Integrating Covariance Intersection into Bayesian multi-target tracking filters

Daniel E. Clark *Senior Member, IEEE* and Mark A. Campbell

Abstract

Multi-target tracking systems typically provide sets of estimated target states as their output. It is challenging to be able to integrate these outputs as inputs to other tracking systems to gain a better picture of the area under surveillance since they do not conform to the standard observation model. Moreover, in cyclic distributed systems, there may be common information between state estimates that would mean that fused estimates may become overconfident and corrupt the system. In this paper we develop a Bayesian multi-target estimator based on the covariance intersection algorithm for multi-target track-to-track data fusion. The approach is integrated into a multi-target tracking algorithm and demonstrated in simulations. The approach is able to account for missed tracks and false tracks produced by another tracking system.

I. Introduction

Target tracking systems typically use the Kalman filter [1] or some variant of it as the basis for online estimation. Measurements are received sequentially in time and the state estimate is updated according to Bayes' rule. If measurements are not available from a sensor and instead a mean and a covariance from an external tracking system is given, the Kalman filter is no longer applicable. Such systems are known as distributed target tracking systems, and such distributed data fusion algorithms combine the state estimates that are generated by a number of fusion centres or nodes. (Note that this is distinct from distributed systems that are not cyclic, e.g. [2].)

A distributed tracking system has numerous advantages over a centralised system [3, 4] including robustness, scalability and modularity. Such systems often require fusing the outputs from heterogeneous platforms with differing sensing and processing capabilities. Methods for fusing the outputs of different tracking systems, have been developed for over 3 decades [5, 6]. One of the most successful approaches for distributed track-to-track fusion is the covariance intersection method of Julier and Uhlmann [7]. This is highly appealing due to its robustness, simple structure, and applicability to any tracking system that uses Gaussians as the basis for tracking. It has been widely applied to Decentralised Data Fusion (DDF) problems [8], and on the Mars rover [9].

Mark A. Campbell is with Heriot-Watt University, UK. Email: mc318@hw.ac.uk and Daniel Clark is with Telecom SudParis, France. Email: daniel.clark@telecom-sudparis.eu This work was supported by the Joint AFRL-Dstl Basic-Research Grant in Autonomous Signal Processing AFOSR grant FA9550-19-1-7008 and Dstl Task No. 1000133068. It was also supported by the Engineering and Physical Sciences Research Council (EPSRC) under grant reference EP/L016834/1 and was undertaken in the EPSRC Centre for Doctoral Training in Robotics and Autonomous Systems at Heriot-Watt University and the University of Edinburgh.

Generalisations of the covariance intersection algorithm to non-Gaussian systems have been proposed based on the exponential mixture density (weighted geometric mean) structure of the algorithm [10, 11]. Extensions for fusing two Gaussian mixtures has also been proposed [12]. Following the proposed multi-target generalisation in [10], practical exponential mixture density implementations based on particular multi-target processes [13, 14] have been developed. These have been extended to sensor localisation in distributed fusion networks [15], and distributed localisation of sensors with partially overlapping field-of-views in fusion networks [16]. Recent applications of distributed multi-target tracking based on exponential mixture densities include consensus exponential mixture density [17, 18] human tracking [19], asynchronous radar system [20], orbit determination and space debris tracking [21], tracking based on multi-static Doppler shifts [22], and localization and tracking of mobile networks [23].

Recently, problems with cardinality estimation using exponential mixture density-based fusion of point processes have been highlighted [24]. The approach taken in this paper is distinct from these exponential mixture density based approaches in distributed multi-target tracking in the following respect: it does not attempt to fuse two processes via the generalised version of covariance intersection. Instead, it develops a Bayesian fusion rule that can be integrated at the track level, which enables the exploitation of the robust and effective covariance intersection algorithm for fusion of two multivariate Gaussian distributions. This has a number of advantages over the exponential mixture density approaches. Firstly, the approach can take sets of tracks from any type of multi-target tracking algorithm as the input. Secondly, it can be integrated into any multi-target tracking algorithm which uses Gaussians to represent the track state, either per-track or within a mixture, using the robust and fast covariance intersection algorithm approach. Thirdly, the approach naturally deals with the fact that fields of view may be different via the detection probability inherited from the multi-target tracker. Fourthly, it can deal with potentially false information through the inheritance of a clutter process for tracks.

The next section describes Bayesian disintegration and presents the central idea of interpreting the Covariance Intersection algorithm as a form of Bayes rule and calculates the marginal needed to integrate it into Bayesian target tracking filters. Section II.C describes how to use it in specific multi-target tracking filters. The approach is illustrated with a multi-target tracking algorithm. Section IV presents a study with a particular multi-target tracking algorithm.

II. Bayesian estimation and Chernoff fusion

A. Application of the Kalman filter in tracking applications

In this section we review the Bayesian interpretation of the Kalman filter as used in target tracking applications. The presentation is designed to facilitate the connection with the Bayesian interpretation of Chernoff fusion rule and the covariance intersection algorithm presented in a novel context in the next section.

Conditional probability and Bayesian estimation relies on the following disintegration of a joint distribution $p(x, z)$ into conditionals and marginals [25, 26].

$$\underbrace{p(z \mid x)}_{\substack{\downarrow \\ \text{likelihood}}}\underbrace{p(x)}_{\substack{\downarrow \\ \text{prior}}} = \underbrace{p(x, z)}_{\substack{\downarrow \\ \text{joint}}} = \underbrace{p(x \mid z)}_{\substack{\downarrow \\ \text{update}}}\ \underbrace{p(z)}_{\substack{\downarrow \\ \text{marginal}}}. \tag{1}$$

On the left, we consider $p(x)$ to be the prior, and $p(z \mid x)$ to be a likelihood. The essence of Bayesian estimation is to start from the prior and determine a conditional update based on observations of the random variable z, i.e. $p(x \mid z) = \frac{p(z|x)p(x)}{p(z)}$. In the calculation of the conditional update, we need to compute the marginal $p(z)$, which can be determined by integrating over the joint, so that we can rewrite the decomposition as follows

$$\underbrace{p(z \mid x)}_{\substack{\downarrow \\ \text{likelihood}}} \underbrace{p(x)}_{\substack{\downarrow \\ \text{prior}}} = \underbrace{p(x, z)}_{\substack{\downarrow \\ \text{joint}}} = \underbrace{p(x \mid z)}_{\substack{\downarrow \\ \text{update}}} \underbrace{\int p(z \mid x) p(x) \mathrm{d}x}_{\substack{\downarrow \\ \text{marginal}}}. \tag{2}$$

We now consider the Bayesian disintegration for Gaussians, as used in the Kalman filter, i.e. given matrices H, R, and P and vector m of appropriate dimensions, the following identity holds.

$$\underbrace{\mathcal{N}(z; Hx, R)}_{\substack{\downarrow \\ \text{likelihood}}} \underbrace{\mathcal{N}(x; m, P)}_{\substack{\downarrow \\ \text{prior}}} = \underbrace{\mathcal{N}(x; \tilde{m}, \tilde{P})}_{\substack{\downarrow \\ \text{update}}} \underbrace{\mathcal{N}\left(z; Hm, S\right),}_{\substack{\downarrow \\ \text{marginal}}} \tag{3}$$

where $\mathcal{N}(x; m, P)$ is a multivariate Gaussian in vector-valued variable x with mean m and covariance matrix P, and the Kalman updated mean, state covariance, innovation covariance, and gain terms are given by

$$\begin{aligned} \tilde{m} &= m + K(z - Hm), \\ \tilde{P} &= (I - KH)P, \\ S &= HPH^T + R, \\ K &= PH^T S^{-1}. \end{aligned} \tag{4}$$

In the target tracking literature, the marginal $p(z) = \mathcal{N}\left(z; Hm, S\right)$ corresponds to the probability that the measurement z originated from the target. This is an important quantity in target tracking algorithms since it is used to determine weights of components. For instance,

- in Alspach and Sorenson's Gaussian sum filter, it is used to calculate component weights in the Gaussian mixture posterior density (see (3.3) in [27]);
- in Reid's algorithm for multi-target tracking [28], this is used to to determine the hypothesis weights (cf. equations (15) and (16));
- in the Probabilistic Data Association algorithm of Bar-Shalom, it is used to determine the likelihood ratio that a measurement belongs to a target rather than clutter (cf. equation (38) in [29]);
- in the Gaussian mixture Probability Hypothesis Density filter of Vo and Ma [30] it is used to determine the weights of different intensity components (see equation (41)).

The exponent of the Gaussian is also used to determine a validation region for gating, i.e. [29]

$$\mathcal{V}(\gamma) = \left\{ z : (z - Hm)' \, S^{-1} \, (z - Hm) \leq \gamma \right\}, \tag{5}$$

where the gate threshold γ corresponds to a gate probability that the validation region contains the true measurement if detected.

B. Chernoff fusion as Bayesian estimation

We now apply this classical approach to the Chernoff fusion rule for Gaussians, i.e. the covariance intersection algorithm. To do this, we reinterpret the Chernoff fusion rule as a Bayesian update rule and determine the analogous quantities to use the result as a tracking filter update. The covariance intersection algorithm provides the updated mean and covariance and thus we only require to determine the equivalent marginal after the update.

In distributed data fusion, it is common to use a fusion rule [10, 11] based on Chernoff information [31] to determine a density $p_\omega(x)$ based on two densities describing the same random variable, $p(x)$ and $q(x)$ and a mixing parameter ω, given by

$$p_\omega(x) = \frac{q(x)^\omega p(x)^{1-\omega}}{\int q(x)^\omega p(x)^{1-\omega} \mathrm{d}x}. \tag{6}$$

Such descriptions are also known as escort distributions in the statistical physics literature [32], and have been used to determine information-related quantities e.g. [33].

When $p(x)$ and $q(x)$ are multivariate Gaussian distributions, this is equal to the covariance intersection algorithm [7]. The structure of the Chernoff update rule is similar to the Bayes update rule if we permit ourselves to consider the term $\left(\frac{q(x)}{p(x)}\right)^\omega$ as a kind of unnormalised likelihood. We start with the prior, and determine the fused posterior $p_\omega(x)$ with the relations

$$\underbrace{\left(\frac{q(x)}{p(x)}\right)^\omega}_{\substack{\downarrow\\ \text{unnormalised likelihood}}} \underbrace{p(x)}_{\substack{\downarrow\\ \text{prior}}} = \underbrace{p_\omega(x)}_{\substack{\downarrow\\ \text{update}}} \underbrace{\int \left(\frac{q(x)}{p(x)}\right)^\omega p(x)\mathrm{d}x}_{\substack{\downarrow\\ \text{unnormalised marginal}}}. \tag{7}$$

We note, of course, the non-standard nature of the proposed (unnormalised) likelihood in that we have not specified a random variable, and that it depends on a whole posterior as input $q(x)$, as well as the prior $p(x)$. This clearly does not factorise into the familiar likelihood-prior and update-marginal relations due to the dependence on $q(x)$ and $p(x)$. We also note that the likelihood and proposed marginal are both in unnormalised forms, so that we have not explicitly defined the random variable nor, by extension, the joint distribution. However, in Bayesian disintegration [26, p325], such convenient decompositions do not always exist. We can see that in this case, if we consider $p(x)$ as the prior and $p_\omega(x)$ as the update, then we do not have the classical Bayesian decomposition.

The objective in the current work is to determine a Bayesian disintegration that would permit us to use the expression $\left(\frac{q(x)}{p(x)}\right)^\omega$ as a likelihood, and find the related marginal. If this is possible, then we can integrate the approach into mixture based filters. Due to nice properties of Gaussians, and their utility for target tracking applications, we shall concentrate on Gaussian descriptions of $q(x)$ and $p(x)$.

In this case, we have the nice property that all of the terms are in the form of multivariate Gaussians, which enables us to determine analytic expressions. Let us now consider a disintegration for the Chernoff update rule with two Gaussians

$\mathcal{N}(x;a,A)$, $\mathcal{N}(x;b,B)$ with the same dimensions for the means a,b and covariances A,B, i.e.

$$\underbrace{\left(\frac{\mathcal{N}(x;a,A)}{\mathcal{N}(x;b,B)}\right)^{\omega}}_{\substack{\downarrow\\ \text{unnormalised likelihood}}}\underbrace{\mathcal{N}(x;b,B)}_{\substack{\downarrow\\ \text{prior}}} \tag{8}$$

$$=\underbrace{\mathcal{N}(x;d,D)}_{\substack{\downarrow\\ \text{update}}}\underbrace{\int\left(\frac{\mathcal{N}(x;a,A)}{\mathcal{N}(x;b,B)}\right)^{\omega}\mathcal{N}(x;a,A)\mathrm{d}x}_{\substack{\downarrow\\ \text{unnormalised marginal}}},$$

where d and D are the updated mean and covariance determined by the covariance intersection algorithm [7],

$$D=\left(\omega A^{-1}+(1-\omega)B^{-1}\right)^{-1}, \tag{9}$$

$$d=D\left(\omega A^{-1}a+(1-\omega)B^{-1}b\right). \tag{10}$$

If we can find a suitable normalisation for the marginal on the right, then we have specified a joint distribution and a prospective Bayesian disintegration. Note that the unnormalised marginal can be determined from the Chernoff information [34], so that we can write it explicitly as

$$\frac{|A|^{\frac{\omega}{2}}\,|B|^{\frac{1-\omega}{2}}}{|\omega A+(1-\omega)B|^{\frac{1}{2}}}\mathrm{e}^{-\frac{\omega(1-\omega)}{2}(a-b)^{\top}(\omega A+(1-\omega)B)(a-b)}, \tag{11}$$

which is equal to

$$C^{\omega}(A,B)\mathcal{N}(a;b,a/(1-\omega)+A/\omega), \tag{12}$$

where we define $C^{\omega}(A,B)$ to be

$$C^{\omega}(A,B)=\frac{(2\pi)^{d/2}\ |A|^{\omega/2}|B|^{(1-\omega)/2}}{\sqrt{\omega(1-\omega)}}. \tag{13}$$

If we consider the marginal to be a probability density in a, then this is a scaled Gaussian distribution in a, and we can write

$$\underbrace{\left(C^{\omega}(A,B)\right)^{-1}\left(\frac{\mathcal{N}(x;a,A)}{\mathcal{N}(x;b,B)}\right)^{\omega}}_{\substack{\downarrow\\ \text{normalised likelihood}}}\underbrace{\mathcal{N}(x;b,B)}_{\substack{\downarrow\\ \text{prior}}} \tag{14}$$

$$=\underbrace{\mathcal{N}(x;d,D)}_{\substack{\downarrow\\ \text{update}}}\ \underbrace{\mathcal{N}(a;b,V)}_{\substack{\downarrow\\ \text{normalised marginal}}}\ ,$$

where

$$V=A/(1-\omega)+B/\omega. \tag{15}$$

This description of the joint distribution is the key result in the paper that enables us to integrate covariance intersection into target tracking filters. More specifically, the specification of the marginal $\mathcal{N}(a;b,V)$ permits us to integrate the covariance intersection algorithm into Bayesian tracking filters since it enables us to determine weight calculations determined via Bayes' rule. See Appendix A for an example of how to employ the result in the Gaussian sum filter of Alspach and Sorenson [27].

While the update is the same as the covariance intersection algorithm for determining the updated mean and covariance, the calculation of the marginal enables us to calculate weights in target tracking filters in a Bayesian manner in the same way as described in the previous section. We note here that here we have taken the marginal to be probability density function in vector a and consider the covariance A to be a parameter for the likelihood i.e. we do not have a probability distribution over mean and covariance pairs (a, A). The appropriateness of this choice of likelihood function will be assessed through statistical simulations in the following sections.

This line of reasoning has enabled us to recast the Chernoff update rule as a Bayesian update rule. With this new interpretation, we can take the mean a and covariance A of another target track estimate, and use them to determine an updated distribution. In the Gaussian case, this is exactly the covariance intersection rule. The exponent of the Gaussian is also used to determine a validation region for gating, i.e.

$$\mathcal{V}(\gamma) = \left\{(a, A) : (a-b)'V^{-1}(a-b) \leq \gamma\right\}.$$

The covariance intersection algorithm provides the updated mean d and covariance D in the updated Gaussian $\mathcal{N}(x;d,D)$.

C. Application to target tracking algorithms

We now describe how to apply the approach in different tracking filters by replacing the Kalman fusion rule with the new rule formed by the new joint distribution with reference to particular algorithms. In a multi-target context, we assume that we are given a set of thresholded tracks from another tracking system. By analogy with classical multi-target tracking models, we assume that we have a detection probability that gives the probability that a target under surveillance gives rise to a track, and that a clutter model describes the distribution of false tracks.

- In Alspach and Sorenson's Gaussian sum filter, we use the covariance intersection algorithm for the updated means and covariances in the Gaussian mixture and use the normalised marginal $\mathcal{N}(a;b,V)$ in the calculations of the component weights in the Gaussian mixture posterior density (see (3.3) in [27] and Appendix A);
- In Reid's algorithm for multi-target tracking in 1979 [28], the new validation region is used for gating tracks in Equation (6) in [28]), the covariance intersection algorithm is used for the updated means and covariances in for each target track in equation (4) in [28]), the related normalised marginal $\mathcal{N}(a;b,V)$ is used to compute the hypothesis weights (cf. Equations (15) and (16) in [28]);
- In the Probabilistic Data Association (PDA) algorithm of Bar-Shalom, the Kalman filter is replaced with the covariance intersection algorithm, the new validation region is used for gating tracks (a, A) in Equation (6) in [29], and the normalised marginal $\mathcal{N}(a;b,V)$ it is used to determine the likelihood ratio that a track report (a, A) belongs to a true target rather than falsely reported target (cf. Equation (38) in [29]);

- In the Gaussian Mixture Probability Hypothesis Density filter of Vo and Ma [30], the covariance intersection algorithm is applied between each thresholded track (a, A) and components in the mixture, the normalised marginal $\mathcal{N}(a; b, V)$ is used to determine the weights of different intensity components (cf. (41) of [30]).
- In the algorithm by Williams [35], the required modifications to the algorithm include replacing the Kalman filter updated means and covariances (Figure 8 lines 21, 28) with the covariance intersection updates means and covariances, the Kalman filter marginal is replaced with the new marginal (line 27), and the new innovation replaces the Kalman innovation in Line 19.
- In the algorithm by Campbell, Clark and de Melo recently reported in [36], a similar modification to the Gaussian mixture Linear-Complexity Cumulant (LCC) is made to the terms in Section V.B. We use this algorithm in the following section for experiments.

We note that the key result itself is straightforward and can be applied in multi-target tracking systems of a different nature since they use the same basic Gaussian components. Though this relies on known results for the multiplication of Gaussians, the application of the covariance intersection method has not been applied in this way before. In the next section we illustrate the use of the method with a multi-target tracking algorithm in a simulated study.

III. Gaussian mixture distributions

A. The Gaussian sum filter

In this section we look at different update mechanisms for Gaussian mixture priors and representations of joint distributions. We show that it is possible to determine different conditional distributions depending on the choice of joint distribution. Suppose that $w^{(i)}$ for $i = 1, \ldots, N$ are non-negative and sum to 1. Suppose further that $p^{(i)}(x)$ for $i = 1, \ldots, N$ are probability density functions. Then the usual approach for updating mixture distributions with a likelihood can be described with the following Bayesian disintegration.

$$\underbrace{p(z \mid x)}_{\substack{\downarrow \\ \text{likelihood}}} \underbrace{\sum_{i=1}^{N} w^{(i)} p^{(i)}(x)}_{\substack{\downarrow \\ \text{prior}}} = \underbrace{p(x, z)}_{\substack{\downarrow \\ \text{joint}}} \tag{16}$$

$$= \underbrace{p(x \mid z)}_{\substack{\downarrow \\ \text{update}}} \underbrace{\int \sum_{i=1}^{N} w^{(i)} p(z \mid x) p^{(i)}(x) \mathrm{d}x}_{\substack{\downarrow \\ \text{marginal}}}.$$

For instance, the Gaussian sum filter of Alspach and Sorenson [27] is determined through the relation.

$$\underbrace{\mathcal{N}(z; Hx, R)}_{\substack{\downarrow \\ \text{likelihood}}} \underbrace{\sum_{i=1}^{N} w^{(i)} \mathcal{N}(x; m^{(i)}, P^{(i)})}_{\substack{\downarrow \\ \text{prior}}} \tag{17}$$

$$= \underbrace{p(x \mid z)}_{\substack{\downarrow \\ \text{update}}} \underbrace{\sum_{i=1}^{N} w^{(i)} \int \mathcal{N}(z; Hx, R) \mathcal{N}(x; m^{(i)}, P^{(i)}) \mathrm{d}x}_{\substack{\downarrow \\ \text{marginal}}} .$$

Each integral on the right can be calculated with the Kalman filter relation.

As we have seen in the previous section, the likelihood that we are interested in is inherently coupled to the prior. Hence, we need to carefully consider how to apply the new Bayesian approach for mixture distributions. However, we note that though mixture distributions are conventionally used in the form above, there is no special reasoning needed to define joint mixture distributions in the following way.

$$\sum_{i=1}^{N} w^{(i)} \underbrace{p^{(i)}(z \mid x)}_{\substack{\downarrow \\ i^{th} \text{ likelihood,}}} \underbrace{p^{(i)}(x)}_{\substack{\downarrow \\ i^{th} \text{ prior}}} \tag{18}$$

$$= \underbrace{p(x, z)}_{\substack{\downarrow \\ \text{joint}}} = \underbrace{p(x \mid z)}_{\substack{\downarrow \\ \text{update}}} \underbrace{\int \sum_{i=1}^{N} w^{(i)} p^{(i)}(z \mid x) p^{(i)}(x) \mathrm{d}x}_{\substack{\downarrow \\ \text{marginal}}} .$$

In this formulation, we consider N hypotheses, where each hypothesis has its own likelihood. For instance, in the Gaussian case, we can consider

$$\sum_{i=1}^{N} w^{(i)} \underbrace{\mathcal{N}(z; H^{(i)}x, R^{(i)})}_{\substack{\downarrow \\ i^{th} \text{ likelihood,}}} \underbrace{\mathcal{N}(x; m^{(i)}, P^{(i)})}_{\substack{\downarrow \\ i^{th} \text{ prior}}} \tag{19}$$

$$= \underbrace{p(x \mid z)}_{\substack{\downarrow \\ \text{update}}} \underbrace{\int \sum_{i=1}^{N} w^{(i)} \mathcal{N}(z; H^{(i)}x, R^{(i)}) \mathcal{N}(x; m^{(i)}, P^{(i)}) \mathrm{d}x}_{\substack{\downarrow \\ \text{marginal}}} .$$

For intuition purposes, we could think of this as a scenario where there is uncertainty about which sensor provided observation z, represented with the mixture weights $w^{(i)}$, and where each sensor has its own observation matrix $H^{(i)}$ and noise covariance $R^{(i)}$, described through likelihood $\mathcal{N}(z; H^{(i)}x, R^{(i)})$. We will use this description as a motivation for finding an analogous result for the Chernoff rule in the following section.

B. The Chernoff rule with mixture priors

We now consider relations for mixtures based on different applications of the Chernoff fusion rule. We shall be particularly interested in the following section in fusing a Gaussian distribution with a Gaussian mixture. The usual approach for updating mixture distributions with a the Chernoff rule can be described with the following relation.

$$\begin{aligned} & q(x)^{\omega}\left(\sum_{i=1}^{N} w^{(i)} p^{(i)}(x)\right)^{1-\omega} \\ & = p_{\omega}(x) \int q(x)^{\omega}\left(\sum_{i=1}^{N} w^{(i)} p^{(i)}(x)\right)^{1-\omega} \mathrm{d}x, \end{aligned} \tag{20}$$

and in the case where $q(x)$ is a Gaussian (input track) and $p(x)$ is the prior, we have

$$\begin{aligned} & \mathcal{N}(x;a,A)^{\omega}\left(\sum_{i=1}^{N} w^{(i)} \mathcal{N}(x;b^{(i)},B^{(i)})\right)^{1-\omega} \\ & = p_{\omega}(x) \int \mathcal{N}(x;a,A)^{\omega}\left(\sum_{i=1}^{N} w^{(i)} \mathcal{N}(x;b^{(i)},B^{(i)})\right)^{1-\omega} \mathrm{d}x. \end{aligned} \tag{21}$$

This is problematic since raising the Gaussian mixture to a power does not have a nice analytic form. Julier proposed the following approximation [37] for this scenario.

$$p_{\omega}(x) \approx \sum_{i=1}^{N} \tilde{w}^{(i)} \mathcal{N}\left(x; d_i, D_i\right), \tag{22}$$

where

$$\tilde{w}^{(i)} = \frac{(w^{(i)})^{1-\omega}}{\sum_{j=1}^{N} (w^{(j)})^{1-\omega}}, \tag{23}$$

and

$$\begin{aligned} D_i &= \left(\omega A^{-1} + (1-\omega) B_i^{-1}\right)^{-1} \\ d_i &= D_i \left(\omega A^{-1} a + (1-\omega) B_i^{-1} b_i\right) \\ V_i &= A/(1-\omega) + B_i/\omega \end{aligned} \tag{24}$$

This has been used in some applications [17, 18].

Based on the approach described in the previous section, we propose a different approximation based on the joint distribution for Gaussians given in the previous section, i.e. we consider the relation

$$\begin{aligned} & \sum_{i=1}^{N} w^{(i)} \underbrace{\left(C^{\omega}(A,B_i)\right)^{-1} \left(\frac{\mathcal{N}(x;a,A)}{\mathcal{N}(x;b_i,B_i)}\right)^{\omega}}_{\substack{\downarrow \\ i^{th} \text{ normalised likelihood}}} \underbrace{\mathcal{N}(x;b_i,B_i)}_{\substack{\downarrow \\ i^{th} \text{ prior}}} \\ & = \underbrace{p_{\omega}(x)}_{\substack{\downarrow \\ \text{update}}} \sum_{i=1}^{N} w^{(i)} \quad \underbrace{\mathcal{N}(a;b_i,V_i)}_{\substack{\downarrow \\ i^{th} \text{ normalised marginal}}} \quad , \end{aligned} \tag{25}$$

or, using the relations in the previous section,

$$\sum_{i=1}^{N} w^{(i)} \underbrace{\mathcal{N}(x; d_i, D_i)}_{\substack{\downarrow \\ i^{th} \text{ update}}} \quad \underbrace{\mathcal{N}(a; b_i, V_i)}_{\substack{\downarrow \\ i^{th} \text{ normalised marginal}}} \tag{26}$$

$$= \underbrace{p_\omega(x)}_{\substack{\downarrow \\ \text{update}}} \sum_{i=1}^{N} w^{(i)} \underbrace{\mathcal{N}(a; b_i, V_i)}_{\substack{\downarrow \\ i^{th} \text{ marginal}}}.$$

where

$$D_i = \left(\omega A^{-1} + (1-\omega) B_i^{-1}\right)^{-1},$$
$$d_i = D_i \left(\omega A^{-1} a + (1-\omega) B_i^{-1} b_i\right),$$
$$V_i = A/(1-\omega) + B_i/\omega.$$

This leads to the following description for $p_\omega(x)$.

$$p_\omega(x) = \sum_{i=1}^{N} \hat{w}^{(i)} \mathcal{N}(x; d_i, D_i), \tag{27}$$

where the updated weights are determined with

$$\hat{w}^{(i)} = \frac{w^{(i)} \mathcal{N}(a; b_i, V_i)}{\sum_{j=1}^{N} w^{(j)} \mathcal{N}(a; b_j, V_j)}. \tag{28}$$

This does not take the original form of the Chernoff fusion rule. However, it has the advantage that it does not have the complication of raising a Gaussian mixture to a power. As we have seen, this takes the form of a conditional probability update. However, since the different modelling approach does not impose the same likelihood for each term in the mixture, we are able to achieve a closed form solution in terms of Gaussian mixtures. In the next section we use these results in the context of Bayesian estimation for point processes.

IV. Experiments

In this section, we shall analyse the performance of the new covariance intersection update method through two scenarios using simulated data. To avoid the computational complexities when calculating the ω weighting fusion parameter, a fast intersection method is used to compute the exponential weighting parameter [38]. Each of the multi-target filters used in the experiments below (with the exception of Scenario. A) are LCC [39] filters, following an optimised Gaussian mixture multi-target filter implementation as described in [36]. The Mahalanobis distance is used to evaluate potential similar components in the merging step with threshold τ_{merge}.

Each scenario shares the same underlying target dynamic and sensor model with shared parameters as seen in Table I. It should be noted that some of these parameters are only applicable for the multi-target scenarios. The target space is over a 2D region of dimensions $X \times Y$ (m). M targets are initially generated uniformly across the region at time $k = 1$ and are detected with a probability P_d with survival probability P_s. New targets are birthed every timestep with a Poisson process with mean μ_γ. These targets follow a near-constant velocity motion model with state dimensions $[x, \dot{x}, y, \dot{y}]$, transition matrix F_k, process

Parameter	Symbol	Value
State Space Dimensions	$X \times Y$	1000×1000
Timesteps	T	100
Gating Threshold	τ_{gate}	0.99
Pruning Threshold	τ_{prune}	0.0001
Merging Threshold	τ_{merge}	10
Extraction Threshold	τ_{track}	0.1
Measurement Noise (m)	σ_r	1
Target Process Noise (m)	σ_q	0.01

Table I: General simulation & tracking parameters

noise Q_k and process noise standard deviation σ_q. Targets are observed by the sensor with an observation matrix H_k, noise R_k and measurement noise standard deviation σ_r.

Each simulation is ran for T timesteps with uniformly spatially distributed false alarms being generated at each timestep according to a Poisson process with mean μ_λ. All results shown are averaged over 100 Monte Carlo (MC) runs of each scenario. The Optimal SubPattern Assignment (OSPA) metric is used to assess the performance [40]. In the following scenarios, an OSPA cut-off c parameter of 100 and an order p value of 2 are used. Only targets that possess a weight above an estimation threshold τ_{track} are used in calculating the OSPA. The lines in the graphs represent the averaged results over the MC runs. For the execution time per timestep and OSPA results, the shaded areas show the 2σ standard deviation of the runs. For the cardinality results, the shaded areas show the mean of the 2σ standard deviation of the multi-object filter estimate. Each of the simulations has been ran on a PC with a twelve-core AMD Ryzen 3900X CPU with $64\,\mathrm{GB}$ of RAM.

A. *Scenario A*

Scenario A integrates the new rule into a Gaussian sum filter [27] and compares with the centralised Gaussian sum filter and the previously proposed approach by Julier [12]. Since we use linear-Gaussian dynamic and observation models, the filters typically converge very quickly. Hence, rather than compare their performance over time, we analyse their performance at the first time step i.e. we focus on the accuracy of the Bayesian estimation step. In this scenario, we consider the performance of the update for varying measurement noise of the filter feeding into the update. 10 values of σ_r, the standard deviation of the measurement noise, were logarhythmically spaced between 0.01 and 1.

At time $k = 1$, a single target is generated and for each Gaussian sum filter, $N = 1000$ Gaussian components are uniformly distributed across the state space. No Gaussian mixture reduction techniques are performed and every component is updated with each obtained measurement.

Four scenarios are considered with the results presented in Figure 1: 1) A Gaussian sum filter with a measurement update from one sensor (solid blue line); 2) A ”centralised” Gaussian sum filter with a measurement update from two different sensors (dotted blue line); 3) A Gaussian sum filter with one measurement update that is then updated with Julier’s method with a Kalman filtered mean and covariance from another sensor (red line); 4) A Gaussian sum filter with one measurement update that is then updated with the new method with a Kalman filtered mean and covariance from another sensor (green line).

The weighted average estimate is then extracted from each filter and compared in terms of Root Mean Square Error (RMSE) from the ground truth target in Figure 1. The results indicate that the new rule provides more accurate estimates than the previous proposed approach i.e. that the calculation of new mixture weights enables more accurate estimates of the filter, as in the previous approach distant components would be given the same weight are closer components. It should also be noted that whilst the two covariance intersection methods perform significantly better than the single Gaussian filter, they do perform worse than the centralised filter for each measurement noise level.

B. Scenario B

Scenario B simulates a challenging scenario in order to test the new fusion method. Here, the simulated scenario has been extended to a multi-target case following the parameters and structure detailed above, and where the probability of detection $P_d = 0.7$, the probability of survival $P_s = 0.99$, the initial number of targets is set at 100 with clutter rate $\mu_\lambda = 200$.

The results indicate that the cardinality estimates (Figure 2a) of the new approach enables more accurate calculations than a single Gaussian mixture filter, though it is not as accurate as the centralised approach, as indicated in the OSPA calculations in Figure 2b. The results for the cardinality estimate indicate that in the scenario considered, there is not a large difference between the new approach and the centralised approach, though the centralised approach does outperform the new approach.

C. Scenario C

Scenario C analyses the effect of varying the probability of detection on the fusion methods. Initially $M = 10$ targets are generated with a high probability of survival $P_s = 0.999$ and zero clutter rate $\mu_\lambda = 0$. Starting from a low probability of detection $P_d = 0.5$, 25 values are linearly spaced up to $P_d = 0.9$. For each value of P_d, the scenario is ran for T timesteps. The OSPA metric for the multi-target filter is calculated at each timestep and then are averaged over the T timesteps for each P_d, as displayed in Figure 3. Three multi-target filter variations are considered: 1) A LCC filter is updated with a single set of measurements (solid red line); 2) A "centralised" LCC filter, which is updated twice at each timestep with two separate sets of measurements (dashed blue line); 3) A LCC filter is updated with a single set of measurements and then updated using the new rule and a set of Gaussian means and covariances obtained from the first filter (solid green line).

The results in Figure 3 show that as to be expected, both the centralised and new fusion approach multi-target filters perform noticeably better than the single multi-target filter and that there is no significant difference in performance between the centralised filter and the new approach.

D. Scenario D

Scenario D follows the same simulation structure as the previous scenario, except now the clutter rate μ_λ is varied instead of the probability of detection. Still for $M = 10$ targets, μ_λ is gradually increased from $\mu_\lambda = 1$ to $\mu_\lambda = 20M = 200$ over 50 linearly spaced values. Similarly to Scenario B, the OSPA metrics shown in Figure 4 show the averaged OSPA metric over the T timesteps for each value of μ_λ.

The results in Figure 4 indicate that for low clutter, the centralised approach performs best. However, as the clutter level increases, the fusion approach can give more accurate estimates than the centralised approach. This is likely due to the fact that the new approach is using data that has already filtered out the clutter and has a better estimate of the number of targets.

We would typically expect that a centralised tracker should outperform a distributed tracker. However, we need to consider what is being fused in order to draw such a conclusion:

1) In the centralised tracker, we take the output of two sensors and estimate the target population based on applying Bayes' rule twice, where the observations are sets of measurements from the two different sensors (including false alarms and missed detections).

2) In the distributed scenario, the targets are updated with the measurements from one sensor with Bayes' rule, and then again with the new fusion rule (also based on Bayes' rule at the target number level) with tracks that have been estimated over time with another tracker.

We can reasonably expect that this other tracker has a better estimate the number of targets better than a full measurement set from another sensor, which is true. Hence, it is actually not surprising that the estimate in the number of targets can better when we apply the new fusion approach which is also based on Bayes rule. If we feed better data in, we can expect a better outcome: the data from the other tracker has already filtered out the false alarms. The authors believe that this is is why there is a better estimate of the number of targets in this scenario as indicated by the experiments.

E. Scenario E

Finally, the effect of the data incest problem is investigated. In a decentralised data fusion context, data incest is when the same information is used multiple times along a data processing chain. A covariance intersection approach [7] has inherent protection against this through the nature of its fusion, however the Gaussian mixture approximations of the multi-target filters used above have no such protection since the approach is Bayesian. So in order to test the robustness of the proposed strategy, the following scenario is used. Two LCC filters are ran simultaneously and are co-located in both spatially and temporally. At each timestep a full tracking cycle is executed: A local prediction and update, followed by broadcasting its local estimate then an update with neighbour's estimates. The second half of this cycle, broadcast and update, is then repeated a R_{CI} number of times. In the following results (Figure 5), $1, 2$ and 5 repeats are used.

This failure of the method in this scenario is confirmed by the cardinality and OSPA results, Figures 5a and 5b, where the 5 repeats run shows a severe degradation in performance when compared to the other runs. This performance degradation is also shown in the 2 repeat run where it consistently overestimates the number of targets present. This overconfidence can be explained by the results shown in Figure 5c, where as more updates are performed, the underlying multi-target filters increase their confidence in the existing targets by increasing the weights and decreasing the Kalman covariances of the corresponding Gaussian components. However, due to the repeated nature of the updates and the lack of new information these Kalman covariances decrease to a point, where the gating operations fail and the input tracks are no longer associated correctly. This leads to the $R_{CI} = 5$ results, where by this time, the weights of the components have fallen so significantly that the components have be pruned from the mixture. These results clearly show that the Gaussian mixture implementations can not adequately deal with the data incest problem alone and therefore should not be used in cyclical networks where there is a large sharing of data between nodes. Instead if the network topology is that of a directed acyclic graph, this approach should prove useful.

V. Conclusion

A new fusion rule for distributed multi-target tracking is proposed based on Chernoff information and covariance intersection. The method has been developed to exploit sets of tracks produced by other tracking systems. The approach has the advantage that it can be integrated directly into a range of existing multi-target tracking systems through the replacement of Kalman filtering updates with covariance intersection updates with a fast fusion weighting calculation. The approach takes advantage of the multi-target tracking system model to account for missed tracks and false tracks produced by the other tracking system hence providing a robust approach for distributed sensor fusion. It has also has the advantage of being able to integrate tracks from multi-target tracking systems of a different nature. Since the approach is Bayesian, there still remain potential problems of data incest in cyclic networks. However, when it can be reasonably assumed that there is low communication, then the approach offers a rapid way of integrating tracks produced by another system.

Acknowledgements

Thanks to Sean O'Rourke from AFRL and Alasdair Hunter from Dstl for helpful comments.

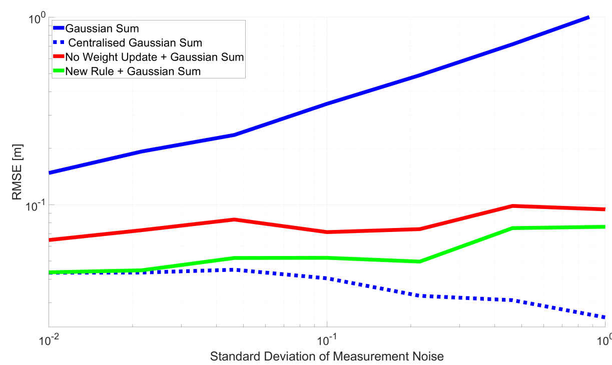


Fig. 1: Results for Scenario A.

References

[1] R. E. Kalman. "A New Approach to Linear Filtering and Prediction Problems". In: *Journal of Basic Engineering* 82.1 (1960), pp. 32–45.

[2] O. Hlinka, F. Hlawatsch, and P. M. Djuric. "Distributed particle filtering in agent networks: A survey, classification, and comparison". In: *IEEE Signal Processing Magazine* 30.1 (2012), pp. 61–81.

[3] B. Upcroft, M. Ridley, L.-L. Ong, B. Douillard, T. Kaupp, S. Kumar, T. Bailey, F. Ramos, A. Makarenko, A. Brooks, S. Sukkarieh, and H. Durrant-Whyte. "Multi-level State Estimation in an Outdoor Decentralised Sensor Network". In: vol. 39. Jan. 2006, pp. 355–365.

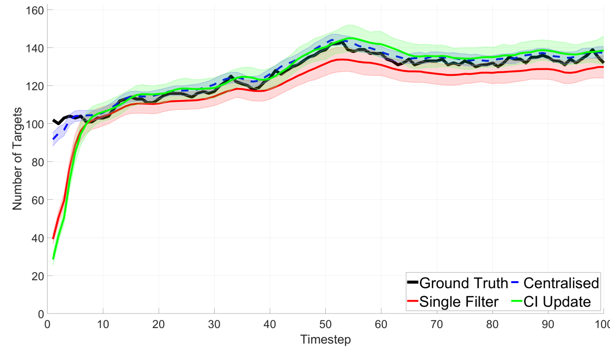
[4] S. Sukkarieh, E. Nettleton, J.-H. Kim, M. Ridley, A. Goktogan, and H. Durrant-Whyte. "The ANSER Project: Data Fusion Across Multiple Uninhabited Air Vehicles". In: *The International Journal of Robotics Research* 22.7-8 (2003), pp. 505–539.

[5] Kuo-Chu Chang, Chee-Yee Chong, and Y. Bar-Shalom. "Joint probabilistic data association in distributed sensor networks". In: *IEEE Transactions on Automatic Control* 31.10 (1986), pp. 889–897.

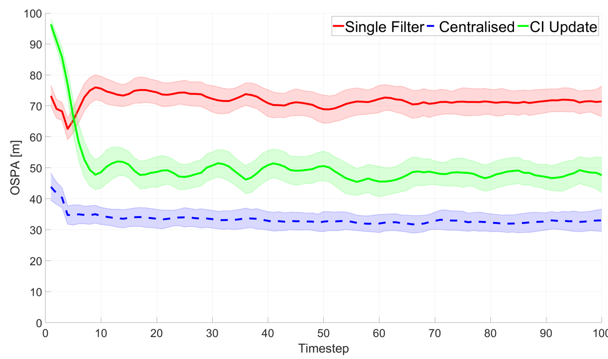
[6] K. Chang, C. Chong, and S. Mori. "Analytical and Computational Evaluation of Scalable Distributed Fusion Algorithms". In: *IEEE Transactions on Aerospace and Electronic Systems* 46.4 (2010), pp. 2022–2034.

[7] S. Julier and J. K. Uhlmann. "General decentralized data fusion with covariance intersection". In: *Handbook of multisensor data fusion*. CRC Press, 2017, pp. 339–364.

[8] B. Noack, J. Sijs, M. Reinhardt, and U. Hanebeck. "Decentralized data fusion with inverse covariance intersection". In: *Automatica* 79 (May 2017), pp. 35–41.



(a) Target cardinality estimates



(b) OSPA

Fig. 2: Results for Scenario B.

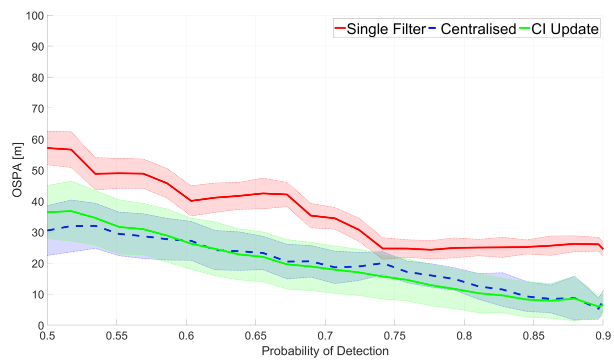


Fig. 3: Results for Scenario C.

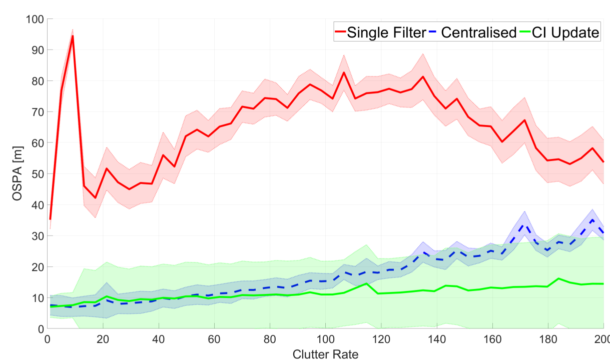
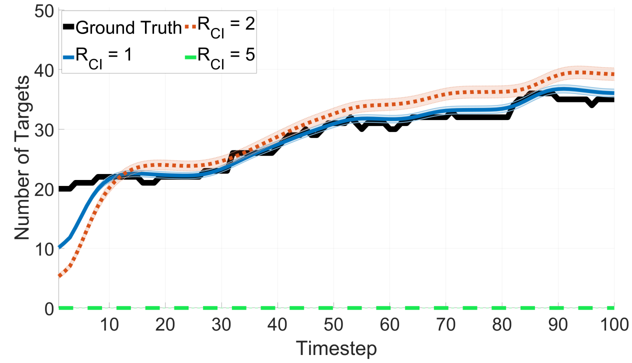
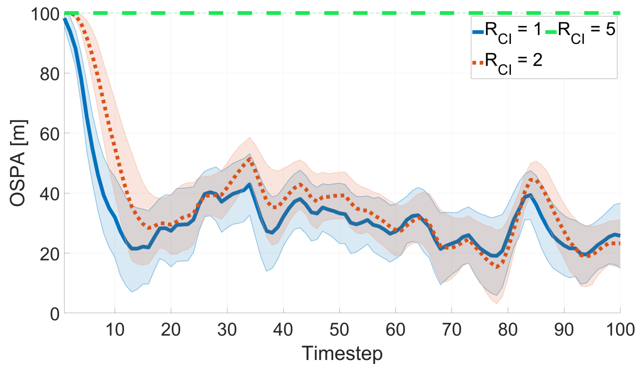


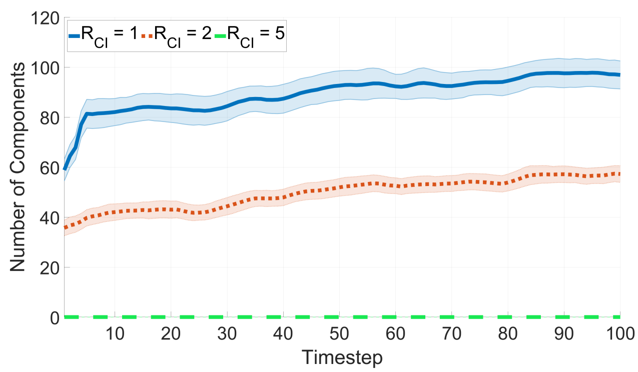
Fig. 4: Results for Scenario D.



(a) Target cardinality estimates



(b) OSPA



(c) Number of Gaussian components in the mixture

Fig. 5: Results for Scenario E.

[9] J. K. Uhlmann, S. J. Julier, B. Kamgar-Parsi, M. O. Lanzagorta, and H.-J. S. Shyu. "NASA Mars rover: a testbed for evaluating applications of covariance intersection". In: *Unmanned Ground Vehicle Technology*. Ed. by G. R. Gerhart, R. W. Gunderson, and C. M. Shoemaker. Vol. 3693. International Society for Optics and Photonics. SPIE, 1999, pp. 140 –149.

[10] R. P. S. Mahler. "Optimal/robust distributed data fusion: a unified approach". In: *Signal Processing, Sensor Fusion, and Target Recognition IX*. Ed. by I. Kadar. Vol. 4052. International Society for Optics and Photonics. SPIE, 2000, pp. 128 –138.

[11] M. B. Hurley. "An information theoretic justification for covariance intersection and its generalization". In: *Proceedings of the Fifth International Conference on Information Fusion. FUSION 2002. (IEEE Cat.No.02EX5997)*. Vol. 1. 2002, 505–511 vol.1.

[12] S. J. Julier. "An Empirical Study into the Use of Chernoff Information for Robust, Distributed Fusion of Gaussian Mixture Models". In: *2006 9th International Conference on Information Fusion*. 2006, pp. 1–8.

[13] D. E. Clark, S. Julier, R. P. S. Mahler, and B. Ristic. "Robust multi-object sensor fusion with unknown correlations". In: *IET Sensor Signal Processing in Defence (SSPD) conference* (2010).

[14] M. Üney, D. Clark, and S. Julier. "Distributed Fusion of PHD Filters Via Exponential Mixture Densities". In: *Selected Topics in Signal Processing, IEEE Journal of* 7 (June 2013), pp. 521–531.

[15] M. Üney, B. Mulgrew, and D. Clark. "A Cooperative Approach to Sensor Localisation in Distributed Fusion Networks". English. In: *IEEE Transactions on Signal Processing* 64.5 (Mar. 2016), pp. 1187–1199.

[16] M. Üney, B. Mulgrew, and D. Clark. "Distributed localisation of sensors with partially overlapping field-of-views in fusion networks". In: *2016 19th International Conference on Information Fusion (FUSION)*. 2016, pp. 1340–1347.

[17] G. Battistelli, L. Chisci, C. Fantacci, A. Farina, and A. Graziano. "Consensus CPHD Filter for Distributed Multitarget Tracking". In: *IEEE Journal of Selected Topics in Signal Processing* 7.3 (2013), pp. 508–520.

[18] M. Gunay, U. Orguner, and M. Demirekler. "Chernoff fusion of Gaussian mixtures based on sigma-point approximation". In: *IEEE Transactions on Aerospace and Electronic Systems* 52.6 (2016), pp. 2732–2746.

[19] Z. Fu, F. Angelini, J. Chambers, and S. M. Naqvi. "Multi-Level Cooperative Fusion of GM-PHD Filters for Online Multiple Human Tracking". In: *IEEE Transactions on Multimedia* 21.9 (2019), pp. 2277–2291.

[20] G. Li, W. Yi, M. Jiang, and L. Kong. "Distributed fusion with PHD filter for multi-target tracking in asynchronous radar system". In: *2017 IEEE Radar Conference (RadarConf)*. 2017, pp. 1434–1439.

[21] J. S. McCabe and K. J. DeMars. "Fusion Methodologies for Orbit Determination with Distributed Sensor Networks". In: *2018 21st International Conference on Information Fusion (FUSION)*. 2018, pp. 1323–1330.

[22] M. B. Guldogan. "Consensus Bernoulli Filter for Distributed Detection and Tracking using Multi-Static Doppler Shifts". In: *IEEE Signal Processing Letters* 21.6 (2014), pp. 672–676.

[23] F. Meyer, O. Hlinka, H. Wymeersch, E. Riegler, and F. Hlawatsch. "Distributed Localization and Tracking of Mobile Networks Including Noncooperative Objects". In: *IEEE Transactions on Signal and Information Processing over Networks* 2.1 (2016), pp. 57–71.

[24] M. Üney, J. Houssineau, E. Delande, S. J. Julier, and D. E. Clark. "Fusion of finite-set distributions: Pointwise consistency and global cardinality". In: *IEEE Transactions on Aerospace and Electronic Systems* 55.6 (2019), pp. 2759–2773.

[25] K. Cho and B. Jacobs. "Disintegration and Bayesian inversion, both abstractly and concretely". In: *See arxiv. org/abs/1709.00322* (2017).

[26] L. Le Cam. *Asymptotic methods in statistical decision theory*. Springer Science & Business Media, 2012.

[27] D. L. Alspach and H. W. Sorenson. "Nonlinear Bayesian estimation using Gaussian sum approximations". In: *Automatic Control, IEEE Transactions on* 17.4 (1972), pp. 439–448.

[28] D. Reid. "An algorithm for tracking multiple targets". In: *IEEE transactions on Automatic Control* 24.6 (1979), pp. 843–854.

[29] Y. Bar-Shalom, F. Daum, and J. Huang. "The probabilistic data association filter". In: *IEEE Control Systems Magazine* 29.6 (2009), pp. 82–100.

[30] B.-N. Vo and W.-K. Ma. "The Gaussian mixture probability hypothesis density filter". In: *IEEE Transactions on signal processing* 54.11 (2006), pp. 4091–4104.

[31] H. Chernoff. "A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations". In: *The Annals of Mathematical Statistics* 23.4 (1952), pp. 493–507.

[32] C. Beck and F. Schögl. *Thermodynamics of chaotic systems: an introduction*. 4. Cambridge University Press, 1995.

[33] O. Kharazmi and N. Balakrishnan. "Jensen-information generating function and its connections to some well-known information measures". In: *Statistics & Probability Letters* 170 (2021), p. 108995.

[34] F. Nielsen. "Generalized Bhattacharyya and Chernoff upper bounds on Bayes error using quasi-arithmetic means". In: *Pattern Recognition Letters* 42 (2014), pp. 25–34.

[35] J. L. Williams. "Marginal multi-Bernoulli filters: RFS derivation of MHT, JIPDA, and association-based MeMBer". In: *IEEE Transactions on Aerospace and Electronic Systems* 51.3 (2015), pp. 1664–1687.

[36] M. A. Campbell, D. E. Clark, and F. de Melo. "An Algorithm for Large-Scale Multitarget Tracking and Parameter Estimation". In: *IEEE Transactions on Aerospace and Electronic Systems* 57.4 (2021), pp. 2053–2066.

[37] S. J. Julier. "An empirical study into the use of Chernoff information for robust, distributed fusion of Gaussian mixture models". In: *2006 9th International Conference on Information Fusion*. IEEE. 2006, pp. 1–8.

[38] D. Franken and A. Hupper. "Improved fast covariance intersection for distributed data fusion". In: *2005 7th International Conference on Information Fusion*. Vol. 1. 2005, 7 pp.–.

[39] D. E. Clark and F. D. Melo. “A Linear-Complexity Second-Order Multi-Object Filter via Factorial Cumulants”. In: *2018 21st International Conference on Information Fusion (FUSION)*. July 2018.

[40] D. Schuhmacher, B.-N. Vo, and B.-T. Vo. “A Consistent Metric for Performance Evaluation of Multi-object filters”. In: *Signal Processing, IEEE Transactions on* 56.8 (2008), pp. 3447–3457.