Kalman Filter

Zhang Hongxin zhx@cad.zju.edu.cn

State Key Lab of CAD&CG, ZJU 2006-04-13

Rudolf Emil Kalman

- Born 1930 in Hungary
- BS and MS from MIT
- PhD 1957 from Columbia
- Filter developed in 1960-61
- Now retired





What is a Kalman Filter?

- Just some applied math.
- A linear system: f(a+b) = f(a) + f(b).
- Noisy data in :: hopefully less noisy out.
- But delay is the price for filtering...
- Pure KF does not even adapt to the data.



What is it used for?

- Tracking missiles
- Tracking heads/hands/drumsticks
- Extracting lip motion from video
- Fitting Bezier patches to point data
- Economics
- Navigation





A really simple example



The Process to be Estimated

- Discrete-time controlled process
 - State estimation:

 $\mathbf{X}_{k} \in \mathfrak{R}^{n}$ $\mathbf{x}_k = A\mathbf{x}_{k-1} + B\mathbf{u}_{k-1} + \mathbf{w}_{k-1}$ Measurement: $\mathbf{Z}_k \in \mathfrak{R}^m$ $\mathbf{Z}_{k} = H\mathbf{X}_{k-1} + \mathbf{V}_{k-1}$ \succ Process noise covariance: Q $p(\mathbf{w}) \sim N(0, Q)$ > Measurement noise covariance: R $p(\mathbf{v}) \sim N(0, R)$

The computational Origins of the Filters

• Priori state estimation error at step k

$$\mathbf{e}_k^- \coloneqq \mathbf{x}_k - \hat{\mathbf{x}}_k^- \qquad P_k^- = E[\mathbf{e}_k^- \mathbf{e}_k^{-1}]$$

Posteriori estimation error

$$\mathbf{e}_k \coloneqq \mathbf{x}_k - \hat{\mathbf{x}}_k \qquad P_k = E[\mathbf{e}_k \mathbf{e}_k']$$

Posteriori as a linear combination of an Priori

$$\mathbf{x}_{k} = A\mathbf{x}_{k-1} + B\mathbf{u}_{k-1} + \mathbf{w}_{k-1}$$
$$\hat{\mathbf{x}}_{k} = \hat{\mathbf{x}}_{k}^{-} + K(\mathbf{z}_{k} - H\hat{\mathbf{x}}_{k}^{-})$$

Measurement *innovation* or *residual*

The computational Origins of the Filters



$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + K(\mathbf{z}_k - H\hat{\mathbf{x}}_k^-)$$

• The gain or blending factor that minimizes the a posteriori error covariance $P_k = E[\mathbf{e}_k \mathbf{e}_k^T]$

$$K_k = \frac{P_k^- H^T}{H P_k^- H^T + R}$$

$$\lim_{R \to 0} K_{k} = H^{-1} \qquad \lim_{P_{k}^{-} \to 0} K_{k} = 0$$

The Probabilistic Origins of the Filter

$$E[\mathbf{x}_{k}] = \hat{\mathbf{x}}_{k}$$
$$E[(\mathbf{x}_{k} - \hat{\mathbf{x}}_{k})(\mathbf{x}_{k} - \hat{\mathbf{x}}_{k})^{T}] = P_{k}$$

- The a posteriori state estimate reflects the mean of the state distribution
- The *a posteriori* state estimate error covariance reflects the variance of the state distribution

$$p(\mathbf{x}_{k} | \mathbf{z}_{k}) \sim N(E[\mathbf{x}_{k}], E[(\mathbf{x}_{k} - \hat{\mathbf{x}}_{k})(\mathbf{x}_{k} - \hat{\mathbf{x}}_{k})^{T}])$$
$$= N(\mathbf{x}_{k}, P_{k})$$



The Discrete Kalman Filter Algorithm



$$\hat{\mathbf{x}}_{k}^{-} = A\hat{\mathbf{x}}_{k} + B\mathbf{u}_{k-1}$$
$$P_{k}^{-} = AP_{k}A^{T} + Q$$



- Measurement update equations $K_{k} = \frac{P_{k}^{-}H^{T}}{HP_{k}^{-}H^{T} + R}$ $\hat{\mathbf{x}} = \hat{\mathbf{x}}^{-} + K (\mathbf{z} - H\hat{\mathbf{x}}^{-})$
 - $\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + K_k(\mathbf{z}_k H\hat{\mathbf{x}}_k^-)$
 - $P_k = (I K_k H) P_k^-$



Filter Parameters and Tuning



- The measurement noise covariance *R* is usually measured prior to operation of the filter.
- *Q* and *R* are generally constants during filtering. Superior filter performance can be obtained by tuning them, referred to as system identification.



Example: 2D Position-Only

• Apparatus: 2D Tablet



Process Model



State *k* State State *k-1* Noise transition

$$\mathbf{x}_k = A\mathbf{x}_{k-1} + \mathbf{w}_{k-1}$$



Measurement Model





Measurement *k* Measurement State *k* Noise matrix

$$\mathbf{z}_k = H\mathbf{x}_k + \mathbf{v}_k$$

Preparation

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

State Transition

$$Q = E\left\{\mathbf{w} \cdot \mathbf{w}^{T}\right\} = \begin{bmatrix} Q_{xx} & 0\\ 0 & Q_{yy} \end{bmatrix}$$
$$R = E\left\{\mathbf{v} \cdot \mathbf{v}^{T}\right\} = \begin{bmatrix} R_{xx} & 0\\ 0 & R_{yy} \end{bmatrix}$$

Process

Noise Covariance

Measurement

Noise Covariance

Initialization



$$\mathbf{x}_0 = H\mathbf{z}_0$$
$$P = \begin{bmatrix} \varepsilon & 0 \\ 0 & \varepsilon \end{bmatrix}$$

Predict

$$\mathbf{x}_{k}^{-} = A\mathbf{x}_{k-1}$$

$$P_{k}^{-} = AP_{k-1}A^{T} + Q$$
transition uncertainty

Correct

$$K_{k} = P_{k}^{-}H^{T}(HP_{k}^{-}H^{T}+R)^{-1}$$

$$\hat{\mathbf{x}}_{k} = \hat{\mathbf{x}}_{k}^{-} + K_{k} (\mathbf{z}_{k} - H\hat{\mathbf{x}}_{k}^{-})$$
$$P_{k} = (I - K_{k} H) P_{k}^{-}$$

Results: XY Track







Y Track: Moving then Still





Motion-Dependent Performance



The Extended Kalman Filter

- Nonlinear Process (Model)
 - Process dynamics: A becomes a(x)
 - Measurement: H becomes h(x)

Filter Reformulation

- Use functions instead of matrices
- Use Jacobians to project forward, and to relate measurement to state



Jacobian?

- Partial derivative of measurement with respect to state
- If measurement is a vector of length M
- And state has length N
- Jacobian of measurement function will be MxN matrix of numbers (not equations)
- Often evaluating h(x) and Jacobian(h(x)) at the same time cost only a little extra



New Approaches



• Several extensions are available that work better than the EKF in some circumstances

Summary



- A set of mathematical equations that provides an efficient computational (recursive) means to estimate the state of a process.
- Minimizes the mean of the squared error
- Powerful:
 - supports estimations of past, present, and even future states,
 - can do so even when the precise nature of the modeled system is unknown

The End of Kalman Filter



Before the end of this course

- Many techniques I cannot mention yet:
 - Neural network
 - SVM
 - Graphical model
 - Genetic methods

• • • •

• It is just a beginning ...



