Decentralised multi-sensor target tracking with limited field of view via possibility theory

Jeremie Houssineau
Division of Mathematical Sciences
Nanyang Technological University
Singapore
jeremie.houssineau@ntu.edu.sg

Chenbao Xue
School of Aerospace Engineering
Beijing Institute of Technology
Beijing, China
chenbaospace@bit.edu.cn

Han Cai
School of Aerospace Engineering
Beijing Institute of Technology
Beijing, China
caihanspace@gmail.com

Murat Uney
School of Electrical Engineering, Electronics and Computer Science
University of Liverpool
Liverpool, UK
m.uney@liverpool.ac.uk

Emmanuel Delande
Centre National d'Études Spatiales
Toulouse, France
emmanuel.delande@cnes.fr

***Abstract*—Quantifying negative information in an efficient way is a challenging task, especially when this information has to be communicated on a network. In this article we leverage the unique properties offered by possibility theory to quantify and approximate the negative information arising in the context of tracking a target with a sensor that has a limited field of view. We also verify experimentally that the corresponding target tracking methodology can be applied in a decentralised manner to a sensor network, while maintaining a performance close to the idealised case where the initial location of the target is better-known.**

***Index Terms*—Uncertainty modelling; negative information; information fusion; Bernoulli filter; inverted Gaussian mixture**

I. Introduction

The specific problem of tracking targets over a sensor network, where each sensor has a limited field of view (FOV), has attracted a lot of attention in the past few years [1]–[6]. Beyond the practical relevance of this problem, it is interesting at a methodological level since it truly tests the efficiency of the different techniques used to handle the sharing of information over sensor networks. Indeed, limited FOVs make such information sharing crucial to maintaining the track continuity for the target(s) of interest.

The standard probabilistic techniques for fusing the incoming information at a node of a sensor network [7], [8], such as covariance intersection (geometric averages) [9]–[11] or arithmetic averages [12], consider different principles to approximate the true (i.e., non-distributed) Bayesian posterior. This posterior distribution is inaccessible in practice as it would require gathering all the observations obtained on the network at a single node and performing a standard Bayesian update there, which goes against the objective of distributed fusion. Therefore, we will refer to the non-distributed Bayesian posterior as the "oracle" from now on.

Recently, a decentralised fusion method [13] leveraging the special properties of possibility theory [14], [15] has been introduced, which recovers a possibilistic version of the oracle (when performed exactly) while satisfying the requirements of decentralised inference. The context in [13] is restricted to the possibilistic Bernoulli filter [16], i.e., for a single target whose existence is uncertain, but the method has been showed to generalise to multiple targets in [17] via the possibilistic labelled multi-Bernoulli filter, which is also introduced in [17].

In this work, we consider limited FOVs in the context where a single target might or might not exist from the first time step onward. This is a simplified setting compared to the Bernoulli filter as there is no target birth, only a potential preexisting target. Yet, this is still a challenging scenario because of the importance of negative information which has to be represented compactly in order to being communicated efficiently on the network. This negative information comes from the fact that we often do not know where the target is at first and, in any case, we need to model that it might not have been observed at all because of the presence of false positives. Our objective is to find a way of handling the prediction, update and fusion of this negative information with as few approximations as possible in order to retain as much of the efficiency of decentralised fusion method of [13] while dealing with limited FOVs.

II. Possibility theory

This section provides a concise introduction to possibility theory; more information and insight about this theory in the context of target tracking can be found, e.g., in [18] and [13]. We will work on Euclidean spaces such as $\mathbb{R}^d$ for some $d > 0$ although generalisations to more sophisticated spaces are straightforward.

The objective of possibility theory is to represent *epistemic* uncertainty, that is, uncertainty which originates from a lack of knowledge rather than from randomness. This type of uncertainty is usually modelled by *subjective* probability distributions, with practical limitations emerging in cases where there is little to no information available, as exemplified by

the well-known paradoxes [19] arising when using improper prior distributions to model the total absence of information.

In possibility theory, probability distributions are replaced by possibility functions, which are non-negative functions with a maximum (or supremum) equal to 1. It can be proved [20] that these possibility functions behave similarly to probability distributions under suitable assumptions; in particular, Bayesian inference for hidden Markov models can be performed almost as usual, by simply replacing integrals with maxima, leading to an analogue of the Kalman filter [21] which enjoys the same computational efficiency and robust behaviour as its probabilistic counterpart.

To make statements about possibility theory easier to formulate, we introduce as is usual a notion of *uncertain variable* which stands as a replacement for random variables. If the quantity of interest is an unknown true state $x^\star \in \mathbb{R}^d$, then we represent the possible values that $x^\star$ take via a function $\boldsymbol{x} : \Omega \to \mathbb{R}^d$, with Ω the set of all possible states of the entire environment. We denote by $\omega^\star$ the true state of the environment so that $\boldsymbol{x}(\omega^\star) = x^\star$. The possibility function describing $\boldsymbol{x}$ is denoted $f_{\boldsymbol{x}}$ and the possibility of an event $\boldsymbol{x} \in B$ for some $B \subseteq \mathbb{R}^d$ is $\max_{x \in B} f_{\boldsymbol{x}}(x)$. As in the probabilistic context, an event with a possibility of 0 is known to be impossible but, as opposed to probabilities, an event with a possibility of 1 is an event against which there is no information. This difference between the two frameworks helps representing negative information, as required in the context of interest, by setting $f_{\boldsymbol{x}}(x) = 1$ for all the values $x \in \mathbb{R}^d$ about which there is no information. In the case where there is no information at all, the possibility function $f_{\boldsymbol{x}}$ can be set to 1 everywhere, which we denote by $f_{\boldsymbol{x}} \equiv 1$. Such an uninformative possibility function is proper and can be shown to avoid the above-mentioned paradoxes. Since possibility functions become less informative as they increase in value, we can define a partial order between them: If $f_{\boldsymbol{x}}$ and $\tilde{f}_{\boldsymbol{x}}$ are two possibility functions describing two different levels of information about $\boldsymbol{x}$,[1] then $f_{\boldsymbol{x}}$ can be said to be more informative than $\tilde{f}_{\boldsymbol{x}}$ if it holds that $f_{\boldsymbol{x}}(x) \leq \tilde{f}_{\boldsymbol{x}}(x)$ for all $x \in \mathbb{R}^d$. The inequality cannot be strict since it must hold that $\tilde{f}_{\boldsymbol{x}}(x) = 1$ whenever $f_{\boldsymbol{x}}(x) = 1$. This is practically relevant since one can replace $f_{\boldsymbol{x}}$ by $\tilde{f}_{\boldsymbol{x}}$ when the latter is less informative for reasons such as computational convenience, especially when $\tilde{f}_{\boldsymbol{x}}$ takes a simple parametric form. Although an information loss occurs in such a replacement, it does not introduce any positive information whatsoever.

If $\boldsymbol{x}$ and another uncertain variable $\boldsymbol{y}$ on $\mathbb{R}^{d'}$ are jointly described by a possibility function $f_{\boldsymbol{x},\boldsymbol{y}}$, then the *marginal* possibility function describing $\boldsymbol{x}$ can be obtained via

$$f_{\boldsymbol{x}}(x) = \max_{y \in \mathbb{R}^{d'}} f_{\boldsymbol{x},\boldsymbol{y}}(x, y), \qquad x \in \mathbb{R}^d,$$

and similarly for $f_{\boldsymbol{y}}$. Such a marginalisation allows for performing predictions when $\boldsymbol{x}$ and $\boldsymbol{y}$ represent the state of a target at different time steps. If it happens that $f_{\boldsymbol{x},\boldsymbol{y}}(x, y) = f_{\boldsymbol{x}}(x) f_{\boldsymbol{y}}(y)$ for all $(x, y) \in \mathbb{R}^d \times \mathbb{R}^{d'}$ then $\boldsymbol{x}$ and $\boldsymbol{y}$ are said to be *independent*. One can also obtain the posterior possibility function $f_{\boldsymbol{x}}(\cdot \,|\, \boldsymbol{y} = y)$, in which case y is typically considered as an observation. Instead, we will consider here a more sophisticated setting [20] where the observation process is random and characterised by a random variable $Y \,|\, \boldsymbol{x} = x$ on $\mathbb{R}^{d'}$ distributed according to $p(\cdot \,|\, x)$.[2] In this case, the posterior possibility function $f_{\boldsymbol{x}}(\cdot \,|\, Y = y)$ is characterised by

$$f_{\boldsymbol{x}}(x \,|\, Y = y) = \frac{p(y \,|\, x) f_{\boldsymbol{x}}(x)}{\max_{x' \in \mathbb{R}^d} p(y \,|\, x') f_{\boldsymbol{x}}(x')}, \qquad x \in \mathbb{R}^d,$$

where $f_{\boldsymbol{x}}$ describes $\boldsymbol{x}$. If $f_{\boldsymbol{x}}$ is the uninformative prior, i.e., $f_{\boldsymbol{x}} \equiv 1$, and if $p(\cdot \,|\, x)$ is a normal distribution with mean x and covariance Σ then the posterior possibility function is

$$\begin{aligned} f_{\boldsymbol{x}}(x \,|\, Y = y) &= \exp\left(-\frac{1}{2}(x - y)^\top \Sigma^{-1} (x - y) \right) \\ &=: \overline{\mathrm{N}}(x; y, \Sigma), \end{aligned}$$

where $\cdot^\top$ is the transpose and where $=:$ indicates that the r.h.s. is defined by the l.h.s. of the equation. The possibility function $\overline{\mathrm{N}}(\cdot\,; \mu, \Sigma)$ can be meaningfully referred to as the Gaussian possibility function with mean $\mu \in \mathbb{R}^d$ and covariance Σ as it satisfies very similar properties to its probabilistic counterpart. One such property is that of conjugacy: if the prior $f_{\boldsymbol{x}}$ is a normal possibility function and $p(\cdot \,|\, x)$ is a normal distribution of the form $\mathrm{N}(\cdot\,; Hx, \Sigma)$ for some $d' \times d$ matrix H, then the posterior $f_{\boldsymbol{x}}(\cdot \,|\, Y = y)$ is also a normal possibility function. The uninformative possibility function is part of this conjugate prior family, so that we can initialise the information about the target state to be uninformative about all or some of the components of the state.

III. The possibilistic Bernoulli filter

A. Original form

We consider the possibilistic Bernoulli filter (PBF) [16] as it allows, in principle, to perform optimal decentralised fusion [13]. More sophisticated methods such as the possibilistic labelled Bernoulli filter [17] will be considered in future work.

We assume that the target's state takes value in $\mathbb{R}^d$, for some $d > 0$. Since the state space is not bounded, it is not possible to define uniform prior probability distribution for the target, and negative information would be equally difficult to represent with a probabilistic approach. At time step $k - 1$, we consider an uncertain variable $\boldsymbol{X}_{k-1}$ defined on the set X of subsets of $\mathbb{R}^d$ with no more than one elements, i.e. $X \in \mathsf{X}$ is either the empty set of a set of the form $\{x\}$ for some $x \in \mathbb{R}^d$. The uncertain variable $\boldsymbol{X}_{k-1}$ models the fact that the target may or may not exist at time step $k-1$, with the event $\boldsymbol{X}_{k-1} = \emptyset$ corresponding to the case where it does not exist. Based on all the observations collected up to time step $k-1$, the information available about the uncertain variable $\boldsymbol{X}_{k-1}$

[1] This is meaningful since information about an unknown but fixed quantity, as opposed to the distribution of a random variable, is not unique.

[2] The notation $Y \,|\, \boldsymbol{x} = x$ implies that Y is indeed random once the deterministic but unknown quantity $\boldsymbol{x}$ is set to x.

can be described by a possibility function F_{k-1} on X of the form

$$F_{k-1}(X) = \begin{cases} \beta_{k-1} & \text{if } X = \emptyset \\ \alpha_{k-1} f_{k-1}(x) & \text{if } X = \{x\} \text{ for some } x \in \mathbb{R}^d. \end{cases}$$

By construction, it holds that f_{k-1} is a possibility function on $\mathbb{R}^d$ and that the set $\{(0, \beta_{k-1}), (1, \alpha_{k-1})\}$ is the graph of a possibility function describing the cardinality of $\boldsymbol{X}_{k-1}$ (ALTERNATIVE: the possibility function defined by $f(0) = \beta_{k-1}$, $f(1) = \alpha_{k-1}$ describes the cardinality of $\boldsymbol{X}_{k-1}$). In general, instead of defining F_{k-1} explicitly as above, we will simply say that F_{k-1} has parameters α_{k-1}, β_{k-1} and f_{k-1}.

To describe the possibility of birth and death of the target, we introduce a 2×2 transition matrix τ with τ_{ij} corresponding to the transition from cardinality i to cardinality j, e.g., τ_{10} corresponds to the possibility of a target death, which we assume to be such that $\tau_{10} \ll 1$. Since we are considering the case where there is no target birth, we set $\tau_{01} = 0$. It follows that $\tau_{00} = \tau_{11} = 1$, since it must hold that $\max\{\tau_{i0}, \tau_{i1}\} = 1$ for any $i \in \{0, 1\}$. It can then be shown that, under the considered assumptions, the *predicted* possibility function $F_{k|k-1}$ has parameters α_{k-1}, $\beta_{k|k-1} = \max\{\beta_{k-1}, \alpha_{k-1}\tau_{10}\}$ and

$$f_{k|k-1}(x) = \sup_{x' \in \mathbb{R}^d} g(x \,|\, x') f_{k-1}(x'),$$

with $g(\cdot \,|\, x')$ the possibility function describing the target's state transition.

When receiving the next observation set Y_k, we compute the *updated* possibility function F_k, with parameters

$$\beta_k \propto \kappa(Y_k)\beta_{k|k-1} \tag{1}$$

$$\alpha_k \propto \alpha_{k-1} \sup_{x \in \mathbb{R}^d} \ell(Y_k \,|\, x) f_{k|k-1}(x) \tag{2}$$

$$f_k(x) = \frac{\ell(Y_k \,|\, x) f_{k|k-1}(x)}{\sup_{x \in \mathbb{R}^d} \ell(Y_k \,|\, x) f_{k|k-1}(x)}, \tag{3}$$

where κ is the distribution of the set of false alarms and where

$$\ell(Y_k \,|\, x) = \max\left\{\beta_{\mathrm{d}}(x)\kappa(Y_k), \max_{y \in Y_k} \alpha_{\mathrm{d}}(x) p(y \,|\, x) \kappa(Y_k \backslash \{y\})\right\},$$

with $\alpha_{\mathrm{d}}(x)$ and $\beta_{\mathrm{d}}(x)$ defining a binary possibility function corresponding to detection and detection failure, respectively, and with $p(\cdot \,|\, x)$ the probability distribution characterising the (conditionally random) observation process. We assume throughout that the observation space is $\mathbb{R}^{d'}$, for some $d' > 0$.

To implement this recursion in the linear-Gaussian case, we assume that f_{k-1} takes the form of a Gaussian max-mixture, that is

$$f_{k-1}(x) = \max_{i \in \{1, \ldots, N_{k-1}\}} w_{k-1,i} \overline{\mathrm{N}}(x; \mu_{k-1,i}, P_{k-1,i}) \tag{4}$$

where, for each index $i \in \{1, \ldots, N_{k-1}\}$, $w_{k-1,i}$ is a non-negative weight, $\mu_{k-1,i}$ is a vector in $\mathbb{R}^d$ and $P_{k-1,i}$ is a positive semi-definite matrix. Under some additional assumptions, e.g., supposing that $\beta_{\mathrm{d}}(x)$ is in fact constant, the prediction and update steps can be found to be closed-form, i.e., the predicted and updated possibility functions $f_{k|k-1}$ and f_k are also Gaussian max-mixtures.

B. Decentralised fusion for the PBF

We consider a network with n sensor nodes S_i, $i \in \{1, \ldots, n\}$. The objective in this section is to track the target in a decentralised manner by processing the observation at each sensor node before exchanging the obtained posterior possibility functions on the network and performing fusion at each node. The communication and fusion steps are repeated L times in order to ensure that the information from all sensors has reached all nodes. The properties of possibility functions allow to recover the possibilistic oracle when L goes to infinity [13], a result that is unique to this approach. Examples of these properties are:

1) if f is a possibility function then f^ω is also a possibility function for any $\omega \in [0, 1]$
2) in particular, the normal possibility function is closed under power, i.e., $\overline{\mathrm{N}}(x; \mu, \Sigma)^\omega = \overline{\mathrm{N}}(x; \mu, \Sigma/\omega)$
3) if f is a possibility function representing some information, then f^ω and $f^{1-\omega}$ are two independent possibility functions, i.e., there is no redundancy between the information in f^ω and the one in $f^{1-\omega}$

These properties can be used to ensure the independence of the information maintained by each sensor node, by following two principles:

1) The Markov transition at S_i allowing to predict $\boldsymbol{X}_k$ given $\boldsymbol{X}_{k-1}$ is discounted by raising it to the power of $\omega_i \in [0, 1]$, with $\sum_{i=1}^n \omega_i = 1$.
2) If $F_k^{(i)}$ is the posterior possibility function obtained at sensor node S_i, $i \in \{1, \ldots, n\}$, and if $\mathcal{N}_i$ is the set of neighbours of S_i, then S_i communicates $(F_k^{(i)})^{\gamma_j}$ to the neighbour $j \in \mathcal{N}_i$, with $\sum_{j \in \mathcal{N}_i} \gamma_j = 1$.

Additional motivation, details and proofs can be found in [13].

IV. Fusion with limited FOVs

The Gaussian mixture implementation mentioned in Section III-A assumes that the possibility of detection failure $\beta_{\mathrm{d}}(x)$ is constant. This is not often the case in a multi-sensor context since sensors generally do not have a FOV that covers the entire area of interest, and rarely have a constant probability of detection over their FOV in any case. In addition, the probability of detection is often assumed to be approximately constant across the likely states of a target in a given term of the underlying Gaussian mixture. However, that would not the case for a term corresponding to a potential target that has never been detected since the beginning of the scenario, especially in cases where the initial information about the target location is scarce or unavailable. Indeed, in these situations, the corresponding possibility functions would have to model negative information, by representing the fact that the target can be anywhere *but* close to a sensor. The objective in this section is to devise a way to represent and approximately propagate such negative information.

In the remainder of this section, we distinguish between a *component* of a vector, e.g., the first component x_i of the vector $x \in \mathbb{R}^d$ for some $i \in \{1, \ldots, d\}$, and a *mixture term*, e.g., a term $w_i \overline{\mathrm{N}}(x; \mu_{k-1,i}, P_{k-1,i})$ of the mixture (4),

though we will also consider non-Gaussian mixture terms later on. In addition, we distinguish between *observable/hidden components* of the state x, i.e., components that are/aren't measured by an observation y, and *observed/unobserved mixture terms*, i.e., mixture terms that have/haven't been updated by an observation at least once throughout the previous time steps.

For a given sensor $s \in \{1, \dots, n\}$, we consider a probability of detection of the form $p_{\mathrm{d}}(x) = p_{\mathrm{d,max}}\overline{\mathrm{N}}(Hx; Hx_s, \sigma^2_{\mathrm{FOV}} I_{d'})$ where $x_s \in \mathbb{R}^d$ is the state of Sensor s and $I_{d'}$ is the identity matrix of dimension d'. Considering an arbitrary diagonal matrix as covariance matrix would be a straightforward generalisation. In addition, we assume that the initial possibility function, i.e., the possibility function at time $k = 1$ when no observation is available, denoted $f_{1|0}$, is a product of two possibility functions $f^{\mathrm{p}}_{1|0}$ and $f^{\mathrm{v}}_{1|0}$, one on the observable components $p \in \mathbb{R}^{d_{\mathrm{p}}}$ of x (positions in our context) and one on the hidden components $v \in \mathbb{R}^{d_{\mathrm{v}}}$ of x (velocities in our context), with $d_{\mathrm{p}} + d_{\mathrm{v}} = d$. In particular, we assume that $f^{\mathrm{p}}_{1|0} \equiv 1$. We also assume that the possibility of detection is $\alpha_{\mathrm{d}} \equiv 1$, and that $\beta_{\mathrm{d}}(x) = 1 - p_{\mathrm{d}}(x)$ for any $x \in \mathbb{R}^d$. Under these assumptions, a subsequent possibility function f_{k-1} cannot easily be assumed to be a Gaussian max-mixture and, instead, we assume that it is of the form $f_{k-1}(x) = \max\{w_{\mathrm{u},k-1} f_{\mathrm{u},k-1}(x), f_{\mathrm{o},k-1}(x)\}$, where $w_{\mathrm{u},k-1} \in [0, 1]$, where $f_{\mathrm{o},k-1}$ corresponds to observed mixture terms (detected at least once in times steps up to time $k-1$) and $f_{\mathrm{u},k-1}$ corresponds to unobserved mixture terms, i.e., those that have never been detected up to time $k-1$. The possibility function $f_{\mathrm{o},k-1}$ is assumed to be a Gaussian max-mixture, i.e., it is of the form

$$f_{\mathrm{o},k-1}(x) = \max_{i \in \{1,\dots,N_{k-1}\}} w_{k-1,i}\overline{\mathrm{N}}(x; \mu_{k-1,i}, P_{k-1,i}),$$

whereas $f_{\mathrm{u},k-1}$ is assumed to be of a similar product form as $f_{1|0}$. In particular, we assume that $f_{\mathrm{u},k-1}(x) = f^{\mathrm{p}}_{\mathrm{u},k-1}(p) f^{\mathrm{v}}_{\mathrm{u},k-1}(v)$, with $f^{\mathrm{v}}_{\mathrm{u},k-1}(v) = \overline{\mathrm{N}}(v; 0, \sigma^2_{\mathrm{v}} I_{d_{\mathrm{v}}})$, and with

$$\begin{aligned} f^{\mathrm{p}}_{\mathrm{u},k-1}(p) &= 1 - \max_{i \in \{1,\dots,\breve{N}_{k-1}\}} \breve{w}_{k-1,i}\overline{\mathrm{N}}(p; \breve{\mu}_{k-1,i}, \breve{\sigma}^2_{k-1,i} I_{d_{\mathrm{p}}}) \\ &= \min_{i \in \{1,\dots,\breve{N}_{k-1}\}} 1 - \breve{w}_{k-1,i}\overline{\mathrm{N}}(p; \breve{\mu}_{k-1,i}, \breve{\sigma}^2_{k-1,i} I_{d_{\mathrm{p}}}), \end{aligned}$$

which can be seen as an "inverted" Gaussian max-mixture. This form is useful to represent *negative* information, e.g., when we only know that a target is unlikely to be close to a sensor but could be anywhere else otherwise. However, the usual Gaussian (max-)mixture properties regarding prediction and update in the linear-Gaussian case cease to hold with such an inverted mixture and approximations have to be introduced. Since we will assume the independence of the components in the possibility functions related to unobserved mixture terms, one-dimensional techniques will be sufficient. In particular, for any term of the form $1 - \breve{w}\overline{\mathrm{N}}(p; \breve{\mu}, \breve{\sigma}^2 I_{d_{\mathrm{p}}})$, we will treat each marginal separately, that is for each component $j \in \{1, \dots, d_{\mathrm{p}}\}$, we will consider the one-dimensional possibility function $f_{\mathrm{p},j}(p_j) = 1 - \breve{w}\overline{\mathrm{N}}(p_j; \breve{\mu}_j, \breve{\sigma}^2)$. We then define support points $p_{j,1}, \dots, p_{j,N}$ and approximate $f_{\mathrm{p},j}$ via the set of pairs $\{(p_{j,l}, \alpha_{j,l})\}_{l=1}^N$ with $\alpha_{j,l} = f_{\mathrm{p},j}(p_{j,l})$, as shown in Figure 1a where one support point is placed at $\breve{\mu}_j$ and the others are placed arbitrarily at $\breve{\mu}_j \pm \breve{\sigma}$ and $\breve{\mu}_j \pm 2\breve{\sigma}$. Although very crude, this type of approximation will prove sufficient for our purpose. Similar schemes based on a larger number of random support points [22] could also be considered. Based on the support points $\{(p_{j,l}, \alpha_{j,l})\}_{l=1}^N$, we can recover the possibility function $f_{\mathrm{p},j}$ *exactly*, as long as one of the support points is at $\breve{\mu}_j$, via the optimisation problem

$$\begin{aligned} &\min \tau \qquad (5) \\ &\text{subject to } 1 - \breve{w}\overline{\mathrm{N}}(p_{j,l}; \breve{\mu}_j, \tau^{-1}) \geq \alpha_{j,l} \text{ for } l \in \{1, \dots, N\}, \end{aligned}$$

where τ is the inverse variance, also known as the precision. The quantities $\breve{w}$ and $\breve{\mu}_j$ can be recovered from $\{(p_{j,l}, \alpha_{j,l})\}_{l=1}^N$ as $\breve{w} = 1 - \alpha_{j,l^*}$ and $\breve{\mu}_j = p_{j,l^*}$ with $l^* = \arg\min_l \alpha_{j,l}$. The optimisation problem (5) can be solved easily as

$$\tau^* = \max_{l \neq l^*} \frac{2(\log \breve{w} - \log(1 - \alpha_{j,l}))}{(p_{j,l} - \breve{\mu}_j)^2}. \qquad (6)$$

If the components of p were not independent, the optimisation problem to find the precision matrix, i.e. the inverse covariance matrix, would require to solve a (convex) optimisation problem on the cone of positive definite matrices, which would require a proper optimisation procedure [23], as opposed to simply evaluating $N - 1$ quantities as in (6).

A. Prediction

The prediction of terms in $f_{\mathrm{o},k-1}$ can be carried out as usual. For terms in $f_{\mathrm{u},k-1}$, we first have to derive a suitable motion model that preserves the independence between position and velocity. To this end, we consider a specific motion model $g(x \,|\, x') = \overline{\mathrm{N}}(x; Gx', Q)$ with

$$G = I_2 \otimes \begin{bmatrix} 1 & \Delta \\ 0 & 1 \end{bmatrix} \quad \text{and} \quad Q = \sigma^2 I_2 \otimes \begin{bmatrix} \Delta^4/4 & \Delta^3/2 \\ \Delta^3/2 & \Delta^2 \end{bmatrix},$$

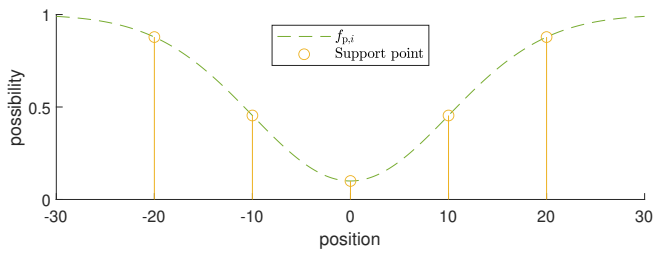
where Δ is the time step. We derive the marginal motion models for the positions and velocities respectively as

$$\begin{aligned} g_{\mathrm{p}}(p \,|\, p') &= \sup_{v,v'} \overline{\mathrm{N}}(x; Gx', Q) f^{\mathrm{v}}_{\mathrm{u},k-1}(v) \\ &= \overline{\mathrm{N}}(p; p', \sigma^2_{\mathrm{p}} I_2) \\ g_{\mathrm{v}}(v \,|\, v') &= \sup_{p,p'} \overline{\mathrm{N}}(x; Gx', Q) f^{\mathrm{p}}_{\mathrm{u},k-1}(p) \\ &= \overline{\mathrm{N}}(v; v', \Delta^2 I_2), \end{aligned}$$

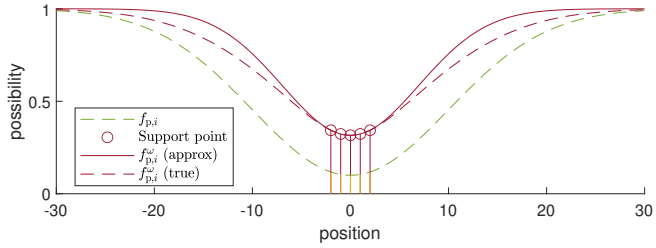
where $\sigma^2_{\mathrm{p}} = \Delta^2 \sigma^2_{\mathrm{v}} + \sigma^2 \Delta^4/4$ and where x and x' are implicit functions of p, v and p', v' respectively. Since the components of $g_{\mathrm{p}}(\cdot \,|\, p')$ are independent, we can easily get the Markov transition of the j-th component of p, $j = 1, 2$, as the one-dimensional Markov transition $g_{\mathrm{p},j}(p_j \,|\, p'_j) = \overline{\mathrm{N}}(p_j; p'_j, \sigma^2_{\mathrm{p}})$.

In order to apply the marginal Markov transition $g_{\mathrm{p}}(\cdot \,|\, p')$ to the possibility function $f^{\mathrm{p}}_{\mathrm{u},k-1}$, we can use a support-point approximation $\{(p'_{i,j,l}, f_{\mathrm{p},i,j}(p'_{i,j,l}))\}_{l=1}^N$ for the j-th component $f_{\mathrm{p},i,j}$ of the i-th Gaussian term $f_{\mathrm{p},i} = 1 - \breve{w}_{k-1,i}\overline{\mathrm{N}}(\breve{\mu}_{k-1,i}, \breve{\sigma}^2_{k-1,i} I_{d_{\mathrm{p}}})$ of $f^{\mathrm{p}}_{\mathrm{u},k-1}$. Applying the one-dimensional Markov transition $g_{\mathrm{p},j}(\cdot \,|\, p'_j)$ step to $f_{\mathrm{p},i}$ gives

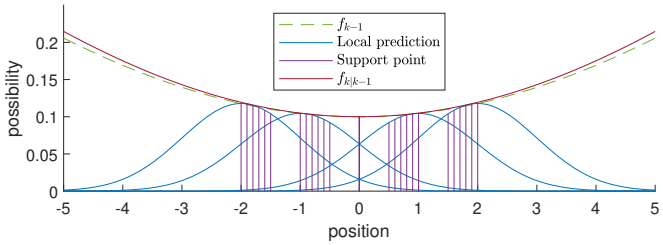
$$\max_{l \in \{1,\dots,N\}} f_{\mathrm{p},i,j}(p'_{i,j,l})\overline{\mathrm{N}}(p_j; p'_{i,j,l}, \sigma^2_{\mathrm{p}}), \qquad (7)$$



(a) Approximation of an unobserved mixture term



(b) Discount of an unobserved mixture term



(c) Prediction for an unobserved mixture term

Fig. 1: Approximation, discount and prediction mechanisms for a one dimensional inverted Gaussian of the form $1 - 0.9\overline{\mathrm{N}}(\cdot\,; 0, 100)$.

which is a Gaussian max-mixture as shown in blue in Figure 1c. To obtain an approximation for the j-th mixture term of the predicted possibility function $f^{\mathrm{p}}_{\mathrm{u},k|k-1}$ corresponding to $f^{\mathrm{p}}_{\mathrm{u},k-1}$, we once more define support points for each term in the mixture (7) and define the set of weighted support point $\{(p_{i,j,l}, \alpha_{i,j,l})\}_{l=1}^{N^2}$ as the union of all obtained sets of points, hence the total of N^2 points. We then fit a new Gaussian possibility function via an optimisation of the same form as (5) to obtain the predicted possibility function as shown in red in Figure 1c. The prediction for the hidden components of $f_{\mathrm{u},k|k-1}$ is standard, that is $f^{\mathrm{v}}_{\mathrm{u},k|k-1}(v) = \max_{v' \in \mathbb{R}^{d_{\mathrm{v}}}} g_{\mathrm{v}}(v \,|\, v') f^{\mathrm{v}}_{\mathrm{u},k|k-1}(v')$.

B. Update and fusion

Update and fusion can be treated in very similar ways, which makes the following generalisation useful: in both cases, we multiply together mixture terms of two distinct forms, some being weighted Gaussian possibility functions (with relatively small variances) and some being inverted Gaussian possibility functions at the level of the observable components (with relatively large variance). We define these terms in general as

$$f_{i,\mathrm{o}}(x) = w_i \overline{\mathrm{N}}(x; \mu_i, P_i)$$

and $f_{i,\mathrm{u}}(x) = f^{\mathrm{p}}_{i,\mathrm{u}}(p) f^{\mathrm{v}}_{i,\mathrm{u}}(v)$ with

$$\begin{aligned} f^{\mathrm{p}}_{i,\mathrm{u}}(p) &= \min_{j \in \{1,\ldots,\breve{N}\}} 1 - \breve{w}_{i,j} \overline{\mathrm{N}}(p; \breve{\mu}_{i,j}, \breve{\sigma}^2_{i,j} I_{d_{\mathrm{p}}}) \\ &=: \min_{j \in \{1,\ldots,\breve{N}\}} f^{\mathrm{p}}_{i,j} \end{aligned}$$

and $f^{\mathrm{v}}_{\mathrm{u}}(v) = \overline{\mathrm{N}}(v; \mu_{i,\mathrm{v}}, P_{i,\mathrm{v}})$, with $i = 1, 2$. The above-mentioned assumption about the variance can be translated as $\det(H P_i H^\top) \ll \det(\breve{\sigma}^2_{i',j} I_{d_{\mathrm{p}}}) = \breve{\sigma}^{2d_{\mathrm{p}}}_{i',j}$ for any $i, i' \in \{1, 2\}$ and any $j \in \{1, \ldots, \breve{N}\}$, based on which we make the approximation

$$f_{i,\mathrm{o}}(x) f_{i',\mathrm{u}}(x) \approx f^{\mathrm{p}}_{i',\mathrm{u}}(\mu^{\mathrm{p}}_i) f_{i,\mathrm{o}}(x) f^{\mathrm{p}}_{i',\mathrm{u}}(v) \tag{8}$$

where μ^{p}_i is the vector of observable components of μ_i. This approximation is based on the fact that $f_{j,\mathrm{u}}$ is nearly constant where $f_{i,\mathrm{o}}$ is non-negligible. The product $f_{1,\mathrm{o}}(x) f_{2,\mathrm{o}}(x)$ is a weighted Gaussian possibility function whose expected value and variance can be computed exactly. For products of the form $f^{\mathrm{p}}_{1,\mathrm{u}}(p) f^{\mathrm{p}}_{2,\mathrm{u}}(p)$, we assume that one of the two terms can be treated as a reference, say $f^{\mathrm{p}}_{1,\mathrm{u}}$, and support points $\{(p_{1,j,l}, f^{\mathrm{p}}_{1,j}(p_{1,j,l}))\}_{l=1}^{N}$ can be introduced for each mixture term $f^{\mathrm{p}}_{1,j}$, $j \in \{1, \ldots, \breve{N}\}$. The weight is then adjusted to $f^{\mathrm{p}}_{2,\mathrm{u}}(p_{j,l}) f^{\mathrm{p}}_{1,j}(p_{1,j,l})$ and a new inverted Gaussian is refitted via (5).

In the case of an update, we can set $f_{1,\mathrm{o}}$ as one of the mixture terms in the predicted possibility function $f_{\mathrm{o},k|k-1}$ and $f_{2,\mathrm{o}}$ as the information produced by the observation y, that is

$$f_{2,\mathrm{o}}(x) = \frac{p(y \,|\, x)}{\max_{x' \in \mathbb{R}^{d'}} p(y \,|\, x')}.$$

We can also set $f^{\mathrm{p}}_{1,\mathrm{u}}$ as one of the mixture terms in the predicted possibility function $f^{\mathrm{p}}_{\mathrm{u},k|k-1}$ and $f^{\mathrm{p}}_{2,\mathrm{u}}$ as the possibility of detection failure β_{d} (which does not depend on v). For instance, in this context, (8) corresponds to the usual approach of considering the probability of detection constant over the region where the target is likely to be.

In the case of fusion, all the considered terms are simply discounted mixture terms from different sensors. However, to implement a discount on an inverted Gaussian possibility function, we resort once more to the support point method, as illustrated in Figure 1b.

C. Algorithm

The proposed method is summarised in Algorithms 1, 2 and 3, which are adapted from their original form, given in [13]. In Algorithm 1, the $n \times n$ matrix Γ is such that $\Gamma_{i,j}$ is the discount when node S_i sends information to node S_j, $i, j \in \{1, \ldots, n\}$. In Algorithm 2, each observation in the set $\{y_1, \ldots, y_M\}$ is assumed to be drawn from the normal distribution $\mathrm{N}(\cdot\,; Hx, R)$ when the target is at state $x \in \mathbb{R}^d$, with H the observation matrix and R the covariance matrix of the observation noise. Also, σ_{v} and $f^{\mathrm{p}}_{\mathrm{u},j}$ are implicitly induced by the possibility function $f_{\mathrm{u},j}$, with the former being the standard deviation of the hidden components and the latter being the possibility function describing the observable

Algorithm 1 Decentralised fusion at the sensor node S_i

Input: Sequence of observation sets $(Y_k)_{k=1}^K$ at S_i; Total number of time steps K; Discount factor $\omega_i \in [0,1]$; Weight matrix Γ; Number of network iterations L

$N \leftarrow 0$
$\hat{\alpha} \leftarrow 1$, $\hat{\beta} \leftarrow 1$
$w_\mathrm{u} \leftarrow 1$
for all $k \in \{1, \dots, K\}$ **do**
  # Discounted prediction
  $\alpha \leftarrow \hat{\alpha}$
  $\beta \leftarrow \max\{\hat{\beta}, \hat{\alpha}\tau_{10}^{\omega_i}\}$
  for all $j \in \{1, \dots, N\}$ **do**
    $\mu_j \leftarrow G\mu_i$
    $P_j \leftarrow GP_jG^\top + Q/\omega_i$
  end for
  for all $j \in \{1, \dots, \breve{N}\}$ **do**
    $f_{\mathrm{u},j} \leftarrow \mathrm{PredictUnobserved}(f_{\mathrm{u},j})$
  end for
  # Update
  $F^{(i)} \leftarrow$ $\mathrm{LocalUpdate}\big(Y_k, \alpha, \beta, (w_j, \mu_j, P_j)_{j=1}^N, w_\mathrm{u}, (f_{\mathrm{u},j})_{j=1}^{\breve{N}}\big)$
  # Fusion
  for all $l \in \{1, \dots, L\}$ **do**
    Broadcast $F^{(i)}$
    Receive $\{F^{(i')} : i' \in \mathcal{N}_i\}$ from neighbours
    for all $i' \in \mathcal{N}_i$ **do**
      $F^{(i)} \leftarrow \mathrm{Fusion}(F^{(i)}, F^{(i')}, \Gamma_{i,i}, \Gamma_{i,i'})$
    end for
  end for
  # Unpack information after fusion
  $\big(\hat{\alpha}, \hat{\beta}, (w_j, \mu_j, P_j)_{j=1}^N, w_\mathrm{u}, (f_{\mathrm{u},j})_{j=1}^{\breve{N}}\big) \leftarrow F^{(i)}$
end for

components. Finally, $0_{1,d_\mathrm{v}}$ denotes the 0 row vector of dimension d_v. In all three algorithms, the functions related to the unobserved components such as "PredictUnobserved()" and "MultiplyUnobserved()" are not given explicitly as they correspond to the methods detailed in Section IV-A and IV-B.

Since we consider that the state space is the whole of $\mathbb{R}^d$, the product between two inverted Gaussian possibility functions will always have a supremum equal to 1, which simplifies some of the calculations.

V. Experimental results

Throughout this section, the units are the ones of the international system, e.g., meters and seconds, and are therefore omitted. To assess the performance of the proposed method, we consider a scenario with $K = 50$ time steps of duration $\Delta = 1$, where the target evolves according to the Markov transition $\mathrm{N}(\cdot\,; Gx, Q)$, with G and Q as defined in Section IV, with $\sigma^2 = 1$. The observation matrix is

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix},$$

Algorithm 2 $\mathrm{LocalUpdate}(Y, F)$

Input: Observation set $Y = \{y_1, \dots, y_M\}$; Possibility function under the form $F = \big(\alpha, \beta, (w_j, \mu_j, P_j)_{j=1}^N, w_\mathrm{u}, (f_{\mathrm{u},j})_{j=1}^{\breve{N}}\big)$

for all $j \in \{1, \dots, N\}$ **do**
  # Detection failure
  $\hat{w}_j \leftarrow \beta_\mathrm{d}\kappa(Y)w_j$
  $\hat{\mu}_j \leftarrow \mu_j$
  $\hat{P}_j \leftarrow P_j$
  for all $m \in \{1, \dots, M\}$ **do**
    # Detection of jth component by m-th observation
    $j' \leftarrow mN + j$
    $c_{m,j} \leftarrow |2\pi R|^{-1/2}\overline{\mathrm{N}}(y_m; H\mu_j, HP_jH^\top + R)$
    $\hat{w}_{j'} \leftarrow c_{m,j}\alpha_\mathrm{d}\kappa(Y \setminus \{y_m\})w_j$
    $(\hat{\mu}_{j'}, \hat{P}_{j'}) \leftarrow \mathrm{KalmanUpdate}(\mu_j, P_j)$
  end for
end for
 # First observation of the target
for all $m \in \{1, \dots, M\}$ **do**
  $j' \leftarrow (M+1)N + m$
  $\hat{w}_{j'} \leftarrow 1$
  for all $j \in \{1, \dots, \breve{N}\}$ **do**
    $\hat{w}_{j'} \leftarrow \min\big\{\hat{w}_{j'}, f_{\mathrm{u},j}^\mathrm{p}(y_m)\big\}$
  end for
  $\hat{w}_{j'} \leftarrow |2\pi R|^{-1/2}w_\mathrm{u}\hat{w}_{j'}$
  $\hat{\mu}_{j'} \leftarrow [y_m^\top, 0_{1,d_\mathrm{v}}]^\top$
  $\hat{P}_{j'} \leftarrow \mathrm{blockdiag}(R, \sigma_\mathrm{v}^2 I_{d_\mathrm{v}})$
end for
for all $j \in \{1, \dots, \breve{N}\}$ **do**
  $\tilde{f}_{\mathrm{u},j} \leftarrow \mathrm{MultiplyUnobserved}(f_{\mathrm{u},j}, \beta_\mathrm{d})$
end for
 # Renormalise weights
$u \leftarrow \max\{w_\mathrm{u}, \max_j \hat{w}_j\}$
$\tilde{w}_\mathrm{u} \leftarrow \hat{w}_\mathrm{u}/u$
$(\hat{w}_j)_{j=1}^{(M+1)N} \leftarrow (\hat{w}_j)_{j=1}^{(M+1)N}/u$
 # Pruning and merging
$(\tilde{w}_j, \tilde{\mu}_j, \tilde{P}_j)_{j=1}^N \leftarrow \mathrm{Reduction}\big((\hat{w}_j, \hat{\mu}_j, \hat{P}_j)_{j=1}^{(M+1)N}\big)$
 # Update possibility of non-existence
$\tilde{\alpha} \leftarrow u\alpha/\max\{u\alpha, \kappa(Y)\beta\}$
$\tilde{\beta} \leftarrow \kappa(Y)\beta/\max\{u\alpha, \kappa(Y)\beta\}$

Output: $F = \big(\tilde{\alpha}, \tilde{\beta}, (\tilde{w}_j, \tilde{\mu}_j, \tilde{P}_j)_{j=1}^N, \tilde{w}_\mathrm{u}, (\tilde{f}_{\mathrm{u},j})_{j=1}^{\breve{N}}\big)$

and the covariance of the observation noise is $R = 5^2 I_2$. In the proposed approach, the only information about the target at the first time step is about its velocity, described by $\overline{\mathrm{N}}(\cdot\,; 0, \sigma_\mathrm{v}^2 I_2)$, with $\sigma_\mathrm{v} = 5$. In the generated scenarios, the initial location of the target is actually always in the middle of the scene, at $[500, 500]$, the magnitude of its initial velocity is fixed to 10 and its direction is drawn uniformly at random between 0 and 2π. The target is then propagated until its time of death, at step 45, whereas the possibility of death is set to $\tau_{10} = 0.01$. The sensors are fixed at locations $[200, 200]$, $[200, 800]$, $[800, 200]$

and $[800, 800]$, and their FOV is as assumed above via the definition of p_{d}, with $\sigma_{\mathrm{FOV}} = 200$ and with $p_{\mathrm{d,max}} = 0.9$. A realisation of such a scenario is shown in Figure 2. The number of false alarms is Poisson distributed with parameter 5 and their location is drawn from the probability distribution with the same shape as p_{d}, that is $\mathrm{N}(\cdot\,; Hx_s, \sigma^2_{\mathrm{FOV}} I_2)$ for the s-th sensor, $s \in \{1, \ldots, n\}$; indeed, it makes sense that there are less false alarms, e.g., due to background objects, where the probability of detection is low. The terms of the observed max-mixture are pruned when their weight falls below 10^{-3} and are merged when the possibilistic Hellinger distance [18] between them falls below 0.1.

The corresponding OSPA metric [24], averaged over 50 Monte Carlo runs, is shown in Figure 3. It confirms that the proposed method, which is decentralised and exploits no particular information about the initial target location, can do nearly as well as the probabilistic and possibilistic oracles. These methods are called oracles in this context for two reasons:

1) All the observations from all sensors are collected at each time step and processed in one central node, i.e., these methods are *not distributed*.
2) These methods are given information about the location of the target at the first time step under the form of a normal possibility function / probability distribution with expected value $[500, 500, 0, 0]^\top$ and covariance matrix $\mathrm{diag}([25, 25, 5, 5])^2$.

The difference between the two oracles could be due to differences in parameterisation, especially at the level of the confirmation threshold: The possibilistic method confirm the presence of the target if $\beta_k < 0.05$, i.e., if there is strong evidence that the target is not absent, while the probabilistic oracle uses a threshold of 0.1 for the highest weight in the underlying Gaussian mixture. This very low threshold for the probabilistic case is required because of the low probabilities of detection occurring in the studied scenario, penalising updates even when the model predicts the observation very well. This is not an issue with possibilistic methods since the possibility of detection is always equal to 1. The use of such a low threshold in the probabilistic oracle means that the algorithm takes time to detect the death of the target at time step 45. A higher threshold would shift the problem to the initialisation.

It was shown in [13] that the decentralised PBF could very closely match the performance of the oracle, as opposed to standard geometric/arithmetic average methods such as [12], [25]; these methods will be further affected by the low probabilities of detection that occur in the studied scenario, and are therefore not considered here.

VI. Conclusion

By introducing general principles for representing negative information in tracking, and by applying these principles in the context of multi-sensor distributed fusion, we have shown that it is possible to retain a very strong performance despite the distributed setting and the realistic observation conditions.

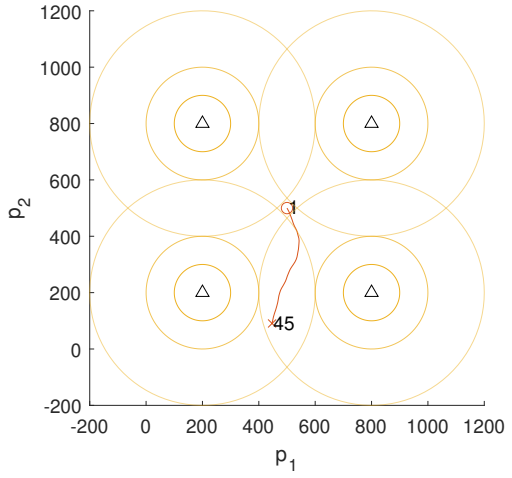


Fig. 2: A realisation of the considered scenario as a function of the two positional components p_1 and p_2 with triangles denoting sensor location, yellow concentric circles denoting the 0.5, 1 and 2 standard deviations contours, with the red circle and red cross denoting the location of at the first time step and at the time of death, respectively, and with the red line denoting the target trajectory.

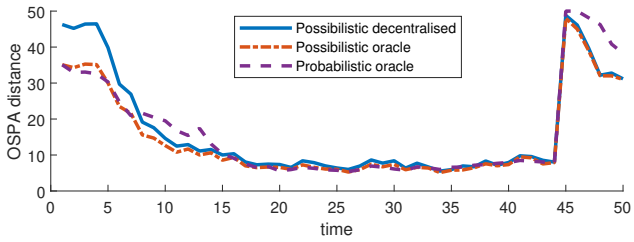


Fig. 3: OSPA distance for the proposed method, "Possibilistic decentralised", compared to the possibilistic and probabilistic oracles, averaged over 50 Monte Carlo runs.

Indeed, we considered limited fields of view induced by an exponentially decaying probability of detection, which are difficult to handle despite being common with real sensors. Future work will consider more sophisticated settings with a general target birth scheme as well as an unknown number of targets.

References

[1] W. Yi, G. Li, and G. Battistelli, "Distributed multi-sensor fusion of PHD filters with different sensor fields of view," *IEEE Transactions on Signal Processing*, vol. 68, pp. 5204–5218, 2020.

[2] A. K. Gostar, T. Rathnayake, R. Tennakoon, A. Bab-Hadiashar, G. Battistelli, L. Chisci, and R. Hoseinnezhad, "Centralized cooperative sensor fusion for dynamic sensor network with limited field-of-view via labeled multi-bernoulli filter," *IEEE Transactions on Signal Processing*, vol. 69, pp. 878–891, 2020.

[3] G. Li, G. Battistelli, W. Yi, and L. Kong, "Distributed multi-sensor multi-view fusion based on generalized covariance intersection," *Signal processing*, vol. 166, p. 107246, 2020.

[4] H. Van Nguyen, H. Rezatofighi, B.-N. Vo, and D. C. Ranasinghe, "Distributed multi-object tracking under limited field of view sensors," *IEEE Transactions on Signal Processing*, vol. 69, pp. 5329–5344, 2021.

[5] G. Li, G. Battistelli, L. Chisci, W. Yi, and L. Kong, "Distributed multi-view multi-target tracking based on CPHD filtering," *Signal Processing*, vol. 188, p. 108210, 2021.

[6] B. Wang, S. Li, G. Battistelli, L. Chisci, and W. Yi, "Multi-agent fusion with different limited fields-of-view," *IEEE Transactions on Signal Processing*, vol. 70, pp. 1560–1575, 2022.

[7] M. E. Liggins, C.-Y. Chong, I. Kadar, M. G. Alford, V. Vannicola, and S. Thomopoulos, "Distributed fusion architectures and algorithms for target tracking," *Proc. of the IEEE*, vol. 85, no. 1, pp. 95–107, 1997.

[8] D. Hall, C.-Y. Chong, J. Llinas, and M. Liggins II, *Distributed data fusion for network-centric operations*. Crc Press, 2017.

[9] S. J. Julier and J. K. Uhlmann, "A non-divergent estimation algorithm in the presence of unknown correlations," in *Proceedings of the 1997 American Control Conference*, vol. 4, pp. 2369–2373, IEEE, 1997.

[10] L. Chen, P. O. Arambel, and R. K. Mehra, "Estimation under unknown correlation: Covariance intersection revisited," *IEEE Transactions on Automatic Control*, vol. 47, no. 11, pp. 1879–1882, 2002.

[11] M. Üney, D. E. Clark, and S. J. Julier, "Distributed fusion of PHD filters via exponential mixture densities," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 3, pp. 521–531, 2013.

[12] T. Li, Z. Liu, and Q. Pan, "Distributed Bernoulli filtering for target detection and tracking based on arithmetic average fusion," *IEEE Signal Processing Letters*, vol. 26, no. 12, pp. 1812–1816, 2019.

[13] J. Houssineau, H. Cai, M. Uney, and E. Delande, "A possibilistic framework for multi-target multi-sensor fusion," *arXiv preprint arXiv:2209.12245*, 2022.

[14] L. A. Zadeh, "Fuzzy sets as a basis for a theory of possibility," *Fuzzy sets and systems*, vol. 1, no. 1, pp. 3–28, 1978.

[15] D. Dubois and H. Prade, "Possibility theory and its applications: Where do we stand?," in *Springer Handbook of Computational Intelligence*, pp. 31–60, Springer, 2015.

[16] B. Ristic, J. Houssineau, and S. Arulampalam, "Target tracking in the framework of possibility theory: The possibilistic Bernoulli filter," *Information Fusion*, vol. 62, pp. 81–88, 2020.

[17] H. Cai, C. Xue, J. Houssineau, and Z. Xue, "Robust multi-sensor multi-target tracking using possibility labeled multi-Bernoulli filter," *arXiv preprint arXiv:2401.10269*, 2024.

[18] J. Houssineau, "A linear algorithm for multi-target tracking in the context of possibility theory," *IEEE Transactions on Signal Processing*, vol. 69, pp. 2740–2751, 2021.

[19] A. P. Dawid, M. Stone, and J. V. Zidek, "Marginalization paradoxes in bayesian and structural inference," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 35, no. 2, pp. 189–213, 1973.

[20] J. Houssineau, "Parameter estimation with a class of outer probability measures," *arXiv preprint arXiv:1801.00569*, 2018.

[21] J. Houssineau and A. N. Bishop, "Smoothing and filtering with a class of outer measures," *SIAM/ASA Journal on Uncertainty Quantification*, vol. 6, no. 2, pp. 845–866, 2018.

[22] B. Ristic, J. Houssineau, and S. Arulampalam, "Robust target motion analysis using the possibility particle filter," *IET Radar, Sonar & Navigation*, vol. 13, no. 1, pp. 18–22, 2019.

[23] C. Kimchaiwong, *Ensemble Kalman filtering with an alternative representation of uncertainty*. PhD thesis, University of Warwick, 2024.

[24] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE transactions on Signal Processing*, vol. 56, no. 8, pp. 3447–3457, 2008.

[25] M. B. Guldogan, "Consensus Bernoulli filter for distributed detection and tracking using multi-static doppler shifts," *IEEE Signal Processing Letters*, vol. 21, no. 6, pp. 672–676, 2014.

Algorithm 3 $\mathrm{Fusion}(F, F', \gamma, \gamma')$

Input: Two possibility functions under the form $F = \big(\alpha, \beta, (w_j, \mu_j, P_j)_{j=1}^{N}, w_{\mathrm{u}}, (f_{\mathrm{u},j})_{j=1}^{\breve{N}}\big)$ and $F' = \big(\alpha', \beta', (w'_j, \mu'_j, P'_j)_{j=1}^{N'}, w'_{\mathrm{u}}, (f'_{\mathrm{u},j})_{j=1}^{\breve{N}'}\big)$; Discount weights γ and γ'

 # Discount unobserved terms
$w_{\mathrm{u}} \leftarrow (w_{\mathrm{u}})^{\gamma}$, $w'_{\mathrm{u}} \leftarrow (w'_{\mathrm{u}})^{\gamma'}$
for all $j \in \{1, \dots, \breve{N}\}$ **do**
  $f_{\mathrm{u},j} \leftarrow \mathrm{DiscountUnobserved}(f_{\mathrm{u},j})$
end for
for all $j \in \{1, \dots, \breve{N}'\}$ **do**
  $f'_{\mathrm{u},j} \leftarrow \mathrm{DiscountUnobserved}(f'_{\mathrm{u},j})$
end for
for all $j \in \{1, \dots, N\}$ **do**
  # Fuse observed terms of Gaussian max-mixture
  for all $j' \in \{1, \dots, N'\}$ **do**
    $m \leftarrow (j-1)N' + j'$
    $\hat{w}_m \leftarrow (w_j)^{\gamma}(w'_{j'})^{\gamma'}\overline{\mathrm{N}}(\mu_j; \mu'_{j'}, P_j/\gamma + P'_{j'}/\gamma')$
    $\hat{P}_m \leftarrow \big(\gamma(P_j)^{-1} + \gamma'(P'_{j'})^{-1}\big)^{-1}$
    $\hat{\mu}_m \leftarrow \hat{P}_m\big(\gamma(P_j)^{-1}\mu_j + \gamma'(P'_{j'})^{-1}\mu'_{j'}\big)$
  end for
  # Fuse some mixed observed/unobserved terms
  for all $j' \in \{1, \dots, \breve{N}'\}$ **do**
    $m \leftarrow NN' + (j-1)\breve{N}' + j'$
    $(\hat{w}_m, \hat{\mu}_m, \hat{P}_m) \leftarrow$ $\mathrm{FuseUnobsObs}(w'_{\mathrm{u}}, f'_{\mathrm{u},j'}, ((w_j)^{\gamma}, \mu_j, P_j/\gamma))$
  end for
end for
 # Fuse other mixed observed/unobserved terms
for all $j' \in \{1, \dots, N'\}$ **do**
  for all $j \in \{1, \dots, \breve{N}\}$ **do**
    $m \leftarrow N(N' + \breve{N}') + (j'-1)\breve{N} + j$
    $(\hat{w}_m, \hat{\mu}_m, \hat{P}_m) \leftarrow$ $\mathrm{FuseUnobsObs}(w_{\mathrm{u}}, f_{\mathrm{u},j}, ((w'_{j'})^{\gamma'}, \mu_{j'}, P'_{j'}/\gamma'))$
  end for
end for
$\hat{N} \leftarrow NN' + N\breve{N}' + N'\breve{N}$
 # Fuse unobserved terms
for all $j \in \{1, \dots, \breve{N}\}$ **do**
  $\tilde{f}_{\mathrm{u},j} \leftarrow \mathrm{MultiplyUnobserved}\Big(f_{\mathrm{u},j}, \min_{j' \in \{1,\dots,\breve{N}'\}} f'_{\mathrm{u},j}\Big)$
end for
 # Renormalise weights
$u \leftarrow \max\{w_{\mathrm{u}}w'_{\mathrm{u}}, \max_j \hat{w}_j\}$
$\tilde{w}_{\mathrm{u}} \leftarrow w_{\mathrm{u}}w'_{\mathrm{u}}/u$
$(\hat{w}_j)_{j=1}^{\hat{N}} \leftarrow (\hat{w}_j)_{j=1}^{\hat{N}}/u$
 # Pruning and merging
$(\tilde{w}_j, \tilde{\mu}_j, \tilde{P}_j)_{j=1}^{N} \leftarrow \mathrm{Reduction}\big((\hat{w}_j, \hat{\mu}_j, \hat{P}_j)_{j=1}^{\hat{N}}\big)$
 # Fuse possibilities of non-existence
$\tilde{\alpha} \leftarrow u\alpha\alpha' / \max\{u\alpha\alpha', \beta\beta'\}$
$\tilde{\beta} \leftarrow \beta\beta' / \max\{u\alpha\alpha', \beta\beta'\}$

Output: $\tilde{F} = \big(\tilde{\alpha}, \tilde{\beta}, (\tilde{w}_j, \tilde{\mu}_j, \tilde{P}_j)_{j=1}^{N}, \tilde{w}_{\mathrm{u}}, (\tilde{f}_{\mathrm{u},j})_{j=1}^{\breve{N}}\big)$