2011 International Conference on Asia Agriculture and Animal
IPCBEE vol.13(2011) © (2011)IACSIT Press, Singapore

Approximated Fusion Filter for Multiple Time Delayed Signals

Seokhyoung Lee [1], Hyuksang Kwon [1], Vladimir Shin [2]

[1] School of Information and Mechatronics, Gwangju Institute of Science and Technology, South Korea
Email: {lee-sh, hyuksang}@gist.ac.kr

[2] Department of Information and Statistics, Gyeongsang National University, South Korea,
Email: vishin@gnu.ac.kr

Abstract. This paper focuses on distributed fusion filtering algorithms and fusion formulas for time delayed multiple signals received from a system. Since local cross-covariances are important values to implement fusion formulas, we present an approximation scheme which uses special steady-state approximation for computing local cross-covariances. Such approximation is useful for calcul
ating matrix weights in real-time. Subsequent analysis of the proposed fusion algorithm is presented through a numerical example demonstrating the computational effectiveness of the proposed fusion algorithm.

Keywords: Multi-sensor, fusion filter, time-delay, low complexity

1. Introduction

In recent years, state estimation problem for signal processing or system control with uncertainties has been investigated widely, and the Kalman filter is well known as an optimal solution to the estimation problem for linear stochastic systems. However, since the Kalman filter is applicable only to linear systems without time-delays, it is subsequence issue to solve estimation problems for linear systems with time-delays. Regarding this, Kwakernnak investigated the optimal solution of the state estimation problem for continuous systems with time-delays [1], but this approach is related to solving partial differential equations which in general do not have explicit solutions. For the case of discrete systems, Priemer and Mishra presented filtering equations using the method of the orthogonal projection and an innovation approach, respectively [2, 3].

However, in multisensory environment, the implementation of the above filtering equations using the augmented measurement vector is computationally expensive, especially when the system dimension is high and the delayed time is large. Thus, distributed filtering process and the fusion of the local estimates are required for saving computational resources. To achieve the distributed fusion filtering, the specific architectures and techniques for data fusion are discussed in [4, 5]. Furthermore the explicit formulas for finding the best linear combination of the local estimates are developed and presented in [6-8].

In the above fusion formulas, the cross-covariances are important factors for estimation fusion. Thus, this paper provides exact equations for the local cross-covariances. However, these equations have also computational burden in real-time computation, especially when the number of sensors is large. Therefore, the main purpose of this paper is to propose a fusion estimation algorithm for reducing the computational time in the implementation.

The rest of the paper is organized as follows. In Section 2, the problem is setting and the main goal is presented. In Section 3, a filtering problem for a single sensor is considered, and the specific filtering equations for the time delayed signals are given. In Section 4, fusion filtering using matrix weighted fusion formula is dealt with. The exact equations of local cross-covariances are presented, and to reduce their computational burden, an approximation approach is discussed. A numerical example demonstrating the concrete accuracies and effectiveness of computation for the discussed fusion filtering are presented in Section 5. Finally, a brief conclusion is given in Section 6.

2. Problem Setting

The system with time-delays considered is described by the state vector difference equation:

$$x(k+1)=\sum_{h=0}^{M}F(k-h)x(k-h)+w(k),\ k=0,1,..., \tag{1}$$

where $x(k)\in\mathbb{R}^n$ is an unknown state, $F(k)\in\mathbb{R}^{n\times n}$ is a time-varying transition matrix, $x(0)\sim\mathbb{N}(\bar{x}_0,P_0)$ and $w(k)\in\mathbb{R}^n$ is a zero-mean white Gaussian noise with covariance $Q(k)$.

Next, N sensor models are given by

$$y^{(i)}(k)=\sum_{d=0}^{L_i}H^{(i)}(k-d)x(k-d)+v^{(i)}(k),\ \ i=1,...,N, \tag{2}$$

where $y^{(i)}(k)\in\mathbb{R}^{m_i}$ represents the i-th sensor measurement, $H^{(i)}(k)\in\mathbb{R}^{m_i\times n}$ is the i-th measurement matrix, $v^{(i)}(k)\in\mathbb{R}^{m_i}$ is a zero-mean white Gaussian noise with covariance $R^{(i)}(k)$, and $\mathrm{cov}(w(k),v^{(i)}(k))=0$ and $\mathrm{cov}(v^{(i)}(k),v^{(j)}(k))=0,\ i\neq j$.

The main problem is to optimally estimate the unknown state $x(k)$ using all sensor measurements $Y(k)=\{y^{(i)}(l),\ l=1,...,k\}_{i=1}^{N}$.

Generally, there are two fusion estimation approaches commonly used to process the measured data. If a central processor directly receives all sensor measurement data $Y(k)$ and processes them in real time, the corresponding result is known as the centralized fusion filtering (CFF) [4, 9]. However, the CFF has several drawbacks, such as heavy computational burdens and requiring big memory sources, especially when $n,N\gg 1$ [9].

The second approach is called distributed fusion filtering (DFF), in which every local sensor is attached to a local processor. In this approach, the processor estimates the state of a system based on its own measurement $y^{(i)}(k)$, and then transmits its local estimates $\hat{x}^{(i)}(k|k)$ to the fusion centre. After that, the fusion centre estimates the object based on received local estimates.

Due to the drawbacks of the CFF, this paper focuses on DFF for the system (1) and (2). Before presenting the DFF, it is needed to explain how local estimates $\hat{x}^{(i)}(k|k)$, $i=1,...,N$ are obtained based on $y^{(i)}(k)$, $i=1,...,N$, respectively. The details are given in the next section.

3. Filtering for a Single Sensor

Let us consider the system (1) and a local sensor (2), i.e., $y^{(i)}(k)$, where the index "i" is fixed. Then, the local estimate can be represented by the following filtering equations [2, 3]:

$$\hat{x}^{(i)}(k|k-1)=\sum_{h=0}^{M}F(k-h-1)\hat{x}^{(i)}(k-h-1|k-1), \tag{3}$$

$$\hat{x}^{(i)}(k|k)=\hat{x}^{(i)}(k|k-1)+G_0^{(i)}(k)\left[y^{(i)}(k)-\sum_{d=0}^{L_i}H^{(i)}(k-d)\hat{x}^{(i)}(k-d|k-1)\right], \tag{4}$$

$$\hat{x}^{(i)}(k-h|k)=\hat{x}^{(i)}(k-h|k-1)+G_h^{(i)}(k)\left[y^{(i)}(k)-\sum_{d=0}^{L_i}H^{(i)}(k-d)\hat{x}^{(i)}(k-d|k-1)\right], \tag{5}$$

where $G_h^{(i)}(k)$, $h=0,1,...,M$ is a gain matrix, and formed by

$$G_h^{(i)}(k)=\sum_{d=0}^{L_i}P^{(ii)}(k-h,k-d|k-1)H^{(i)^T}(k-d)\left[\sum_{d_1,d_2=0}^{L_i}H^{(i)}(k-d_1)P^{(ii)}(k-d_1,k-d_2|k-1)H^{(i)^T}(k-d_2)+R^{(i)}(k)\right]^{-1}, \quad (6)$$

where $P^{(ii)}(k_1,k_2|k)$ is an auto-covariance matrix, i.e.,

$$P^{(ii)}(k_1,k_2|k)\triangleq\operatorname{cov}\left\{\tilde{x}^{(i)}(k_1|k),\tilde{x}^{(i)}(k_2|k)\right\},\quad \tilde{x}^{(i)}(k_1|k)\triangleq x(k)-\hat{x}^{(i)}(k_1|k),\quad k_1,k_2\le k. \quad (7)$$

Moreover, the specific equation for $P^{(ii)}(k_1,k_2|k)$, $k_1,k_2\le k$ and the time-update equations are given by

$$P^{(ii)}(k-h_1,k-h_2|k)=P^{(ii)}(k-h_1,k-h_2|k-1)-G_{h_1}^{(i)}(k)\sum_{d=0}^{L_i}H^{(i)}(k-d)P^{(ii)}(k-d,k-h_2|k-1), \quad (8)$$

$$\begin{aligned}P^{(ii)}(k-h,k+1|k)&=\sum_{h_1=0}^{M}P^{(ii)}(k-h,k-h_1|k)F^T(k-h_1),\\ P^{(ii)}(k+1,k+1|k)&=\sum_{h_1,h_2=0}^{M}F(k-h_1)P^{(ii)}(k-h_1,k-h_2|k)F^T(k-h_2)+Q(k).\end{aligned} \quad (9)$$

where $P^{(ii)}(h,j|0)=P_0$, $h,j=0,-1,-2,...,\bar{M}$, $\bar{M}=\max\{M,L_i\}$.

Then, using the above filtering equations (3)-(9), we have N local estimates $\hat{x}^{(1)}(k|k),\ldots,\hat{x}^{(N)}(k|k)$ and corresponding error-covarinaces $P^{(11)}(k,k|k),...,P^{(NN)}(k,k|k)$. Based on these values, DFF is presented below.

4. Distributed Fusion Filtering

4.1. Fusion Formula with Matrix Weights

Considering N local estimates $\hat{x}^{(1)}(k|k),...,\hat{x}^{(N)}(k|k)$ and corresponding error-covariance $P^{(11)}(k,k|k),\ldots,P^{(NN)}(k,k|k)$, the distributed fusion estimate $\hat{x}^{DFF}(k|k)$ is determined using the following fusion formula with matrix weights [6, 7]:

$$\hat{x}^{DFF}(k|k)=\sum_{i=1}^{N}A_k^{(i)}\hat{x}^{(i)}(k|k),\quad \sum_{i=1}^{N}A_k^{(i)}=I_n, \quad (10)$$

where I_n is an $n\times n$ identity matrix, $A_k^{(i)}$, $i=1,...,N$ are $n\times n$ matrix weights defined as

$$A_k=\left(D^TP_{\tilde{x},k}^{-1}D\right)^{-1}D^TP_{\tilde{x},k}^{-1},\quad D=\left[I_n\ldots I_n\right]^T, \quad (11)$$

where $A_k=\left[A_k^{(1)}\ldots A_k^{(N)}\right]\in\mathbb{R}^{n\times nN}$, $P_{\tilde{x},k}=\left[P^{(ij)}(k,k|k)\right]\in\mathbb{R}^{nN\times nN}$, $P^{(ij)}(k,k|k)=\operatorname{cov}\left\{\tilde{x}^{(i)}(k|k),\tilde{x}^{(j)}(k|k)\right\}$, $i,j=1,...,N,\ i\ne j$.

In order to compute the matrix weights $W_k^{(i)}$, the local cross-covariances $P^{(ij)}(k,k|k)$, $i,j=1,...,N,\ i\ne j$ in (11) are required. Subsequently, the specific equations for the cross- covariance $P^{(ij)}(k,k|k)$ are given by

$$\begin{aligned}P^{(ij)}(k,k|k)&=P^{(ij)}(k,k|k-1)-G_0^{(i)}(k)S_0^{(i)}(k)-S_0^{(j)^T}(k)G_0^{(j)^T}(k)+G_0^{(i)}(k)T^{(ij)}(k)G_0^{(j)^T}(k),\\ S_m^{(i)}(k)&=\sum_{d=0}^{L_i}H^{(i)}(k-d)P^{(ij)}(k-d,k-m|k-1),\\ T^{(ij)}(k)&=\sum_{d_i=0}^{L_i}\sum_{d_j=0}^{L_j}H^{(i)}(k-d_i)P^{(ij)}(k-d_i,k-d_j|k-1)H^{(j)^T}(k-d_j),\end{aligned} \quad (12)$$

$$P^{(ij)}(k-h_i,k-h_j|k)=P^{(ij)}(k-h_i,k-h_j|k-1)-G_{h_i}^{(i)}(k)S_{h_j}^{(i)}-S_{h_i}^{(j)^T}G_{h_j}^{(j)^T}(k)+G_{h_i}^{(i)}(k)T^{(ij)}(k)G_{h_j}^{(j)^T}(k), \quad (13)$$

$$P^{(ij)}(k-h,k+1|k)=\sum_{d=0}^{M}P^{(ij)}(k-h,k-d|k)F^{T}(k-d),$$
$$P^{(ij)}(k+1,k+1|k)=\sum_{d_1,d_2=0}^{M}F(k-d_1)P^{(ij)}(k-h,k-d_2|k)F^{T}(k-d_2)+Q(k). \qquad (14)$$

To reduce the computational burden of (12)-(14), an approximation approach is presented in the next subsection.

4.2. Computation of Cross-Covariances via Approximation

There is a relationship between the covariances, $P^{(ii)}(k,k|k)$, $P^{(jj)}(k,k|k)$, and $P^{(ij)}(k,k|k)$, such that [10]:

$$P^{(ij)}(k,k|k)=S^{(ii)}(k)\Gamma^{(ij)}(k)S^{(jj)}(k),\ \ i,j=1,...,n,\ \ i\neq j, \qquad (15)$$

where $S^{(hh)}(k)=\mathrm{diag}\left[\sigma_{11}^{(hh)}(k)...\sigma_{nn}^{(hh)}(k)\right]$, $\sigma_{ss}^{(hh)}(k)=\sqrt{P_{ss}^{(hh)}(k,k|k)}$, $P^{(hh)}(k,k|k)=\left[P_{sm}^{(hh)}(k,k|k)\right]_{s,m=1}^{n}$, $h=i,j$, $\Gamma^{(ij)}(k)=\left[r_{sm}^{(ij)}(k)\right]_{s,m=1}^{n}$, $r_{sm}^{(ij)}(k)=P_{sm}^{(ij)}(k)/\left[\sigma_{ss}^{(ii)}(k)\sigma_{mm}^{(jj)}(k)\right]$.

Next, we replace the time-varying correlation matrix $\Gamma^{(ij)}(k)$ in (15) by its time-invariant $\hat{\Gamma}^{(ij)}$, which has considerably simplified computational complexity of the cross-covariance calculations (15). Thus, we obtain

$$P^{(ij)}(k,k|k)\approx S^{(ii)}(k)\hat{\Gamma}^{(ij)}S^{(jj)}(k). \qquad (16)$$

Here, the approximation parameter $\hat{\Gamma}^{(ij)}$ can be pre-calculated in steady-state regime, i.e., $\hat{\Gamma}^{(ij)}\approx\Gamma^{(ij)}(\infty)=S^{(ii)^{-1}}(\infty)P^{(ij)}(\infty)S^{(jj)^{-1}}(\infty)$. We refer the distributed fusion filtering using the approximation (15) as DFFA .

5. Numerical Example

Let us consider a autoregressive scalar signal with the order 5, AR(5) with 5 different time-delayed observations. We have the form:

$$\begin{aligned}
x(k)&=0.9x(k)+0.8x(k-1)+0.7x(k-2)+0.8x(k-3)+0.7x(k-4)+w(k),\\
y^{(1)}(k)&=x(k)+0.8x(k-2)+v^{(1)}(k),\\
y^{(2)}(k)&=x(k)+0.5x(k-1)+0.4x(k-2)+v^{(2)}(k),\\
y^{(3)}(k)&=0.8x(k-1)+0.6x(k-3)+0.5x(k-4)+v^{(3)}(k),\\
y^{(4)}(k)&=0.7x(k-2)+0.5x(k-4)+v^{(4)}(k),\\
y^{(5)}(k)&=x(k)+0.7x(k-2)+0.5x(k-3)+v^{(5)}(k),
\end{aligned} \qquad (17)$$

where $x(0)\sim\mathcal{N}(1,5)$, $w(k)\sim\mathcal{N}(0,1)$, $v^{(i)}(k)\sim\mathcal{N}(0,1)$, $i=1,...,5$.

Using the signal model (17), the fusion estimates, $\hat{x}^{DFF}(k|k)$ and $\hat{x}^{DFFA}(k|k)$ based on DFF and DFFA are obtained using computer computations, respectively. To compare the performances (accuracies) of fusion estimates $\hat{x}^{DFF}(k|k)$, $\hat{x}^{DFFA}(k|k)$, the corresponding two mean square errors (MSEs) are considered, i.e.,

$$P_k^{DFF}=\mathbf{E}\left[x(k)-\hat{x}^{DFF}(k|k)\right]^2,\ P_k^{DFFA}=\mathbf{E}\left[x(k)-\hat{x}^{DFFA}(k|k)\right]^2, \qquad (18)$$

and the implementation time is calculated as CPU time using the computer with the following specifications: Intel® Xeon® E5450 3GHz, 1GB RAM.

In Figure 1, we observe that the MSE of DFF, P_k^{DFF} is slightly lower than P_k^{DFFA}. This means, P_k^{DFF} is obviously more accurate than P_k^{DFFA}, since P_k^{DFF} uses the accurate cross-covariances. However, the difference between P_k^{DFF} and P_k^{DFFA} is small, and we see that DFFA is computationally efficient in Table 1. Therefore, we conclude that DFFA is applicable in real-time implementation, especially for multiple time delayed signals.

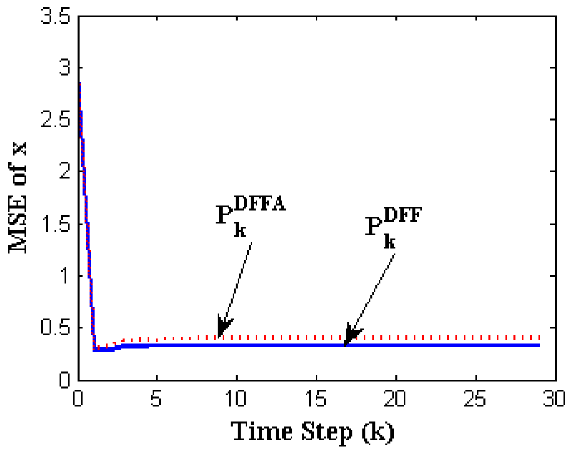

Fig. 1: MSEs of $\hat{x}^{DFF}(k|k)$ and $\hat{x}^{DFFA}(k|k)$

Table 1: Implementation Time of Fusion Filters

Distributed Fusion Filters	CPU time (sec.)
DFF	0.0311
DFFA	0.0188

6. Conclusions

In this paper, a distributed fusion filtering problem is considered for multiple time-delayed signals from a system. To solve this problem, the important values are local cross-covariances, and the equations for them are presented. Further, to overcome their computational burden, an approximation scheme which uses special steady-state approximation for computing local cross-covariances is proposed. Subsequent analysis of the proposed fusion algorithm is presented using a numerical example supporting the computational effectiveness of the proposed fusion algorithm.

7. References

[1] H. Kwakernaak, "Optimal Filtering in Linear Systems with Time Delays," *IEEE Trans. on Automatic Control,* vol. 12, no. 2, pp. 169-173, 1967.

[2] R. Priemer, and A. G. Vacroux, "Estimation in Linear Discrete Systems with Multiple Time Delays," *IEEE Trans. on Automatic Control,* vol. 14, no. 4, pp. 384-387, 1969.

[3] J. Mishra, and V. S. Rajamani, "Least-Squares State Estimation in Time-Delay Systems with Colored Observation Noise: An Innovation Approach," *IEEE Trans. on Automatic Control,* vol. 20, no. 1, pp. 140-142, 1975.

[4] Y. Bar-Shalom, and X. R. Li, *Multi-target Multi-sensor Tracking: Principles and Techniques*, Storrs: YBS Publishing, 1995.

[5] X. R. Li, Y. M. Zhu, J. Wang *et al.*, "Optimal Linear Estimation Fusion - Part I: Unified Fusion Rules," *IEEE Trans. on* Information *Theory,* vol. 49, no. 9, pp. 2192-2208, 2003.

[6] V. Shin, Y. Lee, and T. Choi, "Generalized Millman's Formula and its Applications for Estimation Problems," *Signal Processing,* vol. 86, no. 2, pp. 257-266, 2006.

[7] J. Zhou, Y. Zhu, Z. You *et al.*, "An Efficient Algorithm for Optimal Linear Estimation Fusion in Distributed Multisensor Systems," *IEEE Trans. on Systems, Man, and Cybernetics,* vol. 36, no. 5, pp. 1000-1009, 2006.

[8] Y. M. Zhu, Multisensor *Decision and Estimation Fusion*, Boston: Kluwer, 2002.

[9] A. G. O. Mutambara, *Decentralized Estimation and Control for Multisensor Systems*, Boca Raton: CRC Press, 1998.

[10] S. Lee, and V. Shin, "Computationally Efficient Multisensor Fusion Estimation Algorithms," *Journal of Dynamic Systems,* Measurement*, and Control,* vol. 132, no. 2, pp. 024503.1-024503.4, 2010.