

# Probabilistic Consensus on Feature Distribution for Multi-robot Systems with Markovian Exploration Dynamics

Aniket Shirsat<sup>1</sup>, Shatadal Mishra<sup>2</sup>, Wenlong Zhang<sup>2</sup>, and Spring Berman<sup>1</sup>

**Abstract**—In this paper, we present a consensus-based decentralized multi-robot approach to reconstruct a discrete distribution of features, modeled as an occupancy grid map, that represent information contained in a bounded planar 2D environment, such as visual cues used for navigation or semantic labels associated with object detection. The robots explore the environment according to a random walk modeled by a discrete-time discrete-state (DTDS) Markov chain and estimate the feature distribution from their own measurements and the estimates communicated by neighboring robots, using a distributed Chernoff fusion protocol. We prove that under this decentralized fusion protocol, each robot's feature distribution converges to the ground truth distribution in an almost sure sense. We verify this result in numerical simulations that show that the Hellinger distance between the estimated and ground truth feature distributions converges to zero over time for each robot. We also validate our strategy through Software-In-The-Loop (SITL) simulations of quadrotors that search a bounded square grid for a set of visual features distributed on a discretized circle.

**Index Terms**—Multi-Robot Systems; Mapping; Probability and Statistical Methods

## I. INTRODUCTION

**M**ULTI-ROBOT systems (MRS) composed of multiple mobile robots have been used for various collective exploration and perception tasks, such as mapping unknown environments [1], disaster response [2], and surveillance and monitoring [3], herding a group of robots [4]. The performance of MRS in such applications is constrained by the capabilities of the payloads that the robots can carry on-board, including the power source, sensor suite, computational resources, and communication devices for transmitting information to other robots and/or a central node. These constraints are particularly restrictive in the case of small aerial robots such as multi-rotors that perform vision-guided tasks [5].

Centralized MRS strategies for exploration and mapping, such as the next-best-view planning method in [6], rely on constant communication between all the robots and a central node.

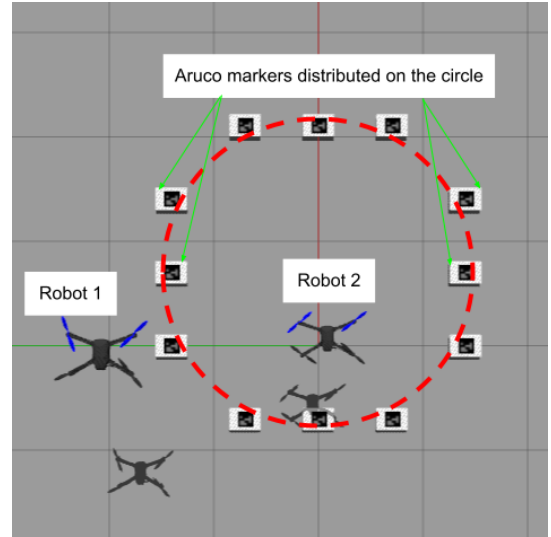
Manuscript Received: October 22, 2021; Revised: January 23, 2022; March 18, 2022; Accepted: April 15, 2022.

This paper was recommended for publication by Editor M. Ani Hsieh upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by the Arizona State University Global Security Initiative.

<sup>1</sup> Aniket Shirsat and Spring Berman are with the School for Engineering of Matter, Transport and Energy, Arizona State University, Tempe, AZ, 85287 USA {ashirsat, Spring.Berman}@asu.edu

<sup>2</sup> Shatadal Mishra and Wenlong Zhang are with the Polytechnic School, Arizona State University, Mesa, AZ 85212, USA {smishr13, Wenlong.Zhang}@asu.edu

Digital Object Identifier (DOI): see top of this page.



**Fig. 1:** Software-In-The-Loop (SITL) setup in Gazebo using the Rotors [8] package. Robots 1 and 2 are quadrotors, and the ArUco markers represent a discrete approximation of a continuous circular feature (the red dotted line).

Scaling up such strategies with the number of robots requires expanding the communication infrastructure and preventing communication failures of the central node. Frontier-based MRS exploration strategies such as [7] rely on a dynamic communication topology in which leaders are responsible for coordinating the team.

Decentralized MRS exploration and mapping strategies that employ only local communication alleviate these drawbacks and are designed to work robustly under inter-robot communication bandwidth constraints [9] and disruptions to communication links by environmental effects [10]. Many decentralized MRS estimation strategies are designed to achieve *consensus* among the robots on a particular variable or property of interest through local inter-robot communication. For example, distributed consensus-based approaches have been designed for spacecraft attitude estimation [11] and space debris tracking [12]. Consensus behaviors also arise in social networks [13] when users reach an agreement on a shared opinion in a distributed fashion. Consensus strategies have been developed for MRS communication networks that are static or dynamic, and that can be represented as directed or undirected graphs [14], as well as random networks [15] and networks with communication delays [16]. However, few works address consensus problems for MRS that follow random mobility models,

often used in MRS exploration strategies as in, e.g., [17], whose communication networks exhibit Markovian switching dynamics as a result. Random exploration strategies have certain advantages for MRS: they do not require centralized motion planning, localization, or communication, and they can be modified to produce more focused or more dispersed coverage.

In our previous work [18], we developed a probabilistic consensus-based strategy for target search by MRS with a discrete-time discrete-state (DTDS) Markov motion model and local sensing and communication. Using this strategy, the robots are guaranteed to achieve consensus almost surely on the presence of a static feature of interest, without any requirements on the connectivity of the robots' communication network. We extended this approach in [19] to an MRS strategy for tracking multiple static features by formulating the tracking procedure as a renewal-reward process on the underlying Markov chain. The robots reach a consensus on the number of features and their locations in a decentralized manner using a Gaussian Mixture approximation of the Probability Hypothesis Density (PHD) filter.

In this paper, we generalize the consensus objective of our probabilistic multi-robot search strategy to agreement on a discrete distribution of static features, modeled as an occupancy grid map, using results on *opinion pools* [20]. We consider a group of robots that move according to a DTDS Markov chain on a finite 2D spatial grid, as shown in Figure 2, and that can detect features using their on-board sensors. The proposed strategy is distributed and asynchronous, and it preserves the required communication bandwidth by relying only on local inter-robot communication. The main contributions of the paper are as follows:

- 1) We present a decentralized, stochastic multi-robot exploration and mapping strategy in which the robots use a consensus protocol, without communication connectivity requirements, to arrive at a common reconstruction of a feature distribution on a 2D grid. Specifically, given a group of robots with a DTDS Markov motion model and local sensing and communication, we prove that if the robots update their estimates of the feature distribution with those of other robots according to a distributed Chernoff fusion protocol, then they will reach consensus almost surely on the ground truth distribution. This extends the result in [21] on opinion consensus over fixed, strongly connected networks to networks with Markovian switching dynamics.
- 2) We validate our theoretical results in numerical simulations that illustrate the pathwise convergence to zero of the Hellinger distance between each robot's estimate of the feature distribution and the ground truth distribution. We also validate our approach in Software-In-The-Loop (SITL) simulations of quadrotors, performed in Gazebo using the Robot Operating System (ROS) with the PX4 autopilot.

The remainder of the paper is organized as follows. We present our probabilistic exploration strategy and information fusion protocol in Section II. We describe some relevant

properties of DTDS Markov chains in Section III, and we derive the main result that guarantees the convergence of each robot's feature distribution to the ground truth distribution in Section IV. We present the results of our numerical simulations in Section V and our SITL simulations in Section VI. Section VII concludes the paper and suggests directions for future work. A video overview of the paper is provided at <https://youtu.be/-Z4-DZrHwSM>.

## II. EXPLORATION AND INFORMATION FUSION STRATEGY

Consider a bounded square environment  $\mathcal{B} \subset \mathbb{R}^2$  with sides of length  $B$ . We discretize  $\mathcal{B}$  into a square grid of nodes spaced at a distance  $\delta$  apart. The set of nodes is denoted by  $\mathcal{S} \subset \mathbb{Z}_+$ , and we define  $S = |\mathcal{S}|$ . A set of  $N$  robots,  $\mathcal{A} = \{1, 2, \dots, N\}$ , each modeled as a point mass, explore the environment by performing a random walk on the grid. We assume that there are no obstacles in the environment that impede the robots' motion. Let  $\mathcal{G}_s = (\mathcal{V}_s, \mathcal{E}_s)$  be an undirected graph associated with this finite spatial grid, where  $\mathcal{V}_s = \mathcal{S}$  is the set of nodes and  $\mathcal{E}_s$  is the set of edges  $(i, j)$ . The edges signify pairs of nodes  $i, j \in \mathcal{V}_s$ , called *neighboring nodes*, between which robots can travel. We assume that the robots can localize on  $\mathcal{G}_s$ .

Let  $Y_k^a \in \mathcal{S}$  be a random variable that defines the node that robot  $a \in \mathcal{A}$  occupies at discrete time  $k$ . Robot  $a$  moves from its current node  $i$  to a neighboring node  $j$  at the next time step with a transition probability  $p_{ij} \in [0, 1]$ . We define  $\mathbf{P} \in \mathbb{R}^{S \times S}$  as the *state transition matrix* consisting of elements  $p_{ij}$  at row  $i$  and column  $j$ . Let  $\pi_k \in \mathbb{R}^{1 \times S}$  denote the probability mass function (PMF) of  $Y_k^a$  for each robot  $a$ , or alternatively, the distribution of the robot population over the grid at time  $k$ . This distribution evolves over time according to a DTDS Markov chain model of order one:

$$\pi_{k+1} = \pi_k \mathbf{P}. \quad (1)$$

We assume that each robot can exchange information with other robots that are within a communication radius  $r_{comm} < 0.5\delta$ . Let  $\mathcal{G}_c[k] = (\mathcal{V}_c, \mathcal{E}_c[k])$  be an undirected graph in which  $\mathcal{V}_c = \mathcal{A}$ , the set of robots, and  $\mathcal{E}_c[k]$  is the set of all pairs of robots  $(a, b) \in \mathcal{A} \times \mathcal{A}$  that can communicate with each other at time  $k$ . Let  $\mathbf{M}[k] \in \mathbb{R}^{N \times N}$  be the *adjacency matrix* with elements  $m_{ab}[k] = 1$  if  $(a, b) \in \mathcal{E}_c[k]$  and  $m_{ab}[k] = 0$  otherwise. For each robot  $a \in \mathcal{A}$ , we define the set of *neighbors* of robot  $a$  at time  $k$  as  $\mathcal{N}_k^a \triangleq \{b \in \mathcal{A} : (a, b) \in \mathcal{E}_c[k]\}$ .

A set of discrete features is distributed over the grid at nodes in the set  $\mathcal{B}^r \subseteq \mathcal{S}$ . The robots know a priori that these features are present in the environment, but do not know their distribution. We assume that when a robot is located at a node in  $\mathcal{B}^r$ , it can detect the presence of a feature at that node using its on-board sensors. Each node in the grid is associated with a binary occupancy value, defined as  $\bar{l} \in (0.5, 1)$  if the robot detects a feature at that node and  $1 - \bar{l}$  if it does not. Setting  $\bar{l} \in (0.5, 1)$  helps produce sharp reconstructions of the features: as the value of  $\bar{l}$  increases, the distinction between occupied and unoccupied nodes becomes clearer.

We define the *occupancy vector* for robot  $a$  at time  $k$  as  $\bar{\theta}_k^a = [\theta_k^a(1) \dots \theta_k^a(S)] \in \mathbb{R}^{1 \times S}$ , where

$$\theta_k^a(s) = \begin{cases} \bar{l}, & s \in \mathcal{B}^r \text{ (occupied)} \\ 1 - \bar{l}, & o.w \text{ (unoccupied)} \end{cases} \quad (2)$$

The occupancy vector for each robot indicates its estimate of the nodes that are occupied by features. The feature PMF, or *occupancy distribution*, estimated by robot  $a$  at time  $k$  from its own sensor measurements is defined as:

$$f_k^a(s) = \frac{\theta_k^a(s)}{\sum_{i \in \mathcal{S}} \theta_k^a(i)} \quad (3)$$

Here,  $f_k^a(s)$  also represents the *opinion* [22] of robot  $a$  at time  $k$  for the occupancy distribution. We denote  $\theta^{ref} \in \mathbb{R}^{1 \times S}$  as the *reference occupancy vector* that is being estimated by all the robots, defined as follows:

$$\theta^{ref}(s) = \begin{cases} \bar{l}, & s \in \mathcal{B}^r \\ 1 - \bar{l}, & o.w \end{cases} \quad (4)$$

Since the robots do not know the occupancy distribution a priori, we specify that they all initially consider the grid to be unoccupied, i.e.,

$$f_0^a(s) = \frac{1 - \bar{l}}{S(1 - \bar{l})}, \quad a \in \mathcal{A}, \forall s \in \mathcal{S} \quad (5)$$

This is defined as the *nominal distribution* for all the robots, denoted by  $f^{nom}(s) = f_0^{(\cdot)}(s)$ . We also denote a vector  $\theta^{nom} \in \mathbb{R}^{1 \times S}$  as the *nominal occupancy vector* for all robots, which represents all nodes as unoccupied (i.e.,  $\theta^{nom}(s) = \theta_0^{(\cdot)}(s) = 1 - \bar{l}$ ,  $\forall s \in \mathcal{S}$ ). We define  $f^{ref}(s)$  as the *reference PMF*, the ground truth feature distribution, corresponding to  $\theta^{ref}(s)$ . We define a fusion weight  $\omega_k^{(a,b)}$  as the following *Metropolis weight* [23]:

$$\omega_k^{(a,b)} = \begin{cases} \frac{1}{1 + |\mathcal{N}_k^a|}, & b \in \mathcal{N}_k^a \setminus \{a\} \\ 1 - \sum_{b \in \mathcal{N}_k^a \setminus \{a\}} \omega_k^{(a,b)}, & a = b, a \in \mathcal{A} \\ 0, & o.w \end{cases} \quad (6)$$

Note that  $\omega_k^{(a,b)} \geq 0$  and  $\sum_{b \in \mathcal{N}_k^a} \omega_k^{(a,b)} = 1$ . We then define the *consensus* or *opinion weighting matrix*  $\Omega_k \in \mathbb{R}^{N \times N}$  at time  $k$ , which consists of elements  $\omega_k^{(a,b)}$  at row  $a$  and column  $b$ .

Given the robot exploration dynamics in Equation (1), each robot  $a$  updates its opinion  $f_k^a(s)$ , computed from Equation (3), to the following PMF  $f_{k+1}^{cher}(s)$  at the next time step, which it computes from the opinions of other robots within its communication range according to the *Chernoff fusion rule* [24]:

$$f_{k+1}^{cher}(s) = \frac{\prod_{b \in \mathcal{N}_k^a \cup \{a\}} [f_k^b(s)]^{\omega_k^{(a,b)}}}{\sum_{s \in \mathcal{S}} \prod_{b \in \mathcal{N}_k^a \cup \{a\}} [f_k^b(s)]^{\omega_k^{(a,b)}}}, \quad a \in \mathcal{A} \quad (7)$$

Applying Theorem 1 from [21], we can say that  $f_{k+1}^{cher}(s)$  is the local neighbor fused feature PMF at time  $k+1$  of the all robots  $a \in \mathcal{A}$  that are in  $\mathcal{N}_k^a$ . When only two robots  $a, b \in \mathcal{A}$  are within communication range, this update rule becomes:

$$f_{k+1}^{cher}(s) = \frac{[f_k^a(s)]^\omega [f_k^b(s)]^{1-\omega}}{\sum_{s \in \mathcal{S}} [f_k^a(s)]^\omega [f_k^b(s)]^{1-\omega}}, \quad (8)$$

**Algorithm 1:** Distributed Chernoff fusion protocol for robots  $a, b \in \mathcal{A}$  computed by robot  $a$  at time  $k$

---

**input :**  $f_k^a, f_k^b, Y_k^a, Y_k^b, k, T, \epsilon$   
**output:**  $f_{k+1}^a$   
**if**  $1 < k \leq T$  **then**  
    **if**  $Y_k^a = Y_k^b$  **then**  
         $\omega = 0.5$   
    **else**  
         $\omega = 1.0$   
    **end**  
     $c = \sum_{s \in \mathcal{S}} [f_k^a(s)]^\omega \cdot [f_k^b(s)]^{1-\omega}$   
     $\log f_{cher} = \omega \log(f_k^a) + (1 - \omega) \log(f_k^b) - \log(c)$   
     $f_{k+1}^a = f_{k+1}^{cher} = \exp(\log f_{cher})$   
**end**

---

where the Metropolis weight  $\omega \equiv \omega_k^{(a,b)}$  is defined in Equation (6). Then each robot  $a$  compares  $f_{k+1}^{cher}(s)$  with  $f^{nom}(s)$  to generate a new fused occupancy vector as follows:

$$\theta_{k+1}^{cher}(s) = \begin{cases} \bar{l} & f_{k+1}^{cher}(s) > f^{nom}(s), s \in \mathcal{S} \\ 1 - \bar{l} & o.w \end{cases} \quad (9)$$

Then, each robot  $a$  generates a new occupancy vector  $\theta_{k+1}^a(s)$  by comparing both its occupancy vector at the previous time step,  $\theta_k^a(s)$ , and the fused occupancy vector  $\theta_{k+1}^{cher}(s)$ , to  $\theta^{nom}(s)$ :

$$\theta_{k+1}^a(s) = \begin{cases} \bar{l} & \theta_k^a(s) > \theta^{nom}(s) \text{ or} \\ & \theta_{k+1}^{cher}(s) > \theta^{nom}(s), s \in \mathcal{S} \\ \theta_k^a(s) & o.w \end{cases} \quad (10)$$

To quantify the convergence of each robot's feature distribution to the reference distribution, we use the *Hellinger metric*, which measures the similarity between two PMFs. The Hellinger distance between the feature PMF  $f_k^a(s)$  of a robot  $a \in \mathcal{A}$  at time  $k$  and the reference PMF  $f^{ref}(s)$  is given by

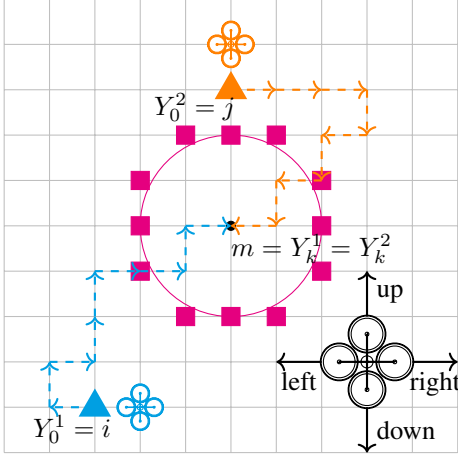
$$\mathbf{D}_H(f_k^a(s), f^{ref}(s)) = \sqrt{1 - \rho(f_k^a(s), f^{ref}(s))}, \quad (11)$$

where  $\rho(f_k^a(s), f^{ref}(s))$  is the *Bhattacharya coefficient*, defined as:

$$\rho(f_k^a(s), f^{ref}(s)) = \sum_{s \in \mathcal{S}} \sqrt{f_k^a(s) \cdot f^{ref}(s)}. \quad (12)$$

This distance takes values in  $[0, 1]$ . We define the vector  $\mathbf{D}_H \in \mathbb{R}_{\geq 0}^{N \times 1}$  with each entry  $a \in \mathcal{A}$  given by  $\mathbf{D}_H(f_k^a(s), f^{ref}(s))$ . The pseudo code in Algorithm 1 implements this fusion strategy for two robots  $a$  and  $b$ . In this algorithm, the normalizing constant  $c$  is the denominator of Equation (8).

Figure 2 illustrates the proposed strategy for a scenario with two quadrotors. The quadrotors start at the spatial grid nodes indexed by  $i$  and  $j$  and move on the grid according to the DTDS Markov chain dynamics in (1). The figure shows sample paths of the quadrotors. The orange quadrotor detects the feature, indicated by a magenta square, when it moves to a node in the set  $\mathcal{B}^r$  (at these nodes, the feature is within the quadrotor's sensing range). The quadrotors meet at grid node



**Fig. 2:** Illustration of our multi-robot exploration strategy, showing sample paths for two quadrotors (orange and blue) on a square grid. The quadrotors search the environment for a set of static features (the magenta squares representing a discretized circle) as they perform a random walk on the grid.

$m$  after  $k = 9$  time steps and fuse occupancy distributions according to Equation (8). They continue to random-walk on the grid and update  $f_k^a(s)$  until a specified final time  $T$ . We chose  $T$  empirically based on feature PMF convergence times from numerical simulation studies, shown in Figure 8, and on flight time constraints of off-the-shelf quadrotors.

### III. PROPERTIES OF DTDS MARKOV CHAINS

The Markov chain in Equation (1) is characterized in terms of the time-invariant state transition matrix  $\mathbf{P}$ , which is defined by the state space of the spatial grid representing the discretized environment. Hence, the Markov chain is *time-homogeneous*, which implies that  $Pr(Y_{k+1}^a = j \mid Y_k^a = i)$  is the same for all robots at all times  $k$ . The entries of  $\mathbf{P}$ , which are the state transition probabilities, can therefore be defined as

$$p_{ij} = Pr(Y_{k+1}^a = j \mid Y_k^a = i), \quad \forall i, j \in \mathcal{S}, \quad k \in \mathbb{Z}_+, \quad \forall a \in \mathcal{A}. \quad (13)$$

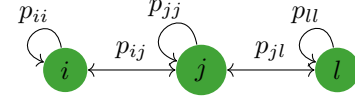
Since each robot chooses its next node from a uniform distribution, these entries can be computed as

$$p_{ij} = \begin{cases} \frac{1}{d_i + 1}, & (i, j) \in \mathcal{E}_s, \\ 0, & o.w, \end{cases} \quad (14)$$

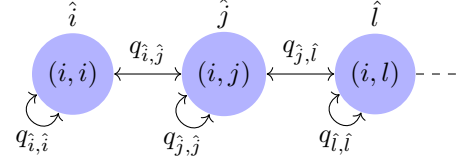
where  $d_i$  is the degree of the node  $i \in \mathcal{S}$ , defined as  $d_i = 2$  if  $i$  is a corner node of the spatial grid,  $d_i = 3$  if it is on an edge between two corner nodes, and  $d_i = 4$  otherwise. Since each entry  $p_{ij} \geq 0$ , we use the notation  $\mathbf{P} \geq 0$ . We see that  $\mathbf{P}^m \geq 0$  for  $m \geq 1$ , and therefore  $\mathbf{P}$  is a *non-negative matrix*. From Theorem 5 in [25], we can conclude that  $\mathbf{P}$  is a stochastic matrix. If  $\pi$  is a stationary distribution of Markov chain (1), then  $\forall k \in \mathbb{Z}_+$ ,

$$\pi \mathbf{P}^k = \pi. \quad (15)$$

From the construction of the Markov chain, each robot has a positive probability of moving from any node  $i \in \mathcal{S}$  to any other node  $j \in \mathcal{S}$  of the spatial grid in a finite number



**Fig. 3:** A graph  $\mathcal{G}_s = (\mathcal{V}_s, \mathcal{E}_s)$  defined on the set of spatial nodes  $\mathcal{V}_s = \{i, j, l\}$ . The arrows signify undirected edges between pairs of distinct nodes and self-edges. The edge set of the graph is  $\mathcal{E}_s = \{(i, i), (j, j), (l, l), (i, j), (j, l)\}$ .



**Fig. 4:** A subset of the composite graph  $\hat{\mathcal{G}} = (\hat{\mathcal{V}}, \hat{\mathcal{E}})$  for 2 robots that move on the graph  $\mathcal{G}_s$  shown in Figure 3.

of time steps. As a result, the Markov chain is *irreducible*, and therefore  $\mathbf{P}$  is an irreducible matrix. By Lemma 8.4.4 (Perron-Frobenius) in [26], there exists a real unique positive left eigenvector of  $\mathbf{P}$  and since  $\mathbf{P}$  is a stochastic matrix, its spectral radius  $\rho(\mathbf{P})$  is 1. Therefore, this left eigenvector is the stationary distribution of the corresponding Markov chain. Since we have shown that the Markov chain is irreducible and has a stationary distribution  $\pi$ , which satisfies  $\pi \mathbf{P} = \pi$ , we can conclude from Theorem 21.12 in [27] that the Markov chain is *positive recurrent*. Thus, all states in the Markov chain are positive recurrent, which implies that each robot will keep visiting every state on the finite spatial grid infinitely often.

### IV. CONSENSUS ON THE FEATURE DISTRIBUTION

By Theorem 1 of [21], Equation (8) achieves opinion consensus over a graph with a fixed and strongly connected topology. We extend this result to graphs with topologies that evolve according to the switching dynamics on the composite Markov chain described in this section. We demonstrate that under our opinion fusion scheme, all the robots will reach a consensus on the feature distribution.

The dynamics of all robots' movements on the spatial grid can be modeled by a composite Markov chain with states  $\mathbf{Y}_k = (Y_k^1, Y_k^2, \dots, Y_k^N) \in \mathcal{M}$ , where  $\mathcal{M} = \mathcal{S}^{\mathcal{A}}$ . Note that  $S = |\mathcal{S}|$  and  $|\mathcal{M}| = S^N$ . We define an undirected graph  $\hat{\mathcal{G}} = (\hat{\mathcal{V}}, \hat{\mathcal{E}})$  that is associated with the composite Markov chain. The vertex set  $\hat{\mathcal{V}}$  is the set of all possible realizations  $\hat{i} \in \mathcal{M}$  of  $\mathbf{Y}_k$ . The notation  $\hat{i}(a)$  represents the  $a^{\text{th}}$  entry of  $\hat{i}$ , which is the node  $i \in \mathcal{S}$  occupied by robot  $a$ . We define the edge set  $\hat{\mathcal{E}}$  as follows:  $(\hat{i}, \hat{j}) \in \hat{\mathcal{E}}$  if and only if  $(\hat{i}(a), \hat{j}(a)) \in \mathcal{E}_s$  for all robots  $a \in \mathcal{N}$ . Let  $\mathbf{Q} \in \mathbb{R}^{|\mathcal{M}| \times |\mathcal{M}|}$  be the state transition matrix associated with the composite Markov chain. The elements of  $\mathbf{Q}$ , denoted by  $q_{\hat{i}\hat{j}}$ , are computed from the transition probabilities defined in Equation (14) as follows:

$$q_{\hat{i}\hat{j}} = \prod_{a=1}^N p_{\hat{i}(a)\hat{j}(a)}, \quad \forall \hat{i}, \hat{j} \in \mathcal{M}. \quad (16)$$

Here,  $q_{\hat{i}\hat{j}}$  is the probability that in the next time step, each robot  $a$  will move from node  $\hat{i}(a)$  to node  $\hat{j}(a)$ .

For example, consider a set of two robots,  $\mathcal{N} = \{1, 2\}$ , that move on the graph  $\mathcal{G}_s$  in Figure 3. In the next time step, the robots can stay at their current node or travel between nodes  $i$  and  $j$  and between nodes  $j$  and  $l$ , but they cannot travel between nodes  $i$  and  $l$ . Figure 4 shows a subset of the resulting composite graph  $\hat{\mathcal{G}}$ , whose entire set of nodes is  $\hat{\mathcal{V}} = \{(i, i), (i, j), (i, l), (j, i), (j, j), (j, l), (l, i), (l, j), (l, l)\}$ . Each node in  $\hat{\mathcal{V}}$  is labeled by an index  $\hat{i}$ , e.g.,  $\hat{i} = (i, j)$ , with  $\hat{i}(1) = i$  and  $\hat{i}(2) = j$ . Given the connectivity of the spatial grid defined by  $\mathcal{E}_s$ , we can for example identify  $((i, j), (i, l))$  as an edge in  $\hat{\mathcal{E}}$ , but not  $((i, j), (l, l))$ . Since  $N = 2$  and  $S = 3$ , we have that  $|\mathcal{M}| = 3^2 = 9$ . For each  $\hat{i}, \hat{j} \in \hat{\mathcal{V}}$ , we can compute the transition probabilities in  $\mathbf{Q} \in \mathbb{R}^{9 \times 9}$  from Equation (16) as follows:

$$q_{\hat{i}\hat{j}} = \Pr(\mathbf{Y}_{k+1} = \hat{j} \mid \mathbf{Y}_k = \hat{i}) = p_{\hat{i}(1)\hat{j}(1)}p_{\hat{i}(2)\hat{j}(2)}, \quad k \in \mathbb{Z}_+. \quad (17)$$

We now prove that all robots will reach consensus on the feature distribution and it will converge to the reference distribution.

**Theorem IV-A.** *Consider a group of  $N$  robots, moving on a finite spatial grid with DTDS Markov chain dynamics Equation (1), that update their opinions  $f_k^a(s)$  for the feature distribution on the grid according to Equations (3), (7), (9) and (10). Then for each robot  $a \in \mathcal{A}$ ,  $f_k^a(s)$  will converge to  $f^{ref}(s)$  as  $k \rightarrow \infty$  almost surely.*

*Proof.* Suppose at initial time  $k_0$ , the locations of the robots on the spatial grid are given by the node  $\hat{i} \in \hat{\mathcal{V}}$ . Consider another set of robot locations at a future time  $k_0 + k$ , given by the node  $\hat{j} \in \hat{\mathcal{V}}$ . The transition of the robots from configuration  $\hat{i}$  to configuration  $\hat{j}$  in  $k$  time steps corresponds to a random walk of length  $k$  on the composite Markov chain  $\mathbf{Y}_k$  from node  $\hat{i}$  to node  $\hat{j}$ . It also corresponds to a random walk by each robot  $a$  on the spatial grid from node  $\hat{i}(a)$  to node  $\hat{j}(a)$  in  $k$  time steps. By construction, the graph  $\mathcal{G}_s$  is strongly connected and each of its nodes has a self-edge. Thus, there exists a discrete time  $n > 0$  such that, for each robot  $a$ , there exists a random walk on the spatial grid from node  $\hat{i}(a)$  to node  $\hat{j}(a)$  in  $n$  time steps. Consequently, there always exists a random walk of length  $n$  on the composite Markov chain  $\mathbf{Y}_k$  from node  $\hat{i}$  to node  $\hat{j}$ , and therefore  $\mathbf{Y}_k$  is an irreducible Markov chain. All states of an irreducible Markov chain belong to a single communication class. In this case, all states are *positive recurrent*; as a result, each state of  $\mathbf{Y}_k$  is visited infinitely often by the group of robots. Moreover, because the composite Markov chain is irreducible, we can conclude that  $\cup_{k \in \mathbb{Z}_+} \mathcal{G}_c[k] = \mathcal{G}_0$ , where  $\mathcal{G}_0$  is the complete graph on the set of robots. Therefore  $\mathcal{G}_0$  is strongly connected. Hence, each robot will meet every other robot at some node  $s \in \mathcal{S}$  infinitely often. Since  $\mathbf{Y}_k$  is irreducible and, from Equation (10), we have that  $\theta_k^a(s) \leq \theta_{k+1}^a(s) \leq \theta^{ref}(s)$ ,  $\forall a \in \mathcal{A}$ ,  $\forall s \in \mathcal{S}$ , it follows from Equations (9) and (10) that  $\theta_k^a(s) \rightarrow \theta^{ref}(s)$  as  $k \rightarrow \infty$ . Consequently,  $f_k^a(s) \rightarrow f^{ref}(s)$  as  $k \rightarrow \infty$  almost surely.  $\square$

## V. NUMERICAL SIMULATION RESULTS

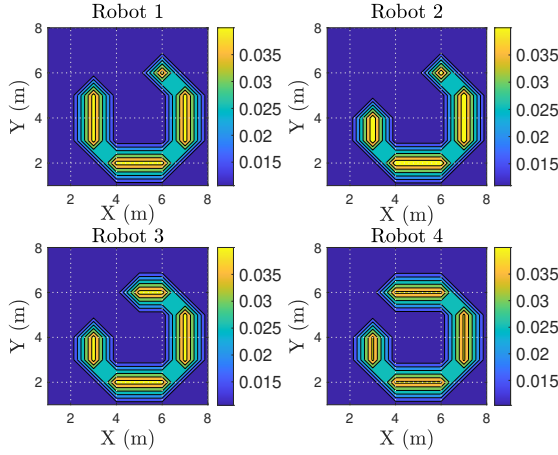
In the numerical simulations, we consider a set of robots  $\mathcal{A} = \{1, 2, 3, 4\}$  moving on a 5.6 m  $\times$  5.6 m domain that is discretized into a square grid with  $c = 8$  nodes on each side, with a distance  $\delta = 0.7$  m between adjacent nodes. The robots switch from one node to another at each time step according to the Markov chain dynamics in Equation (1). The state transition probabilities  $p_{ij}$ ,  $i, j \in \mathcal{S} = \{1, 2, \dots, c^2\}$ , that are associated with the spatial graph  $\mathcal{G}_s$  are computed from Equation (14). We set the value of  $\bar{l} = 0.8$  in Equation (2) in both the numerical and Software-In-The-Loop simulations. We distribute the features on the set of nodes  $\mathcal{B}^r = \{19, 20, 21, 26, 30, 34, 38, 42, 46, 51, 52, 53\}$ , which represents a discrete approximation of a circular distribution on the grid. The set of neighbors  $\mathcal{N}_k^a$  of robot  $a$  at time  $k$  consists of all robots that are located at the same node as robot  $a$  at that time.

All robots are initialized at uniformly random nodes in  $\mathcal{S}$ . Prior to exploration, the robots assume that all the grid nodes are unoccupied by features, and hence the vector of Hellinger distances is initially  $\mathbf{D}_H = \mathbf{0} \in \mathbb{R}^{N \times 1}$ . During their exploration of the grid, when robots encounter each other at the same node, they exchange their current feature PMFs and fuse them according to Equation (7). Figure 5 and Figure 6 show the feature PMFs computed by each robot at  $k = 240$  s and  $k = 500$  s, respectively. We observe in Figure 5 that by  $k = 240$  s, all robots have partially reconstructed the feature PMF, with robot 4 having the closest reconstruction as measured by  $\mathbf{D}_H(f_{240}^4(s), f^{ref}(s)) \approx 0.07$  in Figure 7. In Figure 6, we see that all robots have successfully reconstructed the feature PMF by  $k = 500$  s. The robots' consensus on the reference PMF is also apparent from Figure 7, which shows that all Hellinger distances are zero at that time. Several distances  $\mathbf{D}_H(f_k^a(s), f^{ref}(s))$  increase at times  $k$  between 400 s and 500 s, due to the numerical inaccuracies in the fusion.

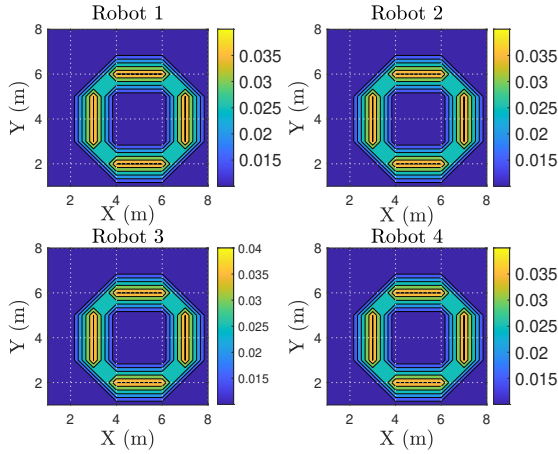
We also ran Monte Carlo numerical simulations with different numbers of robots,  $N = \{4, 8, 12, 16\}$ , to investigate the effect of  $N$  and the effect of consensus on the performance of the strategy in terms of the time for all robots' feature PMFs to converge to the reference PMF. Figure 8 plots the resulting time, averaged over 100 simulations (error bars show standard deviations), until the feature PMFs of all  $N$  robots converge to the reference PMF in the consensus and no-consensus cases. The figure shows that for  $N = 4$  robots, the mean time until convergence for both the consensus and no-consensus strategies are similar, with a significant overlap in their standard deviations. This indicates that for small numbers of robots, both strategies perform similarly. However, as  $N$  increases, there is a widening gap between the mean times until convergence of the consensus and no-consensus strategies, with the times for the strategy with consensus being consistently lower. For  $N = 16$  robots, the strategy with consensus is faster than the one without consensus by a factor of  $\sim 2$  (830 s/438 s).

## VI. SOFTWARE-IN-THE-LOOP RESULTS

We also validate our approach in Software-In-The-Loop (SITL) simulations using Gazebo and ROS Melodic. We sim-

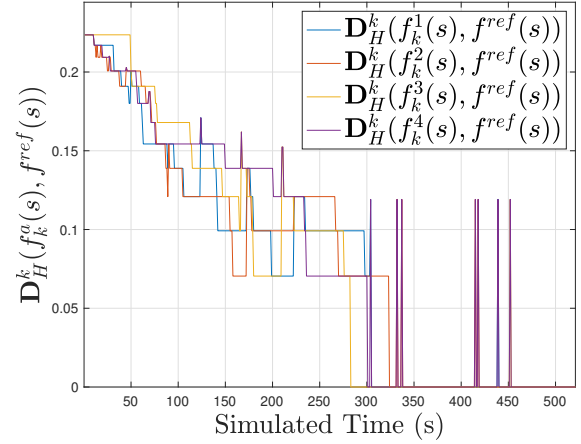


**Fig. 5:** Feature PMF  $f_k^a(s)$  of 4 robots at time  $k = 240$  s in the numerical simulation.

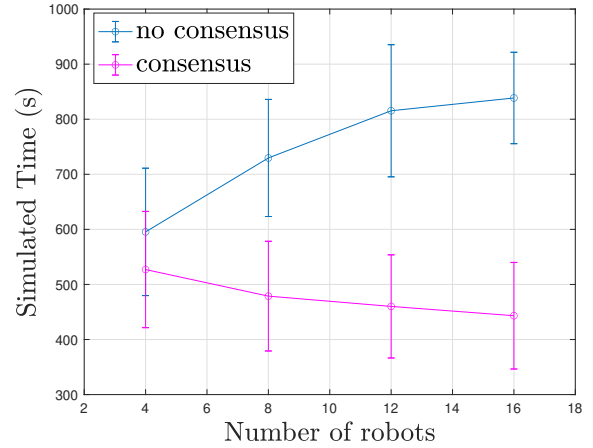


**Fig. 6:** Feature PMF  $f_k^a(s)$  of 4 robots at time  $k = 500$  s in the numerical simulation.

ulate the same scenario as in the numerical simulations, with the set of robots  $\mathcal{A}$  consisting of two quadrotors, *Robot 1* and *Robot 2*. Figure 1 illustrates the simulation setup. Video clips of the SITL simulations are included in the overview video <https://youtu.be/-Z4-DZrHwSM>. The quadrotors are simulated using the Rotors [8] package in Gazebo, and the PX4 flight control stack is implemented in SITL to execute all low-level control tasks for each quadrotor. The discrete feature distribution that the quadrotors must reconstruct is represented by the ArUco markers on the ground plane, located at positions along the red dotted circle. The quadrotors fly at different altitudes (1 m and 2 m) to avoid collisions. This eliminates the need for obstacle detection and avoidance strategies, which are beyond the scope of this work. To detect the ArUco markers, each quadrotor is equipped with a simulated VGA resolution RGB camera that takes images at 30 fps and is oriented to face the ground plane. The image window size is heuristically adjusted to account for the difference in perspective resulting from the robots' difference in altitude. The quadrotors are assigned static IP addresses and exchange



**Fig. 7:** Time evolution of  $D_H(f_k^a(s), f^{ref}(s))$  for each robot  $a \in \{1, 2, 3, 4\}$  in the numerical simulation.



**Fig. 8:** Time until convergence of robots' feature PMFs to the reference PMF in numerical simulations of the feature reconstruction strategy with and without consensus, for  $N = \{4, 8, 12, 16\}$  robots.

their feature distributions over wireless communication when they meet at the  $x, y$  position of the same node (at different altitudes). We chose  $T = 550$  s as the final time of the simulations.

#### A. System architecture

A diagram of our system architecture is shown in Figure 9. We use a hierarchical control scheme composed of low-level and high-level control blocks. The following is a description of each block in Figure 9. The low-level controllers (*Low Level Unit*) use the ROS package MAVROS to generate the control commands for the simulated quadrotors from the high-level controller (*High Level Unit*). Gazebo provides global localization for the simulated quadrotors. It outputs the 3D position of each robot  $a$ ,  $[p_x^a, p_y^a, p_z^a]$ , and its orientation. The 3D quadrotor positions from Gazebo are used to determine the quadrotors' current nodes on the grid and the nodes they should move to in the next time step, according to the DTDS Markov chain in Equation (1). The new node locations are



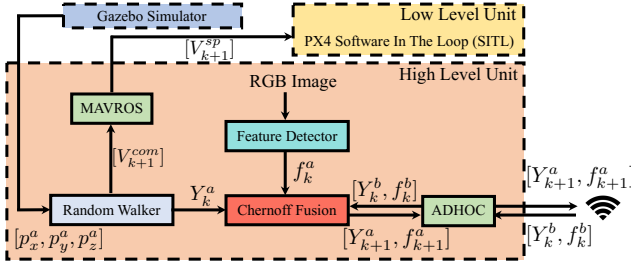


Fig. 9: SITL simulation system architecture block diagram.

mapped to the commanded quadrotor velocities at time  $k+1$ ,  $V_{com}^{k+1}$ , which are converted by *MAVROS* into velocity set points  $V_{sp}^{k+1}$  and sent to the *Low Level Unit*. This block performs the corrections to the quadrotors' poses, which are rendered in *Gazebo*. An RGB image from the quadrotor's bottom-facing camera is used to detect an ArUco marker at the quadrotor's current node, and the feature PMF  $f_k^a(s)$  is computed according to Equation (3). When two quadrotors are located at the same node, they exchange their feature PMFs using ZeroMQ [28]. *Chernoff Fusion* block executes the fusion protocol in Algorithm 1. It computes the quadrotor's feature PMF from its own measurements and from the feature PMFs transmitted by other quadrotors at its current node through the *ADHOC* block.

### B. Simulation results

Figure 10-Figure 13 plot the feature PMFs computed by both quadrotors during two SITL simulation runs. Figure 10 and Figure 11 show the feature PMFs at  $k = 240$  s (4 min) and  $k = 330$  s (5.5 min), respectively, of the first simulation run. The figures indicate that *Robot 1* reconstructs most of the feature distribution within 4 min, and both robots fully reconstruct the distribution within 5.5 min.

Figure 12 and Figure 13 plot the feature PMFs at  $k = 240$  s (4 min) and  $k = 530$  s ( $\sim 8.8$  min), respectively, of the second simulation run. Figure 12 shows that in this simulation run, the two robots do not meet and exchange feature PMFs within the first 4 min, since their feature PMFs are completely distinct at that time. By  $\sim 8.8$  min, Figure 13 shows that both robots have fully reconstructed the feature distribution, which matches the reconstructed distribution in Figure 11. Thus, Figure 10 through Figure 13 demonstrate that our approach ultimately results in accurate feature reconstruction, but that the convergence time to full reconstruction can differ between runs due to the randomness in the robot paths over the grid.

### C. Considerations for real-world implementation

The performance of our approach in real-world environments will be affected by aerodynamic interactions between quadrotors, uncertainty in positioning, and feature occlusion by a quadrotor that enters another's field of view. In the SITL simulations, it is difficult to accurately simulate the aerodynamic disturbance on a quadrotor caused by the downwash of a quadrotor above it. These disturbances can be rejected by incorporating a robust disturbance observer [29], [30] into

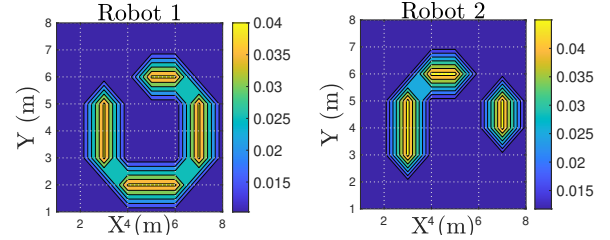


Fig. 10: Feature PMF  $f_k^a(s)$  of 2 robots at time  $k = 240$  s in the first run of the SITL simulation.

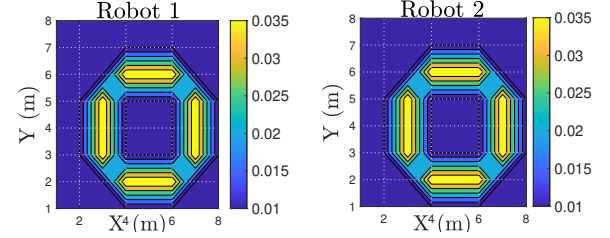


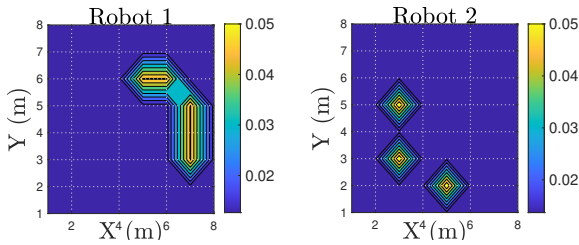
Fig. 11: Feature PMF  $f_k^a(s)$  of 2 robots at time  $k = 330$  s in the first run of the SITL simulation.

the quadrotor's low-level flight control strategy. Alternatively, the quadrotors could fly at the same altitude and employ controllers for inter-robot collision avoidance, e.g. using control barrier functions [31]. Quadrotors may miss feature detections due to misalignment of their camera image window with the ground or to occlusion of the feature by another quadrotor flying below it. Such misalignments and occlusions sometimes occurred in our SITL simulations; however, the accuracy of the feature reconstruction despite these occurrences demonstrates the robustness of our approach to the resulting missed feature detections.

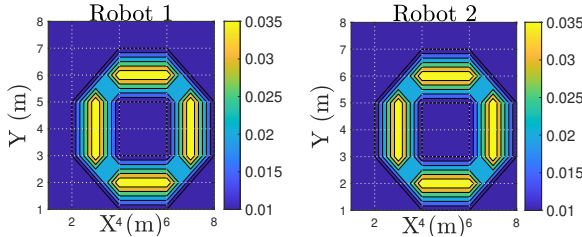
## VII. CONCLUSION AND FUTURE WORK

In this work, we propose a decentralized multi-robot strategy for reconstructing a discrete feature distribution on a finite spatial grid. Robots update their estimate of the feature distribution using their own measurements during random-walk exploration and estimates from nearby robots, combined using a distributed Chernoff fusion protocol. Our strategy extends established results on consensus of opinion pools for fixed, strongly connected networks to networks with Markovian switching dynamics. We provide theoretical guarantees on convergence to the ground truth feature distribution in an almost sure sense, and we validate the strategy in both numerical simulations and SITL simulations with quadrotors.

We note that our strategy is agnostic to the source of the information used to reconstruct the feature distribution; it values the information gained from exploration and other robots equally. This can result in suboptimal convergence rates to consensus on the ground truth distribution, potentially exceeding the operational flight times of small aerial robots with limited battery life. To increase the convergence rate to consensus, we propose to modify the robots' exploration strategy from unbiased random walks to random walks that are biased in directions that increase information gain from individual robots' on-board sensor measurements and also



**Fig. 12:** Feature PMF  $f_k^a(s)$  of 2 robots at time  $k = 240$  s in the second run of the SITL simulation.



**Fig. 13:** Feature PMF  $f_k^a(s)$  of 2 robots at time  $k = 530$  s in the second run of the SITL simulation.

ensure frequent encounters between robots. Another possible way to decrease the time to consensus is by relaxing the constraint  $r_{comm} < 0.5\delta$ , allowing robots to communicate with robots at other nodes. This would require an analysis of whether consensus is still ensured by reformulating the composite Markov chain and determining whether it is irreducible and positive recurrent. In addition, while Pólya's recurrence theorem [32] guarantees that our results on irreducibility and positive recurrence do not extend to random walks on infinite 3D lattices, it would be interesting to investigate whether they are valid for finite 3D grids. Finally, it would be useful to derive an analytical formulation of the expected time until consensus, if possible, which would provide a more rigorous basis for selecting the final time  $T$ .

## REFERENCES

- [1] Wolfram Burgard *et al.* Coordinated multi-robot exploration. *IEEE Transactions on Robotics*, 21(3):376–386, 2005.
- [2] Keiji Nagatani *et al.* Emergency response to the nuclear accident at the Fukushima Daiichi Nuclear Power Plants using mobile rescue robots. *Journal of Field Robotics*, 30(1):44–63, 2013.
- [3] Ricardo Mendonça *et al.* A cooperative multi-robot team for the surveillance of shipwreck survivors at sea. In *OCEANS 2016 MTS/IEEE Monterey*, pages 1–6, 2016.
- [4] Karthik Elamvazhuthi, Zahi Kakish, Aniket Shirsat, and Spring Berman. Controllability and stabilization for herding a robotic swarm using a leader: A mean-field approach. *IEEE Transactions on Robotics*, 37(2):418–432, 2020.
- [5] Shatadal Mishra *et al.* Autonomous vision-guided object collection from water surfaces with a customized multirotor. *IEEE/ASME Transactions on Mechatronics*, 26(4):1914–1922, 2021.
- [6] Fouad Sukkar *et al.* Multi-robot region-of-interest reconstruction with Dec-MCTS. In *2019 International conference on robotics and automation (ICRA)*, pages 9101–9107. IEEE, 2019.
- [7] Nesrine Mahdoui *et al.* Cooperative frontier-based exploration strategy for multi-robot system. In *2018 13th Annual Conference on System of Systems Engineering (SoSE)*, pages 203–210. IEEE, 2018.
- [8] Fadri Furrer *et al.* *RotorS—A Modular Gazebo MAV Simulator Framework*, pages 595–625. Springer International Publishing, Cham, 2016.
- [9] Andrew Howard *et al.* Experiments with a large heterogeneous mobile robot team: Exploration, mapping, deployment and detection. *The International Journal of Robotics Research*, 25(5-6):431–447, 2006.
- [10] Ammar Husain *et al.* Mapping planetary caves with an autonomous, heterogeneous robot team. In *IEEE Aerospace Conference*, pages 1–13, 2013.
- [11] Xianwei Li *et al.* Fully distributed consensus control for linear multiagent systems: A reduced-order adaptive feedback approach. *IEEE Transactions on Control of Network Systems*, 7(2):967–976, 2020.
- [12] Baishen Wei *et al.* Consensus labeled multi-Bernoulli filtering for distributed space debris tracking. In *International Conference on Control, Automation and Information Sciences (ICCAIS)*, pages 203–208, 2017.
- [13] Weiguo Xia *et al.* Structural balance and opinion separation in trust–mistrust social networks. *IEEE Transactions on Control of Network Systems*, 3(1):46–56, 2016.
- [14] Wei Ren and R.W. Beard. Consensus of information under dynamically changing interaction topologies. In *American Control Conference (ACC)*, volume 6, pages 4939–4944, 2004.
- [15] Mehran Mesbahi and Magnus Egerstedt. *Graph theoretic methods in multiagent systems*. Princeton University, Princeton, NJ, 2010.
- [16] Reza Olfati-Saber and Richard M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, 2004.
- [17] Miquel Kegeles *et al.* Random walk exploration for swarm mapping. In *Towards Autonomous Robotic Systems*, pages 211–222, Cham, 2019. Springer International Publishing.
- [18] Aniket Shirsat *et al.* Multi-robot target search using probabilistic consensus on discrete Markov chains. In *IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pages 108–115, 2020.
- [19] Aniket Shirsat and Spring Berman. Decentralized multi-target tracking with multiple quadrotors using a PHD filter. In *AIAA Scitech 2021 Forum*, 2021.
- [20] Tim Bailey *et al.* On conservative fusion of information with unknown non-Gaussian dependence. In *15th International Conference on Information Fusion*, pages 1876–1883, 2012.
- [21] Giorgio Battistelli and Luigi Chisci. Kullback–Leibler average, consensus on probability densities, and distributed state estimation with guaranteed stability. *Automatica*, 50(3):707–718, 2014.
- [22] Morris H. Degroot. Reaching a consensus. *Journal of the American Statistical Association*, 69(345):118–121, 1974.
- [23] Giuseppe C. Calafiore and Fabrizio Abrate. Distributed linear estimation over sensor networks. *International Journal of Control*, 82(5):868–882, 2009.
- [24] William J. Farrell and Chidambar Ganesh. Generalized Chernoff fusion approximation for practical distributed data fusion. In *International Conference on Information Fusion*, pages 555–562, 2009.
- [25] Geoffrey Grimmett and David Stirzaker. *Probability and random processes*. Oxford University Press, 2001.
- [26] Roger A. Horn and Charles R. Johnson. *Matrix analysis*. Cambridge University Press, 1990.
- [27] David A. Levin and Yuval Peres. *Markov chains and mixing times*, volume 107. American Mathematical Society, 2017.
- [28] Pieter Hintjens. *ZeroMQ: messaging for many applications*. O'Reilly Media, Inc., 2013.
- [29] Shatadal Mishra *et al.* Robust attitude control for quadrotors based on parameter optimization of a nonlinear disturbance observer. *Journal of Dynamic Systems, Measurement, and Control*, 141(8), 03 2019. 081003.
- [30] Shatadal Mishra and Wenlong Zhang. A disturbance observer approach with online Q-filter tuning for position control of quadcopters. In *American Control Conference (ACC)*, pages 3593–3598, 2017.
- [31] Li Wang *et al.* Safe certificate-based maneuvers for teams of quadrotors using differential flatness. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3293–3298, 2017.
- [32] Georg Pólya. Über eine aufgabe der wahrscheinlichkeitsrechnung betreffend die irrfahrt im straßennetz. *Mathematische Annalen*, 84(1):149–160, 1921.