Least Squares and the Kalman Filter

16 Nov 2004

We have a series of epochs. At each epoch we have condition equations that relate observations to the current state vector (parameter vector),

$$B_i x_i \approx f_i \qquad (W_0) \qquad (1)$$

Between every pair of adjacent epochs we also have a state transition equation,

$$\Phi_i x_i = x_{i+1}, \quad \text{or} \quad -\Phi_i x_i + x_{i+1} = 0 \qquad (W_t) \qquad (2)$$

If data for all epochs is available, we can solve the whole system simultaneously by conventional, batch Least Squares,

$$\left[\begin{array}{cccc} B_1 & & & \\ -\phi_1 & I & & \\ \hline & B_2 & & \\ & -\phi_2 & I & \\ \hline & & B_3 & \\ & & -\phi_3 & I \\ \hline & & & B_4 \end{array}\right] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \approx \left[\begin{array}{c} f_1 \\ 0 \\ \hline f_2 \\ 0 \\ \hline f_3 \\ 0 \\ \hline f_4 \end{array}\right], \quad W = \begin{bmatrix} W_0 & & & & & & \\ & W_t & & & & & \\ & & W_0 & & & & \\ & & & W_t & & & \\ & & & & W_0 & & \\ & & & & & W_t & \\ & & & & & & W_0 \end{bmatrix} \qquad (3)$$

Forming the normal equations symbolically,

$$\left[\begin{array}{c|c|c|c} B_1^T W_0 B_1 + \phi_1 W_t \phi_1 & -\phi_1^T W_t & 0 & 0 \\ \hline -W_t \phi_1 & W_t + B_2^T W_0 B_2 + \phi_2^T W_t \phi_2 & -\phi_2^T W_t & 0 \\ \hline 0 & -W_t \phi_2 & W_t + B_3^T W_0 B_3 + \phi_3^T W_t \phi_3 & -\phi_3^T W_t \\ \hline 0 & 0 & -W_t \phi_3 & W_t + B_4^T W_0 B_4 \end{array}\right] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} B_1^T W_0 f_1 \\ B_2^T W_0 f_2 \\ B_3^T W_0 f_3 \\ B_4^T W_0 f_4 \end{bmatrix} \qquad (4)$$

Notice the normal equations are block-tridiagonal and one can easily make a "recursive partitioning" scheme to solve them efficiently.

As soon as all equations are available for a given state vector, its contribution can be "folded" into the state vector (see the fold lines in equation (3)) For example, once the first two equations are available, their normals contribution can be constructed,

$$\begin{bmatrix} B_1^T W_o B_1 + \phi_1^T W_t \phi_1 & -\phi_1^T W_t \\ -W_t \phi_1 & W_t \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} B_1^T W_o f_1 \\ 0 \end{bmatrix} \tag{5}$$

or,

$$\begin{bmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} t_1 \\ t_2 \end{bmatrix} \tag{6}$$

By block gauss elimination we can solve for x_1 in the first equation and substitute into the second equation,

$$\begin{aligned} x_1 &= n_{11}^{-1} t_1 - n_{11}^{-1} n_{12} x_2 \\ n_{22} - n_{21} n_{11}^{-1} n_{12} (x_2) &= t_2 - n_{21} n_{11}^{-1} t_1 \end{aligned} \tag{7}$$

Being block tri-diagonal, we only have to modify one diagonal block on the left, and one block in the right hand side vector. After the observations arrive for any epoch, we may solve for that epoch's state vector and covariance, and/or fold its contributions forward to wait for the next epoch. It is very efficient, we only need the latest epoch's data. If you want to go backward, then you must save the matrices for the eliminated epoch state vectors. This sequential technique is equivalent to the Kalman Filter, but to derive the traditional KF equations we approach in a slightly different way.

We will assume that we have a solution for state vector X_{i-1} and its covariance matrix, P_{i-1}, and then we proceed by the two steps,

$$(1.\ \text{prediction}) \qquad X_i = \Phi_{i-1} X_{i-1} \qquad (8)$$

now, using the traditional (Brown & Hwang) variable names, this set of equations has covariance Q. $(X_i = \Phi_{i-1} X_{i-1} + w_{i-1})$. If we do error propagation,

$$X_i = \begin{bmatrix} \Phi_{i-1} & I \end{bmatrix} \begin{bmatrix} X_{i-1} \\ w_{i-1} \end{bmatrix}$$

$$P_i^- = \begin{bmatrix} \Phi_{i-1} & I \end{bmatrix} \begin{bmatrix} P_{i-1} & 0 \\ 0 & Q \end{bmatrix} \begin{bmatrix} \Phi_{i-1}^T \\ I \end{bmatrix} = \Phi_{i-1} P_{i-1} \Phi_{i-1}^T + Q \qquad (9)$$

The super "$-$" in P_i^- indicates that it is the covariance of X_i computed using only <u>prior</u> information. The observation or condition equations constitute the second step. By KF notation $H = B$, and $z = f$.

$$(2.\ \text{observation}) \qquad z_i = H x_i + v_i \qquad (10)$$

The uncertainty here is expressed by covariance matrix R. We want to combine these two steps into one estimation step,

$$\begin{bmatrix} H \\ I \end{bmatrix} (X) \approx \begin{bmatrix} z \\ X^- \end{bmatrix}, \quad \text{Weight} = \begin{bmatrix} R^{-1} & 0 \\ 0 & (P^-)^{-1} \end{bmatrix} \qquad (11)$$

As before, the notation X^- indicates state vector X predicted from prior information, equation (8). Equation (11) is really a least squares problem, in fact it is a unified least squares problem because we are solving for X, but we have a prior estimate and a covariance. Solve the unified LS problem,

$$\begin{bmatrix} H^T & I \end{bmatrix} \begin{bmatrix} R^{-1} & 0 \\ 0 & (P^-)^{-1} \end{bmatrix} \begin{bmatrix} H \\ I \end{bmatrix} (X) = \begin{bmatrix} H^T & I \end{bmatrix} \begin{bmatrix} R^{-1} & 0 \\ 0 & (P^-)^{-1} \end{bmatrix} \begin{bmatrix} z \\ x^- \end{bmatrix} \tag{12}$$

to avoid cumbersome notation $(P^-)^{-1}$, let's use P_m^{-1}. Solve for the solution X, after simplifying,

$$\begin{gathered} \left[H^T R^{-1} H + P_m^{-1} \right] (X) = \left(H^T R^{-1} z + P_m^{-1} X^- \right) \\ X = \left[H^T R^{-1} H + P_m^{-1} \right]^{-1} \left(H^T R^{-1} z + P_m^{-1} X^- \right) \end{gathered} \tag{13}$$

with covariance matrix,

$$P = \left[H^T R^{-1} H + P_m^{-1} \right]^{-1}, \quad \boxed{P^{-1} = H^T R^{-1} H + P_m^{-1}} \tag{14}$$

now we could stop here since we only need to go back to equation (8) and loop continuously as epochs pass. But we need to show that this is <u>equivalent</u> to the KF equations.

At this point we need the famous "matrix inversion lemma" or the Sherman-Morrison-Woodbury-Schur Formula, SMWS,

$$(W + X Z^{-1} Y)^{-1} = W^{-1} - W^{-1} X (Z + Y W^{-1} X)^{-1} Y W^{-1} \tag{15}$$

use this to expand equation (13),

$$X = \left[P_m - \underbrace{P_m H^T (R + H P_m H^T)^{-1}} H P_m\right]\left[H^T R^{-1} z + P_m^{-1} X^-\right] \tag{16}$$

Anticipating future notation let's call the indicated subexpression above, K,

$$\boxed{K = P_m H^T (R + H P_m H^T)^{-1}} \tag{17}$$

Multiplying out equation (16) and substituting, K,

$$X = P_m H^T R^{-1} z + X^- - K H P_m H^T R^{-1} z - K H X^- \tag{18}$$

From (14), (16), and (17), rewrite expression for P

$$P = P_m - K H P_m \, , \quad \boxed{P = (I - KH) P_m} \tag{19}$$

In order to put (18) into the form we need, we must derive an equivalent expression for K (see B+H (3rd) p. 247)

$$K = P_m H^T (R + H P_m H^T)^{-1} \tag{17} \uparrow$$

insert PP^{-1} and RR^{-1},

$$K = P P^{-1} P_m H^T R^{-1} \underbrace{R (H P_m H^T + R)^{-1}} \tag{20}$$

look at sub expression,

$$(R^{-1})^{-1} (H P_m H^T + R)^{-1} \tag{21}$$

combine using $(AB)^{-1} = B^{-1} A^{-1}$,

$$\left([H P_m H^T + R] R^{-1}\right)^{-1} = (H P_m H^T R^{-1} + I)^{-1} \tag{22}$$

plug back into (20),

$$K = P P^{-1} P_m H^T R^{-1} (H P_m H^T R^{-1} + I)^{-1} \tag{23}$$

now plug (14) in for P^{-1}, and simplify

$$K = P \left[P_m^{-1} + H^T R^{-1} H \right] P_m H^T R^{-1} (H P_m H^T R^{-1} + I)^{-1} \tag{24}$$

$$K = P \left[I + H^T R^{-1} H P_m \right] H^T R^{-1} (H P_m H^T R^{-1} + I)^{-1} \tag{25}$$

$$K = P \left[H^T R^{-1} + H^T R^{-1} H P_m H^T R^{-1} \right] (H P_m H^T R^{-1} + I)^{-1} \tag{26}$$

$$K = P H^T R^{-1} \left(I + H P_m H^T R^{-1} \right) (H P_m H^T R^{-1} + I)^{-1} \tag{27}$$

$$\boxed{K = P H^T R^{-1}} \tag{28}$$

now we use (28) and (19) to simplify (18)

$$x = P_m H^T R^{-1} z + x^- - K H P_m H^T R^{-1} z - K H x^- \tag{18}$$

combine terms with common sub-expression

$$x = \underbrace{(I - KH)}_{P} P_m H^T R^{-1} z + x^- - K H x^- \tag{29}$$

Replace sub-expression from (19)

$$x = \underbrace{P H^T R^{-1}}_{K} z + x^- - K H x^- \tag{30}$$

Replace sub-expression from (28)

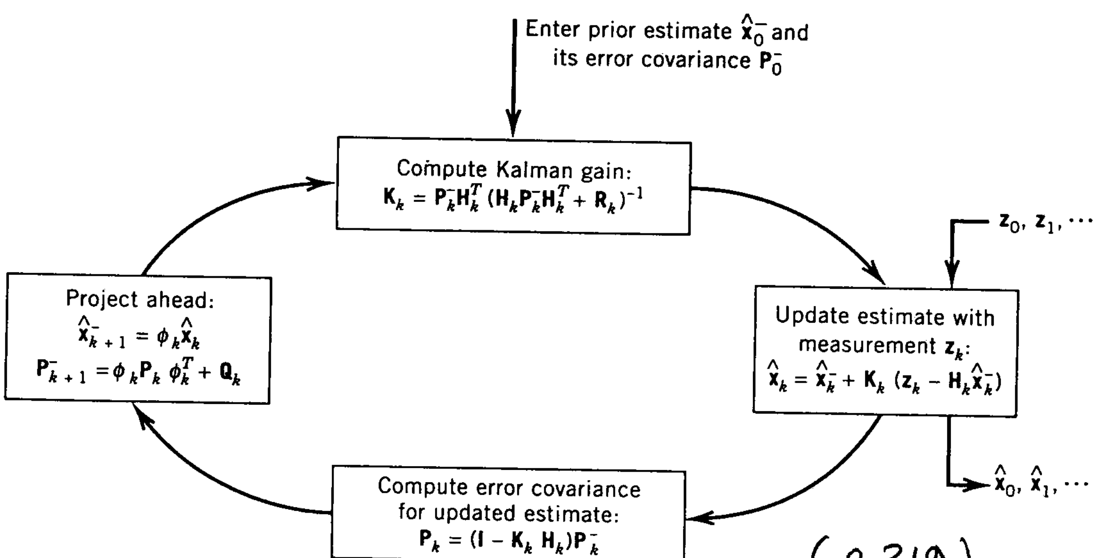
$$x = K z + x^- - K H x^- \tag{31}$$

$$\boxed{X = X^{-} + K(Z - Hx^{-})} \tag{32}$$

This is the famous Kalman update equation, matrix K is called the Kalman gain.

$$\boxed{P = (I - KH)\, P_m} \tag{19}$$

This is the covariance of the new estimate, x. Now we have shown that LS is equivalent to two versions of B+H Kalman Loop,



(p. 219)

Figure 5.8 Kalman filter loop.

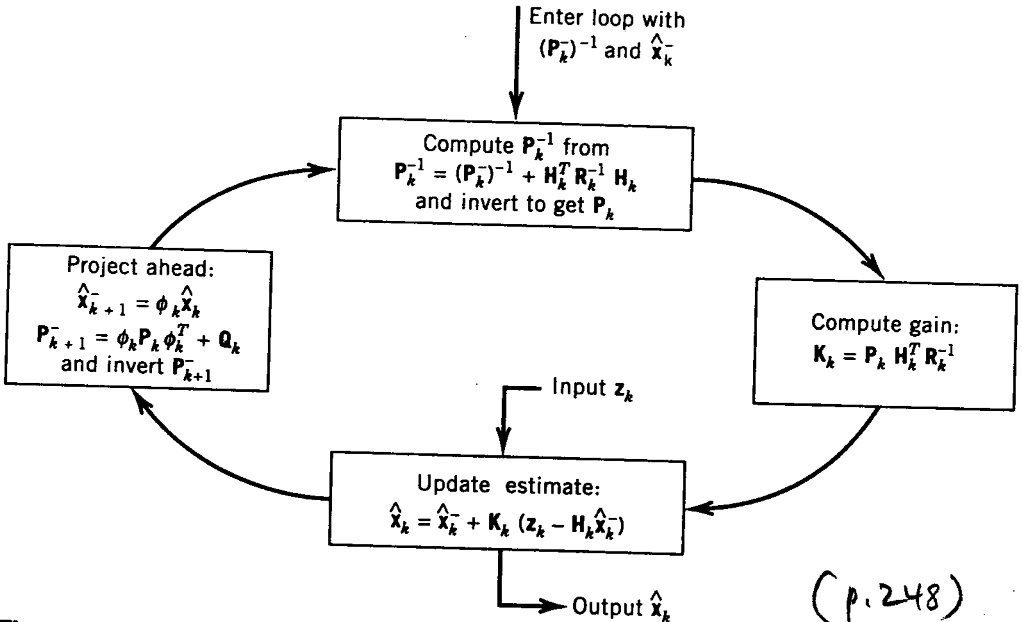


Figure 6.3 Alternative Kalman filter recursive loop.

(p. 248)

References

1. Strang and Borre, Linear Algebra, Geodesy, and GPS, Wellesley-Cambridge Press, 1997

2. Brown and Hwang, Introduction to Random Signals and Applied Kalman Filtering, John Wiley, 1997, 3rd Edition

(next: nonlinear & linearization)