Resilience in multi-robot multi-target tracking with unknown number of targets through reconfiguration

Ragesh K. Ramachandran, Nicole Fronda and Gaurav S. Sukhatme

***Abstract*—We address the problem of maintaining resource availability in a networked multi-robot team performing distributed tracking of an unknown number of targets in a bounded environment. Robots are equipped with sensing and computational resources enabling them to cooperatively track a set of targets using a distributed Probability Hypothesis Density (PHD) filter. We use the trace of a robot's sensor measurement noise covariance matrix to quantify its sensing quality. While executing the tracking task, if a robot experiences sensor quality degradation, the team's communication network is reconfigured such that the robot with the faulty sensor may share information with other robots to improve the team's target tracking ability without enforcing a large change in the number of active communication links. A central system monitor executes the network reconfiguration computations. We consider two different PHD fusion methods and propose four different Mixed Integer Semi-Definite Programming (MISDP) formulations (two formulations for each PHD fusion method) to accomplish our objective. All MISDP formulations are validated in simulation.**

I. Introduction and Related work

MULTI-ROBOT Multi-Target Tracking (MRMTT) problems are of considerable interest due to their significant civilian and military applications [1], [2]. This upsurge of interest in MRMTT problems has spurred the development of various decentralized/distributed tracking strategies which enable robots with limited capabilities (e.g. limited field of view, memory and data processing power) to collaboratively perform the tracking task efficiently and with robustness [3]–[8]. A typical MRMTT framework consists of a set of static or mobile robots ("trackers") which are spatially distributed over a domain of interest while being able to communicate with each other. Each robot runs a local tracking algorithm which estimates the state of the targets in the environment using the measurements received from its field of view. Apart from performing local Multi-Target Tracking (MTT), the robots disseminate their information about the targets to their neighboring robots iteratively. Consequently, each robot refines its estimate on targets' state by appropriately fusing its local target information with the information received from its neighbors. Clearly, for any distributed strategy to function, the communication graph associated with the robots should be connected. Distributed multi-target tracking strategies eliminate the need for a centralized data processing station which was otherwise necessary for the fusion of measurement data collected by the robots to estimate the state of targets present in the environment. Moreover, distributed strategies have shown improved robustness to external noise [9], and are resilient to failures [10].

This work was supported in part by the Army Research Laboratory as part of the Distributed and Collaborative Intelligent Systems and Technology (DCIST) Collaborative Research Alliance (CRA). The authors are with the Department of Computer Science, University of Southern California, Los Angeles, CA 90089, USA rageshku| nfronda|gaurav@usc.edu

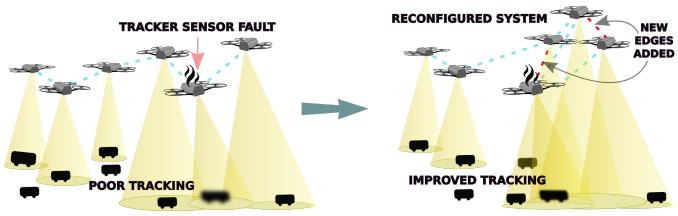


Fig. 1: Our setting for resilient target tracking.

Three challenges are particularly important in multi-target tracking: 1) new targets may appear in the environment and existing ones may leave the environment (number of targets are varying); 2) measurements may be generated by non-target objects (measurement clutter or false alarms); and 3) a robot may fail to detect targets in its Field Of View (FOV) (missed detection). Different multi-target tracking techniques have been proposed which can tackle these challenges e.g. Multiple Hypothesis Tracking (MHT), Joint Probabilistic Data-Association (JPDA) [11], and Random Finite Set (RFS) based Probability Hypothesis Density (PHD) filter [12], [13]. We follow a PHD filter formulation in this paper. The elegant formulation of PHD filters due to Mahler [12] provides a principled way to formulating the target tracking problem as a Bayesian filtering problem. In simple terms, the probability hypothesis density of a target position state (modeled as a random finite set) is the target population density in an environment whose integral over a region yields the expected number of targets in that region. A consensus-based decentralized strategy is employed so that the robots collaboratively estimate the number of targets in the environment [14], [15]. It is noteworthy that, unlike the traditional target tracking algorithms (e.g. MHT, JPDA), the PHD filter does not track individual target tracks. Instead, it estimates the density of the targets over time rather than the motion of individual targets. However, in recent times, using labeled random finite sets, researchers have extended PHD filters to enable tracking of individual targets [16], [17].

We envision a scenario in which a team of robots running a distributed PHD filter on-board cooperatively track a set of unknown number of targets steering according to some known dynamics in an environment. Moreover, the robots are overseen by a central station which intervenes only if a member in the robot team is affected by an event which result in its sensor quality deterioration. We quantify a robot's sensor quality using the trace of robot's sensor measurement noise covariance matrix. The fundamental problem we address in this paper is to attenuate the effect of a robot's sensor quality degradation on its target tracking performance (and hence the tracking performance of the robot team). We attempt

to tackle this by 1) suitably reconfiguring the topology of the robot team's communication graph and 2) through optimal regeneration of weights used for data fusion among the robots (Fig 1). We refer to the event which resulted in robot sensor quality deterioration as a *detrimental event*. From a control theoretic perspective, a target tracking problem can be viewed as a state estimation problem and accuracy of state estimation is related to the observability of the system. The observability of a networked system with respect to its topology has received significant attention in recent times [18]–[20]. These results motivated us to explore the possibility of improving the observability of the system through reconfiguring the communication graph topology, thereby mitigating the effect of sensor quality deterioration on the target tracking performance.

Here, we extend the abstract resilience framework introduced in our previous work [21] to tackle sensor faults in the MRMTT setting. We had previously adopted the abstract resilience framework to handle sensor quality deterioration in the case of single target tracking [22]. In contrast, here we consider the more general multi-target tracking setting where the number of targets is time varying and unknown, the robots may receive clutter measurements or false alarms, and the targets may successfully maneuver within robots' FOV without being detected. Following our abstract resilience framework in [21], we mitigate the impact of a robot's sensor quality deterioration using a two-step approach. In the first step, the robot team's communication graph is modified such that the modified topology is close to the original topology and the multi-target tracking performance of the team (compared to its performance after the "detrimental event") is improved. The details about the multi-target tracking performance metrics used are described in Section IV-A. In addition, at this step, a set of optimal weights to fuse local PHDs among the robots in the team is also computed. The subsequent step computes a set of coordinates for the robots to embed the communication graph in 3D while simultaneously maximizing the robot team's coverage over the domain centered close to the centroid of the estimated targets' state PHD at that instant. Note that, our work does not consider the possibility of a complete robot sensor failure. However, under our framework, an almost-complete sensor failure can be represented as a sensor with extremely large measurement noise covariance. Moreover, we assume that the measurement noise associated with any robot's sensor is always described by a zero mean probability density function. Lastly, the robots are assumed to be able to estimate their sensor quality. This assumption does not impose any unreasonable restriction on the applicability of our strategy as several techniques exist in literature for sensor fault detection [23], [24] and degradation estimation [25].

Step one of our approach uses *mixed integer semi-definite programs* (MISDPs) to formulate and solve the communication network reconfiguration problem and associated local PHD fusion weights generation. In this article, we consider two different kinds of local PHD fusion methods, namely: 1) *Geometric Mean Fusion* (GMF) [14] and 2) *Arithmetic Mean Fusion* (AMF) [26]. The former PHD fusion method (GMF) obtains the resultant PHD as the geometric mean of the fusing local PHDs whereas the latter computes the arithmetic mean of the fusing PHDs. In both fusion strategies, each robot also runs a consensus protocol simultaneously to distributively estimate the number of targets in the domain. The PHD fusion weights obtained from step one are employed to perform the target cardinality consensus update. Furthermore, we consider two target tracking performance criteria for each fusion method which we refer to as *robot-centric* and *team-centric*. As a result, we examine four MISDPs: *Robot-Centric Geometric Mean Configuration generation* (**RCGMC**), *Team-Centric Geometric Mean Configuration Generation* (**TCGMC**), *Robot-Centric Arithmetic Mean Configuration Generation* (**RCAMC**) and *Team-Centric Arithmetic Mean Configuration Generation* (**TCAMC**). In essence, a robot-centric approach optimizes the multi-target tracking performance of the robot affected by a detrimental event and team-centric approach optimizes multi-target tracking performance of the whole robot team, each with respect to a suitable metric consistent with the type of fusion rule employed. Although resilience in multi-robot systems have received tremendous research interest [27], the concept of resilience through reconfiguration to improve task efficacy of the multi-robot system is recent. Through this paper, we introduce the notion of resilience by reconfiguration into multi-robot multi-target tracking.

II. Notation and Background

For any positive integer $z \in \mathbb{Z}^+$, $[z]$ denotes the set $\{1, 2, \cdots, z\}$. $\|\cdot\|$ denotes the standard Euclidean 2-norm and the induced 2-norm for vectors and matrices. $\|M\|_F$ is the Frobenius norm of the matrix $M \in \mathbb{R}^{m_1 \times m_2}$. $Tr(M)$ is the trace of matrix M. $\bar{1}^m$ and $\bar{0}^m$ are the vector of ones and zeros with appropriate dimensions. We drop the superscripts whenever the dimensions of the vectors or matrices are clear from the context. $|\cdot|$ is used to denote the cardinality of a set whenever it encloses a set whereas, the same notation represents the determinant of a matrix if it encompasses a matrix. We use the same notation also to represent the number of Gaussian components in a Gaussian mixture. Also, $\mathbb{E}[\cdot]$ represents the expectation operator. In addition, $\mathcal{N}(\bar{z}; \bar{\mu}, \Sigma)$ denotes the Gaussian probability density of $\bar{z}$ with $\bar{\mu}$ and Σ representing the mean vector and covariance matrix respectively. We use 0^n_i to denote a vector of zeros with one at the index i. $diag(M)$ yields the vector containing the diagonal elements of the matrix M. $M^\top$ or $(M)^\top$ is the transpose of M. $Blkdig(M_1, M_2, \cdots, M_n)$ outputs a block diagonal matrix with the matrices $M_1, M_2, \cdots, M_n$ along its diagonal. $\mathcal{S}^m_+$ denotes the space of $m \times m$ symmetric positive semi-definite matrices. Also, $M \succ 0$ implies M is positive definite. A weighted undirected graph with non negative edge weights $\mathcal{G}$ is defined using the triplet $(\mathcal{V}, \mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}, \mathbf{A} \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}|}_{\geq 0})$, where A is the weighted adjacency matrix of the graph. $\overline{\mathcal{E}} = (\mathcal{V} \times \mathcal{V}) \setminus \mathcal{E}$ is the edge complement of $\mathcal{G}$. A matrix $\mathbf{M}$ is doubly stochastic if its rows and columns sum to unity [28].

A. Random finite set theory

In this section, we review the mathematical background on random finite set theory required to understand the multi-target

framework described in this paper. A rigorous treatment on the subject can be found in [11], [29] and the references therein. A RFS is a random variable whose realizations are sets with finite cardinality. In a multi-tracking application, RFS are used to model the set of target states and the set of measurements obtained from them at various time instants.

A random finite set $\mathcal{X}$ can be characterized using a *multi-object density function* $f(\mathcal{X})$. Unlike a random vector, the multi-object density function associated with a random set is invariant under arbitrary permutation of the elements in the set. Although the multi-object density function completely characterizes a RFS, using the multi-object density function for filtering application is in general intractable due to the high combinatorial complexity involved [12]. Hence, approximate simpler methods are inevitable in practise. A common tractable approximation used to perform filtering on RFS is known as the *probability hypothesis density* or *intensity function* filter. The probability hypothesis density function $v(\bar{x})$ is the first statistical moment of $f(\mathcal{X} = \{x_1, x_2, \cdots, x_n\})$ over the RFS $\mathcal{X}$, where the set integral [14], [29] is applied to compute the statistical moment. An important and useful property of a PHD is that its integral over $\mathcal{R} \subseteq \mathbb{X}$ results in the expected number of targets in $\mathcal{R}$. Specifically, $\int_{\mathcal{R}} v(\bar{x})d\bar{x} = \mathbb{E}[|\mathcal{X} \cap \mathcal{R}|]$. To further reduce the computation burden, it is assumed that a PHD can be approximated using the following weighted finite series expansion, $v(\bar{x}) \approx \sum_{i=1}^{i_{max}} \alpha_i \phi_i(\bar{x})$, with i_{max} non-negative weights α_i and basis functions $\phi_i(\bar{x})$ such that $\int_{\mathbb{X}} \phi_i(\bar{x})d\bar{x} = 1$. It is straightforward to see that,

$$\mathbb{E}[|\mathcal{X}|] \approx \sum_{i=1}^{i_{max}} \alpha_i. \tag{1}$$

When the Gaussian function takes the role of the basis function $\phi_i(\bar{x})$ then PHD filter is referred in literature as *Gaussian Mixture* PHD (GM-PHD) [16]. In the room of this paper, we use a Gaussian mixture representation for the PHD filter. This choice is driven by the fact that the GM-PHD filter equations are similar to the standard Kalman filter equations which is well suited for our MRMTT resilience framework.

III. Problem Formulation

We consider a team of *n* robots whose labels belong to $\{1, 2, \cdots, n\}$ tasked with tracking unknown and time-varying τ_k number of targets (at the k^{th} time step). The team monitors and performs the tracking task over a compact Euclidean space $\mathcal{D}$ for a time period of T epochs. However, the robots in the team can maneuver in the 3D space. To keep the computations simple, we confine $\mathcal{D}$ to be a subset of $\mathbb{R}^2$. Nevertheless, the formulations presented in this paper can be easily generalized to higher dimensions. Since the set of target states can be random, their collection is modeled using a random finite set (RFS) $\mathcal{S}_k = \{\bar{s}_1, \bar{s}_2, \cdots, \bar{s}_{\tau_k}\}$, where $\bar{s}_i \in \mathbb{R}^4$ (position and velocity) models the state of the i^{th} target that exist at time step k. We refer to the robot team that tracks the moving targets as the *tracker team* and the robots as *trackers*. Let $\bar{x}^\iota$ denotes the triplet position vector $[x_k^\iota, y_k^\iota, z_k^\iota] \in \mathbb{R}^3$ of robot $\iota \in [n]$, then the set $\{\bar{x}_{[n]}\}$ contains the positions of all trackers. Also, ρ^ι represents the robot with label $\iota \in [n]$. We assume that the trackers are equipped with localization capabilities which enable them to localize with reasonable accuracy in the environment. Since the tracker team performs Distributed Multi-Target Tracking (D-MTT) through inter-robot communication, they are equipped with resources required for communication. Also, we assume perfect inter-robot communication among the trackers in the tracker team.

In this paper, we model the FOV of ρ^ι as a disc of radius d_{sen}^ι. Now if ρ^ι is stationed at $\bar{x}_k^\iota$, then $p_{D,k}^\iota(\bar{s}|\bar{x}_k^\iota)$ denotes the probability of detection of a target with state $\bar{s}$ by ρ^ι at time k. $p_{D,k}^\iota(\bar{s}|\bar{x}_k^\iota) = 0$ if the target (with state $\bar{s}$) lies outside the FOV of ρ^ι and $p_{D,k}^\iota(\bar{s}|\bar{x}_k^\iota) \leq 1$ otherwise. When ρ^ι flawlessly detects a target, its sensor gives a measurement $\bar{z}$ distributed according to the probability density function $h(\bar{z}|\bar{s}, \bar{x}_k^\iota, k)$. For the computations performed in this paper, we assume that $h(\bar{z}|\bar{s}, \bar{x}_k^\iota, k)$ can be expressed as $\mathcal{N}(\bar{z}; H_k^\iota \bar{s}, R_k^\iota)$, where H_k^ι is the sensor output matrix of ρ^ι. R_k^ι is the covariance matrix of a Gaussian distribution modeling the sensor noise characteristics of ρ^ι. Furthermore, we assume that ρ^ι receives at most one measurement per target present in its FOV at a time instant. Since the set of target measurements received by each tracker robot within its FOV is time-varying, we use a RFS to represent the set of measurements. Let $\mathcal{Z}_k^\iota = \{\bar{z}_{k,1}^\iota, \bar{z}_{k,2}^\iota, \ldots, \bar{z}_{k,|\mathcal{Z}_k^\iota|}^\iota\}$ denote the RFS of the set of measurements obtained by ρ^ι due the targets present in its FOV at time step k. Note that, $|\mathcal{Z}_k^\iota|$ is less than or equal to the number of targets present in ρ^ι's FOV at time k. In addition to the measurements from the targets, a tracker may also gather false measurements due to non-target objects present in the environment. The set of false or clutter measurements acquired by ρ^ι is also modeled using a RFS $\mathcal{A}_k^\iota = \{\bar{c}_{k,1}^\iota, \bar{c}_{k,2}^\iota, \ldots, \bar{c}_{k,|\mathcal{A}_k^\iota|}^\iota\}$. Hence, the total measurements obtained by ρ^ι at time k can be represented using the RFS $\bar{\mathcal{Z}}_k^\iota = \mathcal{Z}_k^\iota \cup \mathcal{A}_k^\iota$. Let $c_k^\iota(\bar{z})$ denotes the PHD associated with $\mathcal{A}_k^\iota$. Furthermore, we account for new targets intruding into the domain using the RFS $\mathcal{B}_k$ and the associated PHD $b_k(\bar{s})$. Finally, $p_{S,k}(\bar{s}_{k-1})$ is the probability that a target with state $\bar{s}$ at time $k-1$ is lingering around in the environment at time k. In essence, $p_{S,k}(\bar{s}_{k-1})$ accounts for targets surviving in the environment.

Let the time varying undirected graph $\mathcal{G}[k] = (\mathcal{V}, \mathcal{E}[k], A_u[k])$ model the communication network of the tracker team at the k^{th} time step ($k \in [T]$). Note that we use "time step", "time" and "epoch" interchangeably in this paper. The node set $\mathcal{V}$ is isomorphic to the tracker team label set $[n]$. An edge (i, j) is included in the edge set $\mathcal{E}[k]$ if i^{th} robot communicates with the j^{th} robot and vice versa at time k. We denote the communication range of trackers as $d_{mc} > 0$. The neighbor set of node i in $\mathcal{G}[k]$ is defined as $\mathcal{N}_{(i)}[k] = \{j \in \mathcal{V} : (i, j) \in \mathcal{E}[k]\}$. The interaction between nodes in a graph can also be represented using an unweighted adjacency matrix. The unweighted adjacency matrix of $\mathcal{G}[k]$ is denoted by $A_u[k]$.

A. Distributed Multi-Target Tracking (D-MTT)

In general, a MTT problem can decomposed into two estimation problems: 1) estimation of the number of targets

present in the environment and 2) estimation of targets' state. The PHD filter is a computationally efficient way to simultaneously solve both estimation problems in a tractable way. If multiple trackers are employed to monitor a region of interest for intruders, then MTT can be performed distributively without the use of a centralized data fusion center. As mentioned earlier, in distributed multi-target tracking, each tracker runs a local PHD filter using its measurements obtained for its field of view and transmits relevant information about its local PHD filter to its neighboring trackers. The neighboring trackers then update their local PHDs by fusing the received information with their PHDs. In this paper, we examine two different PHD fusion methods proposed in literature for our resilience framework, namely, *geometric mean fusion* (GMF) [14] and *arithmetic mean fusion* (AMF) [26]. Interestingly, both fusion strategies can be elegantly derived as optimal solutions to two different minimization problems involving the weighted Kullback-Leibler divergence (KLD) between the fusing multi-object densities [30]. Furthermore, AMF and GMF do not double-count information as long as their fusing weights sum to unity [31]. In the forthcoming subsections, we describe the local PHD filter employed by each tracker and the two different PHD fusing strategies used in this paper.

B. Tracker local PHD filter

Analogous to a Kalman filter, a PHD filter also consists of a *prediction* step and an *update* or *innovation* step. In the prediction step of the PHD filter, PHD associated with the target states RFS is updated based on the target dynamics and the target birth RFS $\mathcal{B}_k$. Subsequently, in the innovation step, the targets' state RFS PHD is refined using the measurements received from the targets. We assume that every target in the environment follows the following standard linear state space dynamics equation

$$\bar{s}_{k+1} = F_k \bar{s}_k + G_k \bar{u}_k + \bar{w}_k, \tag{2}$$

where $\bar{s}_k \in \mathbb{R}^{s_a}$ and $\bar{u}_k \in \mathbb{R}^{u_a}$ are the state and the input vectors of a target respectively. $F_k \in \mathbb{R}^{s_a \times s_a}$ and $G_k \in \mathbb{R}^{s_a \times u_a}$ are the state transition matrix and input matrix of appropriate dimensions respectively. $\bar{w}_k \in \mathbb{R}^{s_a}$ is the zero mean normally distributed random vector with the covariance matrix $Q_k \in \mathbb{R}^{s_a \times s_a}$ ($\bar{w}_k \sim \mathcal{N}(\bar{0}, Q_k)$) modeling the process noise.

In D-MTT, each tracker maintains and updates a local PHD filter. The prediction and update PHD filter equation associated with a tracker ρ^ι can be mathematically expressed as [11], [12],

$$\begin{aligned} v^\iota_{k|k-1}(\bar{s}) &= b_k(\bar{s}) + \\ &\int p_{S,k}(\bar{s}_{k-1}) f(\bar{s}|\bar{s}_{k-1}, \bar{u}_{k-1}) v^i_{k-1|k-1}(\bar{s}_{k-1}) d\bar{s}_{k-1} \end{aligned} \tag{3}$$

$$\begin{aligned} v^\iota_{k|k}(\bar{s}) &= (1 - p^\iota_{D,k}(\bar{s}|\bar{x}^\iota_k)) v^\iota_{k|k-1}(\bar{s}) + \\ &\sum_{\bar{\zeta} \in \bar{\mathcal{Z}}^\iota_k} \frac{p^\iota_{D,k}(\bar{s}|\bar{x}^\iota_k) h(\bar{\zeta}|\bar{s}, \bar{x}^\iota_k, k) v^\iota_{k|k-1}(\bar{s})}{c^\iota_k(\bar{\zeta}) + \int p^\iota_{D,k}(\bar{s}|\bar{x}^\iota_k) h(\bar{\zeta}|\bar{s}, \bar{x}^\iota_k, k) v^\iota_{k|k-1}(\bar{s}) d\bar{s}}, \end{aligned} \tag{4}$$

$f(\bar{s}|\bar{s}_{k-1}, \bar{u}_{k-1})$ is the probability of occurrence of target state $\bar{s}$ at time k given the previous states and inputs, derived using Eq 2. Eq 3 and Eq 4 are the prediction and update or innovation equations respectively.

Despite the fact that, in general, it is hard to further simplify the above equations, a closed form expression can be derived if the PHDs are assumed to be Gaussian Mixtures (GM), and target motion and measurement models are assumed to be linear [32]. As indicated earlier, we adopt a GM approximation for the PHDs used in the paper. Thus $v^\iota_{k|k}(\bar{s}) \approx \sum_{i=1}^{|v^\iota_{k|k}|} \alpha^{\iota,(i)}_{k|k} \mathcal{N}(\bar{s}; \bar{\mu}^{\iota,(i)}_{k|k}, P^{\iota,(i)}_{k|k})$ and $b_k(\bar{s}) \approx \sum_{i=1}^{|b_k|} \alpha^{b,(i)}_k \mathcal{N}(\bar{s}; \bar{\mu}^{b,(i)}_k, P^{b,(i)}_k)$. Also, we prescribe that $c^\iota_k(\bar{\zeta}) = \lambda^\iota_{c,k} |FOV|^\iota p_{FOV}(\bar{\zeta})$, where $p_{FOV}(\bar{\zeta})$ is the probability density function of the occurrence of clutter measurements in ρ^ι's FOV (assumed to be uniform in this paper), $|FOV|^\iota$ is the "volume" of its FOV and $\lambda^\iota_{c,k}$ is the expected number of clutter measurements per unit FOV volume. For clarity of presentation, hereafter, we restrict our attention to tracker state independent $p^\iota_{D,k}(\bar{s}|\bar{x}^\iota_k, k)$ and $p_{S,k}(\bar{s}_{k-1})$, the formula for more general case can be found in [32, Section III-E]. Under these assumptions and approximations Eq 3 and Eq 4 can be written as,

GM-PHD filter prediction:

$$v^\iota_{k|k-1}(\bar{s}) = b_k(\bar{s}) + v^\iota_{S,k|k-1}(\bar{s}) \tag{5}$$

$$v^\iota_{S,k|k-1}(\bar{s}) = p_{S,k} \sum_{i=1}^{|v^\iota_{S,k|k-1}|} \alpha^{\iota,(i)}_{k|k-1} \mathcal{N}(\bar{s}; \bar{\mu}^{\iota,(i)}_{k|k-1}, P^{\iota,(i)}_{k|k-1}) \tag{6}$$

$$\bar{\mu}^{\iota,(i)}_{k|k-1} = F_{k-1} \bar{\mu}^{\iota,(i)}_{k|k-1} + G_{k-1} \bar{u}_{k-1} \tag{7}$$

$$P^{\iota,(i)}_{k|k-1} = Q_{k-1} + F_{k-1} P^{\iota,(i)}_{k-1|k-1} F^\top_{k-1}, \tag{8}$$

GM-PHD filter innovation:

$$v^\iota_{k|k}(\bar{s}) = (1 - p^\iota_{D,k}) v^\iota_{k|k}(\bar{s}) + \sum_{\bar{\zeta} \in \bar{\mathcal{Z}}^\iota_k} v^\iota_{D,k}(\bar{s}; \bar{\zeta}) \tag{9}$$

$$v^\iota_{D,k}(\bar{s}; \bar{\zeta}) = \sum_{i=1}^{|v^\iota_{D,k}|} \alpha^{\iota,(i)}_{k|k}(\bar{\zeta}) \mathcal{N}(\bar{s}; \bar{\mu}^{\iota,(i)}_{k|k}(\bar{\zeta}), P^{\iota,(i)}_{k|k}) \tag{10}$$

$$\alpha^{\iota,(i)}_{k|k}(\bar{\zeta}) = \frac{p^\iota_{D,k} \alpha^{\iota,(i)}_{k|k-1} \mathcal{N}(\bar{\zeta}; H^\iota_k \bar{\mu}^{\iota,(i)}_{k|k-1}, S^{\iota,i}_k)}{c^\iota(\bar{\zeta}) + p^\iota_{D,k} \sum_j \alpha^{\iota,(j)}_{k|k-1} \mathcal{N}(\bar{\zeta}; H^\iota_k \bar{\mu}^{\iota,(j)}_{k|k-1}, S^{\iota,j}_k)} \tag{11}$$

$$S^{\iota,i}_k = R^\iota_k + H^\iota_k P^{\iota,(i)}_{k|k-1} (H^\iota_k)^\top \tag{12}$$

$$\bar{\mu}^{\iota,(i)}_{k|k} = \bar{\mu}^{\iota,(i)}_{k|k-1} + K^{\iota,(i)}_k (\bar{\zeta} - H^\iota_k \bar{\mu}^{\iota,(i)}_{k|k-1}) \tag{13}$$

$$P^{\iota,(i)}_{k|k} = [I - K^{\iota,(i)}_k H^\iota_k] P^{\iota,(i)}_{k|k-1} \tag{14}$$

$$K^{\iota,(i)}_k = P^{\iota,(i)}_{k|k-1} (H^\iota_k)^\top (S^{\iota,i}_k)^{-1}. \tag{15}$$

[32, Table 1] gives a pseudocode for the Gaussian mixture PHD filter implementation. Notice that Eq 7-Eq 8 and Eq 12-Eq 15 are similar to the prediction and innovation steps of a standard Kalman filter respectively [11]. Finally, the number of Gaussian components (GCs) in $v^\iota_{k|k}(\bar{s})$ are reduced for computational efficiency by merging closer GCs and pruning GCs with low weights (see [32, Section III.C, Table II]).

C. Local PHD fusion

In a PHD fusion method, the estimated local targets' state PHD $v^\iota_{k|k}(\bar{s})$ of ρ^ι is fused with the estimated local targets'

state PHDs of other neighboring trackers in the tracker team. From hereon, unless otherwise specified, PHD or local PHD mean local targets' state PHD. Certainly, fusing all the PHD Gaussian components (GCs) of ρ^{ι} with all the PHD Gaussian components received from its neighbours is inefficient in terms of both computational and communication load. To this end, we follow the method proposed in [26]. According to the strategy delineated in [26], each tracker disseminates only the highly weighted GCs or Target likely GCs (T-GCs) (that possibly corresponds to real targets) in its PHD to its neighboring trackers. In our work, we identify the T-GCs in a PHD using the *rank rule* outlined in [26]. Consequently, the numbers of T-GCs selected equals the integer closest to the expected number of targets according to a tracker's local PHD. Recall that, from Eq 1, the sum of the weights of a GM-PHD gives the expected number of targets in the region of interest. Let $\alpha_k^{\iota} = \sum_i \alpha_{k|k}^{\iota,(i)}$ and $\tilde{\alpha}_k^{\iota} = \lceil \alpha_k^{\iota} \rceil$. A neighboring tracker can then fuse its local PHD T-GCs with the communicated T-GCs using any established fusion method of choice [14], [33]. As noted earlier, in this article, we focus on AM and GM based fusion strategies described in latter subsections. In both cases, an additional cardinality consensus scheme is simultaneously executed and each tracker's PHD GCs are rescaled so that the tracker's estimate of the expected number of targets converges to the global average. The cardinality consensus rule can be mathematically written as:

$$\tilde{\alpha}_k^i(l) = \sum_{j \in \mathcal{N}_{(i)}[k] \cup \iota} [\bar{A}[k]]_{i,j} \tilde{\alpha}_k^j(l-1), \tag{16}$$

where $[\bar{A}[k]]_{i,j}$ is the (i,j) entry of a doubly stochastic matrix $\bar{A}[k]$ with the same structure as the unweighted adjacency matrix ($A_u[k]$) of connected graph $\mathcal{G}[k]$ except for the diagonal elements. Specifically, $\bar{\mathbf{A}}[k]$ is non-zero along its diagonal and its off-diagonal elements are non-zero if and only if the corresponding elements of $\mathbf{A}_u[k]$ are unity. In theory, cardinality consensus scheme shown in Eq 16 converges to a common quantity only when l tends to infinity. However, it is known that consensus protocols enjoy an exponential rate of convergence [10]. Thus, a reasonable level of consensus on the global average can be attained by iterating Eq 16 for a sufficient number of consensus steps L. From our simulations, we found that the team reaches a reasonable level of consensus with $L = n/2$. We note that, in this paper we only consider targets having dynamics (Eq 2) comparable with the speed of consensus dynamics. After each consensus step l, along with fusing the local T-GCs with the neighbor's T-GCs (using GMF or AMF), the PHD GCs weights of each tracker are rescaled such that they sum to $\tilde{\alpha}_k^{\iota}(l)$, i.e., $\alpha_{k|k}^{\iota,(i)}(l) = w^{\iota} \alpha_{k|k}^{\iota,(i)}(l)$, with $w\iota = \frac{\tilde{\alpha}_k^{\iota}(l)}{\sum_i \alpha_{k|k}^{\iota,(i)}(l)}$. An analysis on the accuracy of cardinality estimation in the cardinality consensus scheme can be found in [15]. In following two subsections, we will delineate the two PHD (primarily T-GCs of the PHDs) fusion strategy adopted in our paper.

1) Geometric mean fusion: Let $d_1, d_2, \ldots, d_g$ be a set of probability density function defined over some state space, then these probability density functions can be fused into a single probability density function d_{GCI} based on a set of weights $\{\omega_i \geq 0\}$ using the generalized covariance intersection fusion rule (GCI) using $d_{GCI} = C^{-1} \prod_{i \in [g]} d_i^{\omega_i}$, the normalization constant C is given by $\int \prod_{i \in [g]} d_i^{\omega_i}$ and $[g] = \{1, 2, \cdots, g\}$. When $\sum_{i \in [g]} \omega_i = 1$ this fusion rule is also referred as *exponential mixture density*.

Examining formula to compute d_{GCI}, we can observe that if the weights sum to unity then d_{GCI} is proportional to the geometric mean of the densities. Hence, we refer to the GCI fusion rule as *Geometric Mean Fusion* (GMF). A detailed discussion on GCI fusion can be found [14], [30] and the references therein. Since the PHDs can be interpreted as unnormalized probability density functions, the fused PHD can be computed similar to d_{GCI} with $C = 1$.

Suppose $\sum_{i_1 \in [g_1]} \alpha_{i_1} \mathcal{N}(\bar{s}; \bar{\mu}_{i_1}, P_{i_1})$, $\sum_{i_2 \in [g_2]} \alpha_{i_2} \mathcal{N}(\bar{s}; \bar{\mu}_{i_2}, P_{i_2})$, $\cdots$, $\sum_{i_n \in [g_n]} \alpha_{i_n} \mathcal{N}(\bar{s}; \bar{\mu}_{i_n}, P_{i_n})$ are the GM-PHDs to be fused together using GMF based on the normalized set of weights $\{\omega_i \geq 0\}_1^n$. We refer to a set of weights as *normalized* if the weights sum to unity. Also, let ϖ denote the dimension of the mean vectors in the GCs. Under the assumption that, raising the GM-PHD to an exponent approximately equals raising the GCs in the GM-PHD to the exponent and summing them [34] or in mathematical terms

$$\left[\sum_{i \in [g]} \alpha_i \mathcal{N}(\bar{s}; \bar{\mu}_i, P_i) \right]^{\omega} \approx \sum_{i \in [g]} \left[\alpha_i \mathcal{N}(\bar{s}; \bar{\mu}_i, P_i) \right]^{\omega}, \tag{17}$$

and if $[g_{1 \cdots n}] \triangleq [g_1] \times [g_2] \times ... [g_n]$ then, we derive the following formula to compute the resultant GM-PHD (v_{GMF}) obtained by fusing the GM-PHDs according to GMF:

$$v_{GMF} \approx \sum_{(i_1, i_2, \ldots, i_n) \in [g_{1 \cdots n}]} \alpha_{i_1, \cdots, i_n} \mathcal{N}(\bar{s}; \bar{\mu}_{i_1, \cdots, i_n}, P_{i_1, \cdots, i_n}) \tag{18}$$

where

$$\bar{\mu}_{i_1, \cdots, i_n} = (P_{i_1, \cdots, i_n})^{-1} \sum_{j=1}^{n} \omega_j (P_{i_j})^{-1} \bar{\mu}_{i_j} \tag{19}$$

$$P_{i_1, \cdots, i_n} = \left(\sum_{j=1}^{n} \omega_j (P_{i_j})^{-1} \right)^{-1} \tag{20}$$

$$\alpha_{i_1, \cdots, i_n} = K \left(\prod_{j=1}^{n} (\alpha_{i_j})^{\omega_j} \sqrt{\frac{\left| 2\pi \frac{P_{i_j}}{\omega_j} \right|}{|2\pi P_{i_j}|^{w_j}}} \right) \tag{21}$$

$K = \exp(\tilde{K} - \bar{K})$; $\tilde{K} = -\frac{1}{2} \left(n\varpi \ln(2\pi) - \sum_{j=1}^{n} \ln \left| \omega_j (P_{i_j})^{-1} \right| \right.$ $\left. \sum_{j=1}^{n} \omega_j (\bar{\mu}_{i_j})^T (P_{i_j})^{-1} \bar{\mu}_{i_j} \right)$ and $\bar{K} = -\frac{1}{2} \left(\varpi \ln(2\pi) - \log|\Omega| + \bar{q}^T (\Omega)^{-1} \bar{q} \right)$; with $\Omega = \sum_{j=1}^{n} \omega_j (P_{i_j})^{-1}$ and $\bar{q} = \sum_{j=1}^{n} \omega_j (P_{i_j})^{-1} \bar{\mu}_{i_j}$. The derivation of Eq 18 is included in Appendix A. The assumption outlined in Eq 17 holds reasonably well when the centers between the distinct GCs in a GM-PHD are well separated [14], [34]. In practice, the well separatedness of GCs in a GM-PHD are ensured by fusing the GCs which are close in the Mahalanobis distance [14] sense [34]. From Eq 18, one can conclude that the number of GCs in the fused

GM-PHD exponentially increases at each fusion step, which would easily saturate storage and computational capabilities of a tracker. The common approaches devised to control the growth of GCs in GM-PHD are merging GCs which are close is some sense (usually in terms of Mahalanobis distance [14], [32]) and pruning GCs whose weights fall below a pre-defined threshold. Here, we follow the latter and prune GCs in the fused GM-PHD which have low weights ($\alpha^{1,\cdots,n}_{i_1,\cdots,i_n} \ll 1$).

Remark 1. *As noted earlier, an important weakness of GMF is the exponential growth of GCs in the fused GM-PHD. In addition, a more serious defect of GMF as pointed out in [35] is its susceptibility to misdetections. Hence, a misdetection by single tracker could potentially jeopardize the MTT performance of other trackers significantly.*

2) Arithmetic mean fusion: If one utilizes the arithmetic mean fusion (AMF) rule to fuse the probability density functions $\{d_1, d_2, \cdots, d_g\}$ defined over some space to yield a single probability density function over the state space, according to a set of normalized weights, then the resultant fused probability density function d_{AMF} is given by $d_{AMF} = \sum_{i\in[g]} \omega_i d_i$.

Akin to the GMF case, the AMF rule used to fuse probability density functions is applied to fuse GM-PHDs. Fusing n GM-PHDs according to AMF yields,

$$v_{AMF} = \sum_{j\in[n]} \omega_j \left(\sum_{i\in[g_j]} \alpha_i^j \mathcal{N}(\bar{s}; \bar{\mu}_i^j, P_i^j) \right). \tag{22}$$

According to Eq 22, fusing two GM-PHDs with g_1 and g_2 GCs create a GM-PHD with g_1+g_2 GCs whereas, the GMF of the GM-PHDs result in a GM-PHD containing $g_1 \times g_2$ GCs. Hence, in general, the number of GCs resulting from AMF is much smaller compared to the GMF case. The number of GCs in the AMF fused GM-PHD can be further reduced by fusing GCs which potentially describe the same target using the following Gaussian Mixture Reduction (GMR) technique.

Without loss of generality, we assume that the first GC from each GM-PHD to be fused represented as $\alpha_1^1\mathcal{N}(\bar{s}; \bar{\mu}_1^1, P_1^1)$, $\alpha_1^2\mathcal{N}(\bar{s}; \bar{\mu}_1^2, P_1^2)$, $\cdots$, $\alpha_1^n\mathcal{N}(\bar{s}; \bar{\mu}_1^n, P_1^n)$ describe the same target due to their closeness to each other in some sense. In practice, two GCs are considered close if the Mahalanobis distance between them is less than a pre-defined threshold. These GCs are combined and reduced to a single GC $\alpha_1^{GMR}\mathcal{N}(\bar{s}; \bar{\mu}_1^{GMR}, P_1^{GMR})$ using the associated normalized weights $\{\omega_1, \omega_2, \cdots, \omega_n\}$. Then [26],

$$\alpha_1^{GMR} = \sum_{i\in[n]} \alpha_1^i \tag{23}$$

$$\bar{\mu}_1^{GMR} = \frac{\sum_{i\in[n]} \alpha_1^i \bar{\mu}_1^i}{\sum_{j\in[n]} \alpha_1^j} \tag{24}$$

$$P_1^{GMR} = \sum_{i\in[n]} \omega^i P_1^i. \tag{25}$$

It has been shown in [26] that, as long as each GC involved in the fusion process is "conservative/consistent" with respect to an estimate on the target state, the fused GC is also conservative (see [36] for the details on conservative/consistent estimate pair). Roughly speaking, a GC is "conservative" with respect to target state estimate if the covariance of the GC is larger than or equal to the covariance of target state estimate error. More precisely, if $\tilde{a}$ is the target state estimation error then the GC with covariance P is consistent with respect to target state estimate if $P - \mathbb{E}(\tilde{a}\tilde{a}^\top)$ is positive semi-definite. In summary, AMF GC avoids underestimating the actual estimate errors in mean square sense, and are resilient to misdetections [37].

D. Tracking under Sensor Quality Deterioration

As stated in Section I, we consider the problem of attenuating the effect of a tracker's sensor quality deterioration on multi-target tracking performance through appropriate reconfiguration of the tracker team. In this subsection, we will define the notion of tracker team reconfiguration and sensor quality deterioration used in this paper.

We define the tuple $(\mathcal{G}[k], \bar{A}[k])$ as the *configuration* of the tracker team at the k^{th} time step and denote it by $\mathcal{C}[k]$. $\bar{A}[k]$ is a doubly stochastic matrix whose elements are used as normalized weights for the local PHD fusion, and to execute the cardinality consensus operations outlined in Eq 16. During the multi-target tracking task, executed for a time period T, let n_f detrimental events occur independently to arbitrary trackers in the team. We assume that each event results in some sensor quality deterioration. At time k, we say that ρ^ι's sensor quality is deteriorated if the trace of the measurement noise covariance matrix associated with its sensor R_k^ι has increased compared to $Tr(R_{k-1}^\iota)$. In more formal terms, if $Tr(R_k^\iota) > Tr(R_{k-1}^\iota)$, then ρ^ι's sensor quality deteriorated at time k. Recall our assumption that, the detrimental event never introduces any bias in the tracker's sensor measurements. The treatment of detrimental event which results in sensor bias is reserved for future work. Similar to [22], we consider a sequence set $\mathcal{F} = [f_1, f_2, \cdots, f_p, \cdots, f_{n_f}]$, where f_p indicate the time step when the p^{th} sensor fault occurred. We specify that $\mathcal{C}[f_p - 1]$ is the configuration of the tracker team before the p^{th} detrimental event occurred. We now formally define the problems studied in this paper. The first problem (Problem 1) deals with reconfiguration of the tracker team such that target tracking performance is optimal in some reasonable sense. The second problem addresses the issue of realizing the graph topology in 3D space while maximizing the tracker team's coverage over the $\mathcal{D}$.

Problem 1. ***Configuration generation or reconfiguration:*** *Given that ρ^i experienced sensor quality deterioration at some time f_p, $R^i_{f_p^+}$ is the sensor noise covariance matrix immediately after the deterioration event, and $\mathcal{C}[f_p - 1]$ is the tracker configuration prior to the event, determine a new configuration $\mathcal{C}[f_p]$ such that,*

1) *$\mathcal{G}[f_p]$ is a connected graph,*
2) *$\|A_u[f_p] - A_u[f_p - 1]\|_F^2 \leq 2 \times e$, where $e \in \mathbb{Z}^+$ is the number of edges that may be modified in $\mathcal{G}[f_p - 1]$ to obtain $\mathcal{G}[f_p]$, and*
3) *tracking performance is optimized in some appropriate sense.*

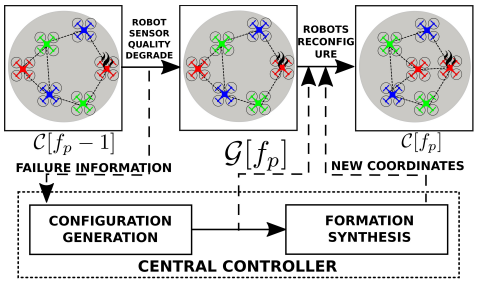


Fig. 2: Basic outline of our approach. When a robot experiences sensor quality degradation, *configuration generation* selects edges to modify the communication graph. Then, *formation synthesis* assigns robots to physical locations that support the desired graph topology.

Problem 2. ***Formation synthesis:*** *Given a tracker team configuration $\mathcal{C}[f_p]$, generate coordinates that best realize the given configuration and maximize the tracker team's coverage over $\mathcal{D}$, subject to constraints. We defer the details of this problem to Section IV-B.*

Graph connectivity constraint is essential for the distributed tracking computation performed over the network and thus is included in Problem 1 [10]. The second condition enables the user to control the communication load on the generated configuration by tuning the parameter e. The final condition assures improved multi-target tracking performance.

IV. Methodology

In this section, we focus on developing various strategies for solving Problem 1 and Problem 2. In our framework, we consider a base station that monitors the activities of the tracker team. The base station intervenes with the team's operation and instructs the multi-target tracking team only when a detrimental event occurs. Similar to [21], [22] , the base station controller adopts a dual-step scheme to arrive at a new tracker team configuration and compute the new tracker coordinates which best realize the new configuration in space. Akin to [21], these steps are referred as *configuration generation* and *formation synthesis*. The whole decision making scheme is depicted in Fig 2. In essence, the solutions to Problem 1 and Problem 2 are the bases for the *configuration generation* and *formation synthesis* steps respectively.

A. Configuration Generation

In general, we classify the configuration generation strategies delineated in this paper into two types: *robot-centric* and *team-centric*. A robot-centric approach aims at improving the tracking performance of the robot which endured the adverse effects of a detrimental event, whereas a team-centric approach optimizes the tracking performance of the whole team. As described earlier, in this article, we restrict our attention to two kinds of local PHD fusion strategies. Hence, for each fusion strategy, we can devise either a tracker-centric or a team-centric approach to configuration generation. We formally formulate all the four configuration generation approaches in this subsection. We refer the four configuration generation approaches as *Robot-Centric Geometric Mean Configuration generation* (**RCGMC**), *Team-Centric Geometric Mean Configuration generation* (**TCGMC**), *Robot-Centric Arithmetic Mean Configuration generation* (**RCAMC**) and *Team-Centric Arithmetic Mean Configuration generation* (**TCAMC**). All four configuration generations result in solving different mixed integer semi-definite programs (MISDPs). The motivation for this stems from the fact that network design problems are often formulated in literature as MISDPs [38]. Note that, in the MISDPs formulations we optimize the tracking performance for one step PHD fusion ($l = 1$). We drop the dependence of variables on time in the MISDPs for brevity. The following theorem guides our design of the connectivity constraint in the MISDPs (See Appendix B for the proof).

Theorem 1. *If a graph containing self loops at every node is equipped with a weighted adjacency matrix A which is doubly stochastic then any graph isomorphic to this graph with or without self loops is connected if and only if $\frac{1}{n}\bar{1}\bar{1}^\top + I \succ A$.*

At time f_p, ρ^ι experienced a sensor fault and $\alpha = \min\{\tilde{\alpha}^1_{f_p}, \ldots, \tilde{\alpha}^\iota_{f_p}, \ldots, \tilde{\alpha}^n_{f_p}\}$. As a result of Eq 16, each tracker should have at least one GC associated with a target up to α targets. In addition, let $\{P^\iota_i\}^\alpha_{i=1}$ be the covariance matrices associated with α T-GCs of ρ^ι and $\tilde{P}^\iota = Blkdig(P^\iota_1, P^\iota_2, \cdots, P^\iota_\alpha)$.

1) ***RCGMC:*** The following MISDP models our robot-centric geometric mean configuration generation approach:

$$\underset{\substack{A\in\mathcal{S}^n_+,\ \nu\in\mathbb{R}_{>0},\\ \Pi\in\{0,1\}^{n\times n}}}{\text{minimize}} \quad -0^{1\times n}_\iota A \begin{bmatrix} \frac{1}{\alpha}\text{Trace}((\tilde{P}^1)^{-1}) \\ \frac{1}{\alpha}\text{Trace}((\tilde{P}^2)^{-1}) \\ \vdots \\ \frac{1}{\alpha}\text{Trace}((\tilde{P}^n)^{-1}) \end{bmatrix} \tag{26}$$

$$\text{subject to} \quad A\cdot\bar{1}^n = \bar{1}^n \tag{27}$$

$$\frac{1}{n}\bar{1}\bar{1}^T + (1-\nu)I \succeq A, \nu \ll 1 \tag{28}$$

$$diag(\Pi) = \bar{1}^n \tag{29}$$

$$\Pi = \Pi^T \tag{30}$$

$$[A]_{i,i} > 0 \ \forall\ i \in [n] \tag{31}$$

$$[A]_{i,j} \geq 0 \forall\ (i,j) \in [n]^2,\ i \neq j \tag{32}$$

$$[A]_{i,j} \leq \Pi_{i,j} \forall\ (i,j) \in [n]^2,\ i \neq j \tag{33}$$

$$\|\Pi - A_u[f_p]\|^2_F \leq 2\times e. \tag{34}$$

The decision variables A and Π model the doubly stochastic matrix used for the consensus protocol and the adjacency matrix of the generated configuration respectively. Constraint 27 and Constraint 31 to Constraint 33 ensures that A is a doubly stochastic matrix that is structurally equivalent to Π. In the light of Theorem 1, Constraint 28 enforces the generated configuration to possess a connected graph. Finally, Constraint 34 encodes the near topology condition (condition 2) in Problem 1 into the MISDP. If ι represents the label of the robot whose sensor quality deteriorated at f_p, then with some simple algebraic manipulation it can be easily shown that Eq 26 results in the average over the trace of the fused GCs according to Eq 20.

2) ***TCGMC:*** Consider the following MISDP formulation encoding the team-centric geometric mean configuration gen-

eration strategy.

$$\underset{\substack{A\in\mathcal{S}_+^n,\ \nu\in\mathbb{R}_{>0},\\ \Pi\in\{0,1\}^{n\times n}\\ \bar{P},\bar{\Delta}\in\mathcal{S}_+^{n\times s_a\times\alpha}}}{\text{minimize}} \quad \text{Trace}(\bar{P}) \tag{35}$$

$$\text{subject to} \quad \begin{bmatrix} \bar{P} & I \\ I & \bar{\Delta} \end{bmatrix} \succeq 0 \tag{36}$$

$$A \otimes I \begin{bmatrix} (\tilde{P}^1)^{-1} \\ (\tilde{P}^2)^{-1} \\ \vdots \\ (\tilde{P}^n)^{-1} \end{bmatrix} = \begin{bmatrix} \Delta_1 \\ \Delta_2 \\ \vdots \\ \Delta_n \end{bmatrix} \tag{37}$$

Constraint 27 – Constraint 34.

Where $\bar{\Delta} = Blkdig(\Delta_1, \Delta_2, \cdots, \Delta_n)$ and $A \otimes I$ results in the *Kronecker product* [28] between A and the identity matrix which matches the size of $\tilde{P}^\iota$. Constraint 37 is the covariance fusion rule Eq 20 for the whole team written compactly as a single equation. In addition, Constraints 27-34 are also required for **TCGMC**. The following lemma proves that minimizing Eq 35 minimizes $\frac{1}{n\times\alpha}\sum_i^n \text{Trace}(P_{gm}^{i,\alpha}))$, where $P_{gm}^{i,\alpha}$ is the block diagonal matrix containing α number of GMF fused GCs' covariance matrices associated with the i^{th} tracker (See Appendix C for the proof).

Lemma 1. *The* $\frac{1}{n\times\alpha}Tr(\bar{P})$ *is an upper bound on* $\frac{1}{n\times\alpha}\sum_i^n Trace(P_{gm}^{i,\alpha})$

3) ***RCAMC****:* Similar to Section IV-A1, we formulated the MISDP for robot-centric arithmetic mean configuration generation as:

$$\underset{\substack{A\in\mathcal{S}_+^n,\ \nu\in\mathbb{R}_{>0},\\ \Pi\in\{0,1\}^{n\times n}}}{\text{minimize}} \quad 0_{\bar{\iota}}^{1\times n} A \begin{bmatrix} \frac{1}{Ws}\text{Trace}((\tilde{P}^1)) \\ \frac{1}{Ws}\text{Trace}((\tilde{P}^2)) \\ \vdots \\ \frac{1}{Ws}\text{Trace}((\tilde{P}^n)) \end{bmatrix} \tag{38}$$

subject to Constraint 27 – Constraint 34.

Here the objective function Eq 38 is a direct result of the application of Eq 25 for ρ^ι.

4) ***TCAMC****:* Finally, the MISDP formulation for the team-centric arithmetic mean configuration generation can be expressed as:

$$\underset{\substack{A\in\mathcal{S}_+^n,\ \nu\in\mathbb{R}_{>0},\\ \Pi\in\{0,1\}^{n\times n}\\ \bar{P},\bar{\Delta}\in\mathcal{S}_+^{n\times s_a\times\alpha}}}{\text{minimize}} \quad \text{Trace}\left(\bar{\Delta}\right) \tag{39}$$

$$\text{subject to} \quad A \otimes I \begin{bmatrix} (\tilde{P}^1) \\ (\tilde{P}^2) \\ \vdots \\ (\tilde{P}^n) \end{bmatrix} = \begin{bmatrix} \Delta_1 \\ \Delta_2 \\ \vdots \\ \Delta_n \end{bmatrix} \tag{40}$$

Constraint 27 – Constraint 34.

Where $\bar{\Delta} = Blkdig(\Delta_1, \Delta_2, \cdots, \Delta_n)$ and therefore minimizing the trace of $\bar{\Delta}$ results in minimizing sum of the traces of AMF fused covariance matrices of the trackers in the team.

B. Formation Synthesis

Once a new configuration is generated, we assign a physical location to each robot to maximize the team's non-overlapping coverage of the space. In this assignment problem, we impose constraints to ensure that connected robot pairs remain within communication distance d_{mc} of each other, and that the distance between all robot pairs exceeds d_s to avoid collision. An additional constraint is added to ensure that each robot is no more than E distance away from the centroid of the T-GCs.

This produces the following constrained optimization problem:

$$\max_{\{X_{|n|}\}} \pi \sum_{i\in\mathcal{V}} \left((d_{sen}^i)^2 - \sum_{j\in\mathcal{V}\neq i} \frac{(2d_{sen}^i - \|X_i - X_j\|)^2}{2} \right) \tag{41}$$

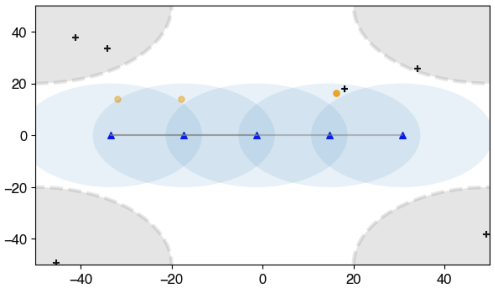
$$\text{subject to} \quad d_s \leq \|X_i - X_j\| \leq d_{mc} \quad \forall\ (i,j) \in \mathcal{E} \tag{42}$$

$$d_s \leq \|X_i - X_j\| \quad \forall\ (i,j) \in \overline{\mathcal{E}} \tag{43}$$

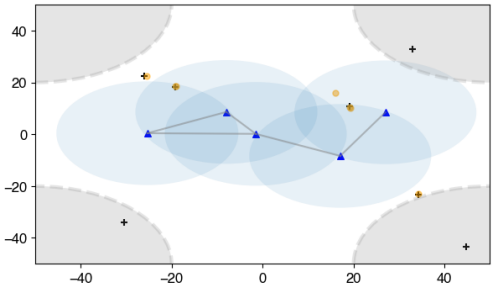
$$B^{\min} \leq X_i \leq B^{\max} \quad \forall\ i \in V \tag{44}$$

$$\|X_{team} - X_{target}\| \leq E \tag{45}$$

where d_{sen}^i is the radius of the circular field of vision of tracker, ρ^i, X_{team} is the average position of the robot team, X_{target} is the centroid of the T-GCs of the failure node, E is the user-defined maximum distance the team can be from X_{target}, and $B^{\min}$, $B^{\max} \in \mathbb{R}^3$ are the minimum and maximum extents of an axis-aligned bounding box, with the operator $\leq$ applied elementwise in Eq 44. We solve the formation synthesis optimization problem Eq 41 - Eq 45 following the simulated annealing approach described in [21].



(a) Before Failure (Overhead)



(b) After Failure (Overhead)

Fig. 3: Screenshots of a simulation in which a robot team of five tracks targets moving below them (overhead view). A robot's sensing area is depicted as a light blue circle. The target birth areas are depicted as light gray circles in the corners. The true target positions are denoted as black '+'s. The target position estimates are denoted as orange circles. The robots themselves are denoted as dark blue triangles. The figure on the top depicts the formation before the occurrence of a sensor deterioration event. The corresponding figure on the bottom portrays the formation after 1) sensor deterioration is detected, 2) a new communication edge is chosen, and 3) the robots move to their new locations.

	OSPA				NMSE		
	Method	Mean Delta	St. Dev Delta		Method	Mean Delta	St. Dev Delta
<50%	RCAMC	0.189	0.425	<50%	RCAMC	0.456	0.374
	GreedyAMC	0.237	0.554		GreedyAMC	-0.208	0.504
	RandomAMC	0.179	0.716		RandomAMC	0.110	0.441
	TCAMC	0.594	0.355		TCAMC	-0.367	0.330
	RCGMC	0.154	0.372		RCGMC	0.567	0.646
	GreedyGMC	0.654	0.470		GreedyGMC	-0.472	0.413
	RandomGMC	0.657	0.503		RandomGMC	-0.432	0.455
	TCGMC	-0.006	0.483		TCGMC	0.183	0.554
>50%	RCAMC	0.296	0.317	>50%	RCAMC	0.450	0.321
	GreedyAMC	0.231	0.544		GreedyAMC	0.494	0.285
	RandomAMC	0.209	0.688		RandomAMC	-0.124	0.556
	TCAMC	0.539	0.608		TCAMC	-0.302	0.519
	RCGMC	0.485	0.631		RCGMC	-0.293	0.522
	GreedyGMC	0.578	0.577		GreedyGMC	-0.352	0.482
	RandomGMC	0.548	0.569		RandomGMC	-0.353	0.488
	TCGMC	0.603	0.235		TCGMC	0.210	0.278

TABLE I: Summary statistics (mean and standard deviation) for the difference in NMSE and OSPA by strategy.

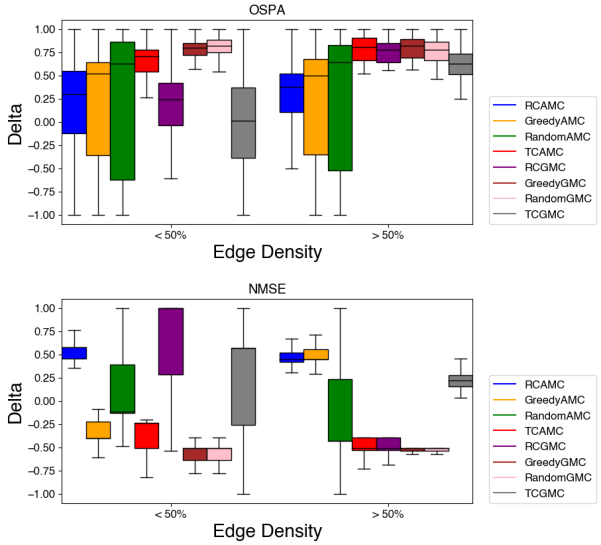


Fig. 4: The distribution of difference (normalized between -1 and 1) in NMSE and OSPA between the *baseline* scenario and each configuration strategy at different graph edge densities. A positive average difference means that the strategy resulted in a performance improvement. Note that results from all robot team sizes are aggregated here. Summary statistics for these results are presented in Table I. For lower edge densities, the, **TCAMC**, **GreedyGMC**, and **RandomGMC** strategies perform best in terms of OSPA, but with large standard deviation in results for **GreedyAMC** and **RandomAMC**. At higher edge densities, the **RCGMC** and **TCGMC** show improved OSPA and are comparable in performance to the **GreedyGMC** and **RandomGMC** strategies. In terms of NMSE, the **RCAMC** strategy shows the most consistency in improved performance over the baseline.

V. Simulation

To validate our approach, we conducted multiple simulations of a robot team tracking multiple moving targets. A target would be born in a random corner of 2D space with bounding box of $x \in [-50, 50]$, $y \in [-50, 50]$. The target would then move in a straight line trajectory to the opposite corner of the bounding box. The number of new targets born at each time step was determined by a random Poisson point process with the Poisson parameter value one. The initial position for a new born target was chosen according to a Gaussian distribution about a 20 unit radius of each corner. Fig 3 illustrates this target birth area.

For the local GM-PHD filter, we set the survival probability of each target to 0.98 and, detection probability of each target to 0.95 if within the field of vision of a robot and 0 otherwise. The clutter generated at each time step was determined by a random Poisson point process in the robot's field of vision with the Poisson parameter value 5. Targets within Mahalanobis distance of 0.2 were merged together at each local GM-PHD filter iteration and during fusion. Targets with weights less than 1e-6 were pruned.

We initiated the same H_k^ι and R_k^ι for each robot in the tracker team. We used $L = n/2$ for the consensus step, where n is the size of the team. Parameters chosen for the configuration generation and formation synthesis problems were $n_e = 1$, $d_s = 10$, $d_{mc} = 25$, and $d_{sen}^\iota = 20\ \forall \iota \in [n]$, with a bounding box of $x \in [-50, 50]$, $y \in [-50, 100]$, and $z \in [0, 100]$.

To simulate deteriorating sensor quality for ρ^ι, we modified its covariance matrix R_k^ι by adding a random positive definite matrix. We generated various deterioration event sequences for robot teams of $n \in \{5, 6, 7, 10, 12, 15, 20, 25, 30\}$ where a random robot was chosen at every f time step of the simulation to experience sensor deterioration. Fig 3 shows an overhead view of a single simulation trial with 5 robots.

For all configuration generation approaches, we simulated 30 deterioration sequences for each size robot team. We evaluate each configuration generation approach against a *baseline* scenario in which no edges are added at failure. Additionally, we compare our approaches to simple random strategy and a greedy strategy. For the random strategy, a single edge is added between the robot experiencing sensor deterioration and another robot selected at random. Depending on the fusion method used, we refer to the random strat-

N		5	7	10	15	20	30
AMC	agent	0.80	0.93	1.67	4.04	4.42	4.00
	team	0.88	1.17	1.78	4.06	4.53	4.07
GMC	agent	1.00	1.11	1.80	4.08	4.65	4.14
	team	7.12	27.52	114.01	451.03	2400.20	10109.2

TABLE II: Average clock time (in seconds) for computation of configuration generation and formation synthesis problems after a single instance of sensor deterioration for varying team sizes. Computation time increases with larger team sizes. All methods take approximately one to four seconds to compute with the exception of the **TCGMC** method, which may take several hours with very large teams.

egy as *Random Arithmetic Mean Configuration Generation* (**RandomAMC**) and *Random Geometric Mean Configuration Generation* (**RandomGMC**).

For the greedy strategy, a single edge is added to the network between the robot which experienced sensor deterioration and the robot with lowest covariance at the time of deterioration. Depending on the fusion method used, we refer to the greedy strategy as *Greedy Arithmetic Mean Configuration Generation* (**GreedyAMC**) and *Greedy Geometric Mean Configuration Generation* (**GreedyGMC**).

Each trial was initialized with a line graph. The target dynamics and distributed GM-PHD filter were implemented in Python. For the agent-centric and team-centric approaches, the MISDP problem was solved using our own custom implementation of branch-and-bound written in Python with PICOS as the optimization problem modeling interface and MOSEK as the semi-definite programming solver. In both approaches, the simulated annealing technique for formation synthesis was implemented in Python.

All simulation computations were performed on a 64-bit Ubuntu 18.04 desktop with two 3GHz Intel Core Xeon Gold 6154 CPU and 256 GB RAM. Additionally, we employed GNU-Parallel to parallelize our computations on this machine. Average clock time for computation on this machine for a single round of configuration generation and formation synthesis of varying team sizes are presented in Table II. We observed that the computation time increases with robot team size as expected, given the general NP-hard nature of MISDP problems [39].

To quantify performance, we used the optimal subpattern assignment (OSPA) distance [40] that evaluates the estimation error of the target positions after PHD fusion. The OSPA metric represents the distance between two sets. In our case, this is the distance between the set of the true target positions and the set of the T-GCs means. For the OSPA calculation, we use cutoff parameter c = 5 and order parameter p = 1.

Additionally, we evaluate the target set cardinality estimate of the team using the normalized mean squared error (NMSE). The results are presented in Fig 4 with summary statistics presented in Table I.

In comparing improvement in OSPA between strategies, the strategies using geometric mean fusion generally performed better than strategies using arithmetic mean fusion. Also, under the OSPA metric, we infer that the team-centric strategies perform better than their robot-centric counterpart. For lower edge densities, the **TCAMC**, **GreedyGMC**, and **RandomGMC** strategies performed relatively better than others. However, the **RCGMC** and **TCGMC** strategies improve considerably in OSPA at higher edge densities. On an average, at higher densities the **TCGMC** strategy outperformed even the **GreedyGMC** and **RandomGMC** and with smaller standard deviation. This is due to the scarcity in the availability of new edges to be added as the network becomes saturated. The random and greedy strategies will only add edges to increase connectivity to the robot experiencing sensor deterioration. In contrast, the team-centric strategy can find the best edge that overall improves the sensing performance of the network. Between the **TCAMC** and **TCGMC** strategies at higher edge densities, **TCGMC** yields better OSPA. Note that, due to the difficulty in formulating and solving optimization problems which directly involve the set distance-based OSPA metric, neither in the "robot-centric" nor in the "team-centric" formulations was the OSPA directly optimized. Instead, in both formulations we used surrogate objective functions based on the covariance of the GCs of GM-PHD associated with the targets. Our results illuminate the interplay between the proposed strategies and the edge density of the tracker network.

In selecting which strategy to use when optimizing for OSPA, we suggest the **TCAMC** strategy at lower edge densities and reserve the **TCGMC** strategy for high edge density situations. Note however that the **TCGMC** strategy may not be suitable for larger networks in which computation time is limited, due to the large time complexity of computing a solution for this strategy. Although as stated earlier AMF is more robust than GMF, this aspect is not reflected in our results. A potential reason for this could be that the GCs used for fusion may not be consistent.

In selecting which strategy to use in situations where only the *number* of detected targets is the prime focus, we suggest the **RCAMC** strategy as it showed the most consistency in performance improvement of NMSE compared to all other strategies across varying edge densities.

VI. Conclusion

This paper presents a novel strategy that facilitates a team of robots performing multi-target tracking to respond to a sensor fault in one of the team members by reconfiguring the team's communication network. The reconfigured team attenuates the adverse effect of sensor quality deterioration on multi-target tracking performance of the team. We presented four different MISDP formulations to compute the new robot team configuration. All formulations were validated in simulation and compared to each other. In future, we plan to validate our approach on our multi-robot testbed [41].

Appendix A

Derivation of v_{GMF} (Eq 18): By definition,

$$v_{GMF} = \prod_{j=1}^{n} \left[\sum_{i_j \in [g_j]} \alpha_{i_j} \mathcal{N}(\bar{s}; \bar{\mu}_{i_j}, P_{i_j}) \right]^{\omega_j}, \qquad (46)$$

due to the assumption in Eq 17 the expression can be approximated as

$$v_{GMF} \approx \prod_{j=1}^{n} \left(\sum_{i_j \in [g_j]} \left[\alpha_{i_j} \mathcal{N}(\bar{s}; \bar{\mu}_{i_j}, P_{i_j}) \right]^{\omega_j} \right). \tag{47}$$

From [14, Eq 36], this can be further simplified to:

$$v_{GMF} \approx \prod_{j=1}^{n} \left(\sum_{i_j \in [g_j]} \left[\alpha_{i_j}^{\omega_j} \mathcal{N}(\bar{s}; \bar{\mu}_{i_j}, \frac{P_{i_j}}{\omega_j}) \sqrt{\frac{\left| 2\pi \frac{P_{i_j}}{\omega_j} \right|}{|2\pi P_{i_j}|^{w_j}}} \right] \right).$$

Due to distributed property, the above equation can be expressed as the summation of various GC products. Mathematically,

$$v_{GMF} \approx \sum_{(i_1,i_2,\ldots,i_n) \in [g_{1\cdots n}]} \left[\alpha_{i_1}^{\omega_1} \mathcal{N}(\bar{s}; \bar{\mu}_{i_1}, \frac{P_{i_1}}{\omega_1}) \frac{\left| 2\pi \frac{P_{i_1}}{\omega_1} \right|^{\frac{1}{2}}}{|2\pi P_{i_1}|^{\frac{w_1}{2}}} \right.$$
$$\alpha_{i_2}^{\omega_2} \mathcal{N}(\bar{s}; \bar{\mu}_{i_2}, \frac{P_{i_2}}{\omega_2}) \frac{\left| 2\pi \frac{P_{i_2}}{\omega_2} \right|^{\frac{1}{2}}}{|2\pi P_{i_2}|^{\frac{w_2}{2}}} \cdots$$
$$\left. \alpha_{i_n}^{\omega_n} \frac{\left| 2\pi \frac{P_{i_n}}{\omega_n} \right|^{\frac{1}{2}}}{|2\pi P_{i_n}|^{\frac{w_n}{2}}} \mathcal{N}(\bar{s}; \bar{\mu}_{i_n}, \frac{P_{i_n}}{\omega_n}) \right],$$

where $[g_{1\cdots n}] \triangleq [g_1] \times [g_2] \times [g_3] \ldots [g_n]$. Thus if we derive the canonical expression for the product of GCs associated with the indices $i_1, i_2, \ldots, i_n$ then, v_{GMF} can be expressed as the summation of GC products. Also, it can be shown that the product of GCs also result in a GC [42]. Let

$$\alpha_{i_1,\cdots,i_n} \mathcal{N}(\bar{s}; \bar{\mu}_{i_1,\cdots,i_n}, P_{i_1,\cdots,i_n}) = \alpha_{i_1}^{\omega_1} \mathcal{N}(\bar{s}; \bar{\mu}_{i_1}, \frac{P_{i_1}}{\omega_1}) \frac{\left| 2\pi \frac{P_{i_1}}{\omega_1} \right|^{\frac{1}{2}}}{|2\pi P_{i_1}|^{\frac{w_1}{2}}}$$
$$\alpha_{i_2}^{\omega_2} \mathcal{N}(\bar{s}; \bar{\mu}_{i_2}, \frac{P_{i_2}}{\omega_2}) \frac{\left| 2\pi \frac{P_{i_2}}{\omega_2} \right|^{\frac{1}{2}}}{|2\pi P_{i_2}|^{\frac{w_2}{2}}} \cdots \alpha_{i_n}^{\omega_n} \mathcal{N}(\bar{s}; \bar{\mu}_{i_n}, \frac{P_{i_n}}{\omega_n}) \frac{\left| 2\pi \frac{P_{i_n}}{\omega_n} \right|^{\frac{1}{2}}}{|2\pi P_{i_n}|^{\frac{w_n}{2}}}$$

. Now by applying the formula for product Gaussian functions [42, Eqn 7-9] we obtain,

$$P_{i_1,\cdots,i_n} = \left(\sum_{j=1}^{n} \omega_j (P_{i_j}^j)^{-1} \right)^{-1} \tag{48}$$

$$\bar{\mu}_{i_1,\cdots,i_n} = (P_{i_1,\cdots,i_n})^{-1} \sum_{j=1}^{n} \omega_j (P_{i_j}^j)^{-1} \bar{\mu}_{i_j}^j \tag{49}$$

$$\alpha_{i_1,\cdots,i_n} = K \left(\prod_{j=1}^{n} (\alpha_{i_j}^j)^{\omega_j} \sqrt{\frac{\left| 2\pi \frac{P_{i_j}^j}{\omega_j} \right|}{|2\pi P_{i_j}^j|^{w_j}}} \right). \tag{50}$$

Hence,

$$v_{GMF} \approx \sum_{(i_1,i_2,\ldots,i_n) \in [g_{1\cdots n}]} \alpha_{i_1,\cdots,i_n} \mathcal{N}(\bar{s}; \bar{\mu}_{i_1,\cdots,i_n}, P_{i_1,\cdots,i_n}).$$

APPENDIX B

Proof of Theorem 1: Let $L = I - A$, then since A is doubly stochastic $L\bar{1}^n = \bar{0}^n$ and $L^T\bar{1}^n = \bar{0}^n$. Also, as the spectrum of A is real and less than or equal to one in magnitude, the spectrum of L is real and less than or equal to zero. Now, from the above statement we conclude that L is a positive semi-definite matrix. Furthermore, note that L can be interpreted as the Laplacian of a weighted undirected graph $\mathcal{G}_L$ having the same topology of the graph associated with A except for self loops. Since the connectivity properties of an undirected graph does not depend on the existence of self loops, original graph(graph associated with A) is connected if and only if $\mathcal{G}_L$ is connected. From [43, Proposition 1], we infer that $\mathcal{G}_L$ is connected if and only if $L + \frac{1}{n}\bar{1}\bar{1}^\top \succ 0$. Therefore, substituting $L = I - A$ in the equation yields $\frac{1}{n}\bar{1}\bar{1}^\top + I \succ A$.

APPENDIX C

Proof of Lemma 1: According to Schur complement lemma [44, Chapter 2], the following *linear matrix inequality*(LMI) [44], $\begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \succeq 0$ is equivalent to $R \succeq 0, Q - SR^{-1}S^T \succeq 0$. Therefore, the LMI Eq 36 is equivalent to $\bar{P} - \bar{\Delta}^{-1} \succeq 0$. Also, it is straightforward to see that Trace($\bar{P}$) $\geq Tr(\bar{\Delta}^{-1})$. Since, $\bar{\Delta}$ is the block diagonal matrix containing the posterior information matrices of all tracking robots, $\bar{\Delta}^{-1}$ is a block diagonal matrix with $\{P_{gm}^{1,\alpha}, P_{gm}^{2,\alpha}, \cdots, P_{gm}^{n,\alpha}\}$ along its diagonal. Therefore, $\frac{1}{n\times\alpha} Tr(\bar{P}) \geq \frac{1}{n\times\alpha} Tr(\bar{\Delta}^{-1})$ is equivalent to $\frac{1}{n\times\alpha} Tr(\bar{P}) \geq \frac{1}{n} \sum_i^{n\times\alpha} Tr(P_{gm}^{i,\alpha})$

REFERENCES

[1] K. Hausman, J. Müller, A. Hariharan, N. Ayanian, and G. S. Sukhatme, "Cooperative multi-robot control for target tracking with onboard sensing," *The International Journal of Robotics Research*, vol. 34, no. 13, pp. 1660–1677, 2015.

[2] G. Foderaro, P. Zhu, H. Wei, T. A. Wettergren, and S. Ferrari, "Distributed optimal control of sensor networks for dynamic target tracking," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 1, pp. 142–153, 2018.

[3] M. Liggins II, C.-Y. Chong, D. Hall, and J. Llinas, *Distributed data fusion for network-centric operations*. Crc Press, 2012.

[4] M. Andreetto, M. Pacher, D. Macii, L. Palopoli, and D. Fontanelli, "A distributed strategy for target tracking and rendezvous using uavs relying on visual information only," *Electronics*, vol. 7, no. 10, p. 211, 2018.

[5] P. M. Dames, "Distributed multi-target search and tracking using the phd filter," *Autonomous Robots*, pp. 1–17, 2019.

[6] K. Rudd, G. Foderaro, P. Zhu, and S. Ferrari, "A generalized reduced gradient method for the optimal control of very-large-scale robotic systems," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1226–1232, 2017.

[7] H. Wei, W. Lu, P. Zhu, S. Ferrari, M. Liu, R. H. Klein, S. Omidshafiei, and J. P. How, "Information value in nonparametric dirichlet-process gaussian-process (dpgp) mixture models," *Automatica*, vol. 74, pp. 360 – 368, 2016.

[8] 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.

[9] T. Sarkar, M. Roozbehani, and M. A. Dahleh, "Asymptotic network robustness," *IEEE Transactions on Control of Network Systems*, vol. 6, no. 2, pp. 812–821, 2018.

[10] F. Bullo, *Lectures on Network Systems*, 1st ed. Kindle Direct Publishing, 2019, with contributions by J. Cortes, F. Dorfler, and S. Martinez. [Online]. Available: http://motion.me.ucsb.edu/book-lns

[11] L. D. Stone, R. L. Streit, T. L. Corwin, and K. L. Bell, *Bayesian multiple target tracking*. Artech House, 2013.

[12] R. P. S. Mahler, "Multitarget bayes filtering via first-order multitarget moments," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1152–1178, Oct 2003.

[13] O. Erdinc, P. Willett, and Y. Bar-Shalom, "Probability hypothesis density filter for multitarget multisensor tracking," in *2005 7th International Conference on Information Fusion*, vol. 1, 2005, pp. 8 pp.–.

[14] G. Battistelli, L. Chisci, C. Fantacci, A. Farina, and A. Graziano, "Consensus cphd filter for distributed multitarget tracking," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 3, pp. 508–520, 2013.

[15] T. Li, F. Hlawatsch, and P. M. Djurić, "Cardinality-consensus-based phd filtering for distributed multitarget tracking," *IEEE Signal Processing Letters*, vol. 26, no. 1, pp. 49–53, 2018.

[16] K. Panta, D. E. Clark, and B.-N. Vo, "Data association and track management for the gaussian mixture probability hypothesis density filter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, no. 3, pp. 1003–1016, 2009.

[17] F. Papi, B. Vo, B. Vo, C. Fantacci, and M. Beard, "Generalized labeled multi-bernoulli approximation of multi-object densities," *IEEE Transactions on Signal Processing*, vol. 63, no. 20, pp. 5487–5497, Oct 2015.

[18] F. Pasqualetti, S. Zampieri, and F. Bullo, "Controllability metrics, limitations and algorithms for complex networks," in *2014 American Control Conference*, June 2014, pp. 3287–3292.

[19] D. Leitold, Á. Vathy-Fogarassy, and J. Abonyi, "Controllability and observability in complex networks–the effect of connection types," *Scientific reports*, vol. 7, no. 1, p. 151, 2017.

[20] R. K. Ramachandran and S. Berman, "The effect of communication topology on scalar field estimation by large networks with partially accessible measurements," in *2017 American Control Conference (ACC)*, May 2017, pp. 3886–3893.

[21] R. K. Ramachandran, J. A. Preiss, and G. S. Sukhatme, "Resilience by reconfiguration: Exploiting heterogeneity in robot teams," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Nov 2019, pp. 6518–6525.

[22] R. K. Ramachandran, N. Fronda, and G. S. Sukhatme, "Resilience in multi-robot target tracking through reconfiguration," in *IEEE/ICRA Int'l. Conf. on Robotics and Automation (ICRA)*, 2019, to appear. [Online]. Available: https://arxiv.org/abs/1910.01300

[23] A. B. Sharma, L. Golubchik, and R. Govindan, "Sensor faults: Detection methods and prevalence in real-world datasets," *ACM Transactions on Sensor Networks (TOSN)*, vol. 6, no. 3, p. 23, 2010.

[24] P. Pierpaoli, D. Sauter, and M. Egerstedt, "Fault tolerant control for networked mobile robots," *2018 IEEE Conference on Control Technology and Applications (CCTA)*, Aug 2018.

[25] L. Jiang, D. Djurdjanovic, J. Ni, and J. Lee, "Sensor degradation detection in linear systems," in *Engineering Asset Management*. Springer, 2006, pp. 1252–1260.

[26] T. Li, J. M. Corchado, and S. Sun, "Partial consensus and conservative fusion of gaussian mixtures for distributed phd fusion," *IEEE Transactions on Aerospace and Electronic Systems*, 2018.

[27] L. Guerrero-Bonilla, A. Prorok, and V. Kumar, "Formations for resilient robot teams," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 841–848, April 2017.

[28] R. A. Horn and C. R. Johnson, Eds., *Matrix Analysis*. New York, NY, USA: Cambridge University Press, 1986.

[29] R. P. Mahler, *Statistical multisource-multitarget information fusion*. Artech House, Inc., 2007, vol. 685.

[30] A. E. Abbas, "A kullback-leibler view of linear and log-linear pools," *Decision Analysis*, vol. 6, no. 1, pp. 25–37, 2009.

[31] T. Bailey, S. Julier, and G. Agamennoni, "On conservative fusion of information with unknown non-gaussian dependence," in *2012 15th International Conference on Information Fusion*. IEEE, 2012, pp. 1876–1883.

[32] B.-N. Vo and W.-K. Ma, "The gaussian mixture probability hypothesis density filter," *IEEE Transactions on signal processing*, vol. 54, no. 11, pp. 4091–4104, 2006.

[33] 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, Dec 2019.

[34] 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.

[35] W. Yi, M. Jiang, S. Li, and B. Wang, "Distributed sensor fusion for rfs density with consideration of limited sensing ability," in *2017 20th International Conference on Information Fusion (Fusion)*, July 2017, pp. 1–6.

[36] J. K. Uhlmann, "Dynamic map building and localization: New theoretical foundations," Ph.D. dissertation, University of Oxford, 1995.

[37] ——, "Covariance consistency methods for fault-tolerant distributed data fusion," *Information Fusion*, vol. 4, no. 3, pp. 201–215, 2003.

[38] M. Rafiee and A. M. Bayen, "Optimal network topology design in multi-agent systems for efficient average consensus," in *49th IEEE Conference on Decision and Control*. IEEE, 2010, pp. 3877–3883.

[39] B. Korte and J. Vygen, *Combinatorial Optimization: Theory and Algorithms*, 5th ed. Springer Publishing Company, Incorporated, 2012.

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

[41] J. A. Preiss*, W. Hönig*, G. S. Sukhatme, and N. Ayanian, "Crazyswarm: A large nano-quadcopter swarm," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 3299–3304.

[42] P. Bromiley, "Products and convolutions of gaussian probability density functions," http://www.lucamartino.altervista.org/2003-003.pdf, 2003.

[43] M. Sundin, A. Venkitaraman, M. Jansson, and S. Chatterjee, "A connectedness constraint for learning sparse graphs," in *2017 25th European Signal Processing Conference*. IEEE, 2017, pp. 151–155.

[44] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear matrix inequalities in system and control theory*. Siam, 1994, vol. 15.