

Unified Optimal Linear Estimation Fusion— Part II: Discussions and Examples

X. Rong Li *

Department of Electrical Engineering
University of New Orleans
New Orleans, LA 70148
Phone: 504-280-7416, Fax: -3950, xli@uno.edu

Jie Wang

School of Electronic & Information Engr
Xi'an Jiaotong University
Xi'an, Shaanxi, 710049, P. R. China
czhan@xjtu.edu.cn

Abstract—Several unified optimal estimation/track fusion rules in the sense of best linear unbiased estimation (BLUE) and weighted least squares (WLS) have been presented in Part I [6] for centralized, distributed, and hybrid fusion architectures. This paper discusses their pros and cons, verifies these rules, and demonstrate via simulation examples how these fusion rules can be used in cases with either complete, incomplete, or no prior information about the estimatee (i.e., the quantity to be estimated).

Keywords: Estimation fusion, track fusion, fusion rules, BLUE, least squares, MMSE estimation

1 Introduction

The following has been presented in Part I [6]: a general and systematic approach to estimation/track fusion, which is much more general and flexible than the previously available approaches; a variety of fusion rules in unified forms that are optimal in the linear class for centralized, distributed, and hybrid fusion architectures. These rules are optimal for an arbitrary number of sensors in the presence of the various cross correlation in the sense of either the weighted least-squares (WLS) or best linear unbiased estimation (BLUE) sense—i.e., linear minimum variance (LMS) or mean-square error (LMMSE). This present paper discusses the pros and cons, and a variety of issues relevant to these fusion rules. Numerical examples are provided here to verify the theoretical results presented in Part I, and to illustrate the usefulness of these rules and their relationships.

Table 1 summarizes most of the unified BLUE and optimal WLS rules for centralized, standard, and the simple nonstandard distributed fusion presented in Part I. Table 2 summarizes BLUE, optimal WLS, and optimal generalized WLS fusers for the unified data model of Part I.

It should be clear that these fusion rules are all very easily implementable—there is no equation to solve. The pseudoinverse required in some of these rules can be computed by such software as Matlab.

*Supported in part by ONR Grant N00014-00-1-0677, NSF via Grant ECS-9734285, LEQSF via Grant (1996-99)-RD-A-32.

2 On Unified Optimal Fusion Rules

2.1 Centralized Fusion vs. Distributed Fusion

When the estimatee x and the data $Z = \{z_1, \dots, z_n\}$ are jointly Gaussian, the local estimates are sufficient statistics of the local data about the estimatee if the data errors are uncorrelated with each other and with the estimatee. Under this condition, it is clear that the distributed fusion rules presented in Part I are equivalent to the optimal centralized fusion rules in terms of optimality, and in fact, they are algebraically equivalent [7]. If x and z are not jointly Gaussian but z_i are linear (more rigorously, affine) functions of x , and the local estimates are BLUE estimates of x , then it can be shown that the distributed fusion rules presented in Part I are also algebraically equivalent to the optimal linear centralized fusion rules [7].

The fusion rules presented in Part I are unified in the sense that they are valid for all centralized, distributed, and hybrid fusion architectures. The differences of these fusion architectures lie in the quantities used in the fusion rules, as highlighted below.

The fusion rules presented for all three types of fusion architectures need knowledge of $C = \text{cov}(v^n)$. In fact, their primary differences lie in what the covariance C actual is and how easily it can be obtained:

- For the centralized (measurement) fusion, $C = R = \text{cov}(\eta_1, \dots, \eta_n)$, which is the covariance of all local sensors' observation noises.
- For the standard distributed (estimate) fusion, $C = \Sigma = \text{cov}(\tilde{x}_1, \dots, \tilde{x}_n)$, which is the *a priori* covariance of estimation errors of local estimates.
- For the simple nonstandard distributed fusion, $C = KRK'$, where $K = \text{diag}(K_1, \dots, K_n)$, which is determined by the covariance of all local sensors' observation noises and all local estimators' gains K_i .

The following general remarks can be made concerning the computation of C for the three fusion architectures.

R is usually much easier to obtain than Σ in practice and is often block-diagonal. It is usually time-invariant even for a dynamic system.

Σ is not block-diagonal and is dependent on how many data points are used in each local sensor. For example, it is usually time-variant when the estimatee is a random

process, such as when it is the state of a dynamic system. Thus, the key in the application of the fusion rules presented in Part I for the standard distributed fusion is how to obtain the covariance matrix Σ . In general, three approaches may be used:

- For some systems, Σ can be obtained exactly. Formulas for computation of Σ for general linear systems is presented in a forthcoming paper (a simple example is given in this paper for the use in the Numerical Examples section).
- For some other systems, Σ can be determined numerically, or by an experiment or simulation.
- For other systems, Σ can be tuned; that is, select Σ that yields the best fusion results for the problem under consideration. This is similar to the tuning of the process noise covariance Q in the Kalman filter, which is almost never known in practice. Of course, Σ often has a higher dimension than Q and its off-diagonal blocks are usually not zero. Note, however, that the diagonal blocks $\Sigma_{ii} = P_i = \text{cov}(\hat{x}_i)$ are known, where P_i is the error covariance of the local estimate \hat{x}_i .

Σ is in general not block diagonal, even in the case where the observation errors are independent across sensors, because the same *random estimatee* is estimated by all local estimators. For example, even if the sensor observation noises are uncorrelated, the local estimation errors of the state of a stochastic system are correlated because of such factors as correlated initial estimates (e.g., common feedback from fusion center) or the common process noise in the system dynamics on which each local estimator is based. Σ in this case and in some more general and coupled cases can be easily obtained (given in a forthcoming paper). In general, Σ quantifies the coupling among local estimates. It, or its equivalent, is needed for optimal standard distributed fusion. This availability assumption for Σ guarantees the optimality of the fusion rules presented given the coupling among local estimates. When this coupling is neither known, obtainable, nor tunable, the above fusion results are not applicable directly but may facilitate the development of the corresponding optimal fusion rules.

KRK' is easier to obtain than Σ but less convenient to use than R . It is often block-diagonal since R is often block-diagonal. It is, however, dependent on local estimators' gains K_i , which depend on how many data points are used in each local sensor. KRK' is usually time-variant when the estimatee is a random process.

In addition, correlation $A = \text{cov}(x, v)$ between noise v and estimatee x is often zero for centralized fusion, but nonzero for distributed fusion.

Of course, channel requirements to send $\{Z, R\}$ for centralized fusion and $\{X, \Sigma\}$ for standard distributed fusion are different, depending on the dimensions of the observation and state vectors, where $X = \{\hat{x}_1, \dots, \hat{x}_n\}$.

2.2 Usefulness of Fusion Rules Presented

It should be emphasized that the fusion rules for the standard distributed fusion is very general. They are valid

for the “data” set Y as *any set* of observations (for the centralized architecture) or unbiased estimates (for the distributed architectures). In general, the BLUE and the optimal generalized WLS fusion rules (with prior knowledge) rely only on the assumption that the covariance matrix Σ , the prior mean \bar{x} , and the associated covariance P_0 (or P_0^{-1} in the incomplete knowledge case) are known. For the WLS fusion rules (and the BLUE fusion rule without prior knowledge) to be unbiased, it also requires that the local estimators be unbiased. Both of these assumptions are fairly reasonable. It is hard to imagine how one can in general have an unbiased estimator by fusing biased local estimates unless the biases are known perfectly. If, however, the biases are known, then one can obtain unbiased local estimates in the first place.

Note that in the standard distributed fusion, \hat{x}_i actually could be any (unbiased) estimate of x . For example, one could define the “equivalent data” set to be

$$Y = [\hat{x}_{k-1}^1, \hat{x}_k^1, \hat{x}_{k-2}^2, \hat{x}_k^2, \dots, \hat{x}_{k-1}^m]'$$

where \hat{x}_i^j denotes the estimate by the j th local estimator using data $\{z_1^j, \dots, z_i^j\}$. With this understanding, most of the advantages of the fusion rules presented in Part I over existing formulas, as discussed below, can be easily appreciated:

- The fusion rules of Part I are valid for cases with observation noises that are coupled across sensors, across time, and/or with the estimatee. Note first that as shown in a forthcoming paper, the observation noises of a sampled system of a continuous-time multiple-sensor system are correlated. This is also useful in practice when the estimatee is observed in a common noisy environment, such as when a target is taking countermeasures, e.g. noise jamming, or when the sensor noises are coupled because of, say, their dependence on the target state. A class of systems that fall into this category is given in the next section. Another important application area is the fusion of estimates based on observations obtained over different time periods from the same sensor. The fusion-based optimal smoothers presented in [5] using measurements corrupted by (not necessarily Markov) autocorrelated noise is a good application example of these fusion rules. Almost all previous fusion results assume that the sensor observations are conditionally independent given the estimatee.
- The fused estimates depend on the network structures or communication patterns only through C . Consequently, the fusion rule is invariant for a given C no matter if there is feedback or not, if the network has a parallel, tandem, tree or general structure, or what the communication bandwidths are. This means that in practice only the coupling among local estimates needs to be obtained. If there is no fusion center, such as for the network structure considered in [4], each estimator can use these fusion rules to obtain the best estimates based on its own observations and any information received from other sensors.

• The local estimators do not have to use the same local model. This is particularly important for extension of the fusion rules for dynamic processes in the asynchronous case. The local estimates may be obtained based on different dynamic models and then be fused using these fusion rules. The use of different dynamic models for the local estimators may be necessary or more effective. It is necessary in many practical situations. One example is when state augmentation is needed for some local estimators with autocorrelated sensor measurement noise. Another is that local estimators operate in different coordinate systems, as is often the case in practice. It may be more effective, for example, when multiple models are used for the process to be estimated. Of course the key to a successful application of these fusion rules in such cases lies in the determination of the covariance matrix Σ . In order to have the best fusion performance, the choice of the local models is of course not arbitrary (see, e.g., [1]).

• There is no requirement on synchronism of the local estimates provided they can be converted to the estimates of the state at the same time (based on observations up to different times) in the sensor-to-sensor track fusion strategy. In reality, it is difficult to synchronize local estimates. However, many existing fusion rules are optimal only for synchronized local estimates and thus require artificial synchronization for the local estimates in application to achieve the best performance in application. This either increases the computational burden considerably or degrades the fusion performance. The fusion rules of Part I provide a more convenient and efficient framework for fusing asynchronous local estimates. For example, they are directly applicable (optimally) for the sensor-to-system track fusion. They are also applicable (optimally) for the sensor-to-sensor track fusion strategy after a simple conversion of all local estimates to the same point in time at which the state is to be estimated by the fusion center, where the conversion could either be prediction or smoothing (retrodiction). More details are given in a forthcoming paper.

• The local estimators need not be of the same type. For example, fusion rules presented are valid if some local estimators are MMSE (minimum mean-square error) estimators, while others are MAP (maximum a posteriori) estimators. This flexibility is useful for some applications [3].

3 A Simple Example of Recursive Computation of Joint Error Covariance Σ

This section presents a simple example of how to compute the joint error covariance matrix Σ recursively for a class of linear data models. It is used for the numerical examples of the next section. More general formulas for computation of Σ are given in a forthcoming paper.

Consider the following model of linear noisy measure-

ments, for all sensor $i \leq n$ and all measurements $k > 0$,

$$z_k^i = H_k^i x + \nu_k^i + e_k^i \quad (1)$$

where the measurement noise is the sum of two zero-mean white sequences ν_k^i and e_k^i (over k):

$$\begin{aligned} E[\nu_k^i] &= 0, & E[\nu_k^i \nu_m^j] &= \delta_{ij} \delta_{km} \bar{R}_k^i \\ E[e_k^i] &= 0, & E[e_k^i e_m^j] &= \delta_{km} \tilde{R}_k^{ij} \\ \text{cov}(x - \bar{x}^i, \nu_k^j) &= 0, & \text{cov}(x - \bar{x}^i, e_k^j) &= 0 \end{aligned}$$

for any $k > 0$ and $i, j = 1, \dots, n$, where δ_{km} is the Kronecker delta function; that is, while ν_k^i are independent across sensors, e_k^i are coupled across sensors, and they are uncorrelated with $x - \bar{x}$. Clearly, this system reduces to the one with independent measurement noise when $e_k^i \equiv 0$. Such a model may be useful when e.g., a target is generating noise jamming or when the estimatee is observed in a common noisy environment. Note that for a linear recursive local estimator

$$\hat{x}_k^i = \hat{x}_{k-1}^i + K_k^i (z_k^i - H_k^i \hat{x}_{k-1}^i) \quad (2)$$

one has

$$\tilde{x}_k^i = x - \hat{x}_k^i = (I - K_k^i H_k^i) \tilde{x}_{k-1}^i - K_k^i (\nu_k^i + e_k^i)$$

where K_k^i is the gain, not necessarily optimal. From the whiteness of $\{\nu_k^i\}$ and $\{e_k^i\}$ and their uncorrelatedness with the initial estimation errors, it thus follows that, for any $k > 0$ and $i, j = 1, \dots, n$,

$$\begin{aligned} \Sigma_{kk}^{ij} &= \text{cov}(\tilde{x}_k^i, \tilde{x}_k^j) \\ &= (I - K_k^i H_k^i) \Sigma_{k-1, k-1}^{ij} (I - K_k^j H_k^j)' \\ &\quad + K_k^i (\delta_{ij} \bar{R}_k^i + \tilde{R}_k^{ij}) K_k^j \\ \Sigma_{k-1, k}^{ij} &= \text{cov}(\tilde{x}_{k-1}^i, \tilde{x}_k^j) = \Sigma_{k-1, k-1}^{ij} (U_k^j)' \\ \Sigma_{k-2, k}^{ij} &= \text{cov}(\tilde{x}_{k-2}^i, \tilde{x}_k^j) = \Sigma_{k-2, k-1}^{ij} (U_k^j)' \\ &\vdots \end{aligned}$$

The matrix form of this off-line recursion for Σ is

$$\Sigma_{ks} = \begin{cases} \Sigma_{k, s-1} U_s' & k < s \\ U_k \Sigma_{k-1, k-1} U_k' + K_k (\bar{R}_k + \tilde{R}_k) K_k' & k = s \\ U_k \Sigma_{k-1, s} & k > s \end{cases} \quad (3)$$

where

$$\begin{aligned} U_k^i &= (I - K_k^i H_k^i), & U_k &= \text{diag}(U_k^1, \dots, U_k^n) \\ K_k &= \text{diag}(K_k^1, \dots, K_k^n), & \bar{R}_k &= \text{diag}(\bar{R}_k^1, \dots, \bar{R}_k^n) \\ \tilde{R}_k &= \begin{bmatrix} \tilde{R}_k^{ij} \end{bmatrix} \end{aligned}$$

The initialization of the above with complete prior knowledge is $\Sigma_0 = \text{cov}(x - \bar{x}^1, \dots, x - \bar{x}^n)$. When there is no prior information, Σ_{ks} should be initialized by

$$\Sigma_{1,1} = E[(x - \hat{x}_1^i)(x - \hat{x}_1^j)'] = K_1 (\bar{R}_1 + \tilde{R}_1) K_1' \quad (4)$$

because $K_1^i H_1^i = I, \forall i$, and

$$x - \hat{x}_1^i = K_1^i H_1^i x - K_1^i (H_1^i x + \nu_1^i + e_1^i) = -K_1^i (\nu_1^i + e_1^i)$$

When there is incomplete prior information, by Theorem 3 of Part I and above, Σ_{ks} should be initialized by

$$\Sigma_{1,1} = K_1 R_1 K_1', \quad R_1 = \text{diag}(R_k^1, \dots, R_k^n) \quad (5)$$

with $K_1 = \text{diag}(V K_1^1, \dots, V K_1^n)$, where K_1^i is the i th local BLUE gain without prior information using

$$\tilde{H}_1^i = \begin{bmatrix} [I_{r \times r}, 0] \\ H_1^i V \end{bmatrix}, \quad R_1^i = \begin{bmatrix} \Lambda_1 & 0 \\ 0 & \bar{R}_1^i + \tilde{R}_1^i \end{bmatrix} \quad (6)$$

4 Illustrative Numerical Examples

In this section, illustrative numerical examples of the BLUE and optimal (generalized) WLS fusers for the centralized, standard distributed, and the simple nonstandard distributed (Subsection 3.3 of Part I) architectures are presented. These examples serve primarily three purposes: (a) to verify the various fusion rules and their relationships presented; (b) to demonstrate the generality and usefulness of the various fusion rules developed; and (c) to illustrate how the cases with incomplete or no prior information of the estimatee can be handled optimally.

Three examples are given for the following three cases: (a) knowing prior mean \bar{x} and its nonsingular covariance P_0 , (b) incomplete prior information (i.e., knowing prior mean \bar{x} and its singular $P_0^{-1} \neq 0$), and (c) without prior knowledge of the estimatee x . In each example, BLUE, optimal WLS, and optimal GWLS (generalized WLS) fused estimates are computed for the centralized and distributed architectures.

The local estimates were always obtained by the BLUE estimator if there is prior information about the estimatee x and by the optimal WLS estimator in the case of no prior information, which, in fact, is still equal to the corresponding BLUE estimator without prior knowledge. All local estimators used the same prior information whenever it is available and thus their error covariance $\Sigma_0 = \text{cov}(x - \bar{x}, \dots, x - \bar{x}) = [\Sigma_0^{ij}]$ with $\Sigma_0^{ij} = P_0, \forall i, j$. Note that this matrix is singular.

4.1 Data Model

In all the examples, each piece of two-dimensional data z_k^i at the i th sensor is related to the estimatee x by

$$z_k^i = H_k^i x + v_k^i = \begin{bmatrix} 1 & 5 \\ 0 & 1 \end{bmatrix} x + \nu_k^i + e_k^i \quad (7)$$

where z_k^i denotes the k th piece of data at the i th sensor, and likewise for others; $\nu_k^i, i = 1, \dots, n, k = 1, 2, \dots$, are i.i.d. (independent and identically distributed) zero-mean Gaussian vectors with covariance:

$$\begin{aligned} \bar{R}_k &= \text{cov}(\nu_k) = \text{cov}(\nu_k^1, \dots, \nu_k^n) = I \\ \text{cov}(\nu_k, \nu_m) &= \delta_{km} \bar{R}_k \end{aligned}$$

e_k^i are zero-mean random vectors, independent across k but correlated across i for the same k :

$$E[e_k] = 0, \quad \text{cov}(e_k, e_m) = \delta_{km} \tilde{R}_k$$

$$\begin{aligned} \tilde{R}_k &= \text{cov}(e_k) = \text{cov}(e_k^1, \dots, e_k^n) = [\tilde{R}_k^{ij}] \\ \tilde{R}_k^{ij} &= \text{cov}(e_k^i, e_k^j) \\ &= \begin{cases} 3^{\min\{i,j\}-1} (0.8\sigma^2)^{i+j} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, & i \neq j \\ \begin{bmatrix} \frac{1}{3} [3(0.8\sigma^2)^2]^i & 0 \\ 0 & 2\sigma^4 \end{bmatrix}, & i = j \end{cases} \quad (8) \end{aligned}$$

Note that the covariance matrix $R_k = \text{cov}(v_k) = \bar{R}_k + \tilde{R}_k$ of the error vector is not block-diagonal. $\{e_k^i, \forall i, k\}$ and $\{\nu_m^j, \forall j, m\}$ are independent.

The observation noise e_k may be generated as follows: First generate i.i.d. zero-mean random vectors \tilde{e}_k with covariance I and then set $e_k = \tilde{R}_k^{1/2} \tilde{e}_k$. It should be noted that the distribution of \tilde{e}_k is arbitrary in this method of generating e_k . In the simulation, \tilde{e}_k were generated as standard Gaussian vectors in this method.

Alternatively, e_k were also generated with a fixed non-Gaussian distribution as follows: $e_k^i = \begin{bmatrix} \eta_k^i \\ (\xi_k^i)^2 - \sigma^2 \end{bmatrix}$, where $\xi_k^i, i = 1, \dots, n, k = 1, 2, \dots$, are i.i.d. zero-mean Gaussian variables with variance σ^2 ; η_k^i are independent across k but correlated across i for the same k :

$$\eta_k^i = 0.8\eta_k^{i-1}(\xi_k^i)^2 = (0.8)^i (\xi_k^1)^2 \dots (\xi_k^i)^2 \eta_k^0$$

η_k^0 is a zero-mean random variable uniformly distributed over $(-1, 1)$; $\{\xi_k^i, \forall i, k\}$ and η_m^0 are mutually independent for any m . Note that $\{e_k^i\}$ generated this way as a sequence of i is not Markov. It can be shown straightforwardly that e_k has a zero mean and the covariance given by (8).

Note that all centralized and distributed fusion rules presented in Part I are optimal in the case where e_k are generated by the first method with \tilde{e}_k being standard Gaussian vectors since the local estimates are sufficient statistics in this case. In the case where the second method is used, however, they are only the best linear centralized and distributed fusers, respectively.

The required joint error covariance matrix was computed by (3) using (8).

4.2 Performance Measures

Results from both single and multiple Monte Carlo runs are presented. The RMS error versus the data size k is computed by $\text{RMS_error}(k) = \sqrt{\frac{1}{M} \sum_{j=1}^M [\tilde{x}_k(j)]' \tilde{x}_k(j)}$, where M is the number of runs and $\tilde{x}_k(j)$ denotes the estimation error of the fused estimate in the j th run using all data up to k : $y^k(j) = [y_1(j)', \dots, y_k(j)']'$, where $y_k(j)$ are from the unified data model; for example, they are the actual data for the centralized architecture and the local estimates for the standard distributed architecture, respectively. An RMS error curve of a single run corresponds to $M = 1$ in the above. The average normalized estimation error squared (ANEES) curve is computed by

$$\text{ANEES}(k) = \frac{1}{\text{dim}(x)} \frac{1}{M} \sum_{j=1}^M [\tilde{x}_k(j)]' [P_k(j)]^{-1} \tilde{x}_k(j)$$

where $P_k(j)$ is the covariance of $\tilde{x}_k(j)$ and $\dim(x)$ is the dimension of x . It is used to check the credibility of an estimator—how credible an estimator is. For a fixed k , it is chi-square distributed with an expected value of 1 and a variance of $2/[\dim(x)M]$ if $\tilde{x}_k^i(j)$ is assumed Gaussian distributed, which is reasonable in many cases. The RMS errors and ANEES for local estimates are defined similarly: with $\tilde{x}_k(j)$, $P_k(j)$, $y^k(j)$ replaced by their local sensor-level counterparts $\tilde{x}_k^i(j)$, $P_k^i(j)$, $z^{k,i}(j)$, respectively.

All results were obtained using Matlab with the following parameters: $n = 3$, $\sigma^2 = 1$, $k = 1, 2, \dots, 20$, $M = 100$.

4.3 Examples

Example 1: Fusion with Complete Prior Information.

The objectives of this example are three-fold: (a) to verify the optimality (in the minimum mean-square error sense) of the fusion rules for distributed as well as centralized architectures in the case where the estimatee and the observation noise are jointly Gaussian; (b) to demonstrate the performance improvement by fusion and by the use of prior information; and (c) to verify the equivalence between optimal GWLS fusion and the BLUE with complete prior information.

The estimatee x is a random variable with prior $\bar{x} = [10, 10]'$ and $P_0 = (1/3)I$. It has different realizations in different Monte Carlo runs. The noise $v_k^i = \nu_k^i$ was generated as Gaussian random vectors, independent of x . Note that v_k^i are uncorrelated across sensors (i.e., $e_k^i = 0$). It is well-known [7] that the optimal centralized and distributed fusers are algebraically equivalent when the observation noises are uncorrelated across sensors.

Parts (a) and (b) of Fig. 1 show the RMS errors of the BLUE, optimal WLS, and optimal GWLS fusers with centralized (CL), standard (SD), and the simple nonstandard distributed (DL) (Subsection 3.3 of Part I) fusion architectures over an arbitrary single run and over 100 runs, respectively, for $k = 1, 2, \dots, 20$. All the optimal WLS fusion results were obtained by ignoring the prior information. In particular, for optimal WLS fusion with the SD architecture, the local estimates were obtained from the corresponding optimal WLS estimators. Also shown is one of the three local BLUE estimates. The corresponding ANEES curves over 100 runs are shown in Part (c). Part (d) shows the curves of the theoretical Σ_{kk}^{ij} , $i, j = 1, 2$ [i.e., the (cross)covariances of the errors of the first two local sensor BLUE estimates], obtained by (3), and the corresponding sample covariances $\tilde{\Sigma}_{kk}^{ij} = \frac{1}{M} \sum_{l=1}^M \tilde{x}_k^i(l)[\tilde{x}_k^j(l)]'$.

Were the fusion rules all different, each part of Fig. 1(a)(b)(c) would have ten curves. Theoretically speaking (confirmed by the simulations), however, there should be no difference between BLUE and optimal GWLS, and between centralized, standard, and the simple nonstandard distributed fusion rules for this example. Consequently, Curve 1 represents the optimal WLS fusion results for CL, SD, and DL fusion architectures. Curve 2 represents the other fusion results. Curve 3 represents one of the three local estimates, which are different but statistically equivalent. Note that the BLUE and optimal WLS fusion results

for centralized fusion are identical to BLUE and optimal WLS estimates using all data points, respectively. The formulas of these estimators are known prior to this study [2] and thus their fusion results serve to verify the validity of the other fusion rules, which are contributions of this work. It can be seen from Fig. 1(d) that the theoretical and sample (cross)covariances are close with each other. This verifies the formulas used for computing Σ .

It is clear from Fig. 1 that the three objectives of this example have been achieved: Centralized and distributed fusers do have identical performance; fusion does enhance the performance; prior information does improve the performance since the optimal GWLS fusers have better performance than the optimal WLS fusers; the optimal GWLS and BLUE do coincide. Also, all the fusers and estimators are very credible since their ANEESs are all within the 2σ region of the χ^2 random variable.

Table 3 gives the computational complexities in terms of relative ratios of CPU time and floating point operation (FLOP) counts for the fusion rules, obtained from 100 runs.

Example 2: Fusion with Incomplete Prior Information.

This example demonstrates how to fuse when incomplete prior information about the estimatee x is available. Specifically, some components of the prior mean of x are known while others are not. This can be treated by setting an unknown component to an arbitrary number with an infinite prior variance, or more appropriately, a zero inverse. As such, in addition to \bar{x} , its corresponding P_0^{-1} is known, nonzero but singular. Note that P_0^{-1} would be the inverse of P_0 should P_0 exist, which does not exist in this case. The use of the symbol P_0^{-1} is thus convenient because it reflects this fact.

The observation errors e_k^i were generated using the second method. The estimatee x is a random variable, which was generated by $x = \bar{x} + \begin{bmatrix} \delta \\ \lambda \end{bmatrix}$, where $\bar{x} = [10, 10]'$; δ is a zero-mean random variable uniformly distributed over $(-1, 1)$ and is independent of observation errors $\{\eta_k\}$ and $\{\xi_k\}$. In fact, there is no knowledge about the second component of the prior mean and this was treated by setting it equal to $10 + \lambda$, where λ is a random variable with a diffuse uniform density [i.e., uniformly distributed over $(-\infty, \infty)$] and is independent of δ , $\{\eta_k^i, \forall i, k\}$ and $\{\xi_k^i, \forall i, k\}$. As a consequence, the covariance P_0 does not exist because $P_0^{-1} = \begin{bmatrix} 3 & 0 \\ 0 & 0 \end{bmatrix}$ and the estimatee x is independent of data errors; that is, $A = 0$. In the simulation, λ was generated first as a random variable uniformly distributed over $(-1, 1)$ and then multiplied by the largest number available to the computer.

Note that in this case, the BLUE fuser was given by Theorem 3 of Part I with $V = I_{2 \times 2}$, $A = 0$ and

$$\check{y}^n = \begin{bmatrix} 10 \\ y^n \end{bmatrix}, \quad \check{H} = \begin{bmatrix} [1, 0] \\ H \end{bmatrix}, \quad \check{C} = \begin{bmatrix} 1/3 & 0 \\ 0 & C \end{bmatrix}$$

It follows from Part I that the optimal GWLS fuser uses the same \check{y}^n , \check{H} , and \check{C}^{-1} .

Parts (a) and (b) of Fig. 2 show the RMS errors of the BLUE, optimal GWLS, and optimal WLS fusers with centralized (CL), standard (SD), and the simple nonstan-

standard distributed (DL) architectures over an arbitrary single run and over 100 runs, respectively. The corresponding ANEES curves over 100 runs are shown in Part (c). Part (d) shows the curves of the theoretical and sample (cross)covariances of the errors of the first two local BLUE estimates. Note that the DL data and data models are those of CL scaled by K and thus they have the same performance whenever the matrix K is invertible, which is true for this example. Consequently, Curve 1 represents BLUE and GWLS fusion with the CL and DL architectures; Curve 2 is from WLS fusion with the CL and DL architectures; Curve 3 is from BLUE and GWLS fusion with the SD architecture; and Curve 4 is from WLS fusion with the SD architecture. In Fig. 2(b), Curves 1 and 3 have very small but noticeable differences (the SD fusion results are very slightly worse), while Curves 2 and 4 have no noticeable difference although they actually do not coincide exactly. In Fig. 2(c), Curves 2 and 4 have no noticeable difference but actually not coincide exactly. In Fig. 2(d), the theoretical covariances TC(1,2) and TC(2,2) coincide. The ANEESs are greater than those in Example 1 because the estimation errors here are not Gaussian.

It can be seen from Fig. 2(a)(c) that the SD fusion results are not identical to the CL and DL fusion results, although they are extremely close on the average. When the observation noises are correlated across sensors, there is no proof in the literature that the optimal centralized and distributed fusers in general must have the same performance. The observation noises are correlated in this example. The simulation results presented here indicate that in such a case, the optimal centralized and distributed fusers are in general not equivalent. This result was not known before and is in direct contrast to the fact that the optimal centralized and distributed fusers are equivalent in the uncorrelated observation noise case. Although using the same “equivalent observations” in the unified model, the optimal standard and the optimal simple nonstandard distributed fusers may differ in general because they have different data models. Note also that the optimal WLS fusers performed worse than their BLUE and optimal GWLS counterparts because the latter two use prior information, albeit incomplete.

It is interesting to note that the BLUE and optimal GWLS fusion with SD architecture has the least credible covariances since its ANEES is the worst (farthest from 1). For the optimal WLS fusion, however, this deterioration in credibility seems not significant.

Example 3: Fusion without Prior Information. The same set-up as Example 2 was used here except that there is no prior information about the estimatee. The true value of the estimatee is fixed at $x = [10, 10]'$ over all runs, but is unknown to the local estimators and the fusers—there is no prior knowledge of x for the local estimators and the fusers. In this case, $A = 0$.

Parts (a) and (b) of Fig. 3 show the RMS errors of the BLUE and optimal WLS fusers with the CL, SD, and DL architectures over an arbitrary single run and over 100 runs, respectively. The corresponding ANEES curves over 100 runs are shown in Part (c). Part (d) shows the curves of the theoretical and sample (cross)covariances of the errors of the first two local estimates. The BLUE and optimal

WLS fusers are algebraically equivalent in this case. It is interesting to note that all three fusion architectures yield the same results in this case, where there is no prior information.

5 Summary

Pros and cons of the unified fusion rules of Part I for centralized, distributed, and hybrid estimation fusion architectures have been presented. Several numerical examples via computer simulation have been presented. They verify or demonstrate the following: (a) the fusion rules developed in Part I are valid for both centralized and various distributed fusion architectures with complete, incomplete, or no prior information; (b) the optimal centralized and distributed fusers have different performance when the observation noises are correlated across sensors, in contrast to the well-known fact that they are algebraically equivalent when observation noises are uncorrelated; (c) Prior information, be it complete or incomplete, does improve the optimal fusion results; and (d) the simple nonstandard distributed fusion outperforms the standard distributed fusion for correlated noise case.

References

- [1] A. T. Alouani and J. D. Birdwell, “Distributed Estimation: Constraints on the choice of the local models,” *IEEE Trans. Automatic Control*, vol. AC-33, pp. 503–506, May 1988.
- [2] Y. Bar-Shalom and X. R. Li, *Estimation and Tracking: Principles, Techniques, and Software*. Boston, MA: Artech House, 1993. (Reprinted by YBS Publishing, 1998).
- [3] O. E. Drummond, X. R. Li, and C. He, “Comparison of Various Static Multiple-Model Estimation Algorithms,” in *Proc. 1998 SPIE Conf. on Signal and Data Processing of Small Targets*, vol. 3373, pp. 510–527, 1998.
- [4] S. Grime and H. F. Durrant-Whyte, “Data Fusion in Decentralized Sensor Networks,” *Control Engineering Practice*, vol. 2, no. 5, pp. 849–863, 1994.
- [5] X. R. Li, C. Z. Han, and J. Wang, “Recursive Smoothing and Prediction for Linear Discrete-Time Systems,” in *Proc. 39th IEEE Conf. on Decision and Control (submitted)*, (Sydney, Australia), Dec. 2000.
- [6] X. R. Li, Y. M. Zhu, and C. Z. Han, “Unified Optimal Linear Estimation Fusion—Part I: Unified Models and Fusion Rules,” in *Proc. 2000 International Conf. on Information Fusion*, (Paris, France), July 2000.
- [7] Y. M. Zhu, J. Zhao, K. S. Zhang, X. R. Li, and Z. S. You, “Performance Analysis for Feedback Track Fusion,” in *Proc. the 3rd Chinese World Congress on Intelligent Control and Intelligent Automation*, (Hefei, China), June 2000.

Table 1: Fusion rules for centralized and distributed architectures.

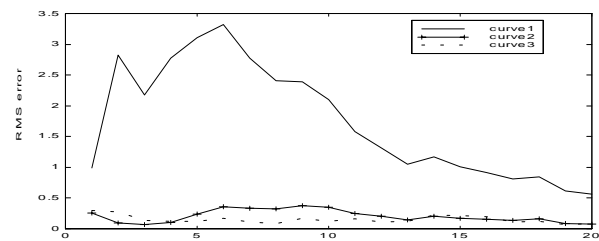
Fusion Method	Centralized	Standard Distributed	Simple Nonstandard Distributed
Data Model	$z_i = h_i x + \eta_i$	$\hat{x}_i = x + (-\tilde{x}_i)$	$y_i = K_i h_i x + K_i \eta_i$
Correlation	$R = \text{cov}(\eta_1, \dots, \eta_n)$	$\Sigma = \text{cov}(\tilde{x}_1, \dots, \tilde{x}_n)$	$R = \text{cov}(\eta_1, \dots, \eta_n)$
Data Received	z_1, \dots, z_n	$\hat{x}_1, \dots, \hat{x}_n$	$y_i = \hat{x}_i - (I - K_i h_i) \tilde{x}_i, i = 1, \dots, n$ K_1, \dots, K_n
Unification	$y = [z'_1, \dots, z'_n]'$ $H = [h'_1, \dots, h'_n]'$ $C = R$	$y = [\hat{x}'_1, \dots, \hat{x}'_n]'$ $H = [I, \dots, I]'$ $C = \Sigma$	$K = \text{diag}(K_1, \dots, K_n)$ $y = [y'_1, \dots, y'_n]'$ $H = \tilde{K} [h'_1, \dots, h'_n]'$, $C = \tilde{K} R \tilde{K}'$
Unification for BLUE	$A_i = \text{cov}(x, \eta_i)$ $A = [A_1, \dots, A_n]$	$P_i = \text{cov}(\tilde{x}_i)$ $A = -[P_1, \dots, P_n]$	$A_i = K_i \text{cov}(x, \eta_i)$ $A = [A_1, \dots, A_n]$
BLUE Fuser	Complete prior: $\bar{x} = E[x], P_0 = \text{cov}(x)$ $S = H P_0 H' + C + H A + (H A)'$ If S^{-1} exists, $K = (P_0 H' + A) S^{-1}$ Else, $K = (P_0 H' + A) S^+$ End if $\hat{x} = \bar{x} + K[y - H \bar{x}]$ $P = P_0 - K S K'$		Incomplete prior: $\bar{x} = E[x], P_0^{-1} \geq 0, P_0^{-1} = 0$ Let $P_0^{-1} = V \text{diag}(\Lambda_1^{-1}, 0) V'$, where $V = [V_1, V_2], V^{-1} = V'$ $\Lambda_1 = \text{diag}(\lambda_1, \dots, \lambda_r)$ $\tilde{H} = \begin{bmatrix} [I_{r \times r}, 0] \\ H V \end{bmatrix}$, $\tilde{C} = \begin{bmatrix} \Lambda_1 & -V_1' A \\ -(V_1' A)' & C \end{bmatrix}$ $T = I - \tilde{H} \tilde{H}^+, K = \tilde{H}^+ - \tilde{H}^+ \tilde{C} [\tilde{C} T]^+ \tilde{H}^+$ $\hat{x} = V K [(V_1' \bar{x}), y^{n'}]'$, $P = V K \tilde{C} K' V'$
WLS Fuser and BLUE Fuser Without Prior			If C^{-1} exists, then $P = [H' C^{-1} H]^+, K = P H' C^{-1}$ Else, let $T = I - H H^+, K = H^+ - H^+ C [T C T]^+, P = K C K'$ End if $\hat{x} = K y$

Table 2: Unified BLUE and optimal (generalized) WLS fusion rules.

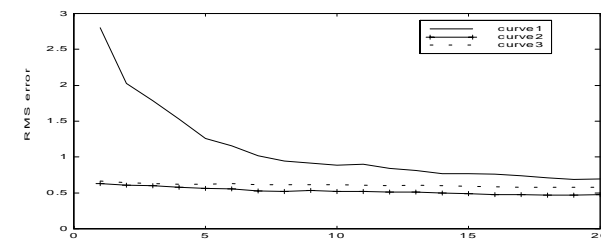
Fuser	BLUE	Optimal WLS	Optimal Generalized WLS
Prior	$\bar{x} = E[x], P_0 = \text{cov}(x)$	None	$\bar{x} = E[x], P_0 = \text{cov}(x)$
Data Model	n -sensor system: $y_i = H_i x + v_i, i = 1, \dots, n, v^i = [v'_1, \dots, v'_i]'$ $\tilde{A}_i = \text{cov}(x, v_i), \tilde{A} = \text{cov}(x, v^n) = [\tilde{A}_1, \dots, \tilde{A}_n], \tilde{C} = \text{cov}(v^n), \tilde{C}_{ij} = \text{cov}(v_i, v_j)$		
	$H = [H'_1, \dots, H'_n]'$		$H = [(I, H'_1), H'_2, \dots, H'_n]'$
	$A_i = \tilde{A}_i, A = \tilde{A}$		$A_i = 0, A = 0$
Correlation	$C = \tilde{C}$ $C_i = \tilde{C}_{ii}$ $C_{ij} = \tilde{C}_{ij}$		$C = \begin{bmatrix} P_0 & -A \\ -\tilde{A}' & \tilde{C} \end{bmatrix}$ $C_1 = \begin{bmatrix} P_0 & -\tilde{A}_1 \\ -\tilde{A}'_1 & \tilde{C}_{11} \end{bmatrix}$ $C_{1i} = [-\tilde{A}'_1, \tilde{C}'_{1i}]', i \neq 1$ $C_{ij} = \tilde{C}_{ij}, C_i = C_{ii}, i, j \neq 1$
Data	$y = [y'_1, \dots, y'_n]'$		$y = [(\bar{x}', y'_1)', y'_2, \dots, y'_n]'$
Fusion Formulas	$S = H P_0 H' + C + H A + (H A)'$ If S^{-1} exists, then $K = (P_0 H' + A) S^{-1}$ Else, $K = (P_0 H' + A) S^+$ End if $\hat{x} = \bar{x} + K[y - H \bar{x}]$ $P = P_0 - K S K'$	If C^{-1} exists, then $P = [H' C^{-1} H]^+, K = P H' C^{-1}$ Else, let $T = I - H H^+$ $K = H^+ - H^+ C [T R T]^+, P = K C K'$ End if $\hat{x} = K y$	
Relations	BLUE = Optimal Generalized WLS Optimal WLS = BLUE without Prior = BLUE with prior $\{\hat{x}_1, P_1\} = \{\hat{x}_1, P_1\}$ Optimal WLS		

Table 3: Computational complexities of various optimal fusion rules for Example 1.

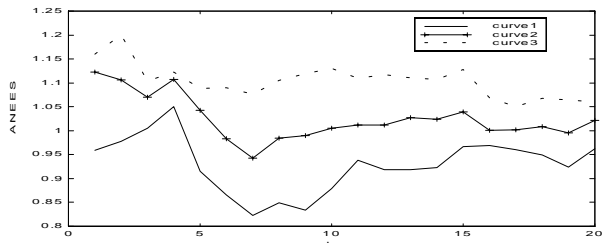
Architecture	Centralized			Standard Distributed			Non-Standard Distributed		
Method	BLUE	GWLS	WLS	BLUE	GWLS	WLS	BLUE	GWLS	WLS
CPU time ratio	1	0.426	0.051	1.971	3.339	3.193	0.999	0.446	0.421
FLOP ratio	1	0.705	0.138	1.931	3.018	2.827	1.016	0.821	0.747



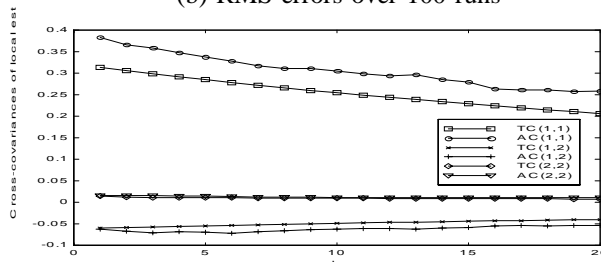
(a) RMS errors over a single run



(b) RMS errors over 100 runs

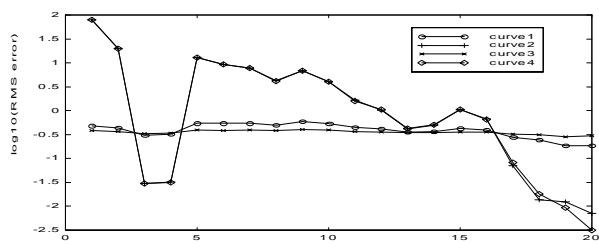


(c) ANEES over 100 runs

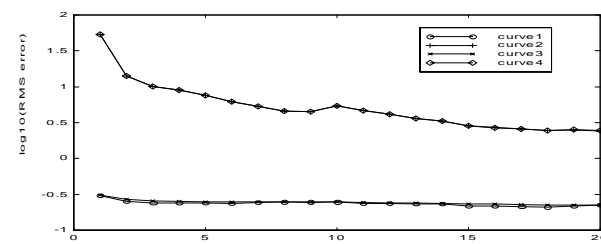


(d) Cross-covariances of local estimates' errors

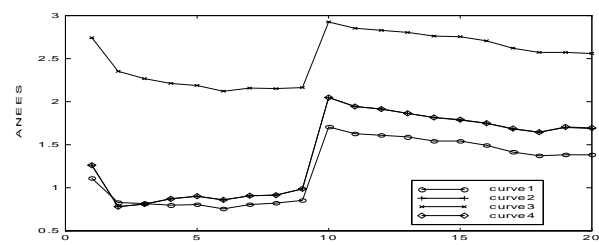
Fig. 1: Verification and optimality of fusion rules for Example 1.



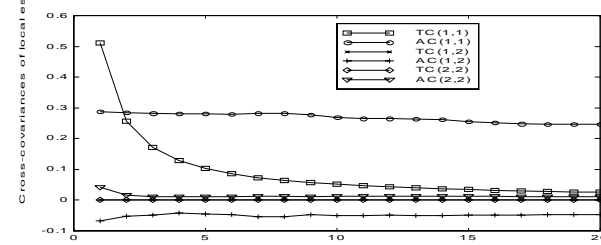
(a) RMS errors over a single run



(b) RMS errors over 100 runs

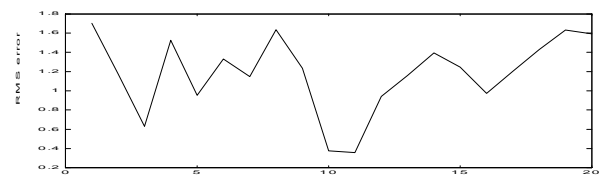


(c) ANEES over 100 runs

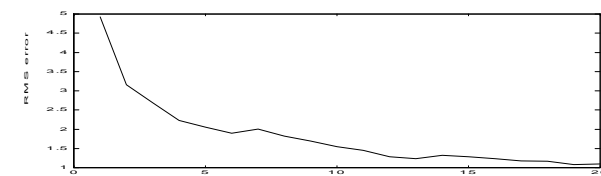


(d) Cross-covariances of local estimates' errors

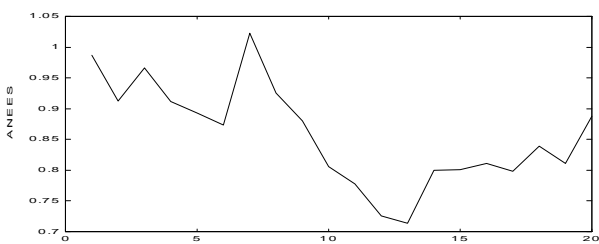
Fig. 2: Fusion with incomplete prior information (Example 2).



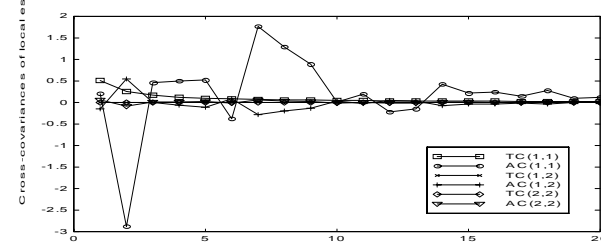
(a) RMS errors over a single run



(b) RMS errors over 100 runs



(c) ANEES over 100 runs



(d) Cross-covariances of local estimates' errors

Fig. 3: Fusion without prior information (Example 3).