EE525 - Notes on Kalman Filter - F. W. DePiero

The Kalman Filter is a powerful method for reducing noise in measurements. It is particularly useful when measuring rigid body motion, for example. The Kalman Filter estimates the state of the physical system that is being measured by modeling system dynamics. This permits an estimate of future states to be predicted, which serves as a basis to mitigate the noise in measurements.

The following models are used to describe changes in the true state and true measurements, for discrete and continuous versions of the system models. These models include state changes due to both system dynamics (deterministic) and process noise (random).

$\dot{x} = Fx + Gu$, Continuous-time

$x_{k+1} = \phi_k x_k + w_k$, Discrete-time

$z_k = H_k x_k + v_k$

$x,\ \dot{x}$	Nx1	True state at time t and time derivative (continuous-time). The value of the true state is unknown the filter estimates it on-line.
x_k	Nx1	True state at time t_k, discrete-time. (Also unknown).
ϕ_k	NxN	State transition matrix describing changes in state due to system dynamics only. The effects of noise disturbances are not included in ϕ_k. State change occurs across the discrete-time interval from t_k to t_{k+1}.
z_k	Mx1	Measurement at time t_k.
H_k	MxN	Measurement matrix, relating state to measurement. No noise effects are included.
w_k	Nx1	Process noise (discrete-time version). A Gaussian white sequence.
u	Nx1	Process noise (continuous-time version). May be Gaussian white noise or other random process.
v_k	Mx1	True value of measurement noise (unknown).

Process noise is included in the system model as a means to describe unmodeled system dynamics or disturbance inputs. Process noise can describe the effect of ocean waves on a boat or wind on an airplane, for example. Process noise can also describe changes to state that are due to simplified, or truncated, dynamic models – such as the use of a [Position Velocity] state rather than [Position Velocity Acceleration] ('[PVA]'). As such, the process noise is somewhat of a 'catch-all' source of disturbances. Another example of unmodeled dynamics is the Captain steering a boat. This action might be deterministic, however it is beyond the simple predictive capabilities of a [PVA] state. So the approach taken is simply to treat this type of state change as process noise.

In the continuous-time system, the process noise u(t) causes changes to the state in an on-going fashion. The cumulative effect of the process noise on the state during the discrete-time interval from t_k to t_{k+1} (Δt seconds) is described by the Gaussian white sequence w_k. The following quantities are defined, and subsequently used, in the on-line version of the Kalman Filter.

e_k	Nx1	Error in state estimate $e_k = x_k - \hat{x}_k$ at time t_k.
P_k	NxN	Covariance matrix of the error in the state estimate. Value updated on-line. $P_k = E[e_k e_k^T]$.
R_k	MxM	Covariance matrix of measurement noise. $R_k = E[v_k v_k^T]$.
Q_k	NxN	Covariance matrix of process noise. $Q_k = E[w_k w_k^T]$.

In a filter with constant R_k and Q_k matrices these are typically computed during the filter design process, as described below.

Implementation, General Case

The following steps describe equations that need to be evaluated on-line for a Kalman Filter.

1) Update Kalman Gain matrix, $K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1}$
2) Acquire new measurement, z_k
3) Update State Estimate using new measurement, $\hat{x}_k = \hat{x}_k^- + K_k (z_k - H_k \hat{x}_k^-)$
4) Use State Estimate, as needed in application
5) Update Error Covariance matrix, $P_k = (I - K_k H_k) P_k^-$
6) Project ahead. Find state based on system dynamics, $\hat{x}_{k+1}^- = \phi_k \hat{x}_k$
7) Project ahead. Find Error Covariance based on dynamics & expected process noise, $P_{k+1}^- = \phi_k P_k \phi_k^T + Q_k$
8) Advance discrete-time index k = k + 1
9) Loop to step 1

$\hat{x}_k$	Nx1	State estimate at time t_k.
$\hat{x}_k^-$	Nx1	State estimate, prior to incorporation of new measurement in step 3.
P_k^-	NxN	Covariance matrix of the error in the state estimate. Value of P_k prior to incorporation of new measurement data.
K_k	NxM	Kalman gain matrix. Weights the effect of new measurements on the changes to the state estimate. (Step 3). Value updated on-line.

Note that since the true state is unknown, the *actual* error in the state estimates and associated covariance matrix, P_k , are also unknown. Therefore and exact value for P_k can never be known. Rather estimates of P_k are updated in an on-line fashion.

Initial Condition

Initial values are needed for P_k^- and $\hat{x}_k^-$. Zero matrices should typically work fine.

A Specific Implementation of the Kalman Filter

- State: Position, Velocity, Acceleration
- Scalar position measurements, variance is not time-varying
- System dynamics are not time-varying
- Gaussian White Noise (Infinite Variance), PSD = W (W is constant)
- Sample Period = Δt

$$\phi_k = \phi = \begin{bmatrix} 1 & \Delta t & \Delta t^2/2 \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{bmatrix} \qquad Q_k = Q = \begin{bmatrix} \frac{W}{20}\Delta t^5 & \frac{W}{8}\Delta t^4 & \frac{W}{6}\Delta t^3 \\ \frac{W}{8}\Delta t^4 & \frac{W}{3}\Delta t^3 & \frac{W}{2}\Delta t^2 \\ \frac{W}{6}\Delta t^3 & \frac{W}{2}\Delta t^2 & W\Delta t \end{bmatrix}$$

Finding The State Transition Matrix

The dynamics of a system are described by the state equation $\dot{x} = Fx$ (in continuous-time) and by $x_{k+1} = \phi_k x_k$ (in discrete-time). Note in both of these cases the forcing functions have been set to zero. The state transition matrix, ϕ_k, may be found by examining the response of the system to non-zero initial conditions due to

$$\dot{x} = Fx$$

taking the Laplace transform,

$$sX - x(t = 0^-) = FX$$

$$sX - FX = (sI - F)X = x(t = 0^-)$$

$$X = (sI - F)^{-1} x(t = 0^-)$$

and then taking the inverse Laplace transform, $L^{-1}[]$, to yield

$$x(t) \;\; = L^{-1}[\,(sI - F)^{-1}\,] \;\; x(t = 0^-)$$

For systems with constant dynamics the state transition matrix is constant, $\phi_k = \phi$ and the change in state from t_k to t_{k+1} may be conveniently evaluated from t=0 to t=Δt. Comparing the form of $x_{k+1} = \phi_k x_k$ with $x(t) \;\; = L^{-1}[\,(sI - F)^{-1}\,] \;\; x(t = 0^-)$ yields

$$\phi \;\; = L^{-1}[\,(sI - F)^{-1}\,], \quad t = \Delta t$$

where Δt is the sample period in seconds.

Example of Finding State Transition Matrix

For a state with position and velocity, $x^T = [p \; v]^T$,

$$\dot{x} = Fx = \begin{bmatrix} \dot{p} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} p \\ v \end{bmatrix}, \text{ and}$$

$$sI - F = \begin{bmatrix} s & -1 \\ 0 & s \end{bmatrix}$$

$$(sI - F)^{-1} = \frac{1}{s^2}\begin{bmatrix} s & 1 \\ 0 & s \end{bmatrix} = \begin{bmatrix} \frac{1}{s} & \frac{1}{s^2} \\ 0 & \frac{1}{s} \end{bmatrix}$$

$$\phi_k = \phi = \left.\begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix}\right|_{t=\Delta t} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}$$

Numerical Evaluation of State Transition Matrix, In MatLab

Just as a first-order differential equation governing changes in a scalar signal has a general solution involving $e^{-t/\tau}$, the first-order matrix differential equation $\dot{x} = Fx$ has solutions involving the matrix exponential

$$\phi_k = \phi(t = \Delta t) = e^{F\Delta t} = I + F\Delta t + \frac{(F\Delta t)^2}{2} + \cdots$$

The state transition matrix for the previous example may be evaluated in MatLab. A specific value for the sample period is used in the calculations, here Δt = 0.1 sec ('dt' below).

```
>>  F = [ 0 1 ; 0 0 ]
>>  dt = 0.1
>>  expm(F * dt)
```

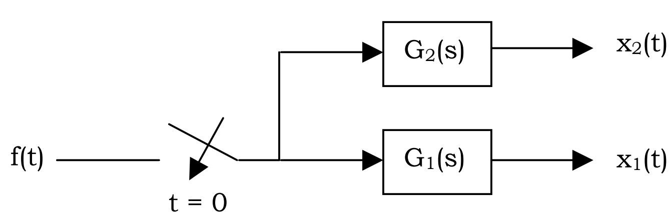
Finding Covariance Matrix of the Process Noise

The covariance matrix of the process noise, Q_k, describes how much the state is expected to wander from time t_k to t_{k+1} due to the cumulative effect of the (continuous-time) process noise. If the process noise is modeled by a random signal having constant PSD (such as Gaussian white noise), and provided $\phi_k = \phi$, then Q_k = Q. For N=3

$$Q_k = E[w_k w_k^T] = \begin{bmatrix} E[x_1x_1] & E[x_1x_2] & E[x_1x_3] \\ E[x_1x_2] & E[x_2x_2] & E[x_2x_3] \\ E[x_1x_3] & E[x_2x_3] & E[x_3x_3] \end{bmatrix}$$

$$E[x_i(t = \Delta t)\; x_j(t = \Delta t)] = R_{x_i x_j}(\Delta t, \Delta t)$$

where $E[x_i\ x_j]$ is the expected value of the state vector components, x_i and x_j, after being subjected to process noise over the interval from t=0 to t=Δt. (Recall w_k gives the cumulative effect suffered by the state on the k_{th} iteration). The following generic block diagram is meant to illustrate the effect of the process noise, f(t), on state elements x_i(t).



The analysis required to find Q_k is a non-stationary, transient analysis. The need for this type of analysis can be appreciated by first observing that if Δt =0 then Q_k = 0 (in the absence of non-zero initial conditions for the process noise of x_i). With Δt =0, there is no time to accumulate any effects of noise that change the state. Hence the analysis begins at t = 0 and proceeds to t = Δt, integrating the noise effect in a transient manner. It can be appreciated that the analysis is non-stationary, by observing that the amount of wandering of the state that can be expected would increase with Δt. Hence

$$R_{x_i x_j}(\Delta t, \Delta t) \neq R_{x_i x_j}(\tau = 0)$$

and the evaluation of Q_k is therefore non-stationary, by definition.

The response of $x_i(t)$, in general, may be found via convolution

$$x_i(t = \Delta t) = \int_0^{\Delta t} g_i(\alpha) f(\Delta t - \alpha)\, d\alpha$$

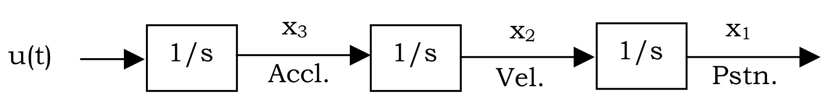
where $G_i(s)$ and $g_i(t)$ are Laplace transform pairs. Finding the expected value,

$$\begin{aligned} E[x_i(t = \Delta t) x_j(t = \Delta t)] &= E\left[\int_0^{\Delta t} g_i(\alpha) f(\Delta t - \alpha)\, d\alpha \int_0^{\Delta t} g_j(\beta) f(\Delta t - \beta)\, d\beta \right] \\ &= \int_0^{\Delta t}\int_0^{\Delta t} g_i(\alpha) g_j(\beta) E[f(\Delta t - \alpha) f(\Delta t - \beta)]\, d\alpha\, d\beta \\ &= \int_0^{\Delta t}\int_0^{\Delta t} g_i(\alpha) g_j(\beta) R_f(\alpha - \beta)\, d\alpha\, d\beta \end{aligned}$$

This method of evaluating Q_k is referred to as the 'transfer-function' approach, because the individual transfer functions that describe the relationship between f(t) and the state variables are used.

Example

Given a [PVA] system and a Gaussian white process noise, u(t), with PSD = W,



The covariance of the process noise

$$Q_k = E[w_k w_k^T] = \begin{bmatrix} E[x_1 x_1] & E[x_1 x_2] & E[x_1 x_3] \\ E[x_1 x_2] & E[x_2 x_2] & E[x_2 x_3] \\ E[x_1 x_3] & E[x_2 x_3] & E[x_3 x_3] \end{bmatrix}$$

may now be found via

$$E[x_i x_j] = \int_0^{\Delta t}\int_0^{\Delta t} g_i(\alpha) g_j(\beta) R_f(\alpha - \beta)\, d\alpha\, d\beta$$

where

$$G_1(s) = G(u \to x_1) = 1/s^3$$
$$G_2(s) = G(u \to x_2) = 1/s^2$$
$$G_3(s) = G(u \to x_3) = 1/s$$

and

$$g_1(t) = 0.5\, t^2, \quad t >= 0$$
$$g_2(t) = t, \quad t >= 0$$
$$g_3(t) = 1, \quad t >= 0$$

which are found via a one-sided inverse Laplace transform. The $E[x_1x_2]$ element is therefore

$$E[x_1 x_2] = \int_0^{\Delta t}\int_0^{\Delta t} g_1(\alpha)\, g_2(\beta)\, R_f(\alpha - \beta)\, d\alpha\, d\beta$$
$$= \int_0^{\Delta t}\int_0^{\Delta t} \frac{\alpha^2}{2}\, \beta\, W\delta(\alpha - \beta) \quad d\alpha\, d\beta$$
$$= W \int_0^{\Delta t} \frac{\alpha^3}{2}\, d\alpha = W\frac{\Delta t^4}{8}$$