Distributed Fusion of Multiple Model Estimators Using Minimum Forward Kullback–Leibler Divergence Sum

ZHENG WEI
ZHANSHENG DUAN ,
Xi'an Jiaotong University , Xi'an, China

UWE D. HANEBECK ,
Karlsruhe Institute of Technology , Karlsruhe, Germany

The problem of distributed fusion of Gaussian mixture models (GMMs) provided by the local multiple model (MM) estimators is addressed in this article. Taking GMMs instead of combined Gaussian assumed probability density functions (pdfs) as the output of local MM estimators can retain more detailed (or internal) information about local estimations, but the accompanying challenge is to perform the fusion of GMMs. For this problem, a distributed fusion framework of GMMs under the minimum forward Kullback–Leibler (KL) divergence sum criterion is proposed first. Then, because the KL divergence between GMMs is not analytically tractable, two suboptimal distributed fusion algorithms are further developed within this framework. These two fusion algorithms all have closed forms. Numerical examples verify their effectiveness in terms of both computational efficiency and estimation accuracy.

This work was supported by National Key R&D Program of China under Grant 2021YFC2202600 and Grant 2021YFC2202603.

Authors' addresses: Zheng Wei and Zhansheng Duan are with the Center for Information Engineering Science Research, Xi'an Jiaotong University, Xi'an 710049, China, E-mail: (weizheng179@stu.xjtu.edu.cn; zsduan@mail.xjtu.edu.cn); Uwe D. Hanebeck is with the Intelligent Sensor-Actuator-Systems Laboratory, Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany, E-mail: (uwe.hanebeck@kit.edu). *(Corresponding author: Zhansheng Duan.)*

I. INTRODUCTION

Maneuvering target tracking refers to sequentially predicting and estimating the motion states of the maneuvering target, which is an indispensable technology in modern information processing systems [1]. It has broad applications in military and civilian fields [2], [3], [4], [5], [6]. A key challenge in maneuvering target tracking is how to estimate the states of the target with uncertain motion models. For this problem, multimodel (MM) estimation is generally considered as one of the most natural and effective estimation approaches [7]. It decomposes a complex problem into multiple subproblems so as to solve the problem of target motion uncertainty [8], [9]. In theory, MM estimators have the potential to obtain globally optimal solutions [7]. Three MM estimators have been developed for maneuvering target tracking: autonomous MM (AMM) [10], interacting MM (IMM) [11], and variable-structure MM (VSMM) [12]. They all run a bank of estimators in parallel to update the target state estimates, each based on a particular model in the model set [10].

Generally, tracking a maneuvering target with a single MM estimator is vulnerable to various factors such as measurement accuracy, reliability, and detection range of sensors. It has been a hot and difficult subject to find an effective fusion approach for a bank of parallel running MM estimators [13]. There are two basic fusion architectures: centralized and distributed, depending on whether the raw measurements are used directly for fusion [14]. In the centralized fusion architecture, all measurements from multiple sensors are sent to the fusion center, and the design of the centralized fusion estimator is equivalent to that of the single-sensor state estimator but with augmented measurements from multiple sensors [15]. In the distributed fusion architecture, individual sensors process their own measurements first to obtain local estimates and then send these local estimates to the other sensors or fusion center to yield a fused state estimate under certain fusion criteria. Compared to centralized fusion, distributed fusion has the advantages of scalability and robustness [16], but is also more challenging. For the distributed fusion of MM estimators, there are two key challenges: 1) how to effectively use structural information of local estimators; 2) how to deal with the correlation among local estimation errors due to common process noise, priori information, etc.

First, taking the IMM estimator as an example, one cycle of the IMM estimator consists of five steps: *calculation of the mixing probabilities, mixing, model-matched filtering, model probability update, and estimate and covariance combination* [17]. After *a model probability update*, we can get a Gaussian mixture model (GMM). Then, in the last step, the GMM is approximated as a single Gaussian assumed probability density function (pdf) using the moment matching method. Nevertheless, on the one hand, only when the distance between the means of the components of the Gaussian mixture is not too far apart, the Gaussian pdf obtained by the moment matching method is close

to the GMM [18]. However, for GMMs representing the state of maneuvering targets, there is no guarantee that this condition is met. On the other hand, for the recursion of the IMM estimator, its last step, *estimate and covariance combination*, is only for output purpose [17]. Comparatively, however, GMM can keep more inner detailed information about the local estimator. Therefore, in order to best utilize the useful information contained in local estimates, a natural idea is that the GMMs obtained by the local estimators are directly used for fusion. Similar problems also exist in the AMM and VSMM estimators.

Second, in distributed fusion, local estimation errors across sensors may be correlated. For this problem, there are many results available [15]. A well-known distributed fusion algorithm is the covariance intersection (CI) [19], which can provide a consistent fused estimate irrespective of the unknown correlations. In [20], an extension of CI fusion for GMMs termed as pairwise component CI (PCCI) was presented, where the CI fusion was performed independently for each cross pair of GMMs. In addition, the generalized CI (GCI) fusion [21], [22] can also be used for the fusion of GMMs. The GCI fusion is sometimes referred to as Chernoff fusion [23], exponential mixture densities (EMD) fusion [24], weighted exponential product (WEP) fusion [25], and geometric mean density (GMD) fusion [26], etc. In [22], an information-theoretic justification for GCI was presented, and it was proposed to obtain the optimal solution of GCI using the Chernoff information minimization criterion. Under this criterion, the optimal fused pdf is the "halfway" between the two pdfs to be fused, where "halfway" is defined in terms of the Kullback–Leibler (KL) divergence. In [26], the GCI was further justified as a conservative fusion algorithm. There are other extensive studies on the approximation of GCI (see, e.g., [27], [28], [29], [30], [31], and [32]).

Although theoretically appealing, the implementation of the GCI, especially for the fusion of GMMs, is computationally extremely expensive. This is mainly caused by two challenges: 1) obtaining the optimal weight for each GMM; 2) computing the noninteger power (the weight) for each GMM. To tackle these two challenges, many suboptimal fusion algorithms were proposed. For example, in [33], a fast and consistent GCI fusion was proposed, which first computes the weights of the local GMMs using a numerical optimization strategy based on Monte Carlo importance sampling and then obtains the fused Gaussian mixture using weighted expectation maximization algorithm. In [34], another GCI approach using a sigma-point approximation was proposed. This approach approximates a noninteger power of a GMM as an unnormalized GMM based on a weighted least squares optimization, and the fused GMM is obtained using a sigma point approximation. In [35], the pseudo-Chernoff fusion (PCF) based on a first-order approximation to Chernoff information was proposed, which approximates the power of each Gaussian mixture as a mixture of the powers of its components to reduce computational complexity and obtains the corresponding weights by minimizing the determinant of the covariance matrix of the fused pdf. The modified pseudo-Chernoff fusion (MPCF) was proposed in [36]. Its main differences from the PCF are twofold: 1) a merging step is performed on close components of each GMM; 2) the weights of the components of the approximated GMM are modified. In [37], a heuristic approximation algorithm, determinant minimization CI for GMMs (DCI-GMM), was developed. The DCI-GMM approximates the power of GMM as a new GMM with a free parameter and obtains this parameter by minimizing the determinant of the covariance matrix of the fused pdf.

Different from the above algorithms, this article is devoted to the distributed fusion of GMMs using KL divergence. The KL divergence (also referred to as relative entropy, cross entropy, KL distance, information for discrimination, etc.) is defined as the expectation of the log-likelihood ratio of two distributions [38]. Since the KL divergence neither satisfies the triangle inequality nor the symmetry, it is not a true distance. But it is still useful to quantify the proximity of probability distributions [39]. For example, the KL divergence between GMMs, as a closeness measure between the current target and template region, has been widely used in speech and image recognition (see, e.g., [40], [41], [42], and [43]). In addition, the KL divergence has been widely applied in the fields of target tracking and information fusion.

In [44], a distributed fusion algorithm using forward KL divergence sum was proposed, which defines the fused pdf as the one that minimizes the sum of the forward KL divergences from local pdfs. If different weights are assigned to local pdfs, then the above is extended to a new fusion algorithm based on weighted forward KL divergence sum, which obtains the fused pdf by a joint optimization with respect to both the fusion weights and fused pdf [45]. The asymmetry of the KL divergence leads to the fact that minimizing the sum of the reverse KL divergence sum yields different fusion results. In [46], a linear fusion algorithm using weighted reverse KL divergence sum, arithmetic average (AA) fusion, was proposed. However, since there is no closed form for the KL divergence between GMMs, it is difficult to obtain the fused pdf by the above fusion algorithms when the local pdfs are all GMMs. More details about these KL divergence sum-based distributed fusion algorithms will be provided in Section III.

In this article, the distributed fusion of GMMs provided by local MM estimators is investigated for the multisensor maneuvering target tracking scenario. The main contributions are twofold.

1) A distributed fusion framework of GMMs under the minimum forward KL divergence sum criterion is proposed.
2) To make this framework analytically tractable, two suboptimal distributed fusion algorithms, all in closed forms are developed using an approximation to the KL divergence between GMMs and a heuristic method.

The rest of this article is organized as follows. Section II formulates the distributed fusion problem of GMMs provided by the MM estimators. Section III briefly reviews existing distributed fusion algorithms using minimum KL divergence sum criterion. Section IV presents a distributed fusion framework for GMMs of local MM estimators using the forward KL divergence sum. Then two closed-form distributed fusion algorithms are developed in Section V to make this framework analytically tractable. Illustrative examples are presented in Section VI. Finally, Section VII concludes this article.

Notation: For clarity, we use italics to denote scalar quantities and boldface for vectors and matrices. A lower or upper case Roman letter represents a name (e.g., "F" for "fusion center," "l" for "local estimator," etc.). We use $\mathbf{x}'$ to denote the transpose of the vector $\mathbf{x}$. $g(\mathbf{x})$ and $p(\mathbf{x})$ denote the GMM and Gaussian pdf of $\mathbf{x}$, respectively.

II. PROBLEM FORMULATION

Consider a distributed tracking system with N sensors, where each sensor runs a MM estimator for handling target maneuvers. For simplicity, assume the following jump Markov linear system [47] in the lth MM estimator:

$$\mathbf{x}_{i,k}^{l} = \mathbf{F}_{i}^{l}\mathbf{x}_{i,k-1}^{l} + \mathbf{w}_{i,k-1}^{l} \tag{1}$$

$$\mathbf{z}_{k}^{l} = \mathbf{H}^{l}\mathbf{x}_{i,k}^{l} + \mathbf{v}_{i,k}^{l} \tag{2}$$

where superscript $l = 1, 2, \ldots, N$ is the sensor index, subscript $i = 1, 2, \ldots, M$ is the index of models running in parallel in the MM estimator, $\mathbf{x}_{i,k}^{l} \in \mathbb{R}^{d_x}$ and $\mathbf{z}_{k}^{l} \in \mathbb{R}^{d_z}$ are the state and measurement vectors at time k, respectively, $\mathbf{F}_{i}^{l}$ and $\mathbf{H}^{l}$ are known parameter matrices of the ith model in estimator l, the process noise $\mathbf{w}_{i,k-1}^{l}$ and measurement noise $\mathbf{v}_{i,k}^{l}$ are assumed to be mutually independent zero-mean Gaussian white noise sequences.

In the local MM estimator l, the posterior state estimate of each model is updated according to the measurement set $\mathbf{Z}_{k}^{l} \triangleq \{\mathbf{z}_{j}^{l}\}_{j=1}^{k}$ in parallel. The intrinsic output of the MM estimator l is a GMM [48], namely

$$g^{l}(\mathbf{x}) = \sum_{i=1}^{M} \mu_{i}^{l} \mathcal{N}\left(\mathbf{x}; \hat{\mathbf{x}}_{i}^{l}, \mathbf{P}_{i}^{l}\right) \tag{3}$$

where the time indices k have been omitted for simplicity, μ_{i}^{l} denote the probability of the ith model in the estimator l, $\hat{\mathbf{x}}_{i}^{l}$ and $\mathbf{P}_{i}^{l}$ are the posterior model-conditioned state estimate and the corresponding variance, respectively.

For a single MM estimator l, in order to provide the user with a state estimate $\hat{\mathbf{x}}^{l}$ and covariance $\mathbf{P}^{l}$ of the maneuvering target, the following mean and covariance of GMM $g^{l}(\mathbf{x})$ can be used [17]:

$$\begin{cases} \hat{\mathbf{x}}^{l} = \sum_{i=1}^{M} \mu_{i}^{l} \hat{\mathbf{x}}_{i}^{l} \\ \mathbf{P}^{l} = \sum_{i=1}^{M} \left[\mathbf{P}_{i}^{l} + (\hat{\mathbf{x}}^{l} - \hat{\mathbf{x}}_{i}^{l})(\hat{\mathbf{x}}^{l} - \hat{\mathbf{x}}_{i}^{l})'\right] \mu_{i}^{l}. \end{cases} \tag{4}$$

Equation (4) is nothing but the last step in one cycle of the MM estimator. This step is only used to output the tracking results to the user, but not a necessary part to make the recursions of MM estimators work [17]. Comparing (3) with (4), it is obvious that (3) can keep more "detailed information," which includes the model probability of each Gaussian mixture component and the corresponding pdf.

Assume that the local MM estimators have the same model sets. Then the GMMs provided by these estimators have the following two characteristics.

1) These GMMs have the same number of components.
2) There is a one-to-one correspondence between the components of different GMMs. That is, components i from different GMMs all refer to the same motion model of the target.

We refer to the above characteristics as the "structure information" of GMMs provided by local MM estimators. In order to make full use of the detailed information and structure information mentioned above, the GMMs should be sent directly to the fusion center as the output of the local MM estimators. In this case, the fusion of GMMs needs to be performed at the fusion center.

Let $g^{F}(\mathbf{x})$ be the fused pdf. It can be naturally modeled as a GMM with the same structure as the local GMMs $g^{l}(\mathbf{x})$, $l = 1, 2, \ldots, N$, due to the fusion of local MM estimators of the same structure. That is, the fused pdf $g^{F}(\mathbf{x})$ is also of the form

$$g^{F}(\mathbf{x}) = \sum_{i=1}^{M} \mu_{i}^{F} \mathcal{N}\left(\mathbf{x}; \hat{\mathbf{x}}_{i}^{F}, \mathbf{P}_{i}^{F}\right) \tag{5}$$

where μ_{i}^{F} is the probability of the ith model at the fusion center, and $\mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}_{i}^{F}, \mathbf{P}_{i}^{F})$ is the corresponding Gaussian pdf.

In this article, our goal is to:

1) obtain the fused GMM $g^{F}(\mathbf{x})$ at the fusion center;
2) compute the fused state estimate $\hat{\mathbf{x}}^{F}$ and the corresponding covariance $\mathbf{P}^{F}$ with respect to the fused GMM $g^{F}(\mathbf{x})$ as the output of the fusion center.

III. REVIEW OF EXISTING DISTRIBUTED FUSION USING KL DIVERGENCE SUM

The KL divergence was first defined in [38]. It describes the information geometry of the density function space and can be used to measure the similarity between two pdfs. For two arbitrary pdfs $p^{m}(\mathbf{x})$ and $p^{n}(\mathbf{x})$, the KL divergence between them is defined as

$$D(p^{m} \| p^{n}) = \int p^{m}(\mathbf{x}) \cdot \log\left(\frac{p^{m}(\mathbf{x})}{p^{n}(\mathbf{x})}\right) d\mathbf{x} \tag{6a}$$

$$= E_{p^{m}(\mathbf{x})}\left[\log\left(\frac{p^{m}(\mathbf{x})}{p^{n}(\mathbf{x})}\right)\right] \tag{6b}$$

where $E_{p^{m}(\mathbf{x})}[\cdot]$ denotes the expectation with respect to $p^{m}(\mathbf{x})$.

According to (6b), the KL divergence can also be interpreted as the expectation (with respect to $p^{m}(\mathbf{x})$) of the log ratio of $p^{m}(\mathbf{x})$ and $p^{n}(\mathbf{x})$. When both $p^{m}(\mathbf{x})$ and $p^{n}(\mathbf{x})$ are

Gaussian pdfs, (6) has an analytic form

$$D(p^m\|p^n)=\frac{1}{2}\Big(\left(\hat{\mathbf{x}}^m-\hat{\mathbf{x}}^n\right)'\left(\mathbf{P}^n\right)^{-1}\left(\hat{\mathbf{x}}^m-\hat{\mathbf{x}}^n\right)-d_{\mathbf{x}}+\operatorname{tr}\left(\left(\mathbf{P}^n\right)^{-1}\mathbf{P}^m\right)+\ln\left(\frac{|\mathbf{P}^n|}{|\mathbf{P}^m|}\right)\Big) \tag{7}$$

where $\hat{\mathbf{x}}^m$ and $\hat{\mathbf{x}}^n$ are the means of the Gaussian pdfs $p^m(\mathbf{x})$ and $p^n(\mathbf{x})$, respectively, $\mathbf{P}^m$ and $\mathbf{P}^n$ are the corresponding covariances, $d_{\mathbf{x}}$ is the dimension of $\mathbf{x}$, and $\operatorname{tr}(\cdot)$ denotes the trace of a matrix.

The KL divergence is not symmetric, i.e., $D(p^m\|p^n)\neq D(p^n\|p^m)$. As a result of this, the distributed fusion using KL divergence sum can be divided into forward and reverse classes.

A. Distributed Fusion Using Forward KL Divergence Sum

Given N local pdfs p^l, $l=1,2,\ldots,N$, the distributed fusion using minimum forward KL divergence sum is defined as [44]

$$\hat{p}^F_{\mathrm{FKL}}(\mathbf{x})\triangleq\arg\min_p\sum_{l=1}^N D\left(p\|p^l\right) \tag{8}$$

where p is an arbitrary pdf to be optimized, p^l is the posterior pdf provided by local estimator l, and $\hat{p}^F_{\mathrm{FKL}}$ is the optimal fused pdf under this criterion.

When the local and fused pdfs are all Gaussian, i.e., $p^l=\mathcal{N}(\mathbf{x};\hat{\mathbf{x}}^l,\mathbf{P}^l)$ and $\hat{p}^F_{\mathrm{FKL}}=\mathcal{N}(\mathbf{x};\hat{\mathbf{x}}^F,\mathbf{P}^F)$, the fused estimate $\hat{\mathbf{x}}^F$ and covariance $\mathbf{P}^F$ can be obtained analytically [44]

$$\begin{cases}\hat{\mathbf{x}}^F=\left(\sum_{l=1}^N\left(\mathbf{P}^l\right)^{-1}\right)^{-1}\sum_{l=1}^N\left(\mathbf{P}^l_i\right)^{-1}\hat{\mathbf{x}}^l\\ \mathbf{P}^F=N\cdot\left(\sum_{l=1}^N\left(\mathbf{P}^l\right)^{-1}\right)^{-1}.\end{cases} \tag{9}$$

When the local pdfs are all Gaussian mixtures, there is no closed form for (8) because the KL divergence between Gaussian mixtures is not analytically tractable. In Section IV of this article, suboptimal fusion algorithms with closed forms will be proposed under this criterion.

Given N local pdfs p^l and the fusion weights ω^l, $l=1,2,\ldots,N$, the distributed fusion using minimum weighted forward KL divergence sum is defined as [49, Definition 1]

$$\hat{p}^F_{\mathrm{WFKL}}\triangleq\arg\min_p\sum_{l=1}^N\omega^l D\left(p\|p^l\right) \tag{10}$$

where $0\leq\omega^l\leq 1$ and $\sum_{l=1}^N\omega^l=1$. We refer to distributed fusion under this criterion as weighted forward KL divergence average (WF-KLDA).

In [49], it has been proven that (10) turns out to be

$$\hat{p}^F_{\mathrm{WFKL}}=\frac{\prod_{l=1}^N\left[p^l(\mathbf{x})\right]^{\omega^l}}{\int\prod_{l=1}^N\left[p^l(\mathbf{x})\right]^{\omega^l}d\mathbf{x}}. \tag{11}$$

Note that (11) is the same as the GCI fusion. So the WF-KLDA actually coincides with the GCI fusion.

When the local and fused pdfs are all Gaussian in (10), the fused estimate $\hat{\mathbf{x}}^F$ and covariance $\mathbf{P}^F$ can be analytically written as [49]

$$\begin{cases}\hat{\mathbf{x}}^F=\mathbf{P}^F\sum_{l=1}^N\omega^l\left(\mathbf{P}^l\right)^{-1}\hat{\mathbf{x}}^l\\ \mathbf{P}^F=\left(\sum_{l=1}^N\omega^l\left(\mathbf{P}^l\right)^{-1}\right)^{-1}.\end{cases} \tag{12}$$

Note that in distributed fusion, the fusion weights w^l, $l=1,2,\ldots,N$, are usually unknown. To determine the fusion weights and fused pdf simultaneously, the following max-min optimization problem was formulated

$$\left(\left\{\hat{\omega}^l\right\}_{l=1}^N,\ \hat{p}^F_{\mathrm{WFKL}}\right)\triangleq\arg\max_{\{\omega^l\}_{l=1}^N}\min_p\sum_{l=1}^N\omega^l D\left(p\|p^l\right). \tag{13}$$

A detailed derivation can be found in [45, Section II].

When the local pdfs are all Gaussian mixtures, it is difficult to obtain the optimal weights $\hat{\omega}^l$, $l=1,2,\ldots,N$, of each Gaussian mixture. Some suboptimal algorithms [33], [34], [35], [36], [37] have been proposed for this.

If the same confidence is assigned to all local pdfs, i.e., $\omega^1=\omega^2=\cdots=\omega^N=1/N$, then weighted forward KL divergence sum criterion is reduced to the unweighted forward one.

B. Distributed Fusion Using Reverse KL Divergence Sum

Given N local pdfs p^l and the fusion weights ω^l, $l=1,2,\ldots,N$, the distributed fusion using minimum weighted reverse KL divergence sum is defined as [50] and [51]

$$\hat{p}^F_{\mathrm{WRKL}}\triangleq\arg\min_p\sum_{l=1}^N\omega^l D\left(p^l\|p\right). \tag{14}$$

It has been proven that (14) leads to a linear fusion

$$\hat{p}^F_{\mathrm{WRKL}}=\sum_{l=1}^N\omega^l p^l. \tag{15}$$

Therefore, distributed fusion under this criterion is also referred to as AA fusion [51], [52].

When the local and fused pdfs are all Gaussian, the fused estimate $\hat{\mathbf{x}}^F$ and covariance $\mathbf{P}^F$ can be analytically written as [46]

$$\begin{cases}\hat{\mathbf{x}}^F=\sum_{l=1}^N\omega^l\hat{\mathbf{x}}^l\\ \mathbf{P}^F=\sum_{l=1}^N\left[\mathbf{P}^l+(\hat{\mathbf{x}}^F-\hat{\mathbf{x}}^l)(\hat{\mathbf{x}}^F-\hat{\mathbf{x}}^l)'\right]\omega^l.\end{cases} \tag{16}$$

However, in distributed fusion, the fusion weights w^l, $l=1,2,\ldots,N$, are usually unknown and hard to be determined in advance. Thus, optimal determination of the fusion weights and fused pdf becomes joint optimization problem with respect to both of them, which can be approximately reformulated as a max–min optimization problem [46]

$$\left(\left\{\hat{\omega}^l\right\}_{l=1}^N,\ \hat{p}^F_{\mathrm{WRKL}}\right)\triangleq\arg\max_{\{\omega^l\}_{l=1}^N}\min_p\sum_{l=1}^N\omega^l D\left(p^l\|p\right). \tag{17}$$

A detailed derivation can be found in [46, Sec. III].

When the local pdfs are all Gaussian mixtures, it is difficult to obtain the optimal weights in (17) because the KL

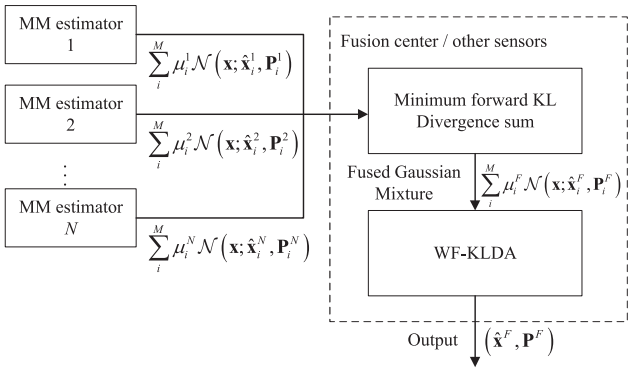

Fig. 1. Distributed fusion framework for multiple MM estimators.

divergence between Gaussian mixtures is not analytically tractable. Two suboptimal algorithms have been proposed in [46].

If $\omega^1 = \omega^2 = \cdots = \omega^N = 1/N$, then the weighted reverse KL divergence sum criterion is reduced to the unweighted reverse KL divergence sum criterion. Correspondingly, (15) boils down to unweighted AA (uAA) fusion. That is, when the local and fused pdfs are all Gaussian pdfs, the fused estimate $\hat{\mathbf{x}}^F$ and covariance $\mathbf{P}^F$ are

$$\begin{cases} \hat{\mathbf{x}}^F = \frac{1}{N}\sum_{l=1}^{N} \hat{\mathbf{x}}^l \\ \mathbf{P}^F = \frac{1}{N}\sum_{l=1}^{N} \left[\mathbf{P}^l + (\hat{\mathbf{x}}^F - \hat{\mathbf{x}}^l)(\hat{\mathbf{x}}^F - \hat{\mathbf{x}}^l)'\right]. \end{cases} \tag{18}$$

When the local pdfs are all Gaussian mixtures, the fused estimate and covariance can be obtained analytically as in [46].

REMARK 1 By comparison, it can be seen that in (9) and (12), the covariance of each local estimate is taken into account when computing the fused state estimate $\hat{\mathbf{x}}^F$. Whereas this is not the case in (16) and (18).

IV. DISTRIBUTED FUSION FRAMEWORK FOR GMMS OF LOCAL MM ESTIMATORS USING FORWARD KL DIVERGENCE

For the maneuvering target tracking with multiple MM estimators, a distributed fusion framework is proposed, as shown in Fig. 1.

This framework includes the following steps.

1) The GMMs $g^l(\mathbf{x}) = \sum_{i=1}^{M} \mu_i^l \mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}_i^l, \mathbf{P}_i^l)$, $l = 1, 2, \ldots, N$, provided by the local MM estimators are sent as input data to the fusion center or other sensors.
2) The fused GMM $g^F(\mathbf{x}) = \sum_{i=1}^{M} \mu_i^F \mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}_i^F, \mathbf{P}_i^F)$ is obtained under the minimum forward KL divergence sum criterion.
3) The state estimate $\hat{\mathbf{x}}^F$ and covariance $\mathbf{P}^F$ of the maneuvering target are computed by WF-KLDA, and output from the fusion center.

In the framework shown in Fig. 1, we fuse local GMMs under the minimum forward KL divergence sum criterion, that is

$$\hat{g}^F(\mathbf{x}) = \arg\min_{g^F} \sum_{l=1}^{N} D\left(g^F \| g^l\right) \tag{19}$$

where $\hat{g}^F(\mathbf{x})$ is the fused GMMs under this criterion, g^F is an arbitrary GMM, and g^l is the local GMM from estimator l as in (3).

Since there is no closed-form solution to the KL divergence between GMMs, how to analytically obtain the solution of (19) is not a trivial problem. This problem is the key to this article, which will be discussed in detail in the next section.

Assume that the solution to (19) has been obtained. Then the next step is how to compute the fused state estimate $\hat{\mathbf{x}}^F$ and covariance $\mathbf{P}^F$ of the maneuvering target from the fused GMM $\hat{g}^F(\mathbf{x})$. One method is to use the moment matching, just like in the last step in one cycle of the MM estimator. That is

$$\begin{cases} \hat{\mathbf{x}}^F = \sum_{i=1}^{M} \mu_i^F \hat{\mathbf{x}}_i^F \\ \mathbf{P}^F = \sum_{i=1}^{M} \left[\mathbf{P}_i^F + \left(\hat{\mathbf{x}}^F - \hat{\mathbf{x}}_i^F\right)\left(\hat{\mathbf{x}}^F - \hat{\mathbf{x}}_i^F\right)'\right] \mu_i^F. \end{cases} \tag{20}$$

The same results can be obtained by weighted reverse KL divergence sum criterion [46].

Another alternative method is the WF-KLDA [18], [53]. That is

$$\begin{cases} \hat{\mathbf{x}}^F = \mathbf{P}^F \sum_{i=1}^{M} \mu_i^F \left(\mathbf{P}_i^F\right)^{-1} \hat{\mathbf{x}}_i^F \\ \mathbf{P}^F = \left(\sum_{i=1}^{M} \mu_i^F \left(\mathbf{P}_i^F\right)^{-1}\right)^{-1}. \end{cases} \tag{21}$$

In (21), the covariance of each model is considered when computing the fused state estimate $\hat{\mathbf{x}}^F$, while it is not in (20). Therefore, the WF-KLDA is used to get the final output.

V. TWO CLOSED-FORM DISTRIBUTED FUSION ALGORITHMS

In this section, in order to obtain the solution to the optimization problem (19) analytically, two closed-form distributed fusion algorithms for GMMs are developed using an approximation of the KL divergence between GMMs, *matched bound approximation* [54], and a heuristic method.

A. Distributed Fusion of GMMs Using Matched Bound Approximation

There exists the following matched bound approximation [54] for the KL divergence between two known GMMs

$$D\left(g^a \| g^b\right) \approx \sum_{i=1}^{M} \int \mu_i^a p_i^a \log g^a - \sum_{i=1}^{M} \int \mu_i^a p_i^a \log g^b \tag{22a}$$

$$\approx \sum_{i=1}^{M} \mu_i^a \int p_i^a \log \mu_i^a p_i^a - \sum_{i=1}^{M} \mu_i^a \max_j \int p_i^a \log \mu_j^b p_j^b \tag{22b}$$

$$= \sum_{i=1}^{M} \mu_i^a \min_j \left(D\left(p_i^a \| p_j^b\right) + \log \frac{\mu_i^a}{\mu_j^b}\right) \tag{22c}$$

$$= \sum_{i=1}^{M} \mu_i^a \left(D(p_i^a \| p_{\delta(i)}^b) + \log \frac{\mu_i^a}{\mu_{\delta(i)}^b} \right) \tag{22d}$$

with

$$\delta(i) = \arg\min_j \left(D\left(p_i^a \| p_j^b\right) + \log \frac{\mu_i^a}{\mu_j^b} \right) \tag{23}$$

where δ is a matching function that matches a single component of the Gaussian mixture g^b to a given component of g^a [54]. The approximation in (22b) is based on the assumption that the component $\mu_j^b p_j^b$ of the Gaussian mixture g^b that is most proximal to $\mu_i^a p_i^a$ dominates the integral $\int \mu_i^a p_i^a \log g^b$ [54].

If the GMMs g^a and g^b are provided by IMM estimators with the same model set, then according to the structural information of the GMMs described in Section II, the same components of g^a and g^b have the following characteristics.

1) Both p_i^a and p_i^b are obtained from model-matched filters with the same motion model. They represent the pdfs of target moving with the ith model.
2) Both μ_i^a and μ_i^b denote the probabilities that the target is moving with model i.

Therefore, the components of the GMMs have clear physical implications. Ideally, if local sensors have no measurement errors, or their errors are equal at any time, then $p_i^a = p_i^b$ and $\mu_i^a = \mu_i^b$. Usually, when these two estimators provide reliable local estimates, it is reasonable to assume that the model probabilities and model-matched filtering pdfs of the same components from different GMMs are closer to each other.

Based on the above analysis, we propose the following assumption.

ASSUMPTION 1 If the Gaussian mixtures g^a and g^b are provided by local MM estimators with the same model set, then the i-th component of g^b is always the matching model of the ith component of g^a, i.e., $j^\star = i$.

Assumption 1 is nothing but an approximate solution for the matching function (23). Its rationality will be verified through numerical examples in Section VI. Note that Assumption 1 can also be equivalently written as

$$\left(D\left(p_i^a \| p_j^b\right) + \log \frac{\mu_i^a}{\mu_j^b} \right) \Big|_{j=i} \leq \left(D\left(p_i^a \| p_j^b\right) + \log \frac{\mu_i^a}{\mu_j^b} \right) \Big|_{j \neq i}. \tag{24}$$

With Assumption 1, (22) becomes

$$\begin{aligned} D\left(g^a \| g^b\right) &\approx \sum_{i=1}^{M} \mu_i^a \min_j \left(D\left(p_i^a \| p_j^b\right) + \log \frac{\mu_i^a}{\mu_j^b} \right) \\ &\approx \sum_{i=1}^{M} \mu_i^a \left[D(p_i^a \| p_i^b) + \log \frac{\mu_i^a}{\mu_i^b} \right]. \end{aligned} \tag{25}$$

It can be seen from (25) that with Assumption 1, the matched bound approximation is equivalent to the upper bound approximation given by Do [55]. This also justifies the rationality of Assumption 1 to a certain extent.

Back to (19), according to the matched bound approximation, one has

$$\begin{aligned} &\min_{g^F} \sum_{l=1}^{N} D\left(g^F \| g^l\right) \\ &\approx \min_{\{\mu_i^F\}_{i=1}^M, \{p_i^F\}_{i=1}^M} \sum_{l=1}^{N} \sum_{i=1}^{M} \mu_i^F \min_j \left(D\left(p_i^F \| p_j^l\right) + \log \frac{\mu_i^F}{\mu_j^l} \right). \end{aligned} \tag{26}$$

Although μ_i^F and $p_i^F, i = 1, 2, \ldots, M$, in (26) are unknown, the fused g^F is modeled as a GMM with the same structure as the local GMM $g^l, l = 1, 2, \ldots, N$. This means that there is a one-to-one correspondence between the components of the fused g^F and each local GMM g^l. Therefore, we use the same approximation in (25) to simplify (26). Then

$$\begin{aligned} &\min_{g^F} \sum_{l=1}^{N} D\left(g^F \| g^l\right) \\ &\approx \min_{\{\mu_i^F\}_{i=1}^M, \{p_i^F\}_{i=1}^M} \sum_{l=1}^{N} \sum_{i=1}^{M} \mu_i^F \min_j \left(D\left(p_i^F \| p_j^l\right) + \log \frac{\mu_i^F}{\mu_j^l} \right) \\ &\approx \min_{\{\mu_i^F\}_{i=1}^M, \{p_i^F\}_{i=1}^M} \sum_{l=1}^{N} \sum_{i=1}^{M} \mu_i^F \left(D\left(p_i^F \| p_i^l\right) + \log \frac{\mu_i^F}{\mu_i^l} \right) \\ &= \min_{\{\mu_i^F\}_{i=1}^M, \{p_i^F\}_{i=1}^M} \left[\sum_{l=1}^{N} \sum_{i=1}^{M} \left(\mu_i^F \log \frac{\mu_i^F}{\mu_i^l} \right) \right. \\ &\quad \left. + \sum_{i=1}^{M} \mu_i^F \sum_{l=1}^{N} D\left(p_i^F \| p_i^l\right) \right] \end{aligned} \tag{27}$$

where $\sum_{i=1}^{M} \mu_i^F = 1$.

As can be seen from (27), p_i^F only exists in $\sum_{l=1}^{N} D(p_i^F \| p_i^l)$. Therefore, we can first compute the optimal $\hat{p}_i^F, i = 1, 2, \ldots, M$, by

$$\hat{p}_i^F = \arg\min_{p_i^F} \sum_{l=1}^{N} D\left(p_i^F \| p_i^l\right). \tag{28}$$

Since both $\hat{p}_i^F$ and p_i^l are known Gaussian as

$$\hat{p}_i^F = \mathcal{N}\left(\mathbf{x}; \hat{\mathbf{x}}_i^F, \mathbf{P}_i^F\right), \; p_i^l = \mathcal{N}\left(\mathbf{x}; \hat{\mathbf{x}}_i^l, \mathbf{P}_i^l\right). \tag{29}$$

Then the mean and covariance of Gaussian pdf $\hat{p}_i^F$ can be computed by [44]

$$\begin{cases} \hat{\mathbf{x}}_i^F = \left(\sum_{l=1}^{N} \left(\mathbf{P}_i^l\right)^{-1} \right)^{-1} \sum_{l=1}^{N} \left(\mathbf{P}_i^l\right)^{-1} \hat{\mathbf{x}}_i^l \\ \mathbf{P}_i^F = N \cdot \left(\sum_{l=1}^{N} \left(\mathbf{P}_i^l\right)^{-1} \right)^{-1}. \end{cases} \tag{30}$$

REMARK 2 Under the minimum forward KL divergence sum criterion, the pdf of component i of the fused GMMs g^F is obtained by fusing the same components of local GMMs.

Let

$$r_i = \sum_{l=1}^{N} D\left(\hat{p}_i^F \| p_i^l\right). \tag{31}$$

Algorithm 1: MBA-KL.

Require: GMMs $g^l,\ l = \{1, 2, \ldots, N\}$, of local MM estimators

Ensure: State estimate $\hat{\mathbf{x}}^F$ and corresponding covariance $\mathbf{P}^F$ of maneuvering target

1: Compute mean $\hat{\mathbf{x}}_i^F$ and covariance $\mathbf{P}_i^F$ of Gaussian pdfs $\hat{p}_i^F,\ i = 1, 2, \ldots, M$, via (30)
2: Compute corresponding model probabilities $\hat{\mu}_i^F$, $i = 1, 2, \ldots, M$, via (34)
3: Obtain the fused state estimate $\hat{\mathbf{x}}^F$ and covariance $\mathbf{P}^F$ by (21)
4: **return** $\hat{\mathbf{x}}^F, \mathbf{P}^F$

By substituting r_i into (27) and considering the constraint $\sum_{i=1}^M \mu_i^F = 1$, we can obtain

$$\begin{aligned}&\left(\hat{\mu}_1^F, \hat{\mu}_2^F, \ldots, \hat{\mu}_M^F\right)\\ &= \arg \min_{\{\mu_i^F\}_{i=1}^M} \left[\sum_{l=1}^N \sum_{i=1}^M \left(\mu_i^F \log \frac{\mu_i^F}{\mu_i^l}\right) + \sum_{i=1}^M \mu_i^F r_i\right] \quad (32a)\\ &\text{subject to } \sum_{i=1}^M \mu_i^F = 1. \quad (32b)\end{aligned}$$

The optimization problem in (32) can be solved using the Lagrange multiplier method. Let

$$\begin{aligned}&f\left(\mu_1^F, \mu_2^F, \cdots \mu_M^F\right)\\ &= \sum_{l=1}^N \sum_{i=1}^M \left(\mu_i^F \log \frac{\mu_i^F}{\mu_i^l}\right) + \sum_{i=1}^M \mu_i^F r_i + \lambda \left(1 - \sum_{i=1}^M \mu_i^F\right)\end{aligned} \quad (33)$$

where λ is the Lagrange multiplier.

To minimize (33), let $\frac{\partial f}{\partial \lambda} = 0$ and $\frac{\partial f}{\partial \mu_i^F} = 0,\ i = 1, 2, \ldots, M$. Then we can get the optimal model probability

$$\hat{\mu}_i^F = e^{\frac{1}{N}\left[-r_i - N + \log\left(\prod_{l=1}^N \mu_i^l\right) - \lambda\right]},\quad i = 1, 2, \ldots, M \quad (34)$$

where

$$\lambda = \log \sum_{i=1}^M e^{\frac{1}{N}\left[-r_i - N + \log\left(\prod_{l=1}^N \mu_i^l\right)\right]}. \quad (35)$$

Therefore, the fused GMM can be obtained by (30) and (34). Then the WF-KLDA is used to get the final output. This algorithm is referred to as MBA-KL. It is summarized in Algorithm 1.

B. Distributed Fusion of GMMs Using Discrete Approximation

It can be seen from (28) that the pdf of each model of the fusion center is obtained by fusing the pdfs of the same model across estimators, and the influence of model probability is not considered in this process. In each local IMM estimator, the computation of the model probabilities and model-matched filtering pdfs are interdependent. That is, when computing the model probabilities, the information from the pdfs has already been taken into account indirectly, and vice versa. Therefore, a heuristic method for the fusion of GMMs is to separately compute the model probability μ_i^F and pdf p_i^F under the minimum forward KL divergence sum criterion. Its advantage is that the complicated joint optimization problem in (19) with respect to both model probabilities and pdfs can be approximately decomposed into suboptimization problems about model probabilities and pdfs alone. In this way, closed form of (19) is readily available.

For this, let $\boldsymbol{\mu}^F = [\mu_1^F, \ldots, \mu_M^F]'$ and $\boldsymbol{\mu}^l = [\mu_1^l, \ldots, \mu_M^l]',\ l = 1, 2, \ldots, N$. Then the optimization problem in (19) can be approximately decomposed into the following suboptimization problems

$$\begin{cases}\hat{\boldsymbol{\mu}}^F = \arg \min_{\boldsymbol{\mu}^F} \sum_{l=1}^N D\left(\boldsymbol{\mu}^F \| \boldsymbol{\mu}^l\right)\\ \text{subject to } \sum_{i=1}^M \mu_i^F = 1,\ \text{and } \mu_i^F \geq 0\end{cases} \quad (36)$$

and

$$\hat{p}_i^F = \arg \min_{p_i^F} \sum_{l=1}^N D\left(p_i^F \| p_i^l\right),\quad i = 1, 2, \ldots, M. \quad (37)$$

The optimization problem in (36) can be solved using the Lagrange multiplier method. Let

$$\begin{aligned}&f\left(\mu_1^F, \mu_2^F, \cdots \mu_M^F\right)\\ &= \sum_{l=1}^N D\left(\boldsymbol{\mu}^F \| \boldsymbol{\mu}^l\right) + \eta \cdot \left(1 - \sum_{i=1}^M \mu_i^F\right)\\ &= \sum_{l=1}^N \sum_{i=1}^M \left(\mu_i^F \log \frac{\mu_i^F}{\mu_i^l}\right) + \eta \cdot \left(1 - \sum_{i=1}^M \mu_i^F\right)\end{aligned} \quad (38)$$

where η is the Lagrange multiplier. The minimization of (38) is a convex optimization problem. Let $\frac{\partial f}{\partial \eta} = 0$ and $\frac{\partial f}{\partial \mu_i^F} = 0, i = 1, 2, \ldots, M$. Then one can get

$$\hat{\mu}_i^F = e^{\frac{1}{N}\left[-N + \log\left(\prod_{l=1}^N \mu_i^l\right) - \eta\right]},\quad i = 1, 2, \ldots, M \quad (39)$$

where

$$\eta = \log \sum_{i=1}^M e^{\frac{1}{N}\left[-N + \log\left(\prod_{l=1}^N \mu_i^l\right)\right]}. \quad (40)$$

The solution of (37) is the same as that of (28). Interestingly, by comparing (34) and (39), it can be seen that the solutions of the model probabilities μ_i^F are very similar. The only difference is that there is no item r_i in the exponential part of (39).

After obtaining the fused GMM, the WF-KLDA is used to get the final output. This algorithm is referred to as DA-KL. It is summarized in Algorithm 2.

VI. ILLUSTRATIVE EXAMPLES

In this section, to evaluate the performance of the two newly developed distributed fusion algorithms for GMMs, some illustrative maneuvering target tracking examples are presented.

Consider a single target tracking problem. Assume that the target performs nearly constant velocity (NCV), nearly coordinated turn (NCT), and NCV motions sequentially in

Algorithm 2: DA-KL.

Require: GMMs g^l, $l = \{1, 2, \ldots, N\}$ of local MM estimators

Ensure: State estimate $\hat{\mathbf{x}}^F$ and corresponding covariance $\mathbf{P}^F$ of maneuvering target

1: Compute mean $\hat{\mathbf{x}}_i^F$ and covariance $\mathbf{P}_i^F$ of Gaussian pdfs $\hat{p}_i^F$, $i = 1, 2, \ldots, M$, via (30)
2: Compute corresponding model probabilities $\hat{\mu}_i^F$, $i = 1, 2, \ldots, M$, via (39)
3: Obtain the fused state estimate $\hat{\mathbf{x}}^F$ and covariance $\mathbf{P}^F$ by (21)
4: **return** $\hat{\mathbf{x}}^F$, $\mathbf{P}^F$

TABLE I
Parameter Settings for Maneuvering Target True Trajectory

Parameter	Value
Mean of initial target position	$[1000, 1000]'\mathrm{m}$
Mean of initial target velocity	$[70, 70]'\mathrm{m/s}$
Covariance of initial target position	$\mathrm{diag}(40^2, 40^2)\mathrm{m}^2$
Covariance of initial target velocity	$\mathrm{diag}(15^2, 15^2)\mathrm{m}^2/\mathrm{s}^2$
Angular rate of NCT motion	$6^\circ/\mathrm{s}$
Duration of the first NCV	$1 \le k \le 35$
Duration of NCT	$35 < k \le 65$
Duration of the second NCV	$65 < k \le 100$

TABLE II
Power Spectral Densities in Two Test Cases

	Case 1	Case 2
$\rho_1(\mathrm{m}^2/\mathrm{s}^3)$	0.1	10
$\rho_2(\mathrm{m}^2/\mathrm{s}^3)$	0.1	10

a 2-D plane, and only the target position can be observed by sensors. The parameters of the target true trajectory are listed in Table I.

The covariance matrices of the process noise of target performing NCV and NCT motions are given by

$$\mathbf{Q}_1 = \rho_1 \cdot \begin{bmatrix} \frac{T^3}{3} & \frac{T^2}{2} & 0 & 0 \\ \frac{T^2}{2} & T & 0 & 0 \\ 0 & 0 & \frac{T^3}{3} & \frac{T^2}{2} \\ 0 & 0 & \frac{T^2}{2} & T \end{bmatrix} \tag{41}$$

$$\mathbf{Q}_2 = \rho_2 \cdot \begin{bmatrix} \frac{2(\omega T - \sin \omega T)}{\omega^3} & 0 & \frac{1-\cos \omega T}{\omega^2} & \frac{\omega T - \sin \omega T}{\omega^2} \\ 0 & \frac{2(\omega T - \sin \omega T)}{\omega^3} & -\frac{\omega T - \sin \omega T}{\omega^2} & \frac{1-\cos \omega T}{\omega^2} \\ \frac{1-\cos \omega T}{\omega^2} & -\frac{\omega T - \sin \omega T}{\omega^2} & T & 0 \\ \frac{\omega T - \sin \omega T}{\omega^2} & \frac{1-\cos \omega T}{\omega^2} & 0 & T \end{bmatrix} \tag{42}$$

where ρ_1 is the power spectral density (PSD) of the process noise of NCV motion, ρ_2 is the PSD of process noise of NCT motion, T is the sampling interval of the sensor, and ω is the angular rate.

The PSDs of the process noise of the target motion for two test cases are given in Table II.

An IMM estimator is run at each sensor to handle target maneuvers. The target state to be estimated is defined as

$$\mathbf{x}_k \triangleq [\ x_k \quad \dot{x}_k \quad y_k \quad \dot{y}_k\ ]' \tag{43}$$

where $[x_k\ y_k]'$ and $[\dot{x}_k\ \dot{y}_k]'$ are the position and velocity of the target in the Cartesian coordinate system, respectively.

In each IMM estimator, the NCV and NCT models are used to track the maneuvering target. The parameter matrices in (1) are as follows:

$$\mathbf{F}_1^l = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{44}$$

$$\mathbf{F}_2^l = \begin{bmatrix} 1 & \frac{\sin \omega T}{\omega} & 0 & -\frac{1-\cos \omega T}{\omega} \\ 0 & \cos \omega T & 0 & -\sin \omega T \\ 0 & \frac{1-\cos \omega T}{\omega} & 1 & \frac{\sin \omega T}{\omega} \\ 0 & \sin \omega T & 0 & \cos \omega T \end{bmatrix} \tag{45}$$

where F_1^l is the state transition matrix of the NCV model in estimator l, F_2^l is the state transition matrix of the NCT model in estimator l. The corresponding used covariance matrices for process noises are the same as in (41) and (42) with the PSDs in Table II.

Since only the target position can be observed by both sensors, the measurement matrix $\mathbf{H}^l$ in (2) is

$$\mathbf{H}^l = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{46}$$

The transition probability matrix of the Markov chain between the NCV and NCT models is taken as

$$\mathbf{\Pi} = \begin{bmatrix} 0.95 & 0.05 \\ 0.05 & 0.95 \end{bmatrix}. \tag{47}$$

To verify the effectiveness of these two new distributed fusion algorithms developed in this article, their tracking performances in terms of both estimation accuracy and computational efficiency are compared with the PCCI [20], MPCF [36], DCI-GMM [37], AA [46], and uAA [46], [52], along with the local IMM estimators, over 2000 Monte Carlo runs.

For MPCF, the merging step is performed on Gaussian mixture components that satisfy the following inequality:

$$\left(\hat{\mathbf{x}}_i^l - \hat{\mathbf{x}}_j^l\right) \mathbf{P}_i^l \left(\hat{\mathbf{x}}_i^l - \hat{\mathbf{x}}_j^l\right) < \Gamma \tag{48}$$

where Γ is the merging threshold. If (48) does not hold, it means that these two components of the Gaussian mixture are separated. As in [36], we let $\Gamma = 4$. In order to verify the effect of the merging step on the fusion algorithm and whether the components corresponding to different models can be well separated, the MPCF without the merging step (i.e., $\Gamma = 0$) is used for comparison. For ease of distinction, we denote the corresponding algorithms as MPCF-4 and MPCF-0, respectively.

The AA fusion is given by [46]

$$\hat{p}_{\mathrm{AA}}^F = \sum_{l=1}^{N} \omega^l g^l = \sum_{l=1}^{N} \omega^l \sum_{i=1}^{M} \mu_i^l p_i^l. \tag{49}$$

TABLE III
Rationality Ratio γ Within $10 \le K \le 100$

	Case 1	Case 2
Minimum	0.9511	0.9525
Maximum	0.9917	0.9798
Average	0.9823	0.9673

In [46], the local GMMs g^l are first approximated as Gaussian pdfs and then the suboptimal weights $\hat{\omega}^l$ are obtained using bound optimization.

If $\omega^1 = \omega^2 = \cdots = \omega^N = 1/N$, then (49) boils down to the uAA fusion. Either AA or uAA results in a Gaussian mixture with $N \times M$ components via (49) for which the mean and covariance are given by [46, Lemma 3]

$$\begin{cases} \hat{\mathbf{x}}^F = \sum_{l=1}^{N} \sum_{i=1}^{M} \mu_i^l \hat{\mathbf{x}}_i^l \\ \mathbf{P}^F = \sum_{l=1}^{N} \sum_{i=1}^{M} \left[\mathbf{P}_i^l + \left(\hat{\mathbf{x}}^l - \hat{\mathbf{x}}_i^l\right)\left(\hat{\mathbf{x}}^l - \hat{\mathbf{x}}_i^l\right)' \right] \mu_i^l. \end{cases} \tag{50}$$

A. Fused Tracking Performance for Maneuvering Target With 2 Sensors

Suppose there are two sensors, and their sampling intervals are $T = 1$ s. The covariances of measurement errors of sensors 1 and 2 are, respectively

$$\mathbf{R}^1 = \begin{bmatrix} 50^2 & 0 \\ 0 & 50^2 \end{bmatrix} \mathrm{m}^2 \tag{51}$$

$$\mathbf{R}^2 = \begin{bmatrix} 30^2 & 0 \\ 0 & 30^2 \end{bmatrix} \mathrm{m}^2. \tag{52}$$

The rationality of Assumption 1 will be verified by the rationality ratio γ defined as

$$\gamma = \frac{\sum_{n=1}^{M_c} c_n}{M_c} \tag{53}$$

where n is the Monte Carlo run index and M_c is the number of Monte Carlo runs. In each Monte Carlo run, if

$$D\left(p_1^1 \| p_1^2\right) + \log \frac{\mu_1^1}{\mu_1^2} \le D\left(p_1^1 \| p_2^2\right) + \log \frac{\mu_1^1}{\mu_2^2} \tag{54}$$

and

$$D\left(p_2^1 \| p_2^2\right) + \log \frac{\mu_2^1}{\mu_2^2} \le D\left(p_2^1 \| p_1^2\right) + \log \frac{\mu_2^1}{\mu_1^2} \tag{55}$$

hold simultaneously, then $c_n = 1$; if only one of them holds, then $c_n = 0.5$; if neither holds, then $c_n = 0$.

Assumption 1 is rational if its rationality ratio is close to 1 (the closer, the more rational); it is not rational if its rationality ratio is close to 0 (the closer, the less rational).

Fig. 2 shows the rationality ratios of Assumption 1 in both cases. First, it can be seen that during the initial time, the rationality of Assumption 1 is relatively smaller, with $\gamma \approx 0.6$. However, with the update on the state estimates and model probabilities, the rationality of the assumption increase quickly. Second, rationality ratios of Assumption 1 decrease slightly when the target is maneuvering.

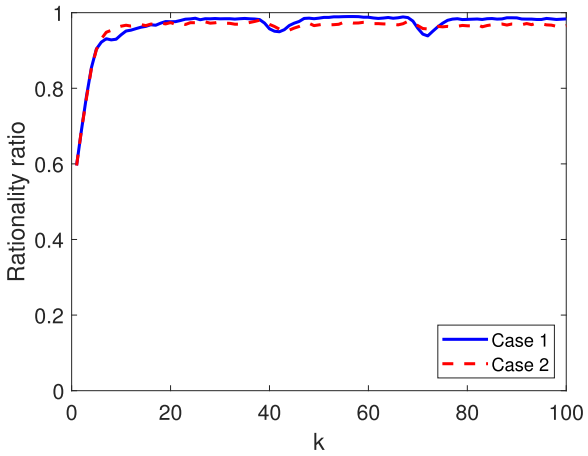

Fig. 2. Rationality ratios for Cases 1 and 2.

Table III lists the minimum, maximum, and average of the rationality ratios within $10 \le k \le 100$. It can be seen that although the PSDs of the process noises of the target motions in Case 2 are increased by 100 times compared to that in Case 1, the average rationality ratios are very close and both are larger than 0.95. Therefore, this assumption is considered rational.

Fig. 3(a) and (b) shows the root mean square (rms) position and velocity errors in Case 1. First, it can be seen that MBA-KL and DA-KL have nearly the same estimation accuracy and both of them are significantly better than the other fusion algorithms. This indicates that the proposed two fusion algorithms make full use of the structural information about the GMMs from the local IMM estimators, which not only simplifies the complexity of the fusion algorithms, but also effectively improves the estimation accuracy. Second, MPCF-4 and MPCF-0 almost overlap in both rms position and velocity errors. This means that in MPCF-4, the merging step is not performed since inequality (48) does not hold. That is, the Gaussian mixture components output by the individual IMM estimators are always separated from each other. This, to some extent, also verifies the rationality of Assumption 1. Third, the estimation accuracy of uAA is better than that of AA, which may be due to the fact that weights computed in AA fusions are suboptimal. In addition, DCI-GMM and MPCF-4/MPCF-0 are also worse than MBA-KL and DA-KL. This means that if the optimal weights are not available, assigning the same degree of confidence to all local pdfs may be a better choice.

Fig. 3(c) shows the average model probabilities of NCV motion from local IMM estimators as well as MBA-KL and DA-KL. The model probabilities of NCT can be easily known because their summations with those of NCV are 1. First, for the model probabilities of the fused GMMs, the only difference between the MBA-KL and DA-KL is that the MBA-KL additionally computes the forward KL divergence sum r_i of pdfs of model-matched filtering. It can be seen that the model probabilities for MBA-KL and DA-KL are almost the same. This shows that r_i has very little effect on the computation of model probabilities. The main

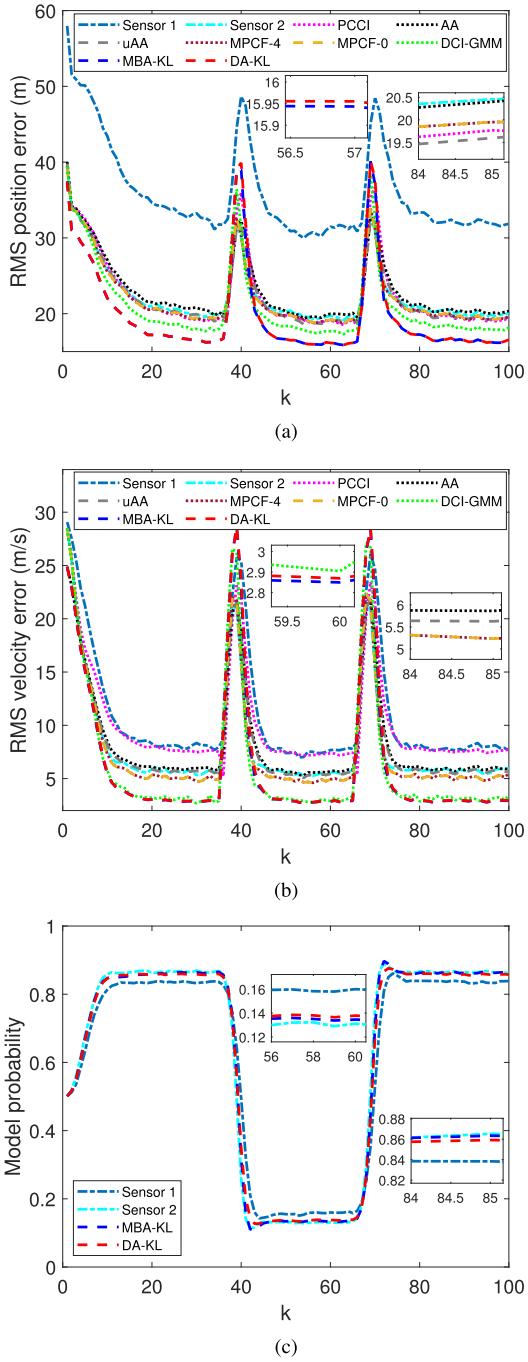

Fig. 3. Case 1 with $\rho_1 = 0.1$ and $\rho_2 = 0.1$. (a) Position rms error. (b) Velocity rms error. (c) Model probability of NCV.

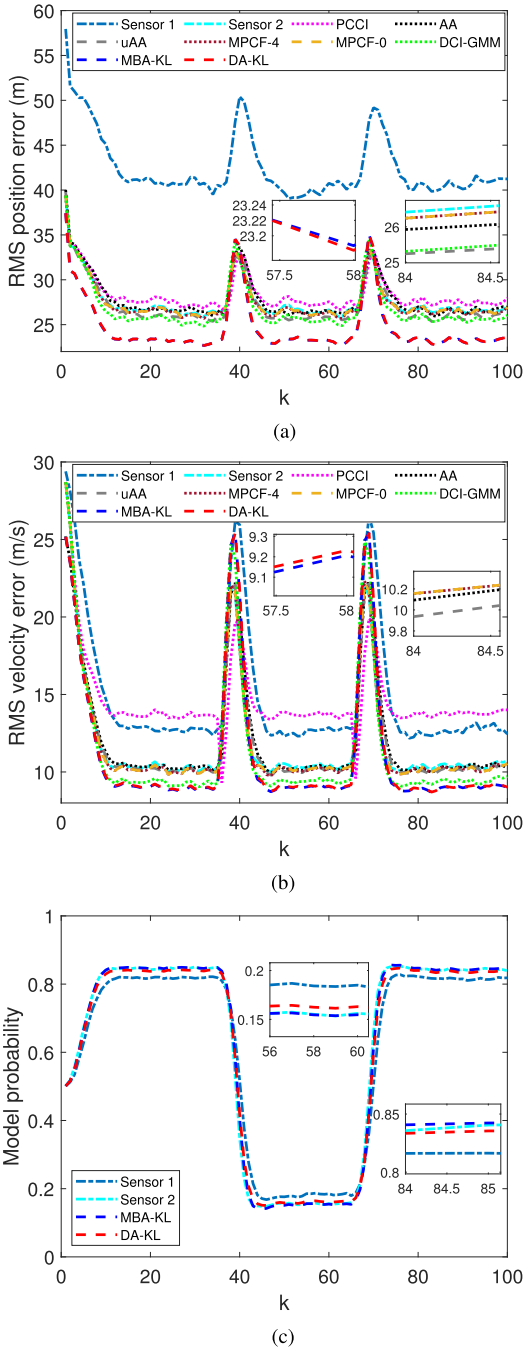

Fig. 4. Case 2 with $\rho_1 = 10$ and $\rho_2 = 10$. (a) Position rms error. (b) Velocity rms error. (c) Model probability of NCV.

reason is that in local IMM estimators, information of the pdfs have already been taken into account indirectly when calculating the model probabilities. Second, when the target performs NCV motion, the probabilities of the NCV model computed by both MAB-KL and DA-KL are significantly greater than 0.5 (about 0.8). This verifies that the model inference capability of the proposed two fusion algorithms is effective.

Fig. 4(a) and (b) shows the rms position and velocity errors and model probabilities of different algorithms in Case 2. Compared with Case 1, the PSDs of the process noises of the target motion are increased by 100 times. First, it can been seen that the estimation accuracies of all algorithms become worse. However, both MBA-KL and DA-KL still outperform the other fusion algorithms. Second, similar to the results in Case 1, the estimation accuracies of MPCF-4 and MPCF-0 are almost the same, and uAA is better than AA. Third, in this case, the estimation accuracy of DCI-GMM, uAA, AA, MPCF-4/MPCF-0, and PCCI are in descending order. Fig. 4(c) shows the average model probabilities of NCV motion from MBA-KL, DA-KL, and local IMM estimators. It can be seen that for the proposed MBA-KL and DA-KL, the model probabilities for various segments are consistent with the truth.

TABLE IV
Execution Time of Fusion Algorithms With Two Sensors

Algorithm	Execution time	Execution time relative to DA-KL
MBA-KL	14.9 ms	1.3545
DA-KL	11.0 ms	1
PCCI	852.0 ms	77.4545
AA	3106.4 ms	282.4000
uAA	3.3 ms	0.3000
MPCF-4	2466.6 ms	224.2364
MPCF-0	2463.1 ms	223.9182
DCI-GMM	2858.0 ms	259.8182

To evaluate the computational efficiency of all fusion algorithms, we compare their execution time relative to that of DA-KL in Case 1. The optimization toolbox of MATLAB (R2020b) is used for weights computation of the PCCI, PCF, MPCF, and DCI-GMM. Tested on a desktop computer with Intel Core i5-9500 CPU at 3.00 GHz and RAM of 8 GB, the average execution time of all fusion algorithms per Monte Carlo run is shown in Table IV.

It can be seen that the MBA-KL, DA-KL and uAA fusion algorithms with closed-form solutions are significantly more computationally efficient than the other fusion algorithms. Compared to DA-KL, MBA-KL requires additional computation of the forward KL divergence sum r_i of pdfs, $i = 1, 2, \ldots, M$, which leads to its slightly larger execution time.

B. Fused Tracking Performance for Maneuvering Target With Three Sensors

In this subsection, the effect of increasing the number of sensors on the estimation accuracy and computational efficiency of distributed fusion algorithms is analyzed.

Suppose there are three sensors, two of which are the same as in Sections VI–VI-A, and the covariance of measurement errors of the additional sensor is

$$\mathbf{R}^3 = \begin{bmatrix} 60^2 & 0 \\ 0 & 60^2 \end{bmatrix} \mathrm{m}^2. \tag{56}$$

The PSDs of the process noise of the target taking NCV and NCT motions are the same as in Case 1, i.e., $\rho_1 = 0.1\,\mathrm{m}^2/\mathrm{s}^3$, $\rho_2 = 0.1\,\mathrm{m}^2/\mathrm{s}^3$. We refer to this case as Case 3. Case 3 just has one more sensor than Case 1.

For Case 3, if none of

$$D\left(p_1^1 \| p_1^2\right) + \log\frac{\mu_1^1}{\mu_1^2} \le D\left(p_1^1 \| p_2^2\right) + \log\frac{\mu_1^1}{\mu_2^2} \tag{57}$$

$$D\left(p_2^1 \| p_2^2\right) + \log\frac{\mu_2^1}{\mu_2^2} \le D\left(p_2^1 \| p_1^2\right) + \log\frac{\mu_2^1}{\mu_1^2} \tag{58}$$

$$D\left(p_1^1 \| p_1^3\right) + \log\frac{\mu_1^1}{\mu_1^3} \le D\left(p_1^1 \| p_2^3\right) + \log\frac{\mu_1^1}{\mu_2^3} \tag{59}$$

$$D\left(p_2^1 \| p_2^3\right) + \log\frac{\mu_2^1}{\mu_2^3} \le D\left(p_2^1 \| p_1^3\right) + \log\frac{\mu_2^1}{\mu_1^3} \tag{60}$$

$$D\left(p_1^2 \| p_1^3\right) + \log\frac{\mu_1^2}{\mu_1^3} \le D\left(p_1^2 \| p_2^3\right) + \log\frac{\mu_1^2}{\mu_2^3} \tag{61}$$

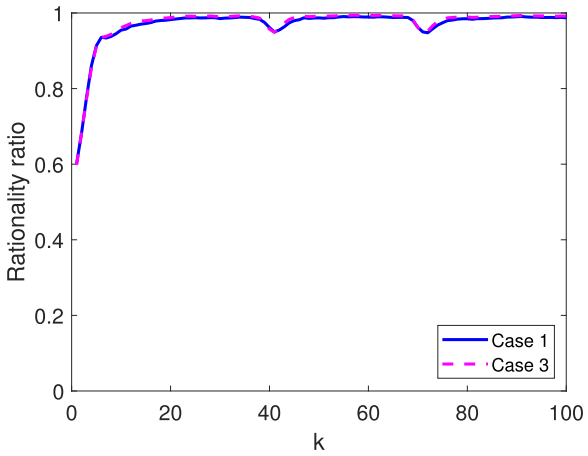

Fig. 5. Rationality ratios for Cases 1 and 3.

TABLE V
Execution Time of Fusion Algorithms With Three Sensors

Algorithm	Execution time	Execution time relative to DA-KL
MBA-KL	18.5 ms	1.2937
DA-KL	14.3 ms	1
PCCI	2049.4 ms	143.3147
AA	4119.0 ms	288.0420
uAA	4.4 ms	0.3077
MPCF-4	5770.5 ms	403.5315
MPCF-0	6122.6 ms	428.1538
DCI-GMM	6674.4 ms	466.7413

$$D\left(p_2^2 \| p_2^3\right) + \log\frac{\mu_2^2}{\mu_2^3} \le D\left(p_2^2 \| p_1^3\right) + \log\frac{\mu_2^2}{\mu_1^3} \tag{62}$$

holds, then $c_n = 0$. If only one of them holds, then $c_n = 1/6$. If only two of them hold then $c_n = 2/6$, and so on.

Fig. 5 shows the rationality ratios of Assumption 1 in both Cases 1 and 3. It can be seen that the rationality ratios of Assumption 1 in these two cases almost overlap. Therefore, when sensors provide reliable local estimates, the rationality of Assumption 1 does not change significantly as the number of sensors increases.

Fig. 6 shows the rms position and velocity errors and model probabilities of different fusion algorithms in Case 3. First, it can be seen from Fig. 6(a) and (b) that, compared to Case 1, when the number of sensors increases, the position estimation accuracies of all fusion algorithms are significantly improved. MBA-KL and DA-KL have nearly the same estimation accuracy and are significantly better than the other fusion algorithms. Note that in this case, the estimation accuracy of PCCI is better than AA and MPCF, which is different from the results in Cases 1 and 2. This shows the potential of PCCI in multisensor ($N > 2$) distributed fusion. Fig. 6(c) shows the model probabilities of the NCV in MBA-KL and DA-KL. It can be seen that both proposed algorithms can quickly detect target maneuvers and provide stable and reliable model probabilities.

Table V shows the average execution time of all fusion algorithms per Monte Carlo run in Case 3. Comparing with

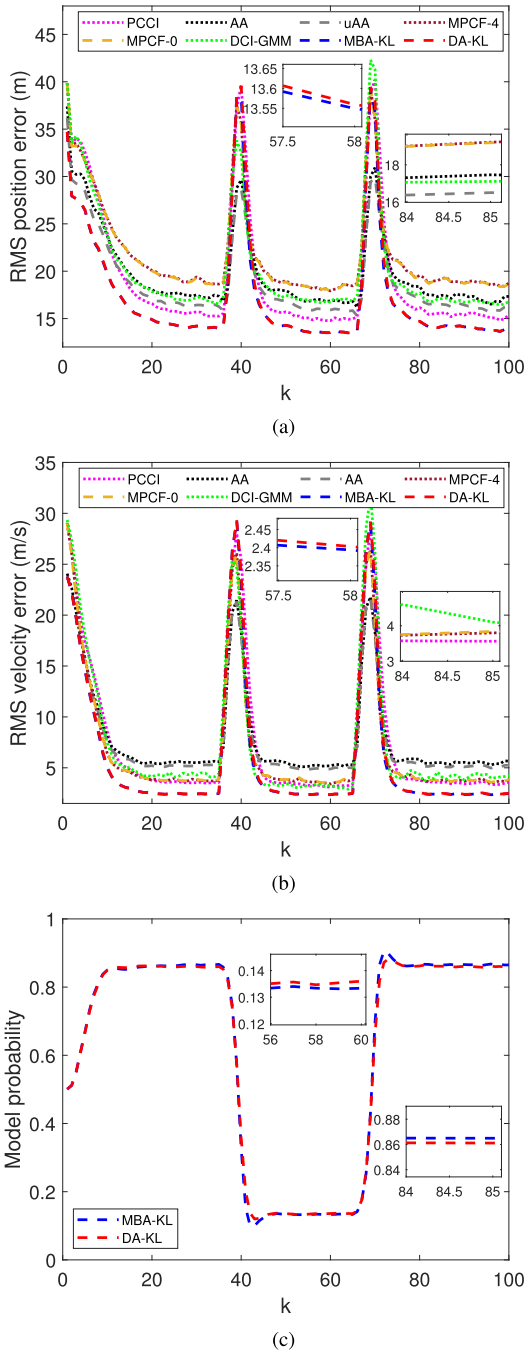

Fig. 6. Case 3 with $\rho_1 = 0.1$ and $\rho_2 = 0.1$. (a) Position rms error. (b) Velocity rms error. (c) Model probability of NCV.

Table IV, it can be seen that when the number of sensors increases, the execution time of PCCI, AA, MPCF-4, MPCF-0, and DCI-GMM increases significantly. For PCCI, it needs to perform CI fusion for all combinations from different GMM components. Therefore, as the number of sensors increases, the number of combinations increases exponentially, resulting in a significant increase in the computational cost of the PCCI. The AA, MPCF, and DCI-GMM fusion algorithms for GMMs are designed for cases with two local sensors. They are extended to the cases with $N > 2$ sensors by the sequential processing mechanism. This means that AA, MPCF, and DCI-GMM all need to perform $N - 1$ pairwise fusion sequentially, leading to a significant increase in their computational costs. For uAA, MBA-KL, and DA-KL, since they have closed forms, their computational cost is only slightly increased.

Overall, the estimation accuracies of both MBA-KL and DA-KL are significantly better than the other fusion algorithms in all three cases. And since MBA-KL and DA-KL have closed forms, their execution time does not increase significantly with the number of sensors. This means that MBA-KL and DA-KL have better scalability and can be used in real-time distributed fusion applications.

VII. CONCLUSION

For the problem of maneuvering target tracking with multiple MM estimators, we have proposed a distributed fusion framework under the minimum forward KL divergence sum criterion, in which the local GMMs rather than their means and covariances from the MM estimators are used for fusion to retain more detailed and structural information of local estimations. Within this framework, we have developed two suboptimal closed-form fusion algorithms, MBA-KL and DA-KL, using an approximation to the KL divergence of GMMs and a heuristic method. Numerical results have demonstrated that, compared with the existing PCCI, MPCF, AA, uAA, and DCI-GMM fusion algorithms, MBA-KL and DA-KL have significantly higher estimation accuracy for maneuvering target tracking.

REFERENCES

[1] X. Wang, Z. Xu, X. Gou, and L. Trajković, "Tracking a maneuvering target by multiple sensors using extended Kalman filter with nested probabilistic-numerical linguistic information," *IEEE Trans. Fuzzy Syst.*, vol. 28, no. 2, pp. 346–360, Feb. 2020.

[2] H. Zhang, X. Zhou, Z. Wang, and H. Yan, "Maneuvering target tracking with event-based mixture Kalman filter in mobile sensor networks," *IEEE Trans. Cybern.*, vol. 50, no. 10, pp. 4346–4357, Oct. 2020.

[3] S. Wang, D. Bi, H. Ruan, and M. Du, "Radar maneuvering target tracking algorithm based on human cognition mechanism," *Chin. J. Aeronaut.*, vol. 32, no. 7, pp. 1695–1704, Jul. 2019.

[4] S. Bu and G. Zhou, "Sequential spatiotemporal bias compensation and data fusion for maneuvering target tracking," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 59, no. 1, pp. 241–257, Feb. 2023.

[5] H. Yan, Y. Tian, H. Li, H. Zhang, and Z. Li, "Input–output finite-time mean square stabilization of nonlinear semi-Markovian jump systems," *Automatica*, vol. 104, pp. 82–89, Jun. 2019.

[6] W. Wu, H. Sun, Y. Cai, S. Jiang, and J. Xiong, "Tracking multiple maneuvering targets hidden in the DBZ based on the MM-GLMB filter," *IEEE Trans. Signal Process.*, vol. 68, pp. 2912–2924, 2020.

[7] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking. Part V. Multiple-model methods," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 4, pp. 1255–1321, Oct. 2005.

[8] H. Qu, L. Pang, and S. Li, "A novel interacting multiple model algorithm," *Signal Process.*, vol. 89, no. 11, pp. 2171–2177, Nov. 2009.

[9] B. Hou et al., "Novel interacting multiple model filter for uncertain target tracking systems based on weighted Kullback–Leibler divergence," *J. Franklin Inst.*, vol. 357, no. 17, pp. 13041–13084, Sep. 2020.

[10] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking. Part II: Motion models of ballistic and space targets," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 46, no. 1, pp. 96–119, Jan. 2010.

[11] H. A. P. Blom and Y. Bar-Shalom, "The interacting multiple model algorithm for systems with Markovian switching coefficients," *IEEE Trans. Autom. Control*, vol. 33, no. 8, pp. 780–783, Aug. 1988.

[12] X. R. Li and Y. Bar-Shalom, "Multiple-model estimation with variable structure," *IEEE Trans. Autom. Control*, vol. 41, no. 4, pp. 478–493, Apr. 1996.

[13] S. Wang, W. Ren, and J. Chen, "Fully distributed dynamic state estimation with uncertain process models," *IEEE Trans. Control Netw. Syst.*, vol. 5, no. 4, pp. 1841–1851, Dec. 2018.

[14] X. R. Li, Y. Zhu, J. Wang, and C. Han, "Optimal linear estimation fusion—Part I. unified fusion rules," *IEEE Trans. Inf. Theory*, vol. 49, no. 9, pp. 2192–2208, Sep. 2003.

[15] C. Y. Chong, "Forty years of distributed estimation: A review of noteworthy developments," in *Proc. Sensor Data Fusion: Trends, Solutions, Appl.*, Bonn, Germany, 2017, pp. 1–10.

[16] S. Wang and W. Ren, "On the convergence conditions of distributed dynamic state estimation using sensor networks: A unified framework," *IEEE Trans. Control Syst. Technol.*, vol. 26, no. 4, pp. 1300–1316, Jul. 2018.

[17] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation With Applications to Tracking and Navigation*. New York, NY, USA: Wiley, 2001.

[18] W. Li and Y. Jia, "An information theoretic approach to interacting multiple model estimation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 51, no. 3, pp. 1811–1825, Jul. 2015.

[19] S. J. Julier and J. Uhlmann, "A non-divergent estimation algorithm in the presence of unknown correlations," in *Proc. Amer. Control Conf.*, Albuquerque, NM, USA, 1997, pp. 2369–2373.

[20] B. Upcroft, L. L. Ong, S. Kumar, M. Ridley, and T. Bailey, "Rich probabilistic representations for bearing only decentralised data fusion," in *Proc. 7th Int. Conf. Inf. Fusion*, Philadelphia, PA, USA, 2005, pp. 1054–1061.

[21] R. P. S. Mahler, "Optimal/robust distributed data fusion: A unified approach," in *Proc. 9th Signal Process., Sensor Fusion, Target Recognit.*, Orlando, FL, USA, 2000, pp. 128–138.

[22] M. B. Hurley, "An information theoretic justification for covariance intersection and its generalization," in *Proc. 5th Int. Conf. Inf. Fusion.*, Annapolis, MD, USA, 2002, pp. 505–511.

[23] K. C. Chang, C. Y. Chong, and S. Mori, "Analytical and computational evaluation of scalable distributed fusion algorithms," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 46, no. 4, pp. 2022–2034, Oct. 2010.

[24] S. J. Julier, T. Bailey, and J. K. Uhlmann, "Using exponential mixture models for suboptimal distributed data fusion," in *Proc. IEEE Nonlinear Statist. Signal Process. Workshop*, Cambridge, U.K., 2006, pp. 160–164.

[25] D. Clarke, "Minimum information loss fusion in distributed sensor networks," in *Proc. 19th Int. Conf. Inf. Fusion*, Heidelberg, Germany, 2016, pp. 1057–1062.

[26] T. Bailey, S. J. Julier, and G. Agamennoni, "On conservative fusion of information with unknown non-Gaussian dependence," in *Proc. 15th Int. Conf. Inf. Fusion*, Singapore, 2012, pp. 1876–1883.

[27] F. Nielsen, "An information-geometric characterization of Chernoff information," *IEEE Signal Process. Lett.*, vol. 20, no. 3, pp. 269–272, Mar. 2013.

[28] G. Liu, M. Li, W. Yi, and L. Kong, "An approximate optimal Chernoff fusion method via importance sampling," in *Proc. Int. Conf. Control, Autom. Inf. Sci.*, Chiang Mai, Thailand, 2017, pp. 128–133.

[29] W. J. Farrell and C. Ganesh, "Generalized Chernoff fusion approximation for practical distributed data fusion," in *Proc. 12th Int. Conf. Inf. Fusion*, Seattle, WA, USA, 2009, pp. 555–562.

[30] O. Tslil, N. Lehrer, and A. Carmi, "Approaches to Chernoff fusion with applications to distributed estimation," *Digit. Signal Process.*, vol. 107, pp. 1–9, Oct. 2020.

[31] K. Lu, R. Zhou, and J. Zhang, "Approximate Chernoff fusion of Gaussian mixtures for ballistic target tracking in the re-entry phase," *Aerosp. Sci. Technol.*, vol. 61, pp. 21–28, Feb. 2017.

[32] N. Lehrer, O. Tslil, and A. Carmi, "Log-linear Chernoff fusion for distributed particle filtering," in *Proc. 22th Int. Conf. Inf. Fusion*, Ottawa, ON, Canada, 2019, pp. 1–9.

[33] N. R. Ahmed and M. Campbell, "Fast consistent Chernoff fusion of Gaussian mixtures for ad hoc sensor networks," *IEEE Trans. Signal Process.*, vol. 60, no. 12, pp. 6739–6745, Dec. 2012.

[34] M. Gunay, U. Orguner, and M. Demirekler, "Chernoff fusion of Gaussian mixtures based on sigma-point approximation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 52, no. 6, pp. 2732–2746, Dec. 2016.

[35] S. J. Julier, "An empirical study into the use of Chernoff information for robust, distributed fusion of gaussian mixture models," in *Proc. 9th Int. Conf. Inf. Fusion*, Florence, Italy, 2006, pp. 1–8.

[36] G. Battistelli, L. Chisci, C. Fantacci, A. Farina, and A. Graziano, "Consensus CPHD filter for distributed multitarget tracking," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 3, pp. 508–520, Mar. 2013.

[37] H. Zhu, K. Guo, and S. Chen, "Fusion of Gaussian mixture models for maneuvering target tracking in the presence of unknown cross-correlation," *Chin. J. Electron.*, vol. 25, pp. 270–276, Feb. 2016.

[38] S. Kullback and R. A. Leibler, "On information and sufficiency," *Ann. Math. Statist.*, vol. 22, no. 1, pp. 79–86, Mar. 1951.

[39] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York, NY, USA: Wiley, 2006.

[40] M. Jiang, S. Song, F. Tang, Y. Li, J. Liu, and X. Feng, "Scan registration for underwater mechanical scanning imaging sonar using symmetrical Kullback–Leibler divergence," *J. Electron. Imag.*, vol. 28, no. 1, pp. 1–11, Feb. 2019.

[41] S. Ji, Z. Zhang, S. Ying, L. Wang, X. Zhao, and Y. Gao, "Kullback-Leibler divergence metric learning," *IEEE Trans. Cybern.*, vol. 52, no. 4, pp. 2047–2058, Apr. 2022.

[42] R. R. Barreira and L. L. Ling, "Kullback–Leibler divergence and sample skewness for pathological voice quality assessment," *Biomed. Signal Process. Control*, vol. 57, Mar. 2020, Art. no. 101697.

[43] L. Moraru et al., "Gaussian mixture model for texture characterization with application to brain DTI images," *J. Adv. Res.*, vol. 16, pp. 15–23, Mar. 2019.

[44] Z. Duan, X. R. Li, and U. D. Hanebeck, "Multi-sensor distributed estimation fusion using minimum distance sum," in *Proc. 17th Int. Conf. Inf. Fusion*, Salamanca, Spain, 2014, pp. 1–8.

[45] M. Üney, J. Houssineau, E. Delande, S. J. Julier, and D. E. Clark, "Fusion of finite-set distributions: Pointwise consistency and global cardinality," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 55, no. 6, pp. 2759–2773, Dec. 2019.

[46] T. Li, Y. Song, E. Song, and H. Fan, "Arithmetic average density fusion—Part I: Some statistic and information-theoretic results," *Inf. Fusion*, vol. 104, Apr. 2024, Art. no. 102199.

[47] C. C. Lutz and D. J. Stilwell, "Stability and disturbance attenuation for Markov jump linear systems with time-varying transition probabilities," *IEEE Trans. Autom. Control*, vol. 61, no. 5, pp. 1413–1418, May 2016.

[48] M. Gunay, U. Orguner, and M. Demirekler, "Chernoff fusion of Gaussian mixtures for distributed maneuvering target tracking," in *Proc. 18th Int. Conf. Inf. Fusion*, Washington, DC, USA, 2015, pp. 870–877.

[49] G. Battistelli and L. Chisci, "Kullback–Leibler average, consensus on probability densities, and distributed state estimation with guaranteed stability," *Automatica*, vol. 50, no. 3, pp. 707–718, Feb. 2014.

[50] A. E. Abbas, "A Kullback-Leibler view of linear and log-linear pools," *Decis. Anal.*, vol. 6, no. 1, pp. 25–37, Feb. 2009.

[51] K. Da, T. Li, Y. Zhu, H. Fan, and Q. Fu, "Kullback–Leibler averaging for multitarget density fusion," in *Proc. 16th Int. Symp. Distrib. Comput. Artif. Intell.*, Avila, Spain, 2019, pp. 253–261.

[52] T. Li, H. Fan, J. García, and J. M. Corchado, "Second-order statistics analysis and comparison between arithmetic and geometric average fusion: Application to multi-sensor target tracking," *Inf. Fusion*, vol. 51, pp. 233–243, Nov. 2019.

[53] G. Wang, X. Wang, and Y. Zhang, "Variational Bayesian IMM-filter for JMSs with unknown noise covariances," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 56, no. 2, pp. 1652–1661, Apr. 2020.

[54] J. Goldberger, S. Gordon, and H. Greenspan, "An efficient image similarity measure based on approximations of KL-divergence between two Gaussian mixtures," in *Proc. 9th Int. Conf. Comput. Vis.*, Nice, France, 2003, pp. 487–493.

[55] M. N. Do, "Fast approximation of Kullback-Leibler distance for dependence trees and hidden Markov models," *IEEE Signal Process. Lett.*, vol. 10, no. 4, pp. 115–118, Apr. 2003.