Sequential Measurement Processing

Glenn D. MacGougan

May 11, 2007

Abstract

In Kalman filtering, it is very useful to be able to process measurements sequentially (i.e. one at a time). This method eliminates the need for matrix inversion operations since sequential processing uses scalar measurement updates [1]. The method requires that measurements are uncorrelated. When this is not the case, diagonalizeation of the measurement variance-covariance matrix and a transformation that is a linear combination of the original measurements into a new measurement set is performed. This whitepaper outlines the steps for a sequential measurement processing technique .

1 Best References

The book by Brown and Hwang [1] discusses sequential measurement processing on pages 250-252.

The book by Grewel and Andrews [2] discusses sequential measurement processing on page 226 in the section entitled “Kalman Implementation with Decorrelation.”

2 The Proposed Method

The standard Kalman filter steps are:

1. Prediction:
   $\hat{x}_{k+1}^{-} = \phi_k \hat{x}_k$
   $P_{k+1}^{-} = \phi_k P_k \phi_k^T + Q_k$

2. Computation of Kalman Gain:
   $K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1}$

3. Update:
   $\hat{x}_k^+ = \hat{x}_k^- + K_k(z_k - H_k \hat{x}_k^-)$
   $P_k = (I - K_k H_k) P_k^-$

The standard Extended Kalman filter steps are:

1. Prediction:
   $\hat{x}_{k+1}^{-} = \phi_k \hat{x}_k$
   $P_{k+1}^{-} = \phi_k P_k \phi_k^T + Q_k$

2. Compute Innovations:
   $w = l - \hat{l}$, (measurements - computed (i.e. predicted) measurements).
   $\hat{l} = H_k \hat{x}_{k+1}^-$

3. Computation of Kalman Gain:
   $K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1}$

4. Update:
   $\widehat{dx}_k = K_k(w_k)$
   $\hat{x}_k^+ = \hat{x}_k^- + \widehat{dx}_k$
   $P_k = (I - K_k H_k) P_k^-$

There is an alternative Kalman filter implementation that propagates P_k^{-1} instead of P_k. Since the state variance-covariance information is required for output purposes at each epoch in many cases for GNSS processing, this will not be used.

This method will use the standard Extended Kalman implementation. The steps are as follows:

1. Perform Prediction:
   $\hat{x}_{k+1}^{-} = \phi_k \hat{x}_k$
   $P_{k+1}^{-} = \phi_k P_k \phi_k^T + Q_k$

2. Compute Innovations:
   $w = l - \hat{l}$, (measurements - computed (i.e. predicted) measurements).
   $\hat{l} = H_k \hat{x}_{k+1}^{-}$

3. Perform suitable innovation testing to remove outliers.

4. For each measurement, i, the following steps are performed.

   (a) Compute the $i^t h$ innovation, w_i using the most recent state estimate.

   (b) Compute the Kalman gain for the i^{th} measurement:

      i. Compute h_i, [1xu] the row of the design matrix corresponding to the observation.
      ii. Compute $B = P_k h_i^T$, [ux1].
      iii. Compute $C = (h_i P_k h_i^T + \sigma_i^2)^{-1} = (h_i B + \sigma_i^2)^{-1}$, [1x1] Hence, the uxu inversion is avoided as only a scalar is inverted. C is the variance of the innovation.
      iv. Compute $k_i = P_k h_i^T (h_i P_k h_i^T + \sigma_i^2)^{-1} = BC$, [ux1].

   (c) Then the state variance-covariance matrix is updated:

      i. Compute $D = k_i h_i P_k$.
      ii. Compute $P_k = P_k - D$ (equivalently $P_k = (I - k_i h_i) P_k$.

   (d) Then the states are updated:

      i. $\widehat{dx}_k = k_i(w_i)$
      ii. $\hat{x}_k^{+} = \hat{x}_k^{-} + \widehat{dx}_k$

3 References

1. Brown, R. G., P. Y. C Hwang (1997). Introduction to Random Signals and Applied Kalman Filtering. Third Edition. John Wiley and Sons Inc. pp. 250-252.

2. Grewal, M. S., A. P. Andrews (2001). Kalman Filtering Theory and Practice Using Matlab. Second Edition. John Wiley and Sons Inc.pp. 226.