

Fixed Memory Filter for Real-Time Estimation of Noise-Corrupted Signals

Rogelio Luck*

Mississippi State University,
Mississippi State, Mississippi 39762

and
Asok Ray†

Pennsylvania State University,
University Park, Pennsylvania 16802

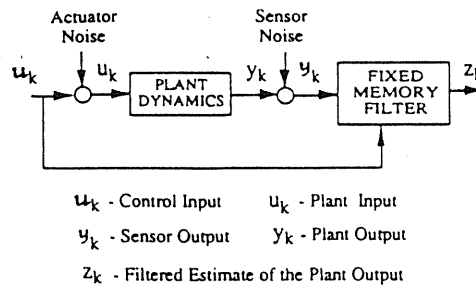


Fig. 1 Interactions between the plant and the filter.

Introduction

IN many engineering applications, such as guidance and navigation of aeronautical vehicles, the signals need to be estimated from noisy measurements in real time. The (minimum-variance) Kalman-Bucy filter^{1,2} that relies on a finite-dimensional state-space model of the plant dynamics and the measurement history is extensively used for this purpose. Techniques of adaptive filtering^{3,4} that are based on the principle of recursive least squares algorithm and rely on an input-output relationship of the plant dynamics are also used. Given a linear (or linearized) time-invariant finite-dimensional model of plant dynamics in the autoregressive moving average (ARMA) setting, we propose a fixed-memory, nonrecursive filter for real-time signal estimation. The key idea is to construct a simple, nonrecursive filter based on weighted averaging of a finite array of past values of measured inputs and outputs of the plant. A fixed memory filter is very useful in the applications of smart sensors and/or active sensors where memory constraints are tight and computationally efficient algorithms are required for real-time implementation on a microchip collocated with the sensor.⁵ The filter memory length depends on the plant dynamics and the allowable bound of estimation error and, for real-time applications, is often selected as a tradeoff between execution time and accuracy of the filter.

This Engineering Note presents the concept and formulation of a fixed-memory filter based on an ARMA model of plant dynamics. The criteria for selection of the filter memory length are not included in this Note. Although the filter algorithm is derived for single-input single-output (SISO) systems, the proposed filter can be extended to multi-input multi-output (MIMO) systems.

Filter Structure

Figure 1 shows a schematic diagram of the SISO plant and the finite-memory filter where u_k and y_k are the actual input and output of the dynamic plant, respectively, u_k is the control input, and y_k is the sensor output at the k th sampling instant. We assume that the sensor noise and the actuator noise are zero-mean, mutually uncorrelated, and stationary white sequences with variances σ_y^2 and σ_u^2 , respectively. The ARMA model of the plant dynamics relating the actual plant output to the actual plant input is expressed in the deterministic setting as

$$y_{k+1} = aY_n(k) + bU_r(k) = [a \ b] \begin{bmatrix} Y_n(k) \\ U_r(k) \end{bmatrix} \quad (1)$$

where

$$a = [a_n a_{n-1} \dots a_1]; \quad b = [b_r b_{r-1} \dots b_1]$$

$$Y_n(k) = [y_{k-n+1} y_{k-n+2} \dots y_k]^T$$

$$U_r(k) = [u_{k-r+1} u_{k-r+2} \dots u_k]^T$$

In the given equation, a is a $1 \times n$ matrix of the autoregressive coefficients, b is a $1 \times r$ matrix of the moving average coefficients, and $U_r(k)$ and $Y_n(k)$ represent the most recent history of r actual plant inputs and n actual plant outputs, respectively, at time k . Similarly, $\mathcal{U}_r(k)$ and $\mathcal{Y}_n(k)$ represent the most recent history of r control inputs and n sensor outputs, respectively, at time k .

The objective is to obtain an estimate, hereafter called the filtered estimate z_{k+1} , of the actual plant output y_{k+1} based on the available history of control inputs and sensor outputs, i.e., $\mathcal{Y}_{n+m}(k)$ and $\mathcal{U}_{r+m}(k)$. The filter memory size m depends on the type of filter used. For example, if $m = k$, i.e., the filter memory grows with time, then a recursive filter such as the Kalman-Bucy filter will yield optimal results in the least mean square sense, such as the best filter (linear or nonlinear) for Gaussian noise and best linear filter for other types of noise distributions.¹ The memory size m has been taken to be a constant integer in the derivation of the proposed nonrecursive filter.

Errors in the Unfiltered Estimate

Design of the filter should account for errors in the plant model due to uncertainties in the model parameter matrices a and b , and also to noise in the sensor and actuator. The ARMA model in Eq. (1) can be used to obtain an estimate \hat{y}_k of y_k as follows:

$$\hat{y}_{k+1} = [a \ \beta] \begin{bmatrix} \mathcal{Y}_n(k) \\ \mathcal{U}_r(k) \end{bmatrix} \quad (2)$$

where a and β are the available unbiased estimates of the model parameter matrices a and b , respectively and $\mathcal{U}_r(k)$ and $\mathcal{Y}_n(k)$ represent the history of the control input and sensor output, respectively, as defined earlier. The following errors are identified in the equation for obtaining the estimate \hat{y}_{k+1} , hereafter called the unfiltered estimate.

$$\Delta a = a - \hat{a} \text{ where } E\{\Delta a\} = 0 \quad (3)$$

$$\Delta b = \beta - \hat{b} \text{ where } E\{\Delta b\} = 0 \quad (4)$$

$$\Delta Y_n(k) = \mathcal{Y}_n(k) - Y_n(k) \text{ where } E\{\Delta Y_n(k)\} = 0 \forall k \quad (5)$$

$$\Delta U_r(k) = \mathcal{U}_r(k) - U_r(k) \text{ where } E\{\Delta U_r(k)\} = 0 \forall k \quad (6)$$

Received Feb. 18, 1993; revision received May 28, 1993; accepted for publication July 20, 1993. Copyright © 1993 by R. Luck and A. Ray. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission.

*Assistant Professor of Mechanical Engineering.

†Professor of Mechanical Engineering. Associate Fellow AIAA.

The estimation error is obtained by manipulating Eqs. (1-6)

$$\begin{aligned} \Delta y_{k+1} &= \hat{y}_{k+1} - y_{k+1} \\ &= [a \ \delta] \begin{bmatrix} \Delta Y_n(k) \\ \Delta U_r(k) \end{bmatrix} + [\Delta a \ \Delta b] \begin{bmatrix} Y_n(k) \\ U_r(k) \end{bmatrix} \end{aligned} \quad (7)$$

The mean square error is then expressed as the second moment of the estimation error

$$E\{|\Delta y_{k+1}|^2\} = E\left\{ \left[[a \ \delta] Y_n(k)^T U_r(k)^T \right] \begin{bmatrix} \Delta Y_n(k) \\ \Delta U_r(k) \\ \Delta a^T \\ \Delta b^T \end{bmatrix} \right\}^2 \quad (8)$$

If $R = W^T W$ is the covariance matrix of the vector containing the Δ terms, then Eq. (8) becomes

$$E\{|\Delta y_{k+1}|^2\} = \|W [a \ \delta] Y_n(k)^T U_r(k)^T\|_2^2 \quad (9)$$

where $\|\cdot\|_2$ is the standard ℓ_2 norm.⁶

Remark 1: If there is no modeling error, i.e., both Δa and Δb are identically zero, then Eq. (9) becomes

$$E\{|\Delta y_{k+1}|^2\} = \sigma_y^2 (aa^T) + \sigma_v^2 (\delta\delta^T) \quad (10)$$

because the actuator noise and sensor noise have been assumed to be white sequences with variances σ_y^2 and σ_v^2 , respectively. Thus, the root mean square error of the prediction is the ℓ_2 norm of the vector $[\sigma_y a \ \sigma_v \delta]$.

Filtered Estimate

It is seen from Eq. (7) that the error Δy_{k+1} in the unfiltered estimate can be interpreted as a weighted sum of the error terms in Eqs. (3-6). Intuitively, if the weighting terms in Eq. (7) are selected correctly and the summation is performed over a longer history of errors it may be possible to average out some of the random fluctuations due to the errors. This motivates the following modification of the unfiltered estimate in Eq. (2) to obtain the filtered estimate:

$$z_{k+1} = [a \ \delta] \begin{bmatrix} \mathcal{Y}_n(k) \\ \mathcal{U}_r(k) \end{bmatrix} + f[\mathcal{Y}_{n+m}(k), \mathcal{U}_{r+m}(k)] \quad (11)$$

where $f[\mathcal{Y}_{n+m}(k), \mathcal{U}_{r+m}(k)]$ is a (scalar) function of the most recent history of $(n+m)$ sensor outputs and $(r+m)$ control inputs to the plant.

Since all of the elements of $\mathcal{Y}_n(k)$ and $\mathcal{U}_r(k)$ are also contained in $\mathcal{Y}_{n+m}(k)$ and $\mathcal{U}_{r+m}(k)$, respectively, the right-hand side of Eq. (11) is solely as a function of $\mathcal{Y}_{n+m}(k)$ and $\mathcal{U}_{r+m}(k)$. Therefore, $f(\cdot, \cdot)$ in Eq. (11) represents the difference between the filtered estimate and the unfiltered estimate, i.e., $f[\mathcal{Y}_{n+m}(k), \mathcal{U}_{r+m}(k)] = z_{k+1} - \hat{y}_{k+1}$.

A mean square criterion is enforced to ensure that the filtered estimate z_k is indeed an improvement over the unfiltered estimate \hat{y}_k .

$$E\{|z_k - y_k|^2\} \leq E\{|\hat{y}_k - y_k|^2\} \quad (12)$$

Also, in the absence of actuator noise and sensor noise, i.e., if $\mathcal{Y}_n(k) = Y_n(k)$, $\mathcal{U}_r(k) = U_r(k)$, we must have $z_k = \hat{y}_k$ for every k . Therefore, by comparing Eqs. (11) and (2), this requirement reduces to

$$f[Y_{n+m}(k), U_{r+m}(k)] = 0 \quad (13)$$

By restricting the choice of the filter structure to linear mappings, $f(\cdot, \cdot)$ can be represented as a function of a

$1 \times (r+n+2m)$ weight, and Eq. (13) is consequently expressed as

$$f[Y_{n+m}(k), U_{r+m}(k)] = q^T \begin{bmatrix} Y_{n+m}(k) \\ U_{r+m}(k) \end{bmatrix} = 0 \quad (14)$$

The $(n+r+2m) \times 1$ matrix q can be identified by using the available information provided by the ARMA model (1) of plant dynamics based on the assumption of perfect model, i.e., $\Delta a = 0$ in Eq. (3) and $\Delta b = 0$ in Eq. (4). This is accomplished by expressing the plant outputs in terms of the initial state

$$U_{r+m}(k) = A Y_n(k-m) + B U_{r+m}(k) = [A \ B] \begin{bmatrix} Y_n(k-m) \\ U_r(k+m) \end{bmatrix} \quad (15)$$

where the matrices A and B are to be constructed in terms of the elements of the model parameter matrices a and b . A procedure to find A and B is illustrated by the following example.

Example: Let $n = 2$, $r = 1$, and $m = 2$. Then,

$$y_{k+1} = a_1 y_k + a_2 y_{k-1} + b_1 u_k$$

$$y_{k+2} = a_1(a_1 y_k + a_2 y_{k-1} + b_1 u_k) + a_2 y_k + b_1 u_{k+1}$$

These equations are expressed in matrix notation as

$$\begin{bmatrix} y_{k-1} \\ y_k \\ y_{k+1} \\ y_{k+2} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ a_2 & a_1 \\ a_1 a_2 & a_1 a_1 + a_2 \end{bmatrix} \begin{bmatrix} y_{k-1} \\ y_k \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ b & 0 & 0 \\ a_1 b & b & 0 \end{bmatrix} \begin{bmatrix} u_k \\ u_{k+1} \\ u_{k+2} \end{bmatrix}$$

Using the same notation as in Eq. (15) and replacing k by $(k-2)$ yields

$$Y_4(k) = A Y_2(k-2) + B U_3(k)$$

The procedure given in the example can be generalized to include arbitrary values of m , n , and r as follows:

$$A_{n+j} = a \begin{bmatrix} A_{n+j-1} \\ A_{n+j-2} \\ A_{n+j-3} \\ \vdots \\ A_j \end{bmatrix}; \quad H = \begin{bmatrix} 0_{n \times (r+m)} \\ b^T 0 \dots 0 0 \\ 0 b^T \dots 0 0 \\ \vdots \\ 0 0 \dots b^T 0 \end{bmatrix}$$

$$B_{n+j} = a \begin{bmatrix} H_{n+j-1} \\ H_{n+j-2} \\ H_{n+j-3} \\ \vdots \\ H_j \end{bmatrix} \quad (16)$$

where

$$A = \begin{bmatrix} I_{n \times n} \\ A_{n+1} \\ A_{n+2} \\ \vdots \\ A_{n+m} \end{bmatrix}; \quad B = \begin{bmatrix} 0_{n \times (r+m)} \\ B_{n+1} \\ B_{n+2} \\ \vdots \\ B_{n+m} \end{bmatrix}$$

and $j = 1, 2, \dots, m$, and A_i, B_i , and H_i denote the i th row of A, B , and H , respectively; a is a $1 \times n$ matrix of the autoregressive coefficients, and b is a $1 \times r$ matrix of the moving

average
ARM
output
dyn
zero
is to
(11).
found
proce
Eq. (1

q^T
[

It fol
 $q^T D$
basis
a line
is a re
Using
ing fil

Since a
tain
side of
and U_r
and β

Substit
express

The n
of ϕ to
ner. Thi
minimiz
necessar
constrain
 ϕ that s
because
equal to
One pos
with all
 $z_k(\phi) = \hat{y}$
will hold
The m
similar to

where W
definite)
right-han
minimiz

average coefficients as defined earlier in Eq. (1) for the ARMA model of plant dynamics. Since the current plant output y_k is not affected by the current plant input u_k in the dynamic model, the last column of the matrix B is identically zero. The rationale for keeping this apparently useless column is to maintain consistency with the notation defined in Eq. (11). Alternatively, the matrices A and B in Eq. (16) can be found using the observability canonical representation⁷ in a procedure similar to that shown in the example. Substituting Eq. (15) into Eq. (14) yields

$$q^T \begin{bmatrix} Y_{n+m}(k) \\ U_{r+m}(k) \end{bmatrix} = q^T D \begin{bmatrix} Y_n(k-m) \\ U_r(k-m) \end{bmatrix} = 0; \text{ with } D = \begin{bmatrix} A & B \\ 0 & I \end{bmatrix} \quad (17)$$

It follows that q must lie in the left null space of D , i.e., $q^T D = 0$. Let N denote any matrix with its rows forming a basis for the left null space of D . Then, q can be expressed as a linear combination of the rows of N , i.e., $q^T = \phi N$, where ϕ is a row vector of the coefficients of the linear combination. Using Eq. (13) and assuming that $f(\cdot, \cdot)$ is linear, the following filter structure is obtained from Eq. (11) as

$$z_{k+1} = [u \ \delta] \begin{bmatrix} Y_n(k) \\ U_r(k) \end{bmatrix} + \phi N \begin{bmatrix} Y_{n+m}(k) \\ U_{r+m}(k) \end{bmatrix} \quad (18)$$

Since all of the elements of $Y_n(k)$ and $U_r(k)$ are also contained in $Y_{n+m}(k)$ and $U_{r+m}(k)$, respectively, the right-hand side of Eq. (18) can be expressed solely in terms of $Y_{n+m}(k)$ and $U_{r+m}(k)$. To this effect we introduce the row vectors α and β as

$$\alpha = [0 \ 0 \ \dots \ 0 \ u] \text{ such that } \alpha Y_{n+m}(k) = u Y_n(k)$$

$$\beta = [0 \ 0 \ \dots \ 0 \ \delta] \text{ such that } \beta U_{r+m}(k) = \delta U_r(k)$$

Substituting α and β in Eq. (18), the filtered estimate can be expressed as

$$z_{k+1} = ([\alpha \ \beta] + \phi N) \begin{bmatrix} Y_{n+m}(k) \\ U_{r+m}(k) \end{bmatrix} \quad (19)$$

The next step in the construction of the filter is the selection of ϕ to satisfy the requirement [Eq. (12)] in an optimal manner. This is done by finding an optimal ϕ , denoted as ϕ^o , that minimizes the mean square error $E\{[z_k(\phi) - y_k]^2\}$. Also, it is necessary to show that the least mean square error satisfies the constraint imposed by Eq. (12). It is only necessary to find one ϕ that satisfies Eq. (12) to show that ϕ^o satisfies Eq. (12) because the least mean square error (due to ϕ^o) is smaller or equal to the mean square error from any other choice of ϕ . One possible choice of ϕ that will satisfy Eq. (12) is the one with all of the coefficients equal to zero; this choice yields $z_k(\phi) = \hat{y}_k$ by comparing Eqs. (18) and (2), and the equality will hold in Eq. (12).

The mean square error is minimized using an expression similar to Eq. (8),

$$E\{|z_{k+1} - y_{k+1}|^2\} = E\left\{ \left\| [([\alpha \ \beta] + \phi N) Y_{n+m}(k)^T U_{r+m}(k)^T] \begin{bmatrix} \Delta Y_{n+m}(k) \\ \Delta U_{r+m}(k) \\ \Delta a^T \\ \Delta b^T \end{bmatrix} \right\|^2 \right\} \\ = \|W[(\alpha \ \beta) + \phi N] Y_{n+m}(k)^T U_{r+m}(k)^T\|^2 \quad (20)$$

where $W^T W = R$ is the covariance matrix (which is positive definite) of the vector containing the Δ terms in Eq. (20). The right-hand side of Eq. (20) is minimized with respect to ϕ by minimizing the ℓ_2 norm of $W[(\alpha \ \beta) + \phi N]^T$. This is done by

solving the overconstrained problem $W[(\alpha \ \beta) + \phi N]^T = 0$ in the least squares sense to obtain

$$(\phi^o)^T = -[(W^T N)^T (W^T N)^T]^{-1} (W^T N)^T W^T [\alpha \ \beta]^T \\ = -[NRN^T]^{-1} NR [\alpha \ \beta]^T \quad (21)$$

The final form of the moving average filter is obtained by substituting Eq. (21) into Eq. (19)

$$z_{k+1} = ([\alpha \ \beta] - [\alpha \ \beta] RN^T [NRN^T]^{-1} N) \begin{bmatrix} Y_{n+m}(k) \\ U_{r+m}(k) \end{bmatrix} \\ = [\alpha \ \beta] (I - RN^T [NRN^T]^{-1} N) \begin{bmatrix} Y_{n+m}(k) \\ U_{r+m}(k) \end{bmatrix} \quad (22)$$

Remark 2: If the covariance matrix R is set equal to the identity, then the idempotent matrix $P = (I - N^T [NRN^T]^{-1} N)$, i.e., $P = P^T$ and $P = P^2$, is recognized to be the orthogonal projection operator^{6,8} onto the column space of the matrix D . This corresponds to ordinary least squares. If R is not the identity matrix, then the projection is orthogonal to the column space of the matrix $W^T D$. This corresponds to the weighted least squares.⁸ It follows that filtering is achieved by canceling all inconsistencies in the sensor outputs relative to the column space of $W^T D$.

Remark 3: As expected, filtering is performed before prediction in Eq. (22). The data, obtained from the history of control inputs and sensor outputs, is filtered by using the weighted projection operator, and then the prediction is obtained by multiplication by $[\alpha \ \beta]$ to obtain $z_{k+1|k}$. A slight modification of Eq. (22) that yields the filtered estimate, i.e., $z_{k|k}$, instead of the predicted estimate, $z_{k+1|k}$ follows.

$$\alpha' = [0 \ 0 \ \dots \ 0 \ 0 \ 1] \text{ and } \beta' = [0 \ 0 \ \dots \ 0 \ 0 \ 0]$$

are set to obtain

$$z_{k|k} = [\alpha' \ \beta'] (I - RN^T [NRN^T]^{-1} N) \begin{bmatrix} Y_{n+m}(k) \\ U_{r+m}(k) \end{bmatrix} \quad (23)$$

Similarly, for $z_{k-1|k}$, i.e., first-order smoothing, $\alpha' = [0 \ 0 \ \dots \ 0 \ 1 \ 0]$ and $\beta' = [0 \ 0 \ \dots \ 0 \ 0 \ 0]$ are set to obtain

$$z_{k-1|k} = [\alpha' \ \beta'] (I - RN^T [NRN^T]^{-1} N) \begin{bmatrix} Y_{n+m}(k) \\ U_{r+m}(k) \end{bmatrix} \quad (24)$$

In this way, for higher order smoothing, it is only necessary to set the corresponding term in α' to 1.

Remark 4: It follows from Eq. (20) that ϕ cannot be used to directly diminish the effect of the multiplicative error $[Y_{n+m}(k)^T \Delta a^T + U_{r+m}(k)^T \Delta b^T]$. Furthermore, for a particular filter realization Δa and Δb may assume constant values, and the multiplicative error can be substantial. Some instances may arise where the length of the filter will worsen the multiplicative error. For example, if $\Delta a = 0$, Δb is a nonzero constant vector, and $u(k) = u = \text{const}$, then the multiplicative error $(r+m)u\Delta b$ is directly proportional to the length of the

filter and an infinite-dimensional filter will diverge. However, it is possible to indirectly diminish the effect of the multiplicative errors by increasing the actuator noise covariance, i.e., by emphasizing the sensor data over the plant model.

The estimation error is obtained by manipulating Eqs. (1-6)

$$\begin{aligned} \Delta y_{k+1} &= \hat{y}_{k+1} - y_{k+1} \\ &= [a \ \delta] \begin{bmatrix} \Delta Y_n(k) \\ \Delta U_r(k) \end{bmatrix} + [\Delta a \ \Delta b] \begin{bmatrix} Y_n(k) \\ U_r(k) \end{bmatrix} \end{aligned} \quad (7)$$

The mean square error is then expressed as the second moment of the estimation error

$$E\{|\Delta y_{k+1}|^2\} = E\left\{ \left[[a \ \delta] Y_n(k)^T U_r(k)^T \begin{bmatrix} \Delta Y_n(k) \\ \Delta U_r(k) \\ \Delta a^T \\ \Delta b^T \end{bmatrix} \right]^2 \right\} \quad (8)$$

If $R = W^T W$ is the covariance matrix of the vector containing the Δ terms, then Eq. (8) becomes

$$E\{|\Delta y_{k+1}|^2\} = \|W [a \ \delta] Y_n(k)^T U_r(k)^T\|^2 \quad (9)$$

where $\|\cdot\|_2$ is the standard ℓ_2 norm.⁶

Remark 1: If there is no modeling error, i.e., both Δa and Δb are identically zero, then Eq. (9) becomes

$$E\{\Delta y_{k+1}\}^2 = \sigma_y^2 (aa^T) + \sigma_v^2 (bb^T) \quad (10)$$

because the actuator noise and sensor noise have been assumed to be white sequences with variances σ_y^2 and σ_v^2 , respectively. Thus, the root mean square error of the prediction is the ℓ_2 norm of the vector $[a \ \delta]$.

Filtered Estimate

It is seen from Eq. (7) that the error Δy_{k+1} in the unfiltered estimate can be interpreted as a weighted sum of the error terms in Eqs. (3-6). Intuitively, if the weighting terms in Eq. (7) are selected correctly and the summation is performed over a longer history of errors it may be possible to average out some of the random fluctuations due to the errors. This motivates the following modification of the unfiltered estimate in Eq. (2) to obtain the filtered estimate:

$$z_{k+1} = [a \ \delta] \begin{bmatrix} \mathcal{Y}_n(k) \\ \mathcal{U}_r(k) \end{bmatrix} + f[\mathcal{Y}_{n+m}(k), \mathcal{U}_{r+m}(k)] \quad (11)$$

where $f[\mathcal{Y}_{n+m}(k), \mathcal{U}_{r+m}(k)]$ is a (scalar) function of the most recent history of $(n+m)$ sensor outputs and $(r+m)$ control inputs to the plant.

Since all of the elements of $\mathcal{Y}_n(k)$ and $\mathcal{U}_r(k)$ are also contained in $\mathcal{Y}_{n+m}(k)$ and $\mathcal{U}_{r+m}(k)$, respectively, the right-hand side of Eq. (11) is solely as a function of $\mathcal{Y}_{n+m}(k)$ and $\mathcal{U}_{r+m}(k)$. Therefore, $f(\cdot, \cdot)$ in Eq. (11) represents the difference between the filtered estimate and the unfiltered estimate, i.e., $f[\mathcal{Y}_{n+m}(k), \mathcal{U}_{r+m}(k)] = z_{k+1} - \hat{y}_{k+1}$.

A mean square criterion is enforced to ensure that the filtered estimate z_k is indeed an improvement over the unfiltered estimate \hat{y}_k .

$$E\{|z_k - y_k|^2\} \leq E\{|\hat{y}_k - y_k|^2\} \quad (12)$$

Also, in the absence of actuator noise and sensor noise, i.e., if $\mathcal{Y}_n(k) = Y_n(k)$, $\mathcal{U}_r(k) = U_r(k)$, we must have $z_k = \hat{y}_k$ for every k . Therefore, by comparing Eqs. (11) and (2), this requirement reduces to

$$f[Y_{n+m}(k), U_{r+m}(k)] = 0 \quad (13)$$

By restricting the choice of the filter structure to linear mappings, $f(\cdot, \cdot)$ can be represented as a function of a

$1 \times (r+n+2m)$ weight, and Eq. (13) is consequently expressed as

$$f[Y_{n+m}(k), U_{r+m}(k)] = q^T \begin{bmatrix} Y_{n+m}(k) \\ U_{r+m}(k) \end{bmatrix} = 0 \quad (14)$$

The $(n+r+2m) \times 1$ matrix q can be identified by using the available information provided by the ARMA model (1) of plant dynamics based on the assumption of perfect model, i.e., $\Delta a = 0$ in Eq. (3) and $\Delta b = 0$ in Eq. (4). This is accomplished by expressing the plant outputs in terms of the initial state

$$U_{r+m}(k) = AY_n(k-m) + BU_{r+m}(k) = [A \ B] \begin{bmatrix} Y_n(k-m) \\ U_r(k+m) \end{bmatrix} \quad (15)$$

where the matrices A and B are to be constructed in terms of the elements of the model parameter matrices a and b . A procedure to find A and B is illustrated by the following example.

Example: Let $n = 2$, $r = 1$, and $m = 2$. Then,

$$y_{k+1} = a_1 y_k + a_2 y_{k-1} + b_1 u_k$$

$$y_{k+2} = a_1(a_1 y_k + a_2 y_{k-1} + b_1 u_k) + a_2 y_k + b_1 u_{k+1}$$

These equations are expressed in matrix notation as

$$\begin{bmatrix} y_{k-1} \\ y_k \\ y_{k+1} \\ y_{k+2} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ a_2 & a_1 \\ a_1 a_2 & a_1 a_1 + a_2 \end{bmatrix} \begin{bmatrix} y_{k-1} \\ y_k \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ b & 0 & 0 \\ a_1 b & b & 0 \end{bmatrix} \begin{bmatrix} u_k \\ u_{k+1} \\ u_{k+2} \end{bmatrix}$$

Using the same notation as in Eq. (15) and replacing k by $(k-2)$ yields

$$Y_4(k) = AY_2(k-2) + BU_3(k)$$

The procedure given in the example can be generalized to include arbitrary values of m , n , and r as follows:

$$A_{n+j} = a \begin{bmatrix} A_{n+j-1} \\ A_{n+j-2} \\ A_{n+j-3} \\ \vdots \\ A_j \end{bmatrix}; \quad H = \begin{bmatrix} 0_{n \times (r+m)} \\ b^T 0 \cdots 0 \\ 0 b^T \cdots 0 \\ \vdots \\ 0 0 \cdots b^T 0 \end{bmatrix} \quad (16)$$

$$B_{n+j} = a \begin{bmatrix} H_{n+j-1} \\ H_{n+j-2} \\ H_{n+j-3} \\ \vdots \\ H_j \end{bmatrix}$$

where

$$A = \begin{bmatrix} I_{n \times n} \\ A_{n+1} \\ A_{n+2} \\ \vdots \\ A_{n+m} \end{bmatrix}; \quad B = \begin{bmatrix} 0_{n \times (r+m)} \\ B_{n+1} \\ B_{n+2} \\ \vdots \\ B_{n+m} \end{bmatrix}$$

and $j = 1, 2, \dots, m$, and A_i , B_i , and H_i denote the i th row of A , B , and H , respectively; a is a $1 \times n$ matrix of the autoregressive coefficients, and b is a $1 \times r$ matrix of the moving

avera
ARM
outp
dyn
zero
is to
(11).
foun
proce
Eq. (1

q^T

It fol
 $q^T D$
basis
a line
is a r
Using
ing fil

Since
tain
side of
and U
and β

Substit
express

The n
of ϕ to
ner. Thi
minimiz
necessar
constrain
 ϕ that s
because
equal to
One pos
with all
 $z_k(\phi) = \hat{y}_k$
will hold
The m
similar to

where W
(definite)
right-hanc
minimizin

average coefficients as defined earlier in Eq. (1) for the ARMA model of plant dynamics. Since the current plant output y_k is not affected by the current plant input u_k in the dynamic model, the last column of the matrix B is identically zero. The rationale for keeping this apparently useless column is to maintain consistency with the notation defined in Eq. (11). Alternatively, the matrices A and B in Eq. (16) can be found using the observability canonical representation⁷ in a procedure similar to that shown in the example. Substituting Eq. (15) into Eq. (14) yields

$$q^T \begin{bmatrix} Y_{n+m}(k) \\ U_{r+m}(k) \end{bmatrix} = q^T D \begin{bmatrix} Y_n(k-m) \\ U_r(k-m) \end{bmatrix} = 0; \text{ with } D = \begin{bmatrix} A & B \\ 0 & I \end{bmatrix} \quad (17)$$

It follows that q must lie in the left null space of D , i.e., $q^T D = 0$. Let N denote any matrix with its rows forming a basis for the left null space of D . Then, q can be expressed as a linear combination of the rows of N , i.e., $q^T = \phi N$, where ϕ is a row vector of the coefficients of the linear combination. Using Eq. (13) and assuming that $f(\cdot, \cdot)$ is linear, the following filter structure is obtained from Eq. (11) as

$$z_{k+1} = [\alpha \ \beta] \begin{bmatrix} Y_n(k) \\ U_r(k) \end{bmatrix} + \phi N \begin{bmatrix} Y_{n+m}(k) \\ U_{r+m}(k) \end{bmatrix} \quad (18)$$

Since all of the elements of $Y_n(k)$ and $U_r(k)$ are also contained in $Y_{n+m}(k)$ and $U_{r+m}(k)$, respectively, the right-hand side of Eq. (18) can be expressed solely in terms of $Y_{n+m}(k)$ and $U_{r+m}(k)$. To this effect we introduce the row vectors α and β as

$$\alpha = [0 \ 0 \ \dots \ 0 \ a] \text{ such that } \alpha^T Y_{n+m}(k) = a Y_n(k)$$

$$\beta = [0 \ 0 \ \dots \ 0 \ b] \text{ such that } \beta^T U_{r+m}(k) = b U_r(k)$$

Substituting α and β in Eq. (18), the filtered estimate can be expressed as

$$z_{k+1} = ([\alpha \ \beta] + \phi N) \begin{bmatrix} Y_{n+m}(k) \\ U_{r+m}(k) \end{bmatrix} \quad (19)$$

The next step in the construction of the filter is the selection of ϕ to satisfy the requirement [Eq. (12)] in an optimal manner. This is done by finding an optimal ϕ , denoted as ϕ^o , that minimizes the mean square error $E\{(z_k(\phi) - y_k)^2\}$. Also, it is necessary to show that the least mean square error satisfies the constraint imposed by Eq. (12). It is only necessary to find one ϕ that satisfies Eq. (12) to show that ϕ^o satisfies Eq. (12) because the least mean square error (due to ϕ^o) is smaller or equal to the mean square error from any other choice of ϕ . One possible choice of ϕ that will satisfy Eq. (12) is the one with all of the coefficients equal to zero; this choice yields $z_r(\phi) = \hat{y}_k$ by comparing Eqs. (18) and (2), and the equality will hold in Eq. (12).

The mean square error is minimized using an expression similar to Eq. (8),

$$E\{|z_{k+1} - y_{k+1}|^2\} = E\left\{ \left\| \begin{bmatrix} ([\alpha \ \beta] + \phi N) Y_{n+m}(k) \\ U_{r+m}(k) \end{bmatrix} \begin{bmatrix} \Delta Y_{n+m}(k) \\ \Delta U_{r+m}(k) \\ \Delta a^T \\ \Delta b^T \end{bmatrix} \right\|^2 \right\}$$

$$= \|W([\alpha \ \beta] + \phi N) Y_{n+m}(k) U_{r+m}(k)^T\|_2^2 \quad (20)$$

where $W^T W = R$ is the covariance matrix (which is positive definite) of the vector containing the Δ terms in Eq. (20). The right-hand side of Eq. (20) is minimized with respect to ϕ by minimizing the ℓ_2 norm of $W([\alpha \ \beta] + \phi N)^T$. This is done by

solving the overconstrained problem $W([\alpha \ \beta] + \phi N)^T = 0$ in the least squares sense to obtain

$$(\phi^o)^T = -[(WN^T)^T(WN^T)]^{-1}(WN^T)^T W[\alpha \ \beta]^T$$

$$= -[NRN^T]^{-1}NR[\alpha \ \beta]^T \quad (21)$$

The final form of the moving average filter is obtained by substituting Eq. (21) into Eq. (19)

$$z_{k+1} = ([\alpha \ \beta] - [\alpha \ \beta]RN^T[NRN^T]^{-1}N) \begin{bmatrix} Y_{n+m}(k) \\ U_{r+m}(k) \end{bmatrix}$$

$$= [\alpha \ \beta](I - RN^T[NRN^T]^{-1}N) \begin{bmatrix} Y_{n+m}(k) \\ U_{r+m}(k) \end{bmatrix} \quad (22)$$

Remark 2: If the covariance matrix R is set equal to the identity, then the idempotent matrix $P = (I - N^T[NN^T]^{-1}N)$, i.e., $P = P^T$ and $P = P^2$, is recognized to be the orthogonal projection operator^{6,8} onto the column space of the matrix D . This corresponds to ordinary least squares. If R is not the identity matrix, then the projection is orthogonal to the column space of the matrix $W^T D$. This corresponds to the weighted least squares.⁸ It follows that filtering is achieved by canceling all inconsistencies in the sensor outputs relative to the column space of $W^T D$.

Remark 3: As expected, filtering is performed before prediction in Eq. (22). The data, obtained from the history of control inputs and sensor outputs, is filtered by using the weighted projection operator, and then the prediction is obtained by multiplication by $[\alpha \ \beta]$ to obtain $z_{k+1|k}$. A slight modification of Eq. (22) that yields the filtered estimate, i.e., $z_{k|k}$, instead of the predicted estimate, $z_{k+1|k}$ follows.

$$\alpha' = [0 \ 0 \ \dots \ 0 \ 0 \ 1] \text{ and } \beta' = [0 \ 0 \ \dots \ 0 \ 0 \ 0]$$

are set to obtain

$$z_{k|k} = [\alpha' \ \beta'](I - RN^T[NRN^T]^{-1}N) \begin{bmatrix} Y_{n+m}(k) \\ U_{r+m}(k) \end{bmatrix} \quad (23)$$

Similarly, for $z_{k-1|k}$, i.e., first-order smoothing, $\alpha' = [0 \ 0 \ \dots \ 0 \ 1 \ 0]$ and $\beta' = [0 \ 0 \ \dots \ 0 \ 0 \ 0]$ are set to obtain

$$z_{k-1|k} = [\alpha' \ \beta'](I - RN^T[NRN^T]^{-1}N) \begin{bmatrix} Y_{n+m}(k) \\ U_{r+m}(k) \end{bmatrix} \quad (24)$$

In this way, for higher order smoothing, it is only necessary to set the corresponding term in α' to 1.

Remark 4: It follows from Eq. (20) that ϕ cannot be used to directly diminish the effect of the multiplicative error $[Y_{n+m}(k)^T \Delta a^T + U_{r+m}(k)^T \Delta b^T]$. Furthermore, for a particular filter realization Δa and Δb may assume constant values, and the multiplicative error can be substantial. Some instances may arise where the length of the filter will worsen the multiplicative error. For example, if $\Delta a = 0$, Δb is a nonzero constant vector, and $u(k) = u = \text{const}$, then the multiplicative error $(r+m)u\Delta b$ is directly proportional to the length of the

filter and an infinite-dimensional filter will diverge. However, it is possible to indirectly diminish the effect of the multiplicative errors by increasing the actuator noise covariance, i.e., by emphasizing the sensor data over the plant model.

The performance of the filter can be expressed as the ratio of the weighted l_2 norm of the filter coefficients as shown next in decibels for the case of no modeling errors (i.e., $\Delta a = 0$ and $\Delta b = 0$), and identically distributed independent sensor noise with variance σ_y^2 and identically distributed independent actuator noise with variance σ_u^2 .

$$10 \log \left[\frac{E[(z_k - y_k)^2]}{E[(\hat{y}_k - y_k)^2]} \right] = 10 \log \left[\frac{\sigma_v^2(\alpha^T \alpha^T) + \sigma_u^2(\beta^T \beta^T)}{\sigma_y^2(\alpha^T) + \sigma_u^2(\beta^T)} \right] \quad (25)$$

where α^T and β^T are of the same dimensions as α and β and are given by

$$[\alpha^T \ \beta^T] = [\alpha \ \beta] (I - RN^T[NRN^T]^{-1}N) \quad (26)$$

Conclusions

A finite memory filter is presented for real-time applications, such as smart sensors and/or active sensors, where memory constraints are tight and computationally efficient algorithms are required for implementation on a microchip collocated with the sensor. It is assumed that an ARMA model of the plant dynamics is available. The key concept is the construction of a nonrecursive filter based on weighted averaging of a finite array of past values of the measured inputs and outputs of the plant. Although the filter algorithm has been derived for SISO systems, it can be extended, in principle, to MIMO systems.

References

¹Jazwinski, A. H., *Stochastic Processes and Filtering Theory*, Academic Press, New York, 1970, Chap. 7.
²Morrison, N., *Introduction to Sequential Smoothing and Prediction*, McGraw-Hill, New York, 1969, Chaps. 2, 12.
³Giordano, A. A., and Hsu, F. M., *Least Square Estimation with Applications to Digital Signal Processing*, Wiley Interscience, New York, 1985, Chap. 3.
⁴Haykin, S., *Adaptive Filter Theory*, Prentice-Hall, Englewood Cliffs, NJ, 1986, Chaps. 3, 4.
⁵Haritos, G. K., and Srinivasan, A. V., eds., *Smart Structures and Materials*, ASME Pub., AD-Vol. 24 and AMD-Vol. 123, Dec. 1991.
⁶Luenberger, D. G., *Optimization by Vector Space Methods*, Wiley, New York, 1969, Chaps. 4, 6.
⁷Kailath, T., *Linear Systems*, Prentice-Hall, Englewood Cliffs, NJ, 1980, Chap. 2.
⁸Strang, G., *Linear Algebra and Its Applications*, 3rd ed., Academic Press, New York, 1988, Chap. 3.

Application of Order- n Formulation to Panel Deployment Problem of a Spacecraft

Shinji Suzuki*

Purdue University, West Lafayette, Indiana 47907

and

Hirohisa Kojima†

University of Tokyo, Tokyo 113, Japan

Introduction

THIS Note presents the application of an order- n formulation and intermittent analysis to calculate the deployment of folded

Received Feb. 27, 1993; revision received July 20, 1993; accepted for publication July 20, 1993. Copyright © 1993 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved.

*Visiting Scholar, Aeronautical and Astronautical Engineering, 1282 Grissom Hall; currently Associate Professor, Department of Aeronautics and Astronautics, University of Tokyo, Bunkyo-ku, Tokyo 113, Japan. Member AIAA.

†Graduate Student, Department of Aeronautics and Astronautics, 3-7-1,

multibody systems in space. When a spacecraft is considered as a serial rigid-body system without closed-loop topology, a new and efficient algorithm, an order- n formulation,¹ can be applied for the dynamic analysis. This formulation requires only an order of n arithmetic operations, where n is the number of degrees of freedom of the system. This Note adapts Rosenthal's algorithm² for the following situations: 1) a system is free in space, 2) a system has a tree topology, and 3) intermittent motion occurs. In spacecraft dynamics, intermittent motion plays an important role in deployment, docking, mass capture, and mass release. This behavior is formulated using the impulse-momentum equations,^{3,4} which are solved recursively by using the order- n formulation. A numerical example demonstrates the validity of the present method. A center arm and two panels of a spacecraft model are connected by revolute hinges and are deployed due to the force of a shrunk spring. When the hinge movement is locked by a ratchet mechanism, intermittent motion occurs.

Model Description

Figure 1 shows the deployment sequence of the model that has four rigid bodies, i.e., a main body, a center arm, and two panels. The main body is considered the base body B_0 , and both the center arm and the panels are labeled as B_i ($i = 1, 2, 3$). The main body is free in space. The revolute hinges are labeled as H_i ($i = 1, 2, 3$), and the hinge angle is measured as shown in Fig. 1.

Geometrical configuration of tree topology is described by using the body connection array⁵ $L(k)$, which represents the label of the adjoining lower numbered body of body B_k . In this model, the body connection array is defined as

$$L(1) = 0, \quad L(2) = L(3) = 1 \quad (1)$$

The deployment process is illustrated in Figs. 1a-1c. Initially, two panels are folded and attached to the center arm ($q_2 = q_3 = 90$ deg), and then the center arm is folded onto the main body ($q_1 = 90$ deg) as shown in Fig. 1a. At $t = 0$ s, the center arm begins to rotate

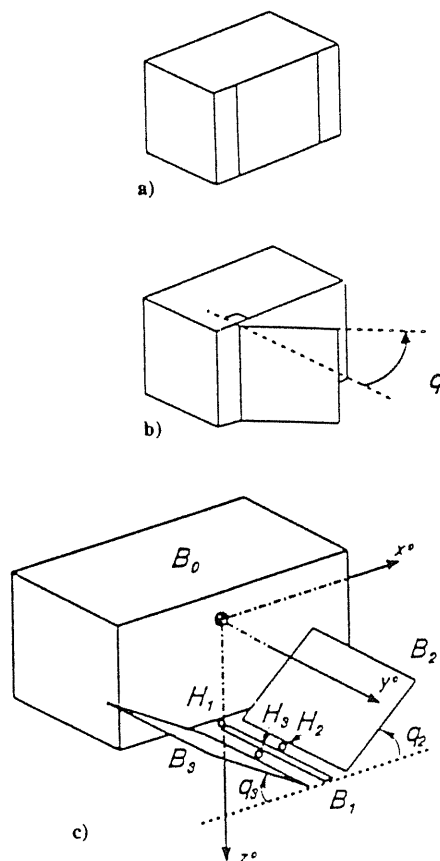


Fig. 1. Spacecraft model and sequence of panel deployment

Body
Mass, kg
Moment of inertia
I_{xx}
I_{yy}
I_{zz}
Size, m

Mainbody Velocities (m/s)

a) Velocity

Mainbody Angular Velocities (rad/s)

b) Angular

Hinge Angles (rad)