Suboptimal FIR Filtering of Nonlinear Models in Additive White Gaussian Noise

Yury S. Shmaliy, Fellow, IEEE

Abstract—The first- and second-order extended finite impulse response (EFIR1 and EFIR2, respectively) filters are addressed for suboptimal estimation of nonlinear discrete-time state-space models with additive white Gaussian noise. It is shown that, unlike the extended Kalman filter (EKF) and EFIR2 filter, the EFIR1 one does not require noise statistics and initial errors. Only within a narrow region around actual noise covariances, EFIR filters fall a bit short of EKF and they demonstrate better performance otherwise. It is shown that the optimal averaging interval for EFIR filters can be determined via measurement without a reference model in a learning cycle. We also notice that the second-order approximation can improve the local performance, but it can also deteriorate it. We thus have no recommendations about its use, at least for tracking considered as an example of applications.

I. INTRODUCTION

Soon after Kalman has published a seminal result in [1], his linear optimal filtering algorithm was extended by Cox [2] and others to nonlinear systems employing the first-order Taylor series expansion in the presence of additive white Gaussian noise both in the process and measurement. Thereafter, the first-order extended Kalman filter (EKF1) has been extensively used in diverse applications such as system state estimation, tracking of moving objects, navigation, Global Positioning System, process control, etc. Overall, whenever the nonlinear system state is of interest, the EKF1 is most often used [3]. That is in spite of the divergence effect [4], [5] particularly brightly pronounced for systems with strong nonlinearities.

A sequential second-order extended Kalman filter (EKF2) was proposed and investigated by Athans, Wishner and Bertolini in [6]. In addition to Jacobian used in EKF1, this filter also employs Hessian that makes it more complicated for the bias errors correction. Over decades, EKF2 has been developed and investigated in details [7]–[9]. But even though its ability to reduce errors was clearly demonstrated in [6], still nothing definitive has been said about its performance in general [10]. Below, we shall show experimentally that it is so.

An important issue is that both EKF1 and EKF2 demonstrate similar divergence if a system is highly nonlinear [4]. Referring to this, Julier and Uhlmann have employed the unscented transform and proposed the unscented KF (UKF) [11]. In spite of the fact that UKF improves the performance substantially, its estimate still may undergo severe deterioration when sufficient care cannot be provided in noise modeling. In the presence of uncertainties, noise tails, and outliers, this may lead to unacceptable results.

Degradation of the KF output in real world is often connected with its infinite impulse response (IR) inherent to recursive computation. This connection was emphasized by Jazwinski in [4] as opposite to the finite impulse response (FIR) filters having limited memory. Later, his conclusion about higher robustness of FIR structures against the unbounded perturbations in systems has become a cornerstone for the theory of receding horizon or model predictive control [12]. A similar opinion was recently expressed by Daum in [13] concerning the Gaussian iterative least squares (LSs): Gauss’s batch LSs often gives accuracy superior to the best available EKF. It can also be noticed that the known finite memory linear optimal estimators, such as those based on the maximum likelihood [4], [14], convolution [12], [15], [16], and regression [17], [18] can be converted to each other [19].

Manuscript received...; revised...

The author is with the Department of Electronics, Universidad de Guanajuato, Mexico (e-mail: shmaliy@ugto.mx).

An important peculiarity of FIR filtering is that its unbiased estimate converges to optimal when the batch length of $N$ points becomes large, $N \geq 1$ [15]. The effect is due to noise reduction by averaging making unbiasedness the principle performance criterion for FIR filters design. Worked out in such a way, the iterative Kalman-like unbiased FIR (UFIR) estimator ignoring noise statistics and initial errors [20] has already proved its efficiency against KF. The first- and second-order extensions (EFIR1 and EFIR2, respectively) of UFIR to nonlinear models are also of great interest, but they still were not addressed.

In this correspondence, we develop the Kalman-like UFIR filter derived in [20] and address the EFIR1 and EFIR2 filtering algorithms for discrete-time state-space models with uncorrelated additive white Gaussian noise sources in the system and measurement. The rest of the paper is organized as follows. In Section II, we discuss the nonlinear model and formulate the problem. The EFIR1 and EFIR2 filters are addressed in Section III along with the engineering algorithms. An application to tracking of a moving object is considered in Section IV and concluding remarks are drawn in Section V.

II. NONLINEAR MODEL AND PROBLEM FORMULATION

Consider a class of nonlinear systems represented in additive white Gaussian noise environment with the state and observation equations, respectively,

$$x_n = f(x_{n-1}) + B_n w_n, \quad y_n = h_n(x_n) + D_n v_n,$$

where $f(x_{n-1})$ and $h_n(x_n)$ are time-variant nonlinear vector functions, $x_n \in \mathbb{R}^k$, $y_n \in \mathbb{R}^m$, $B_n \in \mathbb{R}^{k \times p}$, and $D_n \in \mathbb{R}^{m \times M}$. The noise vectors $w_n \in \mathbb{R}^p$ and $v_n \in \mathbb{R}^M$ are supposed to be zero mean, $E\{w_n\} = 0$ and $E\{v_n\} = 0$, with the covariances, respectively,

$$R_n = E\{w_n w_n^T\}, \quad Q_n = E\{v_n v_n^T\},$$

and the property $E\{w_n v_n^T\} = 0$, for all $i$ and $j$.

Assume that $f_n(x_{n-1})$ and $h_n(x_n)$ are smooth enough to be approximated with the second-order Taylor series expansion. Then let us expand $f_n(x_{n-1})$ around the estimate $\hat{x}_{n-1}$ and $h_n(x_n)$ around the prior estimate $\hat{x}_n$ to be specified later; that is,

$$f_n(x_{n-1}) \approx f_n(\hat{x}_{n-1}) + A_n \varepsilon_{n-1} + \frac{1}{2} B_n \varepsilon_{n-1},$$

$$h_n(x_n) \approx h_n(\hat{x}_n) + C_n \varepsilon_n + \frac{1}{2} \beta_n,$$

where

$$A_n = \frac{\partial f_n}{\partial x} |_{x=\hat{x}_{n-1}}$$

$$C_n = \frac{\partial h_n}{\partial x} |_{x=\hat{x}_n}$$

are both Jacobian, $\varepsilon_{n-1} = x_n - \hat{x}_{n-1}$ is the prior estimation error, and $\varepsilon_n = x_n - \hat{x}_n$ is the estimation error. The second-order terms can be represented as [3]

$$\varepsilon_{n-1} = \sum_{k=1}^{K} \hat{e}_k T G_{kn} \varepsilon_{n-1},$$

$$\beta_n = \sum_{m=1}^{M} \hat{e}_m^T H_{mn} \varepsilon_n,$$

$1^{\hat{x}_n|k}$ means the estimate at $n$ via measurement from the past to $k$. Below, we use the following notations: $\hat{x}_n \equiv \hat{x}_n|n$ and $\hat{x}_n \equiv \hat{x}_n|n-1$. 
where
\[ G_{kn} = \frac{\partial^2 f_{kn}}{\partial x^2} |_{\bar{x}_{n-1}}, \]
\[ H_{mn} = \frac{\partial^2 h_{mn}}{\partial x^2} |_{\bar{x}_{n-1}} \]
are both Hessian and \( f_{kn} \) and \( h_{mn} \) are the \( k \)th and \( m \)th components of \( f_n(x_{n-1}) \) and \( h_n(x_n) \), respectively. Also, \( \epsilon_{kn}^T \in \mathbb{R}^K \) and \( \epsilon_{mn}^T \in \mathbb{R}^M \) are Cartesian basis vectors with the \( k \)th and \( m \)th components unity and all others zeros, respectively.

For the unbiased estimate we have \( E\{ \epsilon_{kj} \} = 0 \) and then the expectation of \( f_n(x_{n-1}) \) gives us the prior expectation of the estimation error covariance.

\[ \bar{x}_n = \frac{1}{2} \epsilon_{kn}^T \text{tr} \{ G_{kn} P_{n-1} \}, \]
where \( \text{tr} \{ B \} \) is trace of \( B \) and \( P_n = E\{ \epsilon_{kn} \epsilon_{kn}^T \} \) is the estimation error covariance.

In view of (1), (5), and (13), the expectation of the prior error \( E\{ \epsilon_n \} = E\{ \bar{x}_n - \bar{x}_n \} \) acquires zero components, and the average of \( h_n(x_n) \) can be found as

\[ \bar{\beta}_n = \frac{1}{2} \epsilon_{kn}^T \text{tr} \{ H_{mn} P_n^{-1} \}, \]
in which \( P_n = E\{ \epsilon_{kn} \epsilon_{kn}^T \} \) is the covariance of the prior estimation error.

Provided with the second-order approximations of (1) and (2), the problem can now be formulated as follows. We would like to extend the linear UFIR filtering algorithm addressed in [20] to the above-extended nonlinear model (Section II) can be written as

\[ \bar{x}_n = f_n(x_{n-1}) + \frac{1}{2} \epsilon_{kn}^T \text{tr} \{ G_{kn} P_{n-1} \} \]
and, using (15) and (16), we assign

\[ q_p(\bar{x}_n) = h_n(\bar{x}_n) + \frac{1}{2} \epsilon_{kn}^T \text{tr} \{ H_{mn} P_n^{-1} \}. \]

To avoid singularities, an iterative variable \( l \) ranges from \( m + K \) to \( n \) and the final estimate is taken at \( l = n \) in each iterative cycle.

As can be seen, the E FIR1 solution results from E FIR2 by neglecting the second order (last) terms in (27) and (28). That means that both the E FIR1 and UFIR algorithms do not involve the noise statistics and initial errors, whereas the E FIR2 one employs the covariances \( P_n \) and \( P_{n-1} \) considered below.

### A. Prior Estimation Error

By simple manipulations involving (1), (5) and (13), and taking into account that \( E\{ \epsilon_{kn} \epsilon_{kn}^T | G_{kn}(\bar{x}_{n-1}) \} \) and other similar vectors are identically zero for zero mean Gaussian processes \( (I_3) \), the prior estimation error can be found to be

\[ P_n^{-1} = \frac{1}{2} \epsilon_{kn} \epsilon_{kn}^T \text{tr} \{ G_{kn} P_{n-1} \} \]

where

\[ \hat{x}_n = \hat{x}_{n-1} + KL \bar{x}_n \]

\[ = f_n(x_{n-1}) + \frac{1}{2} \epsilon_{kn}^T \text{tr} \{ G_{kn} P_{n-1} \} \]

\[ F_m = \frac{1}{2} \epsilon_{kn}^T \text{tr} \{ H_{mn} P_n^{-1} \}. \]

2 Here and in the following, the bar over a function denotes expectation.

### III. ITERATIVE UFIR FILTERING

In the standard formulation of FIR filtering, measurement \( y_n \) is assumed to be conducted on an interval of \( N \) past neighboring points, from \( m = n - N + 1 \) to \( n \). If the state-space model is linear, then the linear iterative Kalman-like UFIR filtering algorithm can be used, as stated in [20] by theorem 2 for \( p = 0 \).

Following the strategy of EKF, the E FIR2 filtering estimate related to the above-extended nonlinear model (Section II) can be written as

\[ \bar{x}_n = \bar{x}_{n-1} + K_n [y_n - q_p(\bar{x}_n)], \]
where, in contrast to the Kalman filter, the gain \( K_n = F_n C_n^T \) does not involve the noise covariances and is defined iteratively by the state transition and measurement matrices via

\[ F_n = [C_n^T C_n + (A_n - 2C_n^T A_n)^{-1}]^{-1}. \]

Auxiliary functions for this algorithm were defined in [20] as

\[ \hat{x}_{(n)} = A_{m+1} \Phi C_{s,m}^T Y_{s,m}, \]
\[ \Phi = (C_{s,m}^T C_{s,m})^{-1}, \]
\[ Y_{s,m} = \begin{bmatrix} Y_{s,1} & Y_{s,2} & \cdots & Y_{s,M} \end{bmatrix}^T, \]
\[ C_{s,m} = \text{diag} \{ C_s, C_{s+1}, \ldots, C_m \}, \]
\[ A_{s,m} = \prod_{i=s}^m A_{m+1}, \]
where \( m = n - N + 1 \) and \( s = M + K - 1 \).

For this filter, the prior estimate is specified by (13) and (14) as

\[ \bar{x}_n = f_n(\bar{x}_{n-1}) + \frac{1}{2} \epsilon_{kn}^T \text{tr} \{ G_{kn} P_{n-1} \} \]
and, using (15) and (16), we assign

\[ q_p(\bar{x}_n) = h_n(\bar{x}_n) + \frac{1}{2} \epsilon_{kn}^T \text{tr} \{ H_{mn} P_n^{-1} \}. \]

To avoid singularities, an iterative variable \( l \) ranges from \( m + K \) to \( n \) and the final estimate is taken at \( l = n \) in each iterative cycle.

As can be seen, the E FIR1 solution results from E FIR2 by neglecting the second order (last) terms in (27) and (28). That means that both the E FIR1 and UFIR algorithms do not involve the noise statistics and initial errors, whereas the E FIR2 one employs the covariances \( P_n \) and \( P_{n-1} \) considered below.
B. Estimation Error

The estimation error $P_n$ can be defined using the extended nonlinearities and estimate (17) if we start with

$$
P_n = E \{ (x_n - \hat{x}_n)(x_n - \hat{x}_n)^T \} = E \{ [f_n(x_n-1) + B_n w_n - f_n(\hat{x}_{n-1})] - \frac{1}{2} \hat{x}_n - K_n (y_n - q_n(\hat{x}_n)) \} [f_n(x_n-1) + B_n w_n - f_n(\hat{x}_{n-1}) - \frac{1}{2} \hat{x}_n - K_n (y_n - q_n(\hat{x}_n)) ]^T \}
$$

$$
= E \{ [A_n \hat{x}_{n-1} + B_n w_n + \frac{1}{2} (\hat{x}_n - \hat{x}_n) - K_n (C_n \hat{x}_n) + B_n w_n - K_n (C_n \hat{x}_n) + \frac{1}{2} (\hat{x}_n - \hat{x}_n)] \} + \frac{1}{2} (\beta_n - \hat{x}_n) + D_n v_n [A_n \hat{x}_{n-1} + B_n w_n + \frac{1}{2} (\hat{x}_n - \hat{x}_n) - K_n (C_n \hat{x}_n) + B_n w_n - K_n (C_n \hat{x}_n) + \frac{1}{2} (\hat{x}_n - \hat{x}_n)] + D_n v_n \} \}
$$

(31)

Now recall that $P_n$, $R_n$, and $Q_n$ are all symmetric, observe that

$$
E \{ \hat{x}_{n-1} \hat{x}_{n-1}^T \} = E \{ (x_n - \hat{x}_n)(x_n - \hat{x}_n)^T \} = E \{ [A_n \hat{x}_{n-1} + B_n w_n - \frac{1}{2} (\hat{x}_n - \hat{x}_n)] \} = A_n P_{n-1},
$$

$$
E \{ w_n \hat{x}_{n-1} \} = R_n B_n^T,
$$

$$
E \{ \hat{x}_n w_n^T \} = P_{n-1} A_n^T,
$$

and the expectations of all other products are matrices with zero components. Next, provide the averaging, account for (29) whenever necessary, and, after the simple but tedious transformations, finally arrive at

$$
P_n = (I - K_n C_n) P_{n-1} (I - K_n C_n)^T + K_n D_n Q_n D_n^T K_n^T
$$

$$
+ \frac{1}{2} [F_n^T C_n^T K_n^T + K_n C_n F_n^T] + K_n \hat{K}_n K_n^T
$$

$$
- \frac{1}{2} (\hat{M}_n K_n^T + K_n \hat{M}_n^T),
$$

(33)

where, by (II.4) and similarly to (30), the $(rg)$th component of $\hat{H}_n = \hat{H}_n - \frac{1}{2} \hat{\gamma}_n \beta_n^T$ is computed as

$$
\hat{H}_{(rg)} = tr \{ [H_n P_n H_{yn} P_n^T] \}
$$

(34)

and, by (II.11), the $(ur)$th component of $\tilde{M}_n = \tilde{M}_n - \frac{1}{2} \hat{x}_n \beta_n^T$ is

$$
\tilde{M}_{(ur)} = \{ G_n P_{n-1} H_n P_{n-1} \} + \frac{K}{\alpha} \sum_{q=1}^{K} \sum_{l=1}^{\alpha} H_{mn}
$$

$$
\times tr \{ [G_n P_{n-1} G_n P_{n-1}] \}
$$

(35)

The covariance $P_n$ can thus be computed via $P_n$. An important issue is that if $P_{n-1}$ is assigned in (29), then both these errors are computed rigorously, as in the Kalman filter. Otherwise, if iterations are required as in (17), the error will be accumulated in (29) and (33) as caused by multiple effect of $Q_n$ and $R_n$. In the latter case, the computed $P_n$ and $P_{n-1}$ should be considered as upper bounds.

It has to be remarked now that different authors prefer using diverse forms of $P_n$ in the EKF2 algorithms. The complete second-order form of $P_{n-1}$ can be met in [6] [7], whereas in [10] [21] only the first-order components are saved. In what follows, we shall base our analysis on the complete forms of (29) and (33).

C. EFR2 Filtering Algorithm

For engineering applications, Table I summarizes the steps in the EFR2 filtering algorithm. Given $N$, $K$, $P_{n-1} = 0$, $R_n$, $Q_n$, and $\alpha = n + N + 1 + K$, a set of auxiliary matrices is computed and updated for each $n$. An iterative variable starts with $l = 0$ and a set of matrices updated for $l$ ranging from $\alpha$ to $n$. The output estimate $x_n|_{n}$ and estimation error covariance $P_n$ are taken when $l = n$. One thus notes once again that, unlike the UIF algorithm, the EFR2 one involves both $P_{n-1}$ and $P_n$. We do not meet this specific in the EFR1 filter.

D. EFR1 Filter

This filter follows straightforwardly from EFR2, if to neglect the second-order terms containing $P_{n-1}$ and $P_1$. Given the discrete-time

<table>
<thead>
<tr>
<th>Stage</th>
<th>Iterative EFR1 Filtering Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Given:</td>
<td>$K$, $N$, $m = n - N + 1$, $\alpha = m + K$, $P_{n-1}$, $R_n$, $Q_n$, $s = \alpha - 1$, $\alpha \leq l \leq n$.</td>
</tr>
<tr>
<td>Set:</td>
<td>$Y_{s,m}$ by (22), $C_{s,m}$ by (23), $A_{n+1}^m$ by (26), $\Phi_{s,m}$ by (21), $P_{n} = A_{n+1}^m \Phi_{n+1}^m A_{n+1}^m + \frac{1}{2} \tilde{F}_n$.</td>
</tr>
<tr>
<td>Update:</td>
<td>$\tilde{x}<em>n = A</em>{n+1}^m \Phi_{n+1}^m Y_{s,m}$.</td>
</tr>
<tr>
<td>$\tilde{x}<em>n = f_l(\hat{x}<em>n) + \frac{1}{2} \sum</em>{k=1}^{K} \frac{\kappa}{\alpha} \sum</em>{\alpha} e_k^T [G_k P_{l-1}],$</td>
<td></td>
</tr>
<tr>
<td>$\tilde{y}<em>n = \sum</em>{m=1}^{\alpha} e_m^T { H_m P_{l-1} }$,</td>
<td></td>
</tr>
<tr>
<td>$\tilde{F}<em>n = { C_l^T + (A</em>{l-1}^T A_{l-1})^{-1}^{-1} },$</td>
<td></td>
</tr>
<tr>
<td>$K_l = F_l C_l^T$,</td>
<td></td>
</tr>
<tr>
<td>$\hat{x}_n = \hat{x}_n + K_l [y_n - q(\hat{x}_n)],$</td>
<td></td>
</tr>
<tr>
<td>$P_l = (I - K_l C_l) P_l (I - K_l C_l)^T + K_l D_l Q_l D_l^T K_l^T + \frac{1}{2} [F_l C_l^T K_l^T + K_l C_l F_l^T] + \frac{1}{2} K_l \hat{H}_n K_l^T - \frac{1}{2} (\hat{M}_l K_l^T + K_l \hat{M}_l^T).$</td>
<td></td>
</tr>
</tbody>
</table>
state-space model, (1) and (2), with the uncorrelated additive white Gaussian noise sources, \( w_n \) and \( v_n \), EFIR1 filtering can be provided iteratively by

\[
\hat{x}_1 = f_1(\hat{x}_{n-1}) + F_1 C_1^T [y_1 - h_1(\hat{x}_{n-1})],
\]

where

\[
F_1 = [C_1^T C_1 + (A_1 F_{l-1} A_1^T)^{-1}]^{-1}.
\]

\( F_{l-1} \) is computed by (20) and \( \hat{x}_{l-1} \) by (19). An iterative variable \( l \) ranges from \( \alpha = m + K \) to \( n \) and the output is taken at \( l = n \).

A generalization of the EFIR1 algorithm is given in Table II. As can be seen, EFIR1 needs only \( N \), \( K \), and \( \alpha = n - N + 1 + K \) to start computing and updating all of the matrices. As well as in the case of EFIR2, an iterative variable \( l \) ranges for each \( m \) as \( \alpha \leq l \leq n \) and the output is taken at \( l = n \) in each cycle.

### IV. Tracking of a Moving Object

Tracking and navigation are most common applications for extended filters. Below, we investigate the trade-off between the EKF and EFIR algorithms based on a typical example of tracking of a moving object. It is assumed that two distance measurement stations (DMSs) are located at \((0,0)\) and \((0,50)\) m as shown in Fig. 1. To simplify the problem, we suppose that the object and DMSs are all in a horizontal plane. Each DMS transmits a pulse that is reflected from the object and returns back to DMS. The transit time is interpreted in terms of distance \( d_1 \) or \( d_2 \).

In the standard interpretation of the object dynamics, we suppose that it moves in the presence of noise along each of the axes and has four states \((K = 4)\): \( x_1 \), is the coordinate \( x \); \( x_2 \), velocity along \( x \); \( x_3 \), coordinate \( y \); and \( x_4 \), velocity along \( y \). The behavior is modeled with (1), in which we set \( f_1(x_{n-1}) = A x_{n-1}, B_n = I, \)

\[
A = \begin{bmatrix}
1 & \tau & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & \tau \\
0 & 0 & 0 & 1
\end{bmatrix}.
\]

It is supposed that white Gaussian noise \( w_n = [0 \; 0 \; 0 \; w_{4n}]^T \) is zero mean with the variances \( \sigma_w^2 = \sigma_{w_1}^2 = \sigma_{w_2}^2 \) and covariance

\[
R = \sigma_w^2 = \begin{bmatrix}
\tau^2/3 & \tau/2 & 0 & 0 \\
\tau/2 & 1 & 0 & 0 \\
0 & 0 & \tau^2/3 & \tau/2 \\
0 & 0 & \tau/2 & 1
\end{bmatrix}.
\]

For measurement organized as in Fig. 1 with noise \( v_n = [v_{in} \; v_{2n}] \) having the variance \( \sigma_v^2 = \sigma_{v_1}^2 = \sigma_{v_2}^2 \), the observation equation (2) is specialized with

\[
h_n(x_n) = \begin{bmatrix}
\sqrt{x_1^2 + y_1^2} \\
\sqrt{(a - x_n)^2 + y_n^2}
\end{bmatrix}
\]

and the covariance

\[
Q = \begin{bmatrix}
\sigma_v^2 & 0 \\
0 & \sigma_v^2
\end{bmatrix}.
\]

In our experiment, measurement has been conducted at 1000 points with step \( \tau = 0.1 \) s. We also allowed \( \sigma_w = 0.01 \) m and \( \sigma_v = 0.2 \) m.

To estimate \( x_n \), at the initial point \( n = \alpha - 1 \) with \( \alpha = m + K = m + 4 \), projections \( s_{xn} \) and \( s_{yn} \) of the measurement to \( x \) and \( y \) were formed as, respectively,

\[
s_{xn} = \frac{1}{2\tau} (y_{1n} - y_{2n} + a^2),
\]

\[
s_{yn} = \sqrt{y_{1n}^2 - s_{xn}^2},
\]

where \( y_{1n} \) and \( y_{2n} \) are time-varied measurement distances \( d_1 \) and \( d_2 \), respectively. The required auxiliary matrices

\[
C_{m,3,m} = \begin{bmatrix}
1 & 3\tau & 0 & 0 \\
0 & 0 & 1 & 3\tau \\
1 & 2\tau & 0 & 0 \\
0 & 0 & 1 & 2\tau
\end{bmatrix},
\]

\[
Y_{m+3,m} = \begin{bmatrix}
S_{x(m+3)} \\
S_{y(m+3)} \\
S_{x(m+2)} \\
S_{y(m+2)} \\
S_{x(m+1)} \\
S_{y(m+1)} \\
S_{xm} \\
S_{ym}
\end{bmatrix}
\]

were filled following (23) and (22). Then both EFR filters (Table I and Table II) were run along with EKF1 and EKF2, which algorithms were taken from [10].
As shown in Fig. 2b. This value is exactly that suggested by envelopes of \( \partial N \) be seen, \( V \) RMSVs as functions of \( N \) x interval of A. Optimal Averaging Interval

\[ N > N \] owing to noise reduction with \( N \) when \( V \) the effect of bias. With \( N \) unavailable and can only be observed via \( x \). Errors in FIR estimates are minimized by the optimal averaging \( N \) opt. As can

A. Optimal Averaging Interval

Errors in FIR estimates are minimized by the optimal averaging interval of \( N \) opt points. Although, \( N \) opt can be found as suggested in [15], [22], there is another way. Recall that the model behavior \( x_n \) is unavailable and can only be observed via \( y_n \). Then exploit projections (42) and (43) and compute the root mean square values (RMSVs) as functions of \( N \),

\[ V_x(N) = \sqrt{E\{(x_n - \bar{x}_N(N))^2\}}, \quad (44) \]

\[ V_y(N) = \sqrt{E\{(y_n - \bar{y}_N(N))^2\}}. \quad (45) \]

It can easily be observed that \( V_x(N) \) and \( V_y(N) \) rise intensively owing to noise reduction with \( N > 4 \) and attain minimum slopes when \( N = N_{\text{opt}} \). With \( N > N_{\text{opt}} \), they intensively rise again due to the effect of bias. With \( N > N_{\text{opt}} \), bias in the estimate dominates the measurement noise and both \( V_x(N) \) and \( V_y(N) \) reach the estimation root mean square errors (RMSEs)

\[ P_x(N) = \sqrt{E\{(x_n - \bar{x}_N(N))^2\}} \quad (46) \]

\[ P_y(N) = \sqrt{E\{(y_n - \bar{y}_N(N))^2\}}. \quad (47) \]

Figure 2 supports this analysis regarding the coordinate \( x \). As can be seen, \( V_x(N) \) behaves monotonously in Fig. 2a and the upper envelopes of \( \frac{\partial}{\partial N} V_x(N) \) and \( \frac{\partial}{\partial N} V_y(N) \) reach minima at \( N_{\text{opt}} = 34 \) as shown in Fig. 2b. This value is exactly that suggested by \( P_x(N) \) at the point of a minimum (Fig. 2a). So, there is a simple way to determine \( N_{\text{opt}} \) in a “learning” cycle and then redetermine and update it whenever necessary. We notice that similar behaviors of the RMSVs were observed from \( N = K \) to \( N = N_{\text{opt}} \) in other state-space models. It was also revealed that \( N > N_{\text{opt}} \) causes RMSVs to approach RMSEs in different ways. This specific, however, does not play any substantial role in the determination of \( N_{\text{opt}} \).

Fig. 3. Effect of the second-order approximation on the EFIR estimates: (a) performance improvement, (b) performance deterioration, (c) local mixed effect, and (d) prolonged mixed effect.

An important applied point is that RMSE is sufficiently flat around \( N_{\text{opt}} \) (Fig. 2a). That means that \( N_{\text{opt}} \) can be ascertained with an error of about \( \pm 15\% \) (29 \( \leq N_{\text{opt}} \leq 39 \), in Fig. 2b) to cause no substantial violation at the output.

B. First- vs. Second-Order Filtering

To learn the difference between estimates provided with EFIR1, EFIR2, EKF1, and EKF2, the process was multiply generated and filtered. Observing the outputs, the only conclusion coming to mind was that made by Simon in [10]: nothing definitive can be said about the performance of the second-order extended filters. Indeed, most of the runs reveal the same trajectories for the first- and second-order filters. Just in a few cases, different behaviors can be observed as shown in Fig. 3 and Fig. 4. This definitely does not allow one to make a preference in favor of a certain filter. In fact, both EFIR2 and EKF2 are able to improve the performance (Fig. 3a and Fig. 4a) in a way similar to that demonstrated in [6]. But they may also deteriorate it (Fig. 3b and Fig. 4b) or produce mixed effects (Fig. 3d and Fig. 4d). Referring to the above experiment, below we focus our attention only on EFIR1 and EKF1.

C. Ideal Operation Conditions

Let us now assume that the model and noise are both known exactly and investigate the EFIR1 and EKF1 estimates. Fig. 1 illustrates this case and Fig. 5 gives us a more precise picture of the estimation
Fig. 4. Effect of the second-order approximation on the EKF estimates: (a) performance improvement, (b) performance deterioration, (c) prolonged mixed effect, and (d) prolonged mixed effect.

D. Effect of Errors in Noise Covariances

Engineers will certainly appreciate that EFIR filters need only one optimization parameter $N_{opt}$ that can be determined much easier than the noise statistics via measurement in a learning cycle (Section IV-A). Unlike the EFIR1 filter, EKF1 needs both the covariance matrices and initial errors. To obtain these, much more investigations are required that may not always be available or afforded. Typically, these matrices are described approximately that sometimes makes the estimates too rough or even unacceptable.

To demonstrate the trade-off between EFIR1 and EKF1 under errors in noise covariances, below we assume that $\sigma_v$ is known exactly, but $\sigma_w$ is uncertain. We thus substitute it in the algorithms with the approximate value $\tilde{\sigma}_w$. Dealing with a single noise component is certainly not enough to make far reaching conclusions, since each of the states undergos effect of noise. Even so, higher robustness of EFIR1 filter against EKF1 will be demonstrated.

Fig. 6a sketches the tracking effects assuming $\tilde{\sigma}_w = 0.1\sigma_w$. It is seen that EKF1 becomes more “inertial” here that results in larger systematic errors and lower noise. The estimation errors are given in Fig. 6b and Fig. 6c for the coordinates $x$ and $y$, respectively. In an opposite case of $\tilde{\sigma}_w = 10\sigma_w$, the EKF1 estimate becomes too “fast”, thus producing more noise with lower systematic errors (Fig. 7a) as shown for $x$ and $y$ in Fig. 7b and Fig. 7c, respectively.

The trade-off between EFIR1 and EKF1 under the uncertain system noise statistics is generalized in Fig. 8. The RMSEs are computed here for each estimate allowing $0 \leq \tilde{\sigma}_w/\sigma_w \leq 10$. Instantly, one infers that there exists only a narrow region around the actual standard deviation, $\tilde{\sigma}_w = \sigma_w$, within which the Kalman filter produces a bit lower errors than in EFIR1. Otherwise and so in practical applications, the latter outperforms it.

E. Effect of Temporary Model Uncertainty

We finally take a look at errors under the model temporary uncertainty. Assuming that the latter is due to

$$A_n = \begin{bmatrix} 1 & \tau + \delta_n & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \tau + \delta_n \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

where $\delta_n \neq 0$ if $400 \leq n \leq 440$ and is zero otherwise, and ignoring this alteration in the algorithms, we run EFIR1 and EKF1 and observe what goes on with errors if we allow different $\tilde{\sigma}_w/\sigma_w \geq 1$. To gain
Fig. 7. Object tracking by EKF1 and EFIR1 with \( \tilde{\sigma}_w/\sigma_w = 10 \): (a) tracking, (b) errors in the coordinate \( x \), and (c) errors in the coordinate \( y \).

Fig. 8. Estimation errors produced by EFIR1 and EKF1 as functions of \( \tilde{\sigma}_w/\sigma_w \): (a) coordinate \( x \) and (b) coordinate \( y \).

Fig. 9. Effect of the temporary model uncertainty occurred from \( n = 400 \) to \( n = 440 \) on the estimation errors with \( \tilde{\sigma}_w \geq \sigma_w \).

the effect, we set \( \delta_n = 10 \) s \( \gg \tau = 0.1 \) s. The result shown in Fig. 9 confirms the statement made in [15], [20] for linear filtering: unlike EFIR1, the EKF1 becomes much lesser robust against the temporary model uncertainties in the presence of errors in noise covariances.

V. CONCLUSION

The first- and second-order extended FIR filters have been developed for suboptimal state estimation in nonlinear models with additive white Gaussian noise. The following critical advantages were pointed out against the EKF. The EFIR1 filter does not require noise covariances and initial conditions. Instead, it relies on an optimal averaging interval that can automatically be determined via measurement in a “learning” cycle. Only within a narrow region around actual (commonly unavailable) system noise covariances, the EFIR estimate falls a bit short of the EKF one. Otherwise it has better performance.

An application to tracking of a moving object has shown the following. If the process noise covariance is estimated to be smaller that actual, the Kalman filter becomes more “inertial”, producing larger systematic errors and lower noise. Otherwise, it becomes too “fast”, yielding smaller systematic errors and larger noise. That does not effect the EFIR estimates. It is also worth to mention that the first- and second-order EKF and EFIR estimates have traced in general along the same trajectories. In some runs, EKF2 and EFIR2 improved the local performance, but they also deteriorated it in some other ones. We thus can give no recommendations about their use.

Although the EFIR1 algorithm is certainly a very attractive tool for engineering applications, the divergence problem occurs here similarly to that in EKF. We are looking into it now, going to turn things around and find some solution. We are also working on fast algorithms for the determination of optimal \( N \).

APPENDIX I

EXPECTATIONS OF PRODUCTS OF QUADRATIC FORMS IN NORMAL VARIABLES

Given the quadratic form \( Q_i = x^T H_i x \), where \( x \sim \mathcal{N}(0, R) \) is Gaussian, \( R = E(xx^T) \), and \( H_i \) is Hessian. Then, by [23], the
low-order expectations of products of $Q$, are

$$ E\{Q_1Q_2\} = \text{tr}(H_1R)\text{tr}(H_2R) + 2\text{tr}(H_1RHR_2R), \quad (I.1) $$

$$ E\{Q_1Q_2Q_3\} = \text{tr}(H_1R)\text{tr}(H_2R)\text{tr}(H_3R) + 2\text{tr}(H_1R)\text{tr}(H_2RHR_3R) + 2\text{tr}(H_1RHR_2R) + 2\text{tr}(H_2R)\text{tr}(H_3RHR_2R), \quad (I.2) $$

$$ E\{xQ_1\} = 0. \quad (I.3) $$

### APPENDIX II

**COVARIANCES OF THE 2-ORDER TERMS**

The forms for the covariances $E\{\kappa_n\kappa_n^T\}$, $E\{\beta_n^\beta_n\}$, and $E\{\kappa_n\beta_n^T\}$ are given below.

#### A. $E\{\kappa_n\kappa_n^T\}$

By (9) and (I.1), the covariance $E\{\kappa_n\kappa_n^T\}$ can be written as

$$ E\{\kappa_n\kappa_n^T\} = 2F_n, \quad (II.1) $$

where $F_n \in \mathbb{R}^{K \times K}$ is a symmetric with the $(u,v)$th component

$$ F_{(u,v)}n = \text{tr}[G_{un}P_{n-1}G_{vn}P_{n-1}] + \frac{1}{2}\text{tr}[G_{un}P_{n-1}tr[G_{vn}P_{n-1}]]. \quad (II.2) $$

#### B. $E\{\beta_n^{\beta_n}\}$

By (10) and (I.1), the covariance $E\{\beta_n^{\beta_n}\}$ becomes

$$ E\{\beta_n^{\beta_n}\} = 2H_n, \quad (II.3) $$

where $H_n \in \mathbb{R}^{M \times M}$ is a symmetric with the $(r,g)$th component

$$ H_{(r,g)n} = \text{tr}[H_{rn}P_{n}^T H_{gn}P_{n}] + \frac{1}{2}\text{tr}[H_{rn}P_{n}^- tr[H_{gn}P_{n}^-]]. \quad (II.4) $$

#### C. $E\{\kappa_n\beta_n^T\}$

To find the cross-covariance $E\{\kappa_n\beta_n^T\}$ first transform $\beta_n$ as

$$ \beta_n = e_n^T H_{rn} e_n = [A_n e_{n-1} + B_n w_n + \frac{1}{2}(\kappa_n - \kappa_n^-)]^T H_{rn} \times [A_n e_{n-1} + B_n w_n + \frac{1}{2}(\kappa_n - \kappa_n^-)]. \quad (II.5) $$

By (II.5), (I.3), and symmetric $H_{rn}$, the $(u,g)$th component

$$ M_{(u,g)n} \text{ of } E\{\kappa_n\beta_n^T\} = 2M_{n} \in \mathbb{R}^{K \times M}$

is written as

$$ M_{(u,g)n} = E\{\kappa_n[e_n^T(H_{rn}e_{n-1})]\} + E\{\kappa_n[w_n^T(H_{rn}w_n)]\} + \frac{1}{4}E\{\kappa_n[H_{rn}^T H_{rn} \kappa_n]\} - \frac{1}{4}E\{\kappa_n[H_{rn}^T \kappa_n H_{rn}]\} + \frac{1}{4}E\{\kappa_n[H_{rn}^T H_{rn} \kappa_n]\}, \quad (II.6) $$

where $H_{rn} = A_n^T H_{rn} A_n$ and $H_{rn} = B_n^T H_{rn} B_n$.

By (I.1), $a_n = E\{\kappa_n[e_n^T(H_{rn}e_{n-1})]\}$ becomes

$$ a_n = 2\text{tr}[G_{un}P_{n-1}H_{rn}P_{n-1}] + \text{tr}[H_{rn}P_{n-1}] \quad (II.7) $$

Because $\kappa_{un}$ originates from $n - 1$ and $w_n$ exists at $n$, they are uncorrelated and $b_n = E\{\kappa_n[w_n^T H_{rn} w_n]\}$ transforms to

$$ b_n = E\{\kappa_n\} E\{w_n^T H_{rn} w_n\} = \kappa_n \text{tr}[H_{rn} R_n]. \quad (II.8) $$

Provided the matrix transformation, $c_n = E\{\kappa_n^T H_{rn} \kappa_n\}$ can be represented as

$$ c_n = \sum_{q=1}^{K} \sum_{l=1}^{K} H_{ql}E\{[\kappa_{qn-1}G_{un}H_{qn-1}]\} \times (\kappa_{qn-1}G_{un}H_{qn-1}). \quad (II.9) $$

that, by (II.7), transforms to

$$ c_n = \sum_{q=1}^{K} \sum_{l=1}^{K} H_{ql} \{8\text{tr}[G_{un}P_{n-1}G_{qn-1}P_{n-1}] | \kappa_{qn-1} \kappa_{ql-1} + 2\text{tr}[G_{qn-1}G_{un}H_{qn-1}] + 2\text{tr}[G_{un}P_{n-1}G_{qn-1}P_{n-1}] \}. \quad (II.10) $$

Next, $E\{\kappa_n \kappa_n^T\}$ can be viewed as the $u$th row vector of $2F_n$ and the term $d_n = E\{\kappa_n \kappa_n^T\} H_{rn}$ transformed to

$$ d_n = \sum_{q=1}^{K} \sum_{l=1}^{K} H_{ql} \text{tr}[G_{un}P_{n-1}G_{qn-1}P_{n-1}] + \text{tr}[H_{qn} P_{n-1}]. \quad (II.10) $$

Substituting (II.7)-(II.10) to (II.6) finally leads to

$$ M_{(u,g)n} = 2\text{tr}[G_{un}P_{n-1}H_{rn}P_{n-1}] + \text{tr}[H_{rn} R_n] + \sum_{q=1}^{K} \sum_{l=1}^{K} H_{ql} \times \{2\text{tr}[G_{qn-1}G_{un}H_{qn-1}] + \text{tr}[G_{qn-1}G_{pn-1}P_{n-1}] + \text{tr}[G_{qn-1}G_{un}H_{qn-1}]\} \quad (II.11) $$

**ACKNOWLEDGMENT**

The author would like to thank Associate Editor Dr. Benoit Champagne and the anonymous reviewers for very useful and highly valuable comments making possible to improve the manuscript substantially.

**REFERENCES**


