Fault-tolerant Bayesian Decentralized Data Fusion Using Reliability Variables and Mixture Models*

Changkyo Shin
Department of Aerospace Engineering
Korea Advanced Institute of Science and Technology
Daejeon, Korea
sinchangkyo@kaist.ac.kr

Ofer Dagan
Smead Aerospace Engineering Sciences Department
University of Colorado Boulder
Boulder, USA
ofer.dagan@colorado.edu

Nisar Ahmed
Smead Aerospace Engineering Sciences Department
University of Colorado Boulder
Boulder, USA
Nisar.Ahmed@colorado.edu

Han-Lim Choi
Department of Aerospace Engineering
Korea Advanced Institute of Science and Technology
Daejeon, Korea
hanlimc@kaist.ac.kr

***Abstract*—In uncertain and dynamic environments, decentralized data fusion (DDF) techniques have been widely used to estimate the states and the uncertainty levels over large mission spaces in a robust and scalable way. In data fusion frameworks using distributed sensor networks, undetected sensor failures can degrade the quality of fusion results of the entire system. Therefore, DDF methods which are robust to inconsistent data are needed. In this paper, a fault-tolerant Bayesian DDF method using Gaussian mixture models is developed. The probability of agent reliability states, which represent consistency of local estimates that agents share with their neighbors, are modeled as weights of mixture models and estimated together with the target process. The target process and reliability states are updated in a decentralized Bayesian way, exploiting the properties of Gaussian mixture models. To prevent the hypothesis explosion problem of Gaussian mixture models, a mixture compression method considering the physical meaning of mixture weights is utilized. A numerical simulation on a 2D dynamic target tracking problem is presented to verify performance of the suggested algorithm and compared with existing DDF methods. It is shown that the suggested algorithm gives more compact fusion results compared to existing fault-tolerant DDF method.**

***Index Terms*—State estimation, Distributed inference, Multi-robot systems, Distributed sensor network, Fault-tolerant Bayesian DDF**

I. Introduction

In uncertain and dynamic environments, it is necessary for autonomous vehicles to precisely estimate their own states and the states of their environments, as well as quantify uncertainty to ensure best overall performance. For estimation tasks, fusing information from multiple distributed sensors in a network enables effective and scalable information gathering. A centralized data fusion strategy enables ideal fusion results, but it suffers from the inherent issues of single point of communications and computation failures. To overcome these issues and make the entire system more scalable, decentralized data fusion (DDF) methods have been suggested. In this sense, a wide range of data fusion algorithms, including Bayesian DDF [1,2], Dempster-Shafer inference [3,4], and fuzzy logic [5,6] have been widely studied. Among these variety of algorithms, Bayesian DDF is capable of tracking the full state distribution of a given process, and can easily and recursively fuse new data shared across multiple platforms with prior knowledge, without the need to store or communicate massive amounts of raw sensor data [2]. These properties and its rigorous theoretical framework have made Bayesian DDF a well-established technique for decentralized state estimation.

Due to aforementioned advantages, multi-robot system architectures are becoming increasingly popular to information gathering tasks. However, major concerns can arise if some robots or their sensors fail during the mission. Once a sensor on a certain platform fails, inconsistent data or estimates are generated, and communicated to the others. If other fusing agents do not know about and account for this inconsistency, the quality of overall fusion result can be quite degraded. As such, the sensor failure problem can affect the credibility of fusion results. However, despite the potential failure, there has been lack of research dealing with fault-tolerant Bayesian fusion methods. Therefore, the goal of this study is to develop an effective Bayesian DDF algorithm, that is robust to inconsistent data.

One strategy for adapting Bayesian data fusion for the case with potential failures is to extend conservative fusion strategies (where the fused posterior pdf does not underestimate the uncertainty relative to the true pdf [21]) like Covariance Intersection (CI) [7]. Since CI guarantees conservative fusion result only if the individual estimates are both consistent, methods like Covariance Union (CU) [8] and Generalized Covariance Union (GCU) [9] try to account for all possible multiple failure hypotheses among the networked platforms, so that agents in the network "hedge" their estimates against potentially inconsistent estimates received by others. Since CU and GCU cover both of potentially inconsistent estimates, these methods naturally give overly conservative fusion results compared to CI and exact Bayesian methods. The authors in [8] suggested utilizing CI after CU to reduce this large uncertainty, but this approach cannot deal with persistent error case (i.e., once a sensor on the platform fails, it stays in failed state afterward). More explicit strategies that model failure probabilities deploy methods akin to those used for data association, by leveraging mixture models to quantify the uncertainty associated with different agents being in different failure/non-failure states over time. This idea can be combined with Standard Mixture Reduction (SMR) [10] techniques to give more compact fusion results compared to CU and GCU, but this method requires prior knowledge of each agent's reliability.

The other way of dealing with inconsistent data in fusion framework is to detect erroneous measurement and eliminate it from fusion process. To find the faulty agent, error detection module (EDM) compares outputs of local fusion blocks and utilize similarity between them as determination metric [11]. Also, a method to eliminate inconsistent data by analyzing the variance level of fusion results, which change according to the consistency of local measurement, has been suggested [12]. However, these deterministic methods using EDM and elimination process are highly dependent on setting of the elimination criteria, and hard to apply for decentralized case, since they inherently require comparisons between local data.

In this study, a fault-tolerant Bayesian DDF method which considers online estimation of agent reliability states is developed. The agent reliability here means the credibility of local estimate that agent shares to its neighbors. The joint distribution over the target process and the reliability states is modeled by Gaussian mixture models, where the probability of different reliability state hypotheses serve as mixture weights. The augmented state vector (defined by the target process and the reliability states) is conditionally factorized, so that the target process and the reliability states can be separately updated. In this process, two distinct channel filters (CF) [1] are utilized to exactly track common information, which enables decentralization of the fusion procedure. The reliability states here can be interpreted as an extension of modes in multiple model estimation [13, 14], representing credibility of distinct local estimators. Each fusing agent updates mixture weights related with their own local measurement, and these posterior reliability distributions are fused in a decentralized way. Therefore, the suggested algorithm can show good scalability especially when there are large number of local estimators. The goal of this algorithm is to have robustness to data inconsistency, and give more compact (i.e. not overly conservative) fusion results compared to existing fault-tolerant fusion methods, with the capability to deal with persistent error cases.

II. Problem Statement

The decentralized data fusion problem considered in this paper is defined as follows, described by simple two fusing agent case for the ease of explanation. Agent i and j estimate a common target process, x. Along with the target state, each agent has its own reliability state r^i and r^j, so the augmented state vector at time k can be defined as

$$s_k = [x_k, r_k] = [x_k, r_k^i, r_k^j]. \tag{1}$$

The joint reliability state r is represented by combinations of binary local reliability states r^i and r^j, and r can have value among 11, 10, 01, and 00.

The agent reliability states may change at every time step, and affect the local measurement on the target process at time k, z_k^i and z_k^j. Each agent calculates local posterior distribution on s using its prior and local measurement, and shares this posterior with its neighboring agents. Therefore, if we define cumulative information set at time k from agent i as Z_k^i, the goal of the decentralized data fusion is to find the fusion rule:

$$p_f(s) = p(s|Z_k^i \cup Z_k^j) = \mathbb{F}(p(s|Z_k^i), p(s|Z_k^j)) \tag{2}$$

It is assumed throughout this paper that a "fault" in one agent results in its local sensor measurement on target process becoming corrupted, so that its local state estimates are no longer reliable. If θ^i denotes failure probability of agent i at each time step, the reliability state of agent i at time k is modeled as:

$$\begin{cases} r_k^i = 0 \text{ if } r_{k-1}^i = 0 \\ r_k^i = Bernoulli\left(1-\theta^i\right) \text{ if } r_{k-1}^i = 1 \end{cases} \tag{3}$$

Equation (3) means reliable agent i can become unreliable ($r^i = 0$) with probability θ^i, and once a sensor on the platform fails during the operation, it cannot be recovered and collects inaccurate data from that point on. This error model is denoted as "persistent error" hereafter. Depending on its reliability state, an agent makes different measurements on the target process. If an agent is unreliable, a known bias b is added to an otherwise accurate measurement. Then unknown true target and reliability states are estimated by using acquired measurements.

Fig. 1 represents Bayesian network for dynamic target process. Since the reliability states of different agents do not affect each other, it can be assumed that reliability states of fusing agents are mutually independent ($r_k^i \perp\!\!\!\perp r_k^j$). Also, the local measurement of each agent is determined by the true target state, its true reliability state and bias. Therefore, it can be naturally assumed that measurements of fusing agents are conditionally independent given target state ($z_k^i \perp\!\!\!\perp z_k^j | x_k$), which is reasonable for general distributed information gathering problems.

III. Fault-tolerant Bayesian DDF

A. Conditional Factorization

When $s = [x, r]$, we can conditionally factorize joint distribution over s as

$$p(s) = p(x, r) = p(x|r)p(r). \tag{4}$$

The decentralized fusion rule at time k can be written as [17]

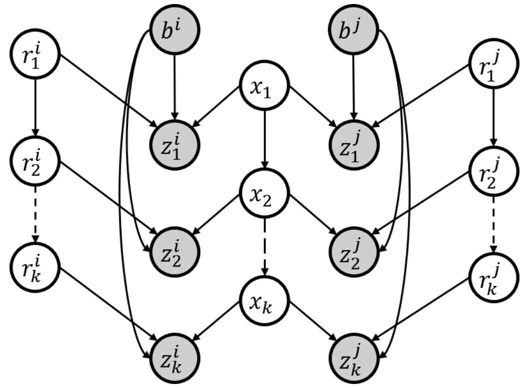

Fig. 1. Bayesian network. White nodes are unobserved and shaded nodes are observed variables. Measurement bias b is known constant.

$$p_f(s_k) = p(s_k|Z_k^i \cup Z_k^j)$$
$$\propto \frac{p(s_k|Z_k^i)p(s_k|Z_k^j)}{p(s_k|Z_k^i \cap Z_k^j)} = \frac{p_i(s_k)p_j(s_k)}{p_c(s_k)} \tag{5}$$

where $p_i(\cdot)$ denotes local posterior distribution from agent i, and $p_c(\cdot)$ denotes common prior distribution of agent i and j. $Z_k^i \cup Z_k^j$ means collection of cumulative information set of agent i and j at time k, and $Z_k^i \cap Z_k^j$ means cumulative common information set of agent i and j at time k.

By applying (4) to (5), we can conditionally factorize joint posterior distribution over s as shown in [18],

$$p_f(s) \propto \frac{p_i(x|r)p_i(r)p_j(x|r)p_j(r)}{p_c(x|r)p_c(r)}$$
$$= \frac{p_i(x|r)p_j(x|r)}{p_c(x|r)} \cdot \frac{p_i(r)p_j(r)}{p_c(r)}. \tag{6}$$

Let $\tilde{p}_f(x|r)$ be unnormalized fused conditional pdf,

$$\tilde{p}_f(x|r) = \frac{p_i(x|r)p_j(x|r)}{p_c(x|r)}. \tag{7}$$

If we normalize this and define $\eta(r)$ as

$$p_f(x|r) = \tilde{p}_f(x|r) \cdot \frac{1}{\eta(r)}$$
$$\eta(r) = \int \tilde{p}_f(x|r)\, dx, \tag{8}$$

then we can rewrite (6) as

$$p_f(s) \propto p_f(x|r)\eta(r)p_f(r). \tag{9}$$

Now we separated Bayesian update process of augmented state vector s into separate updates of the conditional target distribution and the reliability state distribution. If we can exactly track common information, we can update the conditional target distribution in a decentralized way for each reliability hypotheses. Also if we can calculate $\eta(r)$, reliability posteriors can be fused, and the fused joint posterior distribution $p_f(x, r)$ can be calculated in a decentralized way, as can be seen from (9).

B. Fault-tolerant Fusion using Gaussian Mixtures

Since the joint distribution on the augmented state vector s is conditionally factorized as in Section III.*A*, pdf on the target process can be modeled by Gaussian mixture models, where the reliability state probabilities are utilized as mixture weights, representing likelihood of various reliability hypotheses. Therefore, the online reliability states of fusing agents are estimated together with the target process.

Through the conditional factorization, the estimation procedure can be separated into estimation of continuous target process and of discrete reliability states. This makes the estimation problem simpler, and also well-established Bayesian filters for continuous variables can be used for target process estimation. To decentralize the estimation process on the target and the reliability states, two distinct CFs [1] are utilized to track the common information exactly. Since the pdf on the target process is represented by Gaussian mixtures (i.e., components of the pdf is "Gaussian" distributions), the message from local target estimator to reliability estimator (denoted by $\eta(r)$ in (9) and Fig. 2) can be exactly calculated, so that the decentralized fusion process becomes tractable (detail explanation is in Section III.*D*). A Gaussian moment-matching [15] based mixture compression method is utilized to prevent "hypothesis explosion" problem, with combination criteria (of hypotheses) to deal with mixture weights physically representing the reliability states of agents, and have their own dynamics.

The overall fusion procedure can be represented as Fig. 2. Each agent calculates local posterior distributions (e.g. $p_i(x|r)$ and $p_i(r)$ for agent i) using common information and its own local measurement. Local posterior distributions are sent to each other through CFs (target CF and reliability CF). CFs gets local posterior distributions as inputs and send out "new" information to the other fusing agent. Then, each agent fuses communicated "new" information with its own local posterior to calculate $p_f(x|r)$ and $p_f(r)$. In this process, $\eta(r)$ can be exactly calculated since $p_f(x|r)$ is Gaussian distribution. Therefore, fused joint posterior $p_f(x, r)$ and fused marginal distribution $p_f(x)$ (which is obtained by marginalizing out r from $p_f(x, r)$ and represented by Gaussian mixtures) can be calculated. Finally, fused marginal distribution $p_f(x)$ is compressed, and the compressed mixture models are used as common prior information at the next time step.

C. Local Posterior Update

Suppose we have four different conditional prior distributions at time k, $p(x_k|r_{k-1})$ for joint reliability hypotheses (11, 10, 01, 00) of fusing agents at previous time step. From the perspective of agent i, we can calculate local posterior distribution over x_k corresponding to previous joint reliability hypothesis r_{k-1} and current joint reliability hypothesis r_k by using Bayes' rule,

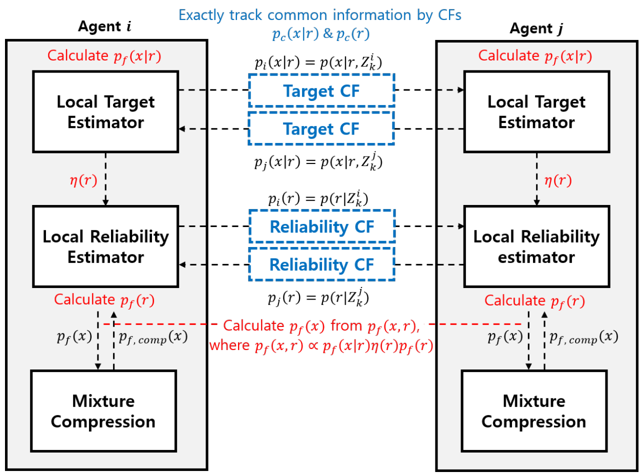

Fig. 2. Overall fusion procedure

$$
\begin{aligned}
p_i(x_k|r_{k-1}, r_k) &= p(x_k|r_{k-1}, r_k, z_k^i, Z_{k-1}) \\
&= \frac{p(z_k^i|x_k, r_{k-1}, r_k, Z_{k-1})p(x_k|r_{k-1}, r_k, Z_{k-1})}{p(z_k^i|r_{k-1}, r_k, Z_{k-1})} \\
&= \frac{p(z_k^i|x_k, r_{k-1}, r_k^i, Z_{k-1})p(x_k|r_{k-1}, r_k, Z_{k-1})}{p(z_k^i|r_{k-1}, r_k^i, Z_{k-1})}.
\end{aligned} \tag{10}
$$

As written in the last line of (10), agent i can update its local posterior only considering its own current reliability hypothesis r_k^i, since agent i's local measurement is independent to agent j's reliability state ($z_k^i \perp\!\!\!\perp r_k^j$).

Likewise, the local posterior on joint reliability states can be calculated again using Bayes' rule,

$$
\begin{aligned}
p_i(r_{k-1}, r_k) &= p(r_{k-1}, r_k|z_k^i, Z_{k-1}) \\
&= \frac{p(z_k^i|r_{k-1}, r_k, Z_{k-1})p(r_{k-1}, r_k|Z_{k-1})}{p(z_k^i|Z_{k-1})} \\
&= \frac{p(z_k^i|r_{k-1}, r_k^i, Z_{k-1})p(r_{k-1}, r_k|Z_{k-1})}{p(z_k^i|Z_{k-1})}.
\end{aligned} \tag{11}
$$

Although it is not specified in (10) and (11), prediction steps of the target and the reliability states are conducted before measurement updates at each local estimators. The prediction and measurement update of the target process, corresponding to (10), can be conducted together by using Bayesian filters in a parallel manner, according to the reliability hypotheses. Also, normalization constant $p(z_k^i|r_{k-1}, r_k^i, Z_{k-1})$ from (10), which will be obtained during local target posterior calculation, can be reused as measurement likelihood while updating local reliability posterior.

D. Decentralized Posterior Fusion

Calculated local posterior distributions over target process and joint reliability state can be fused in a decentralized way by using common information tracked by target CF and reliability CF, respectively. As can be seen from (9), $\eta(r)$ is required to calculate fused joint posterior distribution of augmented state. Unfortunately, calculation of $\eta(r)$ for general pdfs is not an easy task, since it includes integration over all possible $x's$. For the case of fusing Gaussian pdfs of static process, or dynamic process when local estimators and target CF have same system model, it is possible to analytically calculate $\eta(r)$.

The Gaussian pdf in information form is,

$$
\mathcal{N}^{-1}(x; \nu, \Lambda) \triangleq \frac{\exp\left(-\frac{1}{2}\nu^\top \Lambda^{-1} \nu\right)}{(2\pi)^{n/2}\sqrt{\det \Lambda^{-1}}} \exp\left(-\frac{1}{2}x^\top \Lambda x + x^\top \nu\right), \tag{12}
$$

where ν and Λ denote information vector and matrix respectively.

The Channel filter [1] is a method to exactly track common information, $p_c(\cdot)$ between two fusing agents. The CF recursively updates the common information between two fusing agents, by propagating and comparing the predicted distribution with the local posteriors. The Gaussian CF update equation in information form can be written as,

$$
\begin{aligned}
\nu_f &= \nu_i + \nu_j - \nu_c \\
\Lambda_f &= \Lambda_i + \Lambda_j - \Lambda_c.
\end{aligned} \tag{13}
$$

where subscript i and j mean local posterior from each agent, c means (predicted) common prior tracked by target CF, and f means fused posterior.

From (7) and (8), $\eta(r)$ can be rewritten as,

$$
\begin{aligned}
\eta(r) &= \frac{\tilde{p}_f(x|r)}{p_f(x|r)} \\
&= \frac{p_i(x|r)p_j(x|r)}{p_c(x|r)} \cdot \frac{1}{p_f(x|r)}.
\end{aligned} \tag{14}
$$

When both of local posteriors, common prior, and fused posteriors are all Gaussian distributions, $\eta(r)$ can be exactly calculated by using (12). By canceling out common terms, $\eta(r)$ can be simply represented as,

$$
\begin{aligned}
\eta(r) &= \sqrt{\frac{\det \Lambda_i \det \Lambda_j}{\det \Lambda_c \det \Lambda_f}} \\
&\exp\left\{-\frac{1}{2}\left(\nu_i^\top \Lambda_i^{-1}\nu_i + \nu_j^\top \Lambda_j^{-1}\nu_j - \nu_c^\top \Lambda_c^{-1}\nu_c - \nu_f^\top \Lambda_f^{-1}\nu_f\right)\right\}.
\end{aligned} \tag{15}
$$

Now, local posterior distributions of x and r can be calculated by using (10) and (11), and calculated local posterior distributions can be fused in a decentralized way by using (14) and (15).

E. Mixture Compression

From a single prior distribution on the reliability hypothesis, there can be four posterior distributions of current reliability hypotheses, similar to Gaussian sum filters with non-Gaussian measurement noise. Therefore, if we form a Gaussian mixture on x by marginalizing out r in (4) after fusion, the number of mixands increases geometrically every time step, which is called "hypothesis explosion" problem. To resolve this issue, a mixture compression method based on Gaussian moment-matching [15] is utilized.

Gaussian moment-matching is an algorithm to compress Gaussian mixtures to single Gaussian distribution while preserving mean and covariance matrix. The compressed Gaussian distribution of M Gaussian mixtures can be represented as,

$$
\begin{aligned}
w' &= \Sigma_{m=1}^{M} w_m \\
\mu' &= \frac{1}{w'} \Sigma_{m=1}^{M} w_m \mu_m \\
P' &= \frac{1}{w'} \Sigma_{m=1}^{M} w_m (P_m + \mu_m \mu_m^\top) - \mu' \mu'^\top
\end{aligned} \tag{16}
$$

where μ, P and w denote mean, covariance matrix, and weight of mixand, and $(\cdot)'$ means compressed Gaussian distribution.

In general mixture compression problems, criteria like the upper bound of Kullback-Leibluer divergence (KLD) between pre-merge and post-merge Gaussian mixtures [16] have been suggested to select mixands to be combined. In this paper, the physical meaning of the mixture weight is used as a criterion.

The mixture weight in our problem has physical meaning of the likelihood of each reliability hypothesis. Therefore, the prediction step of the mixture weights should reflect dynamics of the agent reliability states. The true dynamics of agent reliability state is defined as (3). Therefore, by utilizing prior knowledge of agent failure mode ("persistent") and failure probability θ of each agent, the prediction step of the probability of reliability states (which is defined by column vector) can be represented as,

$$
p(r_k|Z_{k-1}) = F_r p(r_{k-1}|Z_{k-1})
$$

$$
F_r = \begin{bmatrix} (1-\theta^i)(1-\theta^j) & 0 & 0 & 0 \\ (1-\theta^i)\theta^j & 1-\theta^i & 0 & 0 \\ \theta^i(1-\theta^j) & 0 & 1-\theta^j & 0 \\ \theta^i\theta^j & \theta^i & \theta^j & 1 \end{bmatrix}. \tag{17}
$$

To apply (17), compressed mixture weights should retain their physical meaning, the likelihood of current reliability hypotheses. Therefore, Gaussian moment-matching method is applied to mixands of same "current" hypothesis (i.e. merging four mixands of different "previous" hypotheses) in a parallel manner for each current hypothesis. It can be interpreted as filtering previous trajectories of reliability states. The overall mixture compression procedure can be represented as fig. 3.

The overall procedure of fault-tolerant Bayesian DDF can be represented as Algorithm 1.

IV. Numerical Simulation Result

A. Simulation Setup

For verification of the suggested algorithm, numerical simulation is conducted on a 2D dynamic target tracking problem

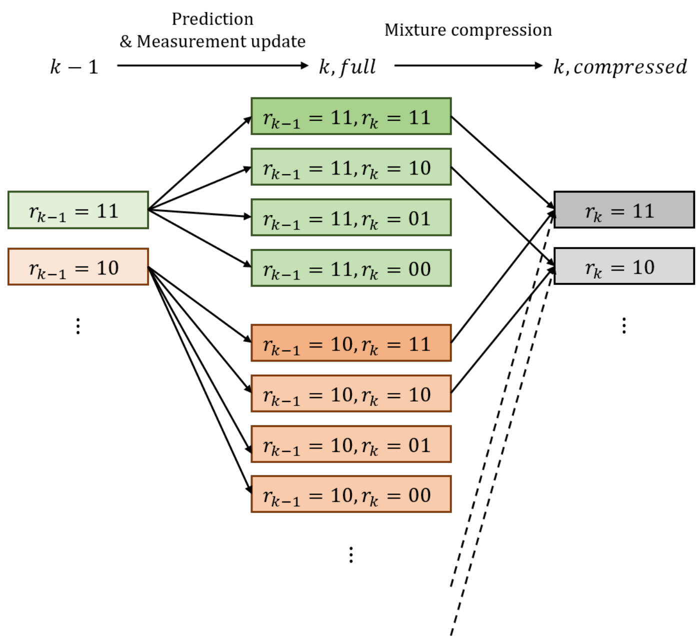

Fig. 3. Mixture compression procedure. Each block means estimation on joint reliability states and corresponding conditional distribution on x given r.

Algorithm 1 Fault-tolerant Bayesian DDF

Define prior pdfs on x and r
for All time steps **do**
  Propagate target process in local estimators
  Make local measurements
  Update local posterior on x and r ▷ Eq. (10) and (11)
  Send local posteriors to each other through CFs
  Propagate prior on target process
  Fuse local posteriors on x
  Calculate $\eta(r)$ ▷ Eq. (15)
  Propagate prior on reliability states ▷ Eq. (17)
  Fuse local posteriors on r
  Calculate fused joint posterior $p_f(x, r)$ ▷ Eq. (9)
  Calculate $p_f(x)$ by marginalizing out r
  Compress Gaussian mixtures ▷ Eq. (16)
  $p(x|r) \leftarrow$ Mixand of compressed Gaussian mixtures
end for

with two fusing agents (denoted by agent 1 and 2). The target process x and joint reliability state r are defined as,

$$
\begin{aligned}
x &= [e, \dot{e}, n, \dot{n}]^\top \\
r &= [r_1, r_2]^\top,
\end{aligned} \tag{18}
$$

where e and n denotes position in easting/northing directions, and $\dot{e}$ and $\dot{n}$ denotes velocity.

For simplicity, it is assumed that the target has linear constant velocity dynamics, and position of the target is directly measured by sensors. However, the suggested algorithm can be applied to a nonlinear problem without loss of generality by replacing local estimators by nonlinear filters, like the unscented Kalman filter. The target dynamics can be written as,

$$x_{k+1} = F_t x_k + G_t u_k + w_k \tag{19}$$

where

$$\begin{gathered} u_k = [1, 0]^\top \\ w_k \sim \mathcal{N}(0, Q_t) \\ F_t = \begin{bmatrix} 1 & dt & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & dt \\ 0 & 0 & 0 & 1 \end{bmatrix}, \; G_t = \begin{bmatrix} \frac{1}{2}dt^2 & 0 \\ dt & 0 \\ 0 & \frac{1}{2}dt^2 \\ 0 & dt \end{bmatrix} \\ Q_t = \begin{bmatrix} 0.01 & 0 & 0 & 0 \\ 0 & 0.04 & 0 & 0 \\ 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 0.04 \end{bmatrix}. \end{gathered}$$

The measurement model of agent i can be described as,

$$z^i_{k+1} = H^i x_{k+1} + (1 - r^i_{k+1}) b^i + v^i_{k+1} \tag{20}$$

where

$$\begin{gathered} v^i_{k+1} \sim \mathcal{N}(0, R^i) \\ H^1 = H^2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \\ R^1 = R^2 = \begin{bmatrix} 0.36 & 0 \\ 0 & 0.36 \end{bmatrix}. \end{gathered}$$

Since the goal of the suggested algorithm is to give consistent and compact fusion result in spite of inconsistent data, it is assumed that at least one fusing agent makes consistent local estimate. Therefore, for the clarity of analysis, agent 1 is set to stay normal (yet fusing agents do not know this), while agent 2 follows failure model defined by (3) with the failure probability $\theta^2 = 0.05$. As can be seen from (20), an additive bias b is added to otherwise accurate but noisy measurement when the true reliability state of an agent is 0. In this case, b is set to be $[2, -2]^\top$.

The same initial prior distribution on the target process is used for all reliability hypotheses. The target and reliability priors are set to be,

$$\begin{gathered} p_{pr}(x|r) = \mathcal{N}(\mu_{pr}, P_{pr}) \\ p_{pr}(r) = [(1-\theta)^2, \theta(1-\theta), \theta(1-\theta), 0]^\top \\ \mu_{pr} = x_0 \\ P_{pr} = 4I_4 \\ \theta = 0.05, \end{gathered}$$

where x_0 means initial target state vector.

To update the common information of target process and reliability states, it is assumed that both of local estimators and target CF have same dynamics model, which guarantee the proper tracking of the common information for dynamic process.

To verify fusion performance of the suggested algorithm, CU/CI [19] and Bayesian DDF method without considering agent reliability (denoted as "naive Bayesian") were used for comparison. Both of the algorithms used same local estimators as the suggested one. For all cases, it is assumed that local agents have no information of the control input of the target.

B. Result and Discussion

Fig. 4 represents one of the fusion results of the suggested algorithm and comparison group. The result of the suggested algorithm, denoted by FT Bayesian, is represented by blue line. The red line represents CU/CI and the yellow line represents naive Bayesian DDF method.

Fig. 4(a) shows the true position trajectory and the position estimates of all algorithms. It is shown that the suggested algorithm can track the target state properly, despite the existence of agent failure at time $k = 12$ (from the true reliability history plot of Fig. 4(d)). Due to the inconsistent local estimate from faulty agent 2, it is found that the fusion result of naive Bayesian DDF is shifted to the direction of the measurement bias of agent 2. Since CU/CI makes fusion result to cover both of local estimates (which are potentially inconsistent), the fused estimate of CU/CI also shifted to the same direction. This phenomenon can also be seen in the absolute error plot of position estimate in Fig. 4(b). Similar to Fig. 4(a), the suggested algorithm showed the smallest error overall, and CU/CI and naive Bayesian DDF showed very similar behaviors in terms of point estimate.

Fig. 4(c) shows the trace of error covariance matrix, which represents the uncertainty level of the fusion result. All algorithms showed similar value before the agent 2 fails, but CU/CI showed much larger level of uncertainty after the failure compared to the others. This is because CU method is utilized due to local inconsistency from agent 2, and the level of uncertainty cannot be recovered because of the persistent error. The naive Bayesian DDF showed similar value with the suggested algorithm, but there was shift in the point estimate when agent 2 is unreliable.

Fig. 4(d) shows the history of true and estimated reliability states. By using the suggested algorithm, the reliability states can be properly estimated in general. However, there is instability of reliability estimates especially in the early stage. This instability can make the fusion result to be inaccurate, so improving this reliability estimation process should be a future work.

Fig. 5 represents the NEES chi-square consistency test [20] result of the suggested algorithm with 100 Monte-Carlo runs and $\alpha = 0.05$. To calculate NEES statistics of the fusion result represented by Gaussian mixtures, the mean and covariance matrix of the Gaussian mixture (calculated by Gaussian moment-matching method) is used. It is shown that the NEES statistics of the fusion result are larger than r_2 bound (except for the early stage of the simulation), which means the fusion result is not statistically consistent. As can

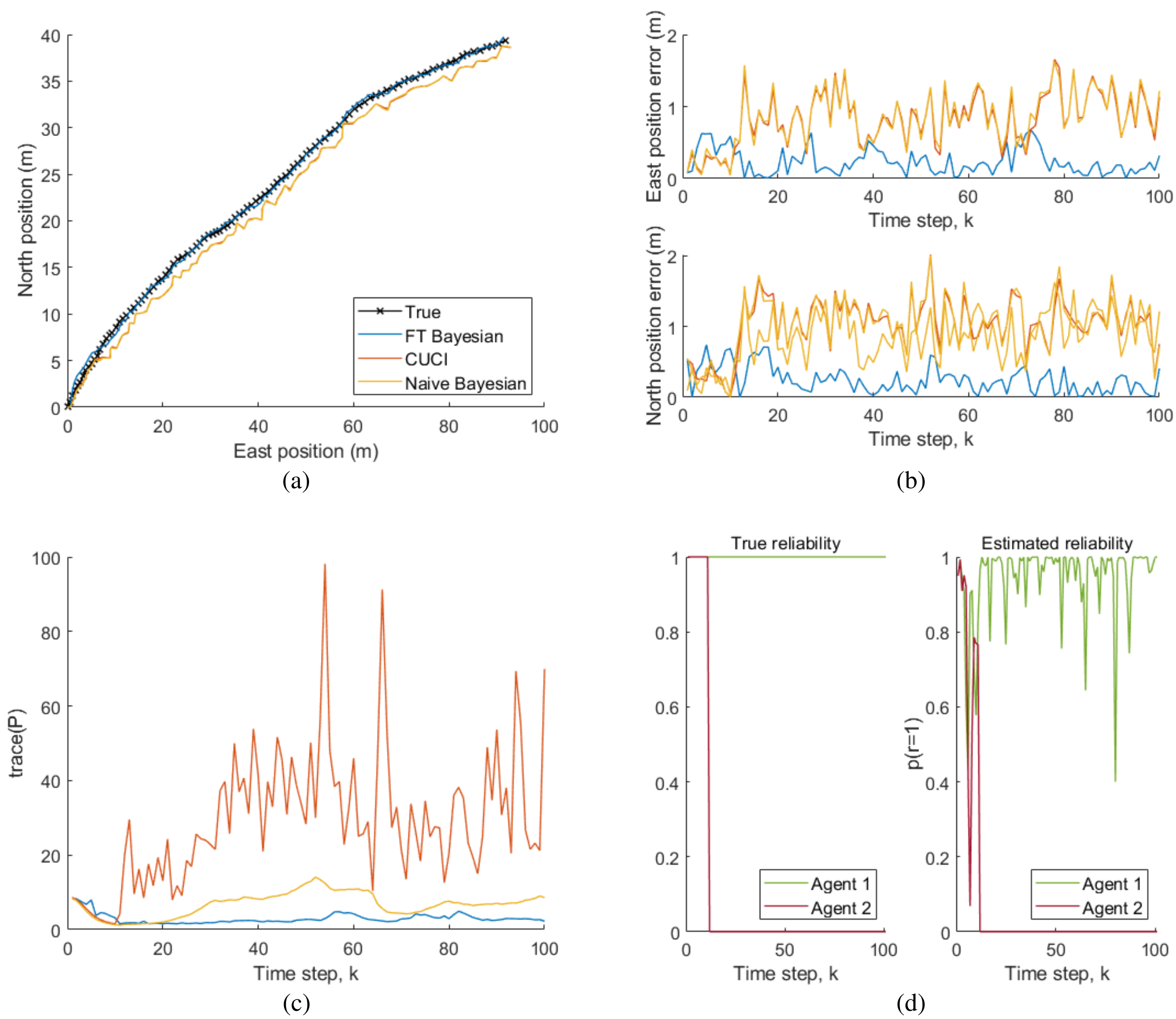

Fig. 4. (a) Position estimates (b) Absolute position error (c) Trace of error covariance matrix (d) True/estimated reliability states

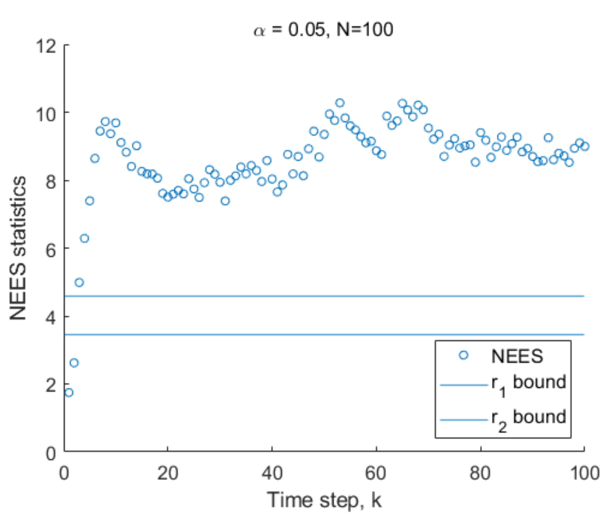

Fig. 5. NEES chi-square test result with 100 Monte-Carlo runs

be seen from Fig. 4(c) and Fig. 5, our Bayesian algorithm can give compact fusion result considering reliability states of fusion agents, but its statistical consistency is not guaranteed according to NEES chi-square test. Therefore, improvement of the fusion algorithm to resolve this issue should be a future work.

A summary of the results is as follows. The fusion performance of the suggested fault-tolerant Bayesian DDF algorithm is compared with existing methods for the problem where agent reliability changes during the system operation. In terms of the accuracy of the point estimate, the suggested algorithm showed better performance than comparison group. Also, the suggested algorithm can give more compact fusion result compared to CU/CI, but it is shown that the statistical consistency is not guaranteed. It was also able to properly estimate reliability states, which is one of the key parts to track the target in the presence of inconsistent local estimate.

V. Conclusion

In this paper, a fault-tolerant Bayesian DDF algorithm using agent reliability variables and Gaussian mixture models was developed. The joint reliability states of fusing agents are used as mixture weights, and locally estimated together with the target process. The local estimates are fused in a decentralized manner, and this process could become tractable by using the advantages of Gaussian mixture models. To prevent hypothesis explosion problem, Gaussian moment-matching method considering the physical meaning and dynamics of weights was utilized to compress the Gaussian mixtures.

To verify fusion performance of the suggested algorithm and compare with the existing methods, numerical simulation on a 2D dynamic target tracking problem was conducted. While CU/CI and Bayesian DDF without considering agent reliability led to biased fusion results due to inconsistent local estimate, our algorithm could track the target more accurately. Along with the target process, reliability states could be properly estimated. In terms of uncertainty, the suggested algorithm gave more compact fusion result compared to CU/CI, and could deal with persistent errors properly. However, statistical consistency of the fusion result is not guaranteed, so further analysis is in need.

The future research directions from this study are as follows:

- Consistency analysis and improvement of fusion method: determine the source of statistical inconsistency of the fusion result (e.g. underestimating error covariance matrix or high state estimation error). If there is a significant non-Gaussian component in the state pdf which breaks the Gaussian assumption for NEES chi-square test, other metrics like Normalized Deviation Squared (NDS) statistics [22] should be utilized.
  optimism of the fusion result, and improve fusion method to resolve this issue.
- Generalize for unknown/partially known erroneous measurement model (e.g. the amount of measurement error increases since the first failure occured): need to determine how accurate prior knowledge for erroneous measurement model is required, and improvement and stabilization of reliability estimate method is needed for unknown measurement error cases.
- Heterogeneous data fusion: factorize state variables according to conditional independence to reduce communication overhead [21].
- More realistic simulation and real-world demonstration: realistic simulation considering system nonlinearities in dynamics and measurement model and complex network topology, and real-world demonstration on robotic platforms.

References

[1] S. Grime and H. F. Durrant-Whyte, “Data fusion in decentralized sensor networks,” Control Engineering Practice, vol. 2, no. 5, pp. 849–863, 1994.

[2] M. E. Campbell and N. R. Ahmed, “Distriubted data fusion: Neighbors, rumors, and the art of collective knowledge,” in IEEE Control Systems Magazine, vol. 36, no. 2, pp.83–109, Aug. 2016.

[3] P. Dempster, “A generalization of Bayesian inference,” Journal of the Royal Statistical Society B, vol. 30, no. 2, pp. 205–247, Nov. 1968.

[4] A. Shafer, “Mathematical theory of evidence,” Princeton University Press, Princeton, NJ, USA, 1976.

[5] R. R. Yager and L. A. Zadeh, Eds., An Introduction to Fuzzy Logic Applications in Intelligent Systems. Norwell, MA: Kluwer, 1991.

[6] J. K. Klir and B. Yuan, Fuzzy Sets and Fuzzy Logic: Theory and Applications. Upper Saddle River, NJ: Prentice-Hall, 1995.

[7] S. J. Julier and J. K. Uhlmann, “A non-divergent estimation algorithm in the presence of unknown correlations,” in Proc. IEEE American Control Conference, vol. 4, Albuquerque, NM, USA, 1997, pp. 2369–2373.

[8] J. K. Uhlmann, “Covariance consistency methods for fault-tolerant distributed data fusion,” Information Fusion, vol. 4, no. 3, pp. 201–205, Sept. 2003.

[9] S. Reece and S. Roberts, “Generalised covariance union: A unified approach to hypothesis merging in tracking,” in IEEE Transactions on Aerospace and Electronic Systems, vol. 46, no. 1, pp. 207–221, Jan. 2010.

[10] D. J. Salmond, “Mixture reduction algorithms for target tracking in clutter,” in Proc. SPIE 1990, vol. 1305, Orlando, FL, USA, 1990, pp. 434–445.

[11] K. Bader, B. Jussier, and W. Schön, “A fault tolerant architecture for data fusion: A real application of Kalman filters for mobile robot localization,” Robotics and Autonomous Systems, vol. 88, pp. 11–23, 2017.

[12] M. Kumar, D. P. Garg, and R. A. Zachery, “A method for judicious fusion of inconsistent multiple sensor data,” IEEE Sensors Journal, vol. 7, no. 5, pp. 723–733, May 2007.

[13] H. A. P. Blom, and Y. Bar-Shalom, “The interacting multiple model algorithm for systems with Markovian switching coefficients,” IEEE Transactions on Automatic Control, vol. 33, no. 8, pp. 780–783, 1988.

[14] E. Mazor, A. Averbuch, Y. Bar-Shalom, and J. Dayan, “Interacting multiple model methods in target tracking: a survey,” IEEE Transactions on aerospace and electronic systems, vol. 34, no. 1, pp. 103–123, 1998.

[15] D. J. Salmond, “Mixture reduction algorithms for target tracking in clutter,” Signal and Data Processing of Small Targets, vol. 1305, pp. 434–445, 1990.

[16] A. R. Runnalls, “Kullback-Leibler approach to Gaussian mixture reduction,” IEEE Transactions on Aerospace and Electronic Systems, vol. 43, no. 3, pp. 989–999, July 2007.

[17] C. Y. Chong, E. Tse, and S. Mori, “Distributed estimation in networks,” in Proc. American Control Conference, 1983, pp. 294–300.

[18] N. Ahmed, “Conditionally factorized DDF for general distributed Bayesian estimation,” 2014 Internal Conference on Multisensor Fusion and Information Integration for Intelligent Systems (MFI), 2014, pp. 1–7.

[19] J. Klingner, N. Ahmed, and N. Correll, “Fault-tolerant covariance intersection for localizing robot swarms,” Robotics and Autonomous Systems, vol. 122, Dec. 2019.

[20] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, “Estimation with applications to tracking and navigation: theory algorithms and software,” John Wiley & Sons, Hoboken, NJ, USA, 2001.

[21] O. Dagan and N. R. Ahmed, “Exact and approximate heterogeneous Bayesian decentralized data fusion,” IEEE Transactions on Robotics, vol. 38, no. 2, pp. 1136–1150, 2022.

[22] N. Ahmed, L. Burks, K. Cabral, and A. B. Rose, “Exact consistency tests for Gaussian mixture filters using normalized deviation squared statistics,” arXiv preprint arXiv:2312.17420, 2023.