13}$	$\frac{24}{13}$
2	$\frac{35}{13}$	$\frac{70}{61}$	$\frac{35}{61}$	$\frac{153}{61}$

Convergence of Kalman Filter

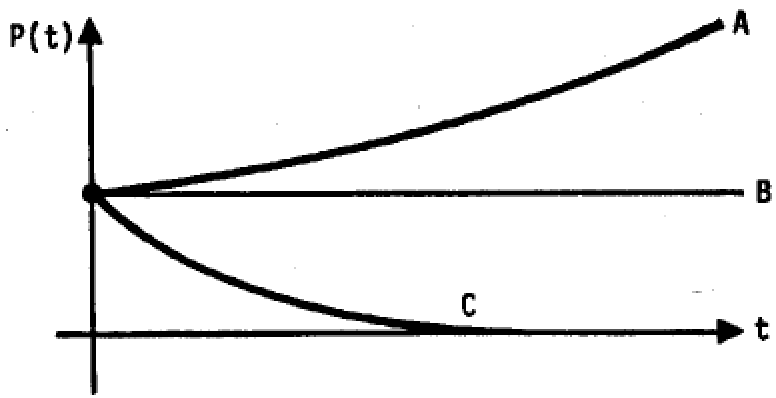
- An optimal filter converges if $\lim_{t=\infty} (\text{Trace } P) = 0$
- Example of typical behavior:



For an optimal filter, convergence or lack of convergence is correctly and fully defined by $P(t)$

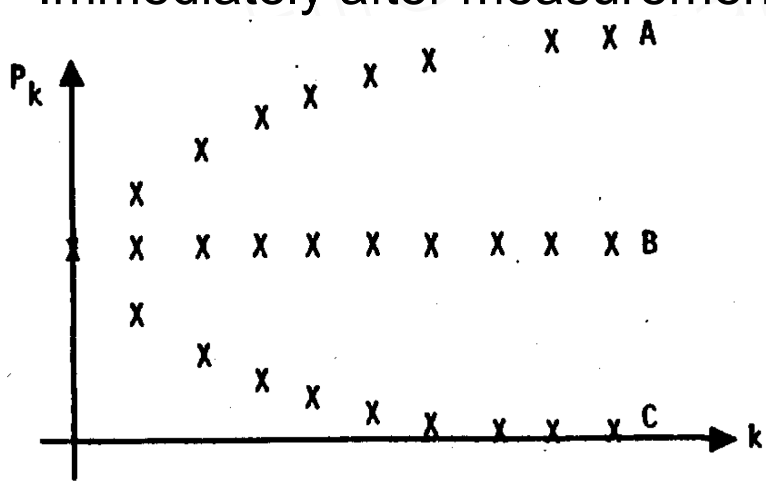
Convergence (cont.)

- Typical behavioral patterns for $P(t)$
  - Between measurement samples



CASE	TYPICAL CAUSES
A	SYSTEM UNSTABLE $\dot{X} = AX \quad A > 0$ OR SYSTEM HAS DRIVING NOISE $\dot{X} = AX + W$
B	SYSTEM HAS CONSTANT STATE $\dot{X} = 0$
C	SYSTEM IS STABLE AND HAS NO DRIVING NOISE $\dot{X} = -AX \quad A > 0$

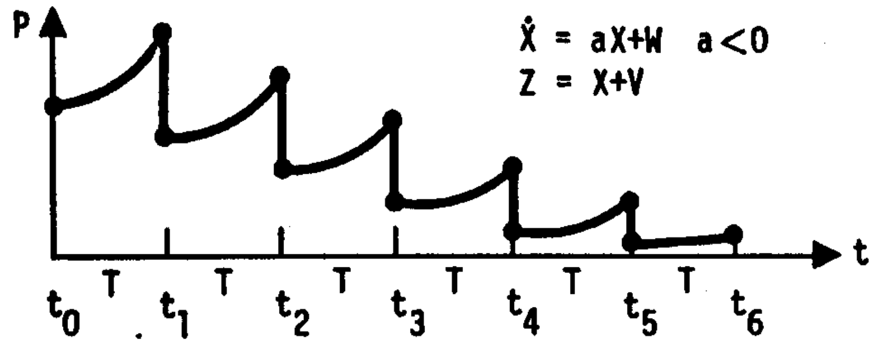
  - Immediately after measurements are processed



CASE	TYPICAL CAUSES
A	SYSTEM DRIVING NOISE AND MEASUREMENT NOISE ARE LARGE RELATIVE TO P_O
B	STATE IS UNOBSERVABLE AND UNCORRELATED WITH OTHER STATES
C	SYSTEM DRIVING NOISE AND MEASUREMENT NOISE ARE SMALL RELATIVE TO P_O

Convergence (cont.)

- An example of a combined behavior patter for $P(t)$
  - Between and at the time of measurements



- Notes
  1. Processing the measurement tends to reduce P
  2. The larger Q parameters are, the lower the overall estimation accuracy becomes
  3. System driving noise tends to increase P
  4. The damping in a stable system tends to reduce P
  5. An unstable system tends to increase P
  6. With white measurement noise, the time between samples can be shortened to reduce P
  7. The behavior of P represents a composite of all these effects and often reaches a "statistical equilibrium"

Causes, Cures of Non-convergence

- Non-convergence categories
  - Non-convergence predicted by P (optimal case)
    - As “natural behavior”
    - Due to non-observability
  - Non-convergence not predicted by P (suboptimal case)
    - Due to bad data *
    - Due to numerical problems
    - Due to mismodeling *

* Here we are caught lying to the filter

Bad Data Rejection

- Data rejection
  - Assuming that adequate knowledge of the innovations—vector $\left(Z - H\hat{X}\right)$ exists—data rejection filters can be implemented
  - For example
    - Excess amplitude

      If $\left|\left(Z - H\hat{X}\right)_{\mathrm{i}}\right| > A_{\mathrm{MAX}} \Rightarrow$ reject data
    - Excess rate (or change)

      If $\left|\left(Z - H\hat{X}\right)_{\mathrm{i+1}} - \left(Z - H\hat{X}\right)_{\mathrm{i}}\right| > \delta A_{\mathrm{MAX}} \Rightarrow$ reject data
    - Other
      - Many ingenious techniques have been used, but often depend on the specifics involved
      - For example, Chi-Squared Distribution

Chi-Squared Statistic

- Detecting anomalous sensor data
  - The Kalman gain matrix $\overline{\mathrm{K}}_k = \mathrm{P}_k(-)\mathrm{H}_k^{\mathrm{T}} \underbrace{\left(\mathrm{H}_k \mathrm{P}_k(-)\mathrm{H}_k^{\mathrm{T}} + \mathrm{R}_k\right)^{-1}}_{\mathrm{Y}_{\nu k}}$
  - includes the factor $\mathrm{Y}_{\nu k} = \left(\mathrm{H}_k \mathrm{P}_k(-)\mathrm{H}_k^{\mathrm{T}} + \mathrm{R}_k\right)^{-1}$, the information matrix of innovations. The innovations are the measurement residuals $\boldsymbol{\nu}_k = \mathrm{z}_k - \mathrm{H}_k \hat{\mathrm{x}}_k(-)$,the differences between the apparent sensor outputs and predicted sensor outputs.
  - The associated likelihood function for innovations is

$$L(\boldsymbol{\nu}_k) = \exp\left(-\frac{1}{2}\boldsymbol{\nu}_k^{\mathrm{T}} \mathrm{Y}_{\nu k} \boldsymbol{\nu}_k\right),$$

  - and the log-likelihood is $\log[\mathrm{L}(\boldsymbol{\nu}_k)] = -\boldsymbol{\nu}_k^{\mathrm{T}} \mathrm{Y}_{\nu k} \boldsymbol{\nu}_k$, which can be easily calculated.

Chi-Squared Statistics (cont.)

- Detecting Anomalous Sensor Data (conc.)
  - The equivalent statistic $$\chi^2 = \frac{\boldsymbol{v}_k^{\mathrm{T}}\, \mathrm{Y}_{vk}\, \boldsymbol{v}_k}{\ell}$$ is nonnegative with a minimum value of zero.
  - If the Kalman filter were perfectly modeled and all white-noise sources were Gaussian, this would be a chi-squared statistic with distribution.
  - An upper limit threshold value on χ^2 can be used to detect anomalous sensor data, but a practical value of that threshold should be determined by the operational values of χ^2, not the theoretical values.
  - That is, first its range of values should be determined by monitoring the system in operation, then a threshold value $\chi^2_{\max}$ chosen such that the fraction of good data rejected when $\chi^2 > \chi^2_{\max}$ will be acceptable.

Kalman Filter Engineering

- Square root filtering
  - Robust against round off
- KF implementation requirements
  - Memory & OPS
- Nuisance Variable Examples
  - Some can be ignored (at some cost)
    - Correlated noise states (e.g., S/A)
    - Anything not appearing elsewhere in model
  - Some cannot be ignored
    - Sensor biases
    - Sensor scale factors

Square Root Filtering

- Riccati equation not well conditioned for solution in finite precision
- Square root filters replace covariance matrix P by Cholesky factor C such that $CC^T = P$.
- Riccati equation reformulated (many ways) in terms of Cholesky factors is more robust against computer roundoff
- Riccati equation problems
  - Asymmetrical P
  - Negative values on diagonal of P
  - Unable to invert (HPH^T+R)
  - Complex values in P
  - Estimates diverge or fail to converge

Triangular Cholesky Factors

$$\begin{vmatrix} c_{11} & 0 & 0 \\ c_{21} & c_{22} & 0 \\ c_{31} & c_{32} & c_{33} \end{vmatrix} \begin{vmatrix} c_{11} & c_{21} & c_{31} \\ 0 & c_{22} & c_{32} \\ 0 & 0 & c_{33} \end{vmatrix} = \begin{vmatrix} p_{11} & p_{21} & p_{31} \\ p_{21} & p_{22} & p_{32} \\ p_{31} & p_{32} & c_{33} \end{vmatrix}$$

$$c_{11}{}^{2} = p_{11}$$

$$c_{11}\, c_{21} = p_{21}$$

$$c_{11}\, c_{31} = p_{31}$$

$$c_{31}{}^{2} + c_{32}{}^{2} + c_{33}{}^{2} = p_{33}$$

$$c_{21}{}^{2}\; c_{22}{}^{2} = p_{22}$$

$$c_{21}\, c_{31} \; + \; c_{22}\, c_{32} = p_{32}$$

Other Cholesky Factors

$$C\,C^{\mathrm{T}} = P$$

$$A\,A^{\mathrm{T}} = I$$

$$M = C\,A$$

$$M\,M^{\mathrm{T}} = C\,A\,A^{\mathrm{T}}$$

- Modified Cholesky factors

$$P = U\,D\,U^{\mathrm{T}}$$

D is diagonal

$$U = \begin{vmatrix} 1 & u_{12} & u_{13} \\ 0 & 1 & u_{23} \\ 0 & 0 & 1 \end{vmatrix}$$

Alternative Implementations

Matrix Format	Corrector	Predictor
Symmetric positive Def.	Kalman	Kalman
Gen. Cholesky factor	Potter	$C = \Phi\, C$
Triangular Cholesky factor	Carlson	Schmidt
Modified Cholesky factor	Bierman	Thornton

Cholesky Factors

- "Virtues"
  - Not unique—can take many forms
  - Forms related through orthogonal transformations (can be exploited)
  - Memory efficient
  - Computationally efficient
  - Better condition for inversion
  - $N^2 \to N(N+1)/2$ matrix elements
  - Condition number $10^x \to 10^{x/2}$
    - "Same performance with half the bits"
  - Non-negative definite P guaranteed
  - Symmetric P guaranteed
  - Enabled Kalman filtering applications
    - Thousands of state variables
    - Poorly conditioned problems

Square Root Riccati Equations

- Observational updates (corrector)
  - Rank-1 modifications of Cholesky factors
    - Potter, Carlson (triangular), Bierman (UD)
- Temporal updates (predictor)
  - $C_{k+1} = \Phi\, C_k$ (Potter)
  - $C_{k+1} = \left[\Phi\, C_k \;\middle|\; C_Q\right] A$ (Schmidt)
  - Modified weighted Gram-Schmidt (Thornton)

Nonlinear Kalman Filter

- Nonlinear Plant and Measurement Models

Model	Continuous Time	Discrete Time
Plant	$\dot{x} = f(x, t) + w(t)$	$x_k = f(x_{k-1}, k-1) + w_{k-1}$
Measurement	$z(t) = h(x(t), t) + v(t)$	$z_k = h(x_k, k) + v_k$
Plant noise	$E\langle w(t)\rangle = 0$	$E\langle w_k\rangle = 0$
	$E\langle w(t)w^T(s)\rangle = \delta(t-s)Q(t)$	$E\langle w_k w_i^T\rangle = \Delta(k-i)Q_k$
Measurement noise	$E\langle v(t)\rangle = 0$	$E\langle v_k\rangle = 0$
	$E\langle v(t)v^T(s)\rangle = \delta(t-s)R(t)$	$E\langle v_k v_i^T\rangle = \Delta(k-i)R_k$

- Dimensions of Vectors and Matrices in Nonlinear Model

Symbol	Dimensions	Symbol	Dimensions
x, f, w	$n \times 1$	z, h, v	$\ell \times 1$
Q	$n \times n$	R	$\ell \times \ell$
Δ, δ	Scalars		

Linearized Kalman Filter

- Partial derivatives evaluated along some "nominal trajectory" of the system.
- Used principally for covariance analysis of expected system performance, when all one has is a nominal trajectory, or set of nominal trajectories.
- Can be used for pre-computing Kalman gains, but depends on following close to nominal trajectory.
- **Reminder:**
  - Linearization is used only in the Riccati equation for computing the Kalman gains.
  - The estimated states are propagated in time by integrating the full nonlinear dynamic model.
  - The predicted measurement is calculated using the full nonlinear sensor model.

Extended Kalman Filtering

- Applies only to nonlinear problems, either nonlinear dynamics or nonlinear sensors, or both.
- All partial derivatives are evaluated at the estimated values of the state variables.
- Requires full nonlinear implementation of state dynamics and dependence of measurements on state variables.

Tables of Kalman Filter Equations

Table 1: Discrete Linearized Kalman Equations	
Nonlinear Nominal Trajectory Model	$x_k^{\text{nom}} = \phi_{k-1}\left(x_{k-1}^{\text{nom}}\right)$
Linearized Perturbed Trajectory Model	$\delta x \overset{\text{def}}{=} x - x^{\text{nom}}, \ \delta x_k \approx \frac{\partial \phi_{k-1}}{\partial x}\big|_{x=x_{k-1}^{\text{nom}}} \delta x_{k-1} + w_{k-1}$ $w_k \sim \mathcal{N}(0, Q_k)$
Nonlinear Measurement Model	$z_k = h_k(x_k) + v_k, \ v_k \sim \mathcal{N}(0, R_k)$
Linearized Approximation Equations	
Linear perturbation prediction	$\hat{\delta x}_k(-) = \Phi_{k-1}^{[1]}\, \hat{\delta x}_{k-1}(+), \ \Phi_{k-1}^{[1]} \approx \frac{\partial f_{k-1}}{\partial x}\big|_{x=_{k-1}^{\text{nom}}}$
Conditioning the predicted perturbation on the measurement	$\hat{\delta x}_k(+) = \hat{\delta x}_k(-) + \overline{K}_k\left[z_k - h_k\left(x_k^{\text{nom}}\right) - H_k^{[1]}\hat{\delta x}_k(-)\right]$ $H_k^{[1]} \approx \frac{\partial h_k}{\partial x}\big|_{x=x_k^{\text{nom}}}$
Computing the *a priori* covariance matrix	$P_k(-) = \Phi_{k-1}^{[1]} P_{k-1}(+) \Phi_{k-1}^{[1]\text{T}} + Q_{k-1}$
Computing the Kalman gain	$\overline{K}_k = P_k(-) H_k^{[1]\text{T}}\left[H_k^{[1]} P_k(-) H_k^{[1]\text{T}} + R_k\right]^{-1}$
Computing the *a posteriori* covariance matrix	$P_k(+) = \left\{I - \overline{K}_k H_k^{[1]}\right\} P_k(-)$

Tables of Kalman Filter Equations

Table 2: Discrete Extended Kalman Equations	
Nonlinear Dynamic Model	$x_k = \phi_{k-1}(x_{k-1}) + w_{k-1}\,,\ \ w_k \sim \mathcal{N}(0, Q_k)$
Nonlinear Measurement Model	$z_k = h_k(x_k) + v_k,\ \ v_k \sim \mathcal{N}(0, R_k)$
Nonlinear Implementation Equations	
Computing the predicted state estimate	$\hat{x}_k(-) = \phi_{k-1}(\hat{x}_{k-1}(+))$
Computing the predicted measurement	$\hat{z}_k = h_k(\hat{x}_k(-))$
Linear approximation	$\Phi^{[1]}_{k-1} \approx \frac{\partial \phi_k}{\partial x}\Big\vert_{x=\hat{x}_{k-1(-)}}$
Conditioning the predicted estimate on the measurement	$\hat{x}_k(+) = \hat{x}_k(-) + \overline{K}_k(z_k - \hat{z}_k),\ \ H^{[1]}_k \approx \frac{\partial h_k}{\partial x}\Big\vert_{x=\hat{x}_k(-)}$
Conditioning the *a priori* covariance matrix	$P_k(-) = \Phi^{[1]}_{k-1} P_{k-1}(+) \Phi^{[1]\mathrm{T}}_{k-1} + Q_{k-1}$
Computing the Kalman gain	$\overline{K}_k = P_k(-) H^{[1]\mathrm{T}}_k \left[H^{[1]}_k P_k(-) H^{[1]\mathrm{T}}_k + R_k \right]^{-1}$
Computing the *a posteriori* covariance matrix	$P_k(+) = \left\{ I - \overline{K}_k H^{[1]}_k \right\} P_k(-)$

Examples of Nonlinear KF

- Example: Discrete Linearized Kalman Filter
  - Consider the discrete-time system

$$
\begin{aligned}
x_k &= x_{k-1}^2 + w_{k-1} \\
z_k &= x_k^3 + v_k \\
Ev_k &= Ew_k = 0 \\
Ev_{k_1}v_{k_2} &= 2\Delta(k_2 - k_1) \\
Ew_{k_1}w_{k_2} &= \Delta(k_2 - k_1) \\
Ex(0) &= \hat{x}_0 = 2 \\
x_k^{\mathrm{NOM}} &= 2 \\
P_0(+) &= 1,
\end{aligned}
$$

    - for which one can use the "nominal" solution equations from Table on pages 12-13.

$$
\begin{aligned}
\Phi^{[1]}(x_k^{\mathrm{NOM}}) &= \frac{\partial}{\partial x}[x^2]\bigg|_{x=x^{\mathrm{NOM}}} \\
&= 4 \\
H^{[1]}(x_k^{\mathrm{NOM}}) &= \frac{\partial}{\partial x}(x^3)\bigg|_{x=x^{\mathrm{NOM}}} \\
&= 12
\end{aligned}
$$

    - to obtain the discrete linearized filter equations

$$
\begin{aligned}
\hat{x}_k(+) &= \widehat{\delta x}_k(+) + 2 \\
\widehat{\delta x}_k(+) &= 4\widehat{\delta x}_{k-1}(+) + \bar{K}_k\left[z_k - 8 - 48\widehat{\delta x}_{k-1}(+)\right] \\
P_k(-) &= 16P_{k-1}(+) + 1 \\
P_k(+) &= [1 - 12\bar{K}_k]P_k(-) \\
\bar{K}_k &= \frac{12P_k(-)}{(144P_k(-) + 2)}
\end{aligned}
$$

Example (conc.)

- Discrete Extended Kalman Filter

Given the measurements z_k, $k = 1, 2, 3$, the values for $P_k(-)$, $\bar{K}_k$, $P_k(+)$ and $\hat{x}_k(+)$ can be computed. If z_k are not given, then $P_k(-)$, $\bar{K}_k$, and $P_k(+)$ can be computed and leads towards covariance analysis results. For large k with very small Q and R, the difference $\hat{x}_k - x_k^{\text{NOM}}$ will not stay small, the results become meaningless.

This situation can be improved by using the extended Kalman filter as discussed in KFTP.

$$
\begin{aligned}
\hat{x}_k(+) &= \hat{x}_{k-1}^2(+) + \bar{K}_k[z_k - (\hat{x}_k(-))^3] \\
P_k(-) &= 4\left[\hat{x}_{k-1}(-)\right]^2 P_{k-1}(+) + 1 \\
\bar{K}_k &= \frac{3P_k(-)\left[(\hat{x}_k(-)\right]^2}{9\left[\hat{x}_k(-)\right]^4 P_k(-) + 2} \\
P_k(+) &= [1 - 3\bar{K}_k(\hat{x}_k(-))^2]P_k(-)
\end{aligned}
$$

These equations are now more complex, but should work, provided Q and R are small.

Sigma Point Kalman Filters (SPKF)

- Unscented and central difference Kalman filters
  - Distinguished by weights and scaling parameters associated with sigma points
  - In contrast to EKF, SPKF does not require an approximation to nonlinear dynamics and measurement models using Jacobian in order to calculate the covariance of a random vector (RV) propagated through the nonlinear models
  - In SPKF, a set of deterministically selected sigma points is chosen which have the same mean and covariance as the original RV
  - These sigma points are propagated through the nonlinear models
  - The mean and cov. of transformed RV is calculated from the sigma points
  - This captures the mean and cov. accurately to the 3^{rd} order for arbitrary nonlinear functions (1^{st} order for EKF)

Sigma Point Kalman Filters (SPKF)

- Comment
  - SPKF may be an option in considering the design of new systems
  - But— a modification of the existing EKF GPS/INS based tightly coupled system is neither required, nor appropriate to improve the performance.

Unscented Kalman Filter (UKF)

$$\overset{n\times 1}{X}_k = f_{k-1}(x_{k-1}) + w_{k-1} \sim N(0, Q_k)$$

$$\overset{l\times 1}{Z}_k = h_k(x_k) + v_k \sim N(0, R_k)$$

1) UKF is initialized

$$\hat{X}_0(t) = E\, X_0$$

$$\mathrm{P}_0(+)$$

2) Time update

a) $$\hat{X}^i_{k-1} = \hat{X}_{k-1}(+) + \tilde{X}^i \quad , \quad i = 1, \ldots 2n$$

$$\tilde{X}^i = \left(\sqrt{n\, P_{k-1}(+)}\right)^{\mathrm{T}}_i \quad , \quad i = 1, \ldots n$$

$$\tilde{X}^{n+i} = -\left(\sqrt{n\, P_{k-1}(+)}\right)^{\mathrm{T}}_i \quad , \quad i = 1 \ldots n$$

– Square roots calculated by Cholesky's decomposition

Unscented Kalman Filter (cont.)

b) $$\hat{X}_k^i = f\left(\hat{X}_{k-1}^i\right)$$

c) A priori state estimate

$$\hat{X}_k(-) = \frac{1}{2n}\sum_{i=1}^{2n} \hat{X}_k^i$$

d) A priori error cov.

$$P_k(-) = \frac{1}{2n}\sum_{i=1}^{2n} \left(\hat{X}_k^i - \hat{X}_k(-)\right)\left(\hat{X}_k^i - \hat{X}_k(-)\right)^{\mathrm{T}} + Q_{k-1}$$

3) Observation update

(Use the $\hat{X}_k^i$ from Part 2 b)

a) $$\hat{Z}_k^i = h\left(\hat{X}_k^i\right)$$

b) $$\hat{Z}_k = \frac{1}{2n}\sum_{i=1}^{2n} \hat{Z}_k^i$$

Unscented Kalman Filter (cont.)

c) Cov. of predicted measurements

$$P_z = \tfrac{1}{2n}\sum_{i=1}^{2n}\left(\hat{Z}_k^i - \hat{Z}_k\right)\left(\hat{Z}_k^i - \hat{Z}_k\right)^{\mathrm{T}} + R_k$$

d) Estimate the cross cov. between $\hat{X}_k(-), \hat{Z}_k$

$$P_{xz} = \tfrac{1}{2n}\sum_{i=1}^{2n}\left(\hat{X}_k^i - \hat{X}_k(-)\right)\left(\hat{Z}_k^i - \hat{Z}_k\right)^{\mathrm{T}}$$

e) Meas. update of state estimate done using normal KF

$$\overline{K}_k = P_{xz}P_z^{-1}$$

$$\hat{X}_k(+) = \hat{X}_k(-) + \overline{K}_k\left[Z_k - \hat{Z}_k\right]$$

$$P_k(+) = P_k(-) - \overline{K}_k P_z \overline{K}_k^{\mathrm{T}}$$

Comment

- System and meas. equations are

$$X_k = f(x_k\ , w_k)$$
$$Z_k = h(x_k\ , v_k)$$

  - Then

$$X_k^a = \begin{bmatrix} x_k & w_k & v_k \end{bmatrix}^{\mathrm{T}}$$
$$\hat{X}_b^a(+) = \begin{bmatrix} EX_0 & 0 & 0 \end{bmatrix}^{\mathrm{T}}$$
$$P_0^a(+) = \begin{bmatrix} P_0(+) & 0 & 0 \\ 0 & Q_0 & 0 \\ 0 & 0 & R_0 \end{bmatrix}$$

  - Use the same process as before

References

- Kalman Filtering Theory & Practice Using MATLAB®, 3rd Edition, Mohinder S. Grewal and Angus P. Andrews, Wiley & Sons, 2008.
- Global Positioning Systems, Inertial Navigation, & Integration, 2nd Edition, Mohinder S. Grewal, Lawrence R. Weill, and Angus P. Andrews, Wiley & Sons, 2007.
- "Application of Kalman Filtering to GPS, INS, & Navigation," Short Course Notes, Mohinder S. Grewal (ed), Kalman Filtering Consultant Associates, 2011.