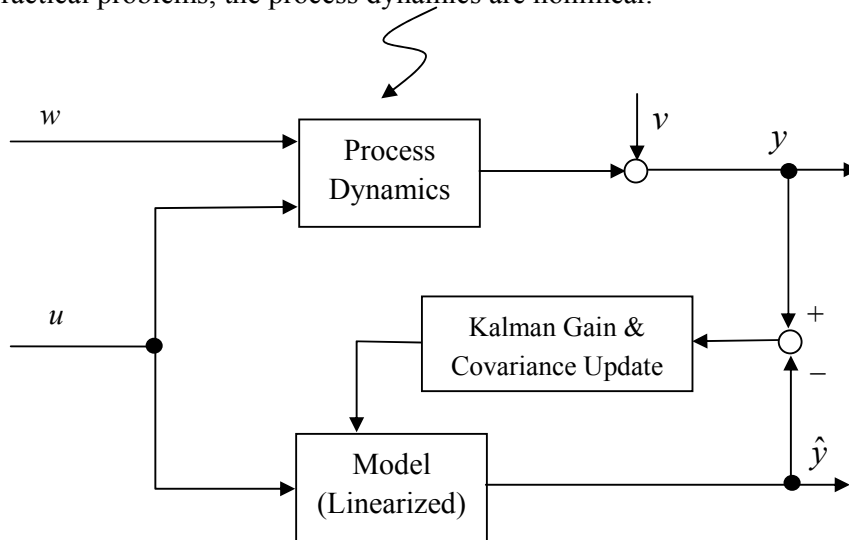**2.160 System Identification, Estimation, and Learning**

Lecture Notes No. 8

March 6, 2006

4.9 Extended Kalman Filter

In many practical problems, the process dynamics are nonlinear.

If the process is nonlinear but smooth, its linearized approximation may be used for the process model.

Extension to non-linear system using linearization
Consider a non-linear, continuous system

$$\dot{x} = f(x,u,t) + w(t) \qquad \dots \quad n-\text{dim} \tag{87}$$

$$y = h(x,t) + v(t) \qquad \dots \quad l-\text{dim} \tag{88}$$

$f(.)$, and $h(.)$: known but non-linear, differentiable functions

u : input (deterministic forcing term; assumed zero)
w , v : uncorrelated process and measurement noises

$$E[w(t)] = 0 , \quad E[v(t)] = 0$$

$$E[w(t)w^{T}(s)] = \begin{cases} 0 & t \neq s \\ Q & t = s \end{cases} \qquad E[v(t)v^{T}(s)] = \begin{cases} 0 & t \neq s \\ Q & t = s \end{cases} \qquad E[w(t)v^{T}(s)] = 0$$

The original Kalman filter is not applicable to this class of non-linear systems.

Approach   Linearize the non-linear system around the state that is either:

      1) Pre-determined…linearized Kalman filter

         e.g. a trajectory to track.

    or

      2) Estimated in real-time using on-line measurement

         …Extended Kalman Filter

4.9.1 Linearized Kalman Filter
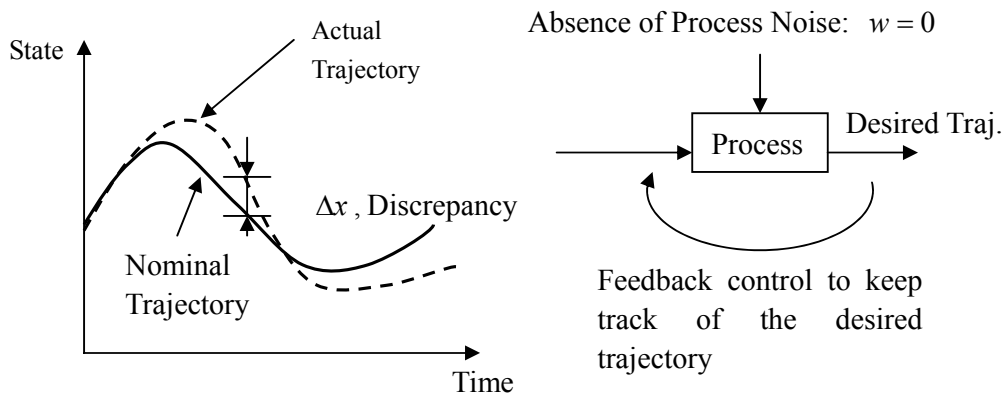
$x^*(t) = a$ nominal trajectory in the state space satisfying the noise-less state equation:

$$\dot{x}^*(t) = f\left(x^*(t), t\right) \tag{89}$$

    absence of process noise

Consider derivation $\Delta x(t)$

$$x(t) = x^*(t) + \Delta x(t) \tag{90}$$

$$\dot{x}(t) = \dot{x}^*(t) + \Delta \dot{x}(t) \tag{91}$$

Taylor expansion

$$f(x,t) = f\left(x^* + \Delta x, t\right)$$
$$\cong f\left(x^*, t\right) + \frac{\partial f}{\partial x}\Big|_{x=x^*} \Delta x \tag{92}$$

where

$$\frac{\partial f}{\partial x}\bigg|_{x=x^*} = \begin{bmatrix} \dfrac{\partial f_1}{\partial x_1} & \dfrac{\partial f_1}{\partial x_2} & \cdots & \dfrac{\partial f_1}{\partial x_n} \\ \dfrac{\partial f_2}{\partial x_1} & \ddots & & \vdots \\ \vdots & & \ddots & \\ \dfrac{\partial f_n}{\partial x_1} & \cdots & & \dfrac{\partial f_n}{\partial x_n} \end{bmatrix}_{x=x^*} \in R^{n \times n} \quad \text{Jacobian} \tag{93}$$

Combining (91) and (92)

$$\dot{x}* + \Delta\dot{x} = \underbrace{f(x^*, t)}_{\dot{x}^* \text{ from (89)}} + \frac{\partial f}{\partial x}\bigg|_{x^*} \Delta x + w(t)$$

$$\Delta\dot{x} = \frac{\partial f}{\partial x}\bigg|_{x^*} \Delta x + w(t)$$

Now replacing Δx by x and $\dfrac{\partial f}{\partial x}\bigg|_{x^*}$ by $F(t) \in R^{n \times n}$
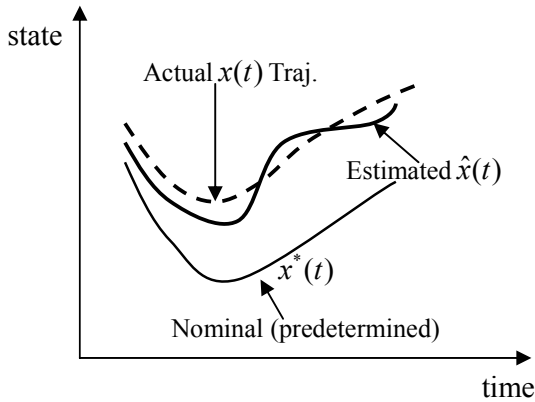
$$\dot{x} = F(t)x + w(t) \tag{94}$$

Similarly, from (88),

$$y = H(t)x + v(t) \qquad\qquad H(t) = \frac{\partial h}{\partial x}\bigg|_{x^*} \in R^{l \times n} \tag{95}$$

$$y^* + \Delta y = h(x^*, t) + \frac{\partial h}{\partial x}\bigg|_{x^*} \Delta x + v$$

Replacing $\quad y^* \quad H(t)$

Δy by y

Note that the above linearized system (94), (95) with $F(t)$ and $H(t)$ are in the same form as that of the original Kalman filter: linear time-varying systems.

4.9.2 Extended Kalman Filter.

state

Actual $x(t)$ Traj.

Estimated $\hat{x}(t)$

$x^*(t)$

Nominal (predetermined)

time

With the measurement of the actual process, the estimated trajectory is deemed to be more accurate. Use the state $\hat{x}(t)$ estimated in real-time for linearizing the dynamics

$\longrightarrow$ the Extended Kalman filter
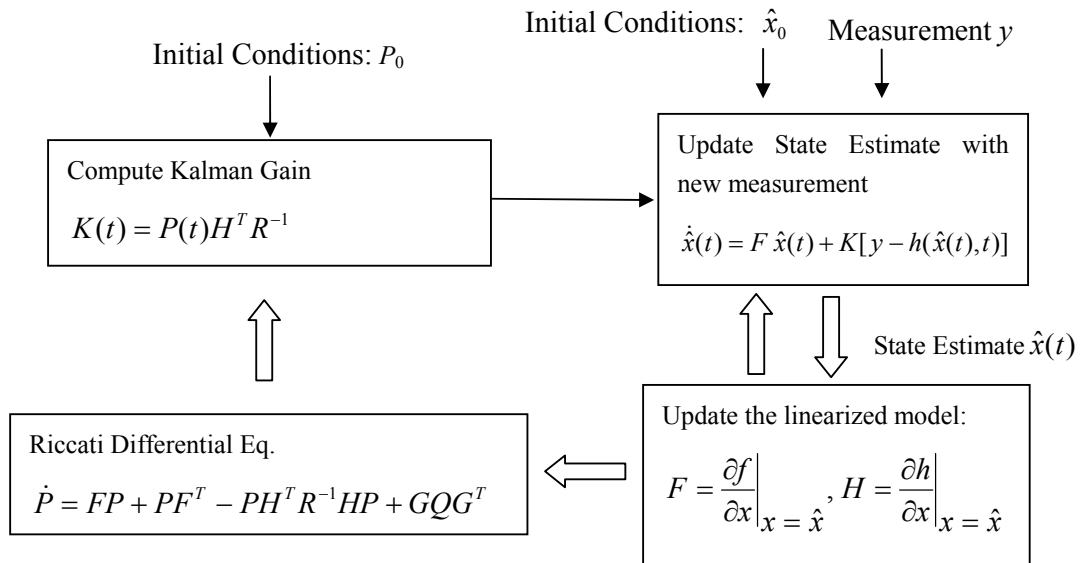
$$\left.\frac{\partial f}{\partial x}\right|_{x=\hat{x}} = F(\hat{x},t)$$

$$\left.\frac{\partial h}{\partial x}\right|_{x=\hat{x}} = H(\hat{x},t) \tag{96}$$

Namely, matrices F and H are evaluated at $x=\hat{x}$, the estimated values of the state in real time, rather than its nominal values. Note that $F(\hat{x},t)$ and $H(\hat{x},t)$ can not be pre-computed in off-line.

For estimating the output, however, we do not have to use the linearized model; the nonlinear output function, eq.(88), can be used:

$$\hat{y}(t) = h(\hat{x}(t),t) \tag{97}$$

Initial Conditions: $\hat{x}_0$ Measurement y

Initial Conditions: P_0

Compute Kalman Gain

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

Update State Estimate with new measurement

$\dot{\hat{x}}(t) = F\,\hat{x}(t) + K[y - h(\hat{x}(t),t)]$

State Estimate $\hat{x}(t)$

Riccati Differential Eq.

$\dot{P} = FP + PF^T - PH^T R^{-1}HP + GQG^T$

Update the linearized model:

$F = \left.\frac{\partial f}{\partial x}\right|_{x=\hat{x}}$, $H = \left.\frac{\partial h}{\partial x}\right|_{x=\hat{x}}$

Extended Kalman Filter

A critical issue of this Extended Kalman Filter is instability. As estimated state $\hat{x}$ deviates from the true state, the linearized model becomes inaccurate, which may lead to an even larger error in state estimation. Care must be taken in implementing extended Kalman Filter.

4.10 Implementation Issues

Kalman filters have been applied to diverse applications since early 60's. Various implementation techniques have also been developed. There are three known failure scenarios in which Kalman filters do not work well:

a) Unobservable or nearly unobservable processes
b) Numerical instability
c) Blind spot

a) Poor observability

The above first issue can be checked with the well-known observability condition. An alternative, more practical method is to examine the error covariance matrix $P(t)$ or P_t. If the process is poorly observable, the variance associated with some unobservable state variables blow up.

If the process is poorly observable, one should change the sensors, or a new sensor must be added.

b) Numerical instability

Asymmetric covariance: By definition, covariance matrices are symmetry, but numerically they may become asymmetric, leading to divergence in recursive computation. Suchan asymmetric covariance often comes from the computation of:

$$P_t = (I - K_t H_t) P_{t|t-1} \tag{41}$$

where $K_t \in R^{n \times \ell}$ and $H_t \in R^{\ell \times n}$ are not square matrices. Some round off errors yield an asymmetric posteriori covariance P_t, although the a priori covariance $P_{t|t-1}$ is symmetry. To resolve this problem, it is efficient to use Joseph's form (40):

$$P_t = (I - KH) P_{t|t-1} (I - KH)^T + K R_t K^T$$

which is equivalent to (41), as discussed in Section 4.5.2. Note that both terms on the right hand side are symmetric matrices.

U-D Factorization: Since the covariance matrix is a real, symmetric, positive-definite matrix, it can be decomposed to the following U-D Factorization form:

$$P = UDU^T$$

$$D = \begin{pmatrix} d_1 & 0 & \cdots & 0 \\ 0 & d_2 & & \vdots \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & \cdots & d_n \end{pmatrix}, \quad U = \begin{pmatrix} 1 & * & \cdots & * \\ 0 & 1 & * & * \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & 0 & 1 \end{pmatrix} \tag{98}$$

where matrix D is a diagonal matrix while matrix U is an upper triangular matrix.

This particular form assures the positive definiteness of the covariance matrix, and implicitly preserves the symmetry of P. Furthermore, if the covariance update formula of Kalman filter is converted to the one in the diagonalized space using the upper triangular matrix U, the dynamic range of computation reduces to 50 % of the original formulation. See more details in Section 9.5 in Brown and Hwang's textbook.

c) Blind spot

Consider another failure scenario: When both process noise and measurement noise covariance matrices are deemed to be very small, the state estimation error-covariance reduces quickly. This is clear from the Riccati differential equation (62):

$$\dot{P} = FP + PF^T - PH^T R^{-1} HP + GQG^T$$

as well as from the covariance propagation and update formulae for discrete Kalman filter: $P_{t+1|t} = A_t P_t A_t^T + G_t Q_t G_t^T$, and $P_t = (I - K_t H_t) P_{t|t-1}$. This implies that the Kalman gain diminishes quickly. Once the Kalman gain diminishes, the subsequent observations are ignored. In other words, the Kalman filter is decoupled from the sensors and the real process. This blind spot problem is often triggered by numerical round off error in computing covariance matrices as well.

4.11 Relationship to Wiener Filter

Prior to Kalman's work in the early 60's, N. Wiener made a significant contribution to the theory of minimum mean-square error filtering in the 40's. You can see his picture and accomplishments in the display posted at the Infinite Corridor. The table below compares Kalman filter and Wiener filter:

Kalman Filter	Wiener Filter
State space	Frequency domain
Non-stationary	Stationary
Continuous and discrete	Continuous

Since Wiener filter can be seen as a special case of Kalman filter, the fomer can be derived from the latter. Assume a stationary, time-invariance process, for which the state estimation error covariance in Kalman filter reduces to:

$$P(t) \to P_\infty = const. \quad \frac{dP}{dt} = 0$$

$$0 = FP_\infty + P_\infty F^T - P_\infty H^T R^{-1} HP_\infty + GQG^T \qquad (67)$$

Using this, the Kalman filter reduces to

$$\dot{\hat{x}} = F\hat{x} + K_\infty (y - H\hat{x}) \qquad (98)$$

where

$$K_\infty = P_\infty H^T R^{-1}. \tag{99}$$

Taking Laplace transform of (98),

$$s\,\hat{x}(s) = F\,\hat{x}(s) - K_\infty H\,\hat{x}(s) + K_\infty y(s)$$

and using (99) yields:

$$\hat{x}(s) = [sI - F + P_\infty H^T R^{-1} H]^{-1} P_\infty H^T R^{-1} \cdot y(s) \tag{100}$$

The above expression is equivalent to Wiener filter.