There are applications for which it may be desirable to base an estimate on data taken over a limited time period during which we have adequate knowledge of the system dynamics and measurement system to yield a near optimal estimate of the state. For example, we may have an onboard filter which bases its estimate of the current state on the immediate past few hours or days of data and ignores prior data.

If we process all available data the filter will saturate and our solution will diverge. One method to prevent divergence is to add process noise to the system. Another solution is the one described here where we limit the amount of data to a finite period of time over which our models for the dynamical and measurement systems are accurate.

A rigorous derivation of the limited memory filter is given by Jazwinski (1970) who derives expressions for the mean and variance-covariance of the conditional density function

$$P\left(\begin{bmatrix} x_n \\ Y_N \end{bmatrix} \right).$$

Here $x_n$ is the desired state at $t_n$ and $Y_N$ are the observations over the time interval of length $N$.

A heuristic derivation of the filter equations is given here. Assume that we have observations $Y_1 \cdots Y_m \cdots Y_n$, and define $N = n - m$. We also have solutions $\hat{x}_n^m$ and $\hat{x}_n^n$ and their associated covariance matrices. Here $\hat{x}_n^m$ is the best estimate of the state deviation vector at $t_n$ based on observations $Y_1 \cdots Y_m$ and $\hat{x}_n^n$ is the best estimate of the state deviation vector at $t_n$ based on all observations $Y_1 \cdots Y_n$.

We may think of $\hat{x}_n^n$ as consisting of two parts; an estimate based on the first $m$ observation plus an estimate based on the final $N$ observations, i.e.,

$$\hat{x}_n^n = \hat{x}_n^m + \hat{x}_n^N.$$

Hence, the solution we desire based on the final $N$ observations is given by

$$\hat{x}_n^N = \hat{x}_n^n - \hat{x}_n^m.$$ (2)

We could simply subtract these two estimates but in doing so we would ignore the statistical knowledge we have contained in $P_n^n$ and $P_n^m$. Consequently, we define a performance index $Q$ which is a weighted sum of squares of the estimation errors in $\hat{x}_n^n$ and $\hat{x}_n^m$. $\hat{x}_n^N$ is then chosen to minimize $Q$, i.e.,
\[ Q = \frac{1}{2} \left[ (\hat{x}_n^m - x_n)^T (P_n^m)^{-1} (\hat{x}_n^m - x_n) - (\hat{x}_n^n - x_n)^T (P_n^n)^{-1} (\hat{x}_n^n - x_n) \right]. \] (3)

In order to minimize \( Q \), we set
\[
\frac{\delta Q}{\delta \hat{x}_n} = 0
\]

\[
\delta Q = - (\hat{x}_n^n - x_n)^T (P_n^n)^{-1} \delta \hat{x}_n + (\hat{x}_n^m - x_n)^T (P_n^m)^{-1} \delta \hat{x}_n = 0.
\] (4)

This must be true for all \( \delta \hat{x}_n \); hence,

\[
- (\hat{x}_n^n - x_n)^T (P_n^n)^{-1} + (\hat{x}_n^m - x_n)^T (P_n^m)^{-1} = 0.
\] (5)

Solving for \( \hat{x}_n^N = x_n \) yields

\[
\hat{x}_n^N = \left[ (P_n^n)^{-1} - (P_n^m)^{-1} \right] \left[ (P_n^n)^{-1} \hat{x}_n^n - (P_n^m)^{-1} \hat{x}_n^m \right]
\] (6)

where \( \hat{x}_n^N \) is the best estimate of \( \hat{x} \) at \( t_n \) based on \( N \) observations. Also,

\[
\hat{x}_n^m = \Phi(t_n, t_m) \hat{x}_m^m
\] (7)

\[
P_n^m = \Phi(t_n, t_m) P_m^m \Phi^T(t_n, t_m)
\] (8)

and \( \hat{x}_n^m \) and \( P_n^m \) are the solution and covariance from processing all \( n \) observations.

Note that the algorithm requires that the filter solution, covariance, and state transition matrix be saved at each stage. The state transition matrix \( \Phi(t_n, t_m) \) is obtained from

\[
\Phi(t_n, t_m) = \Phi(t_n, t_{n-1}) \cdots \Phi(t_{m+2}, t_{m+1}) \Phi(t_{m+1}, t_m)
\] (9)

Reference: