Université Libre de Bruxelles

Institut de Recherches Interdisciplinaires
et de Développements en Intelligence Artificielle

Reducing Uncertainty in Collective Perception using Self-organized Hierarchy

A. Jamshidpey, M. Dorigo, and M.K. Heinrich

IRIDIA – Technical Report Series

Technical Report No.
TR/IRIDIA/2022-011

October 2022
Last revision: May 2023

IRIDIA – Technical Report Series
ISSN 1781-3794

Published by:

IRIDIA, *Institut de Recherches Interdisciplinaires et de Développements en Intelligence Artificielle*
Université Libre de Bruxelles
Av F. D. Roosevelt 50, CP 194/6
1050 Bruxelles, Belgium

Technical report number TR/IRIDIA/2022-011

Revision history:

TR/IRIDIA/2022-011.001	October 2022
TR/IRIDIA/2022-011.002	October 2022
TR/IRIDIA/2022-011.003	October 2022
TR/IRIDIA/2022-011.004	May 2023

Reducing Uncertainty in Collective Perception using Self-organized Hierarchy

Aryo Jamshidpey[1,2], Marco Dorigo[1], and Mary Katherine Heinrich[1*]

[1]IRIDIA, Université Libre de Bruxelles, Brussels, Belgium
[2]University of Ottawa, Ottawa, Canada
[*]Corresponding author. Email: mary.katherine.heinrich@ulb.be

Abstract

In collective perception, agents sample spatial data and use the samples to agree on some estimate. In this paper, we identify the sources of statistical uncertainty that occur in collective perception and note that improving the accuracy of fully decentralized approaches, beyond a certain threshold, might be intractable. We propose self-organized hierarchy as an approach to improve accuracy in collective perception, by reducing or eliminating some of the sources of uncertainty. Using self-organized hierarchy, aspects of centralization and decentralization can be combined: robots can understand their relative positions system-wide and fuse their information at one point, without requiring, e.g., a fully connected or static communication network. In this way, multi-sensor fusion techniques that have been designed for fully centralized systems can be applied to a self-organized system for the first time, without losing the key practical benefits of decentralization. We implement simple proof-of-concept fusion in a self-organized hierarchy approach and test it against three fully decentralized benchmark approaches. We test the perceptual accuracy of the approaches for absolute conditions that are uniform time-invariant, time-varying, and spatially nonuniform with high heterogeneity, and test the scalability and fault tolerance of their accuracy. We show that, in our tested conditions, the self-organized hierarchy approach is generally more accurate, more consistent, and faster than the other approaches, and also that its accuracy is more scalable and comparably fault-tolerant. In the spatially nonuniform conditions, our results indicate that the four approaches are comparable in terms of similarity to the reference samples. In future work, extending these results to additional methods such as collective probability distribution fitting is likely to be much more straightforward in the self-organized hierarchy approach than the decentralized approaches.

Keywords— Self-organization, Swarm intelligence, Collective perception, Collective decision-making

1 Introduction

Increased autonomy in robot swarms is an open challenge. For example, in collective decision making, robot swarms cannot yet autonomously identify when a collective decision needs to be made and trigger the process [1]. Full collective autonomy would require task-general approaches to several constituent capabilities, including accurate collective perception and manageable collective actuation. However, existing approaches are often task-specific and features such as accuracy and manageability are challenging in swarm robotics—innovation is required if swarm autonomy is to increase significantly.

One approach to these challenges would be introducing some hierarchy into an otherwise fully decentralized system. Self-organized hierarchy has been identified as a crucial research direction for the future of robot swarms [2, 3]. However, if hierarchy is implemented in robot swarms, we need to ensure that features such as scalability and fault tolerance are not lost. Indeed, the motivation for studying self-organized hierarchy is to combine aspects of centralized and decentralized control, ideally to get the benefits of both in one system. In this paper, we propose a self-organized hierarchy approach to collective perception, based on the existing concept of mergeable nervous systems (MNS) [4, 5]. We empirically compare it to three fully decentralized approaches as benchmarks, assessing whether accuracy is improved and scalability and fault tolerance are preserved.

In robot swarms, collective perception—i.e., the perception of an environment by a group of agents collaborating in a self-organized way—can be viewed as a type of collective decision-making [6]. The swarm must both collect information and converge on a shared understanding of that information. Swarm robotics approaches to collective perception [e.g., 7, 8] are generally scalable and fault-tolerant because they do not rely on fully connected or static communication networks and do not have single points of failure such as base stations or fixed leaders. These approaches also have strong potential for autonomy, because they do not require access to external infrastructure or extensive prior knowledge. However, because fully decentralized approaches reach a collective decision via consensus, accuracy is challenging (compared to fully centralized approaches) and convergence times can be quite long [9].

Centralized approaches to perception generally make use of information fusion. Multi-sensor and multi-robot fusion problems are well understood, and existing methods are powerful [10–13]. However, these approaches typically know the positions and often also the poses of all robots or sensors in the system, using global positioning infrastructure, predefined or static positions, or other solutions that restrict scalability or fault tolerance (compared to swarm robotics approaches). For most perception problems, fully decentralized information fusion has not yet been developed.

We propose self-organized hierarchy as a way for robots to understand all their relative positions system-wide and fuse their collective information at one point, without relying on restrictive features such as fixed positions, a fully connected or fixed communication topology, external infrastructure, or prior knowledge.

1.1 Related work

In this subsection, we give an overview of the existing fully decentralized approaches to collective perception with robots, in which perception is usually formulated as a best-of-n decision-making problem. We also give an overview of the existing multi-robot perception approaches that are *not* fully decentralized, which usually focus on the information fusion problem. We then discuss the topic of information fusion in self-organized systems, for which there are almost no existing approaches. Finally, we briefly discuss the perception of absolute conditions versus relative conditions.

Collective and multi-robot perception approaches are summarized in Table 1. They are organized according to whether perceptual information is fused at the data-level (e.g., images), feature-level, or decision-level, based on the multi-sensor fusion levels defined in [13]. They are also organized according to the type of perceptual decision being made—either discrete best-of-n decisions or estimates of continuous values, according to [6], or target tracking (e.g., of moving objects or stationary landmarks, including for mapping or SLAM).

Fully decentralized perception All fully decentralized approaches to collective perception use a dynamic communication network that it not fully connected (i.e., there is no global broadcast and local connections are not static). The majority of these approaches are set up as best-of-n decision-making problems [see 6], where a robot swarm compares multiple options according to a given criterion. In one common setup, robots sense colors in an environment and compare them according to the criterion of highest representation [8, 14–18]. In another setup, robots compare discrete zones according to a criterion of *quality* that can be sensed from anywhere in the zone [7, 19, 20]. In similar setups, robots aggregate at a certain type of environmental feature, either by collectively deciding on an appropriate threshold to detect it [21] or sharing detected maximums so robots at a local optimum are triggered to explore further [22]. In [23], robots estimate the density of tiles scattered in the environment by sharing perceptual information, including sample counts and values for time-based decrementation. In [24], robots collaboratively track a moving target by sharing locally-based beliefs about the target position with their neighbors, using incomplete knowledge of robot positions and orientations.

Decision type	Data-level	Feature-level	Decision-level
Tracking	[25, *] [26, *]	**[27, *]** [28, *]	[29, *] [30, *] [31, *] **[24]**
Continuous	**[21]**	**[23]** **[22]**	
Best-of-n	[32, *] [33, *]		**[8]** **[14]** **[17]** **[16]** **[18]** **[15]** **[7]** **[20]** **[19]** [34, *]
	Information fusion type		

Table 1: Existing approaches to collective perception and multi-robot perception, organized according to the level at which perceptual information is fused [see 13] and the type of decision being made [best-of-n, continuous, see 6, and tracking]. In **bold blue** references, the topology of the communication network is dynamic and not fully connected. In starred (*) references, all relative robot positions in the system are either fixed or globally known and are used explicitly or implicitly during information fusion.

Not fully decentralized perception All the approaches that are not fully decentralized use fixed or known robot positions during information fusion. In most of these approaches, robots collaboratively track targets using known robot positions, e.g., by data-level fusion of sensor readings [26], feature-level fusion of perceptual information and associated uncertainties [28], or decision-level fusion of estimated positions [27], which can be supported by manual annotations merged at a base station [30]. In some of the tracking approaches, targets are tracked as part of mapping or SLAM, e.g., by decision-level fusion of estimated positions at a base station [29] and sometimes supported by prior information about the environment [31]. In SLAM, the positions of robots are not known beforehand, but the information needed to estimate all robot positions is available during the process of fusing information about the tracked targets (e.g., stationary landmarks). Besides target tracking, the literature also includes a few non-fully decentralized approaches for best-of-n decision making. In [32], robots establish a fixed communication network to train a system-wide ANN that classifies a global light pattern in the environment. In [33], static robots with known relative positions send IR readings to a base station to identify objects from a predefined set. In [34], robots identify hand gestures from a predefined set and merge their opinions according to known robot positions. In this approach, the communication network is fully connected during information fusion (although the impact of packet loss is studied).

Fusion for robot swarms Access to explicit or implicit position information is required for the majority of existing approaches to multi-sensor fusion [see 10–13], which are quite developed and would be useful for many applications if they could be implemented in robot systems. However, if these approaches are applied to robot systems using global broadcast or networks with fixed topologies, scalability and fault tolerance will be lacking. Ideally, existing approaches that require position information would be implemented in robot systems in a self-organized way, but this combination is quite challenging. Note that in Table 1, there is one approach [27] that both fuses information based on known robot positions (i.e., starred in Table 1) and can operate under dynamic topologies (i.e., blue and bold in Table 1). However, this approach assumes full knowledge of absolute robot positions and orientations, the availability of which cannot always be guaranteed. In short, position-guided information fusion methods cannot currently be used for self-organized collective perception. In this paper, we propose self-organized hierarchy based on the MNS concept as a general framework in which existing multi-sensor fusion techniques could be implemented for collective perception, without using restrictive mechanisms (e.g., fixed central coordinating entity or fixed communication topology) that impede desirable swarm robotics traits.

Perceiving absolutes The fully decentralized approaches in the literature are applied either to the perception of a relative condition (e.g., whether red or blue is more represented) or the completion of a targeted action using sensed information (e.g., aggregation [22]). Results of these studies are not necessarily directly transferable to perception of an absolute condition. For instance, when perceiving a more represented color, if a swarm underestimates or overestimates absolute density, it might do so for each color somewhat consistently and therefore might still be able to accurately determine which color is more represented. The one exception in the literature is [23], in which perception of absolute density is assessed as an auxiliary contribution (the primary focus is completion of a targeted action). In the reported results, we see that the swarm's perception of density diverges widely from the ground truth. In short, collective perception of absolute conditions still requires study. In this paper, we test some of the most accurate approaches for relative conditions in a new experiment setup, to benchmark the approximate error (incorporating both bias and variance) present in fully decentralized approaches when perceiving an absolute condition.

1.2 Paper structure

The remainder of this paper is organized as follows. In Sec. 2, we first discuss statistical uncertainty in collective perception and identify two sources of uncertainty that are not present in all spatial sampling and inference problems, but are crucial when sampling with mobile robots. We then introduce and describe the proposed self-organized hierarchy approach, the three fully decentralized approaches selected as benchmarks for comparison, and the design and setup of the experiments. We report the results of the comparative experiments in Sec. 3, discuss the results and future work in Sec. 4, and finally summarize the conclusions in Sec. 5.

2 Materials and methods

2.1 Uncertainty in collective perception

Collective perception by ground robots is generally a 2D *spatial sampling and inference problem* [cf. 35]. In other words, robots need to collect samples of information that varies spatially in two dimensions and use the samples to make some estimate, such as a mean or total value for an area (e.g., mean temperature, mean noise level, total daylight coverage, or total soil toxicity) or the positions of some objects within a relative coordinate system.

2.1.1 Spatial sampling and inference

In estimation of spatial data, there are three points at which uncertainty and error can originate, following [35]:

1. features of the **stochastic field** ($\mathcal{R}$), i.e., the distribution and variability of the spatial data,
2. the spatial **sampling** method ($\mathcal{T}$), and
3. the statistical **inference** method (ψ).

The main task is to minimize bias that originates in the sampling method $\mathcal{T}$ and bias that originates in inference ψ. Note that biased $\mathcal{T}$ can also be compensated for during ψ.

2.1.2 Collective perception with mobile robots

In principle, collective perception approaches should use parallel sample collection with multiple robots to overcome the bias that would be present in a single robot perceiving by itself.

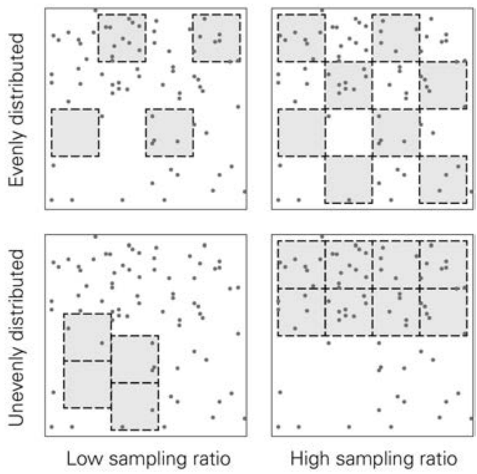


Figure 1: When prior knowledge about a stochastic field is not known, higher sampling ratios and more evenly distributed samples can reduce the potential for bias to be introduced during the sampling method $\mathcal{T}$.

In collective perception, robots do not have any prior knowledge of the stochastic field. A sampling method $\mathcal{T}$ that introduces the least possible uncertainty would have a sampling ratio of 1 with perfectly evenly distributed sampling sites [see 35]. Therefore, parallel sampling in collective perception can minimize bias by increasing the total number of samples that are available to one robot, as well as increasing the spatial dispersion and potentially the density uniformity of the sampling sites (see Fig. 1).

However, because samples are taken by mobile robots that sweep an area over time, not all aspects of $\mathcal{T}$ can be directly defined. For instance, if simple random sampling is desired, this cannot be programmed directly, but instead needs to be targeted indirectly by designing, e.g., a random walk and sampling time protocol. Furthermore, if the stochastic field $\mathcal{R}$ is time-varying, spatiotemporal correlations and variability aspects that are not representative of $\mathcal{R}$ could be introduced during sampling due to the robot's sweeping motion. Also, in collective perception, robots need to incorporate samples collected by their peers in parallel. In most approaches, robots sample the decisions of their peers in addition to sampling the stochastic field directly, and fuse decisions during the inference process ψ.

Therefore, in collective perception with mobile robots, there are five points at which uncertainty and error can originate:

1.–3. $\mathcal{R}$, $\mathcal{T}$, ψ [following 35],

4. the spatial **decision sampling** method ($\mathcal{G}$), and

5. the spatiotemporal **sweeping** method ($\mathcal{P}$).

In collective perception, a random walk is typically used to sweep the environment, because robots are assumed to have no or very little prior knowledge of the layout of the environment or the stochastic field $\mathcal{R}$. Random walks can be tuned or modulated to improve the sampling distribution (refer to Fig. 1), but some interference from obstacles, environment layout, or other robots is unavoidable. An optimal random walk sweep $\mathcal{P}$ would reliably result in uniformly random distribution of sampling sites in $\mathcal{T}$ and $\mathcal{G}$—i.e., $\mathcal{T}$ and $\mathcal{G}$ would be simple random sampling. However, simple random sampling leads to large gaps and therefore bias [35], so even if an optimal random walk free of interference were practical, substantial uncertainty could still originate during $\mathcal{T}$ and $\mathcal{G}$. Consider that, although biased $\mathcal{P}$, $\mathcal{T}$, and $\mathcal{G}$ can be compensated for during inference ψ, it is difficult or impossible to do so without prior knowledge of the stochastic field $\mathcal{R}$ [35]. Therefore, we conclude that it might be intractable to substantially improve the performance of fully decentralized approaches to collective perception beyond the current state of the art [e.g., 7, 8].

In this paper, we assert that, by reducing or eliminating key sources of uncertainty that are usually present in collective perception, a well-designed self-organized hierarchy approach could provide much more accurate estimates without sacrificing scalability and fault tolerance, as compared to fully decentralized approaches. We test this assertion empirically.

2.2 Problem statement

We propose a self-organized hierarchy approach to collective perception, based on the MNS concept, and test it against fully decentralized collective perception in simulated experiments. In the **self-organized hierarchy approach** (Hier), a robot fulfilling the temporary role of "brain" forms one collective opinion on behalf of the group, using collective sensor information merged hierarchically by all robots. In the fully decentralized approaches, each robot partakes (either explicitly or implicitly) in a collective decision-making process to form its own opinion and reach decentralized consensus with its peers.

We test the Hier approach against the following three fully decentralized approaches, as benchmarks for comparison.

- **Voter decision model** (Vote): each robot selects new opinions randomly, from among the current opinions of itself and its neighbors [based on 8].

- **Mean decision model** (MEAN): each robot averages the opinions of itself and its neighbors [based on 8].[1]
- **Stigmergy** (STIG): robots do not communicate explicitly, but leave cues for each other to observe in the environment [based on 36].

In all approaches, simulated ground robots use short-range onboard sensing to detect some objects that are distributed randomly in an arena of unknown size. Their collective goal is to form an accurate opinion on the mean density of objects in the whole arena.

In this paper, robots collect samples by detecting individual objects, and then infer the mean density of the arena using odometry and knowledge of their own sensor ranges.

Absolute object density $\lambda = b/a$ is defined as the number of objects b per unit area a. The true mean density λ in the environment is noted as λ^{true}. Robots are not given any information about the size and shape of the objects nor the size and shape of the arena. Robots only know the dimensions of their own fields of view and must use this knowledge to infer the number of objects per unit area.

2.2.1 Experimental Design

To study the baseline performance of the tested methods, we construct stochastic fields $\mathcal{R}$ with relatively low spatial variability by distributing the features (the objects to be detected) uniformly randomly in a regular polygon area. Still, inference ψ is not negligible even in the case of optimal sampling, because the robots cannot detect the target value (i.e., density) directly, and must instead infer it from representative information (i.e., short-range boolean detection of objects).

We test the approaches under several time-invariant and time-varying $\mathcal{R}$. Each approach is assessed in terms of perceptual accuracy (i.e., the error in the collective opinion, with respect to the true value), the consistency of the accuracy, and the reaction time. The scalability and fault tolerance of each approach is also assessed in time-invariant $\mathcal{R}$, in terms of the change in accuracy under robot failures and group size variations.

2.3 Methods for collective perception

The robots run two parallel processes.

- Process A: robot r **individually** counts detected objects and infers the density in its own field of view.
- Process B: robots influence the **collective** opinion of the group via either the inputs or outputs of Process A, and robot r forms its opinion on the absolute object density (λ) in the arena.

Robots use only local or indirect communication to influence the collective opinion of the group, so the motion routines and communication rules of each approach are presented in the descriptions of Process B.

Process A is the same in each approach (HIER, VOTE, MEAN, STIG), except for the tuning parameters. Process B is different in each approach.

[1] We use average opinion (i.e., taking the mean) instead of majority opinion as in [8] (i.e., taking the mode), because we consider a continuous decision rather than a discrete decision task and the sample size the robots use at each time step is very small. It would often not be possible for a robot to find a unique mode among its available samples, so the performance of the majority model would be poor.

2.3.1 Process A: Robots make individual interpretations

In all approaches, robot r counts sensed objects and infers the density in its (direct or indirect) field of view, outputting the value ϵ_r. The sensing and inference output ϵ_r, which is equal to 0 in the first time step, is defined as the average number of objects per unit area in the robot's field of view in a given time window, calculated as follows for time t:

$$\epsilon_r = \frac{\sigma_t - \sigma_{t-k_t}}{v \cdot s \cdot k_t^{\mathtt{max}}} \cdot P \ , \tag{1}$$

where σ_t is the cumulative number of objects seen from the first time step to the current time t by robot r , k_t is the time window at time t, $k_t^{\mathtt{max}}$ is the maximum time window, and the remaining terms are tuning parameters: v is the view area of robot r, s is the average speed of robot r, and P is a parameter related to the motion routines of the different approaches. (For details on the tuning parameters, see Sec. 2.4.)

The time window k_t keeps track of the elapsed time since the robot started the mission, up to the maximum time window $k_t^{\mathtt{max}}$, i.e.,

$$k_t = \begin{cases} t, & \text{if } t < k_t^{\mathtt{max}} \\ k_t^{\mathtt{max}}, & \text{otherwise} \end{cases} . \tag{2}$$

At each time step, a new value σ_t is saved to a circular buffer $\boldsymbol{\sigma}$ of length $k_t^{\mathtt{max}}$ in the memory of robot r. In other words, after $\boldsymbol{\sigma}$ reaches $k_t^{\mathtt{max}}$ elements (i.e., when $t \geq k_t^{\mathtt{max}}$), then at every subsequent update of σ_t (at position $i = k_t^{\mathtt{max}}$), each element at $i : i \in \{1, \ldots, k_t^{\mathtt{max}} - 1\}$ is replaced by the element at $i + 1$. Buffer $\boldsymbol{\sigma}$ is defined as:

$$\boldsymbol{\sigma} = (\sigma_{t-k_t}, \sigma_{t-k_t+1}, \ldots, \sigma_t), \qquad \sigma_t = \sum_{i=1}^{t} b_i, \tag{3}$$

where b_t is the number of objects detected (directly or indirectly) at the current time step t by robot r. Note that, in order to calculate σ_t, only σ_{t-1} and the most recent b_t values are required; it is not necessary to maintain a memory of all previous b_t values.

In summary, at each time step t, Process A: (1) inputs b_t, the number of objects the robot detects, and (2) outputs ϵ_r, the inferred density in the robot's field of view during the time window.

2.3.2 Process B: Robots influence the collective opinion

In all approaches, all robots influence collective opinion through either the inputs or outputs of Process A. Also, each robot r uses the outputs of Process A to form its opinion λ_r^{app} of the apparent absolute density in the arena.

In the decision model approaches (Vote, Mean), robots modulate the input b_t individually and process the output ϵ_r collectively. In the other two approaches (Stig, Hier), robots do the opposite—they modulate the input b_t collectively and process the output ϵ_r individually. So, in all four approaches, the opinion λ_r^{app} is influenced by a collective process.

2.3.3 Decision model approaches (Vote, Mean): Process B

We test two fully decentralized approaches that use a collective decision-making process to reach consensus about the apparent density in the arena. Both approaches use a basic stochastic motion routine and explicit communication among neighbors. The robots coordinate their opinions using either a voter decision model or a mean decision model.

Stochastic motion routine The VOTE and MEAN approaches use a stochastic motion routine based on RANDOM BILLIARDS [37, 38]. Each robot moves forward at a constant velocity unless it detects the boundary line of the arena. A robot can detect a line's angle relative to its own heading by driving over the line, and can likewise detect the "inside" or "outside" of the arena by driving partially over the respective area. When a robot detects an arena boundary line, it turns away from the line to face a random direction towards the "inside" of the arena. For a boundary line with detected angle θ_1 and the "inside" of the arena towards the direction $\theta_1 + \frac{\pi}{2}$, the robot turns to face a random direction with uniform distribution $U(\theta_1, \theta_1 + \pi)$.

A robot pauses its RANDOM BILLIARDS motion and performs obstacle avoidance when it meets an object or another robot. Robots use line-of-sight sensing and communication to detect (and distinguish between) objects and robots that are within short-range radius ρ_1 and within $\pm\frac{\pi}{3}$ of the heading angle θ_h. When a robot sees either an object or a robot, it turns away from it until it is no longer visible within $\theta_h \pm \frac{\pi}{3}$. If the detection was of an object, then the input of Process A is updated as $b_t \leftarrow b_t + 1$.

Voter decision model (VOTE) In this approach, robots share their individual inference outputs ϵ_r (i.e., the outputs of Process A) using explicit communication. Each robot then uses a voter model process to form its opinion λ_r^{app} of absolute object density.

For explicit communication, each robot r_n maintains and shares a matrix $\boldsymbol{\epsilon}$ as follows:

$$\boldsymbol{\epsilon} = \begin{pmatrix} r_1 & \epsilon_{r_1} & t_{r_1} \\ r_2 & \epsilon_{r_2} & t_{r_2} \\ \vdots & \vdots & \vdots \\ r_m & \epsilon_{r_m} & t_{r_m} \end{pmatrix}, \tag{4}$$

where r_j is the robot ID, ϵ_{r_j} is the most up-to-date ϵ_r value known for robot r_j by robot r_n, and t_{r_j} is the time stamp of the known ϵ_{r_j}. At each time step, robot r_n updates its own ϵ_{r_n} and t_{r_n} according to Equation 1 and then sends its matrix $\boldsymbol{\epsilon}$ to its current neighbors. If a robot receives a matrix that contains a higher t_{r_j} than its current entry for that robot, it updates its row for r_j accordingly. Also, if $t - t_{r_j} < k_t^{\max}$ for any t_{r_j} entry, the robot removes the corresponding row. In this way, the $\boldsymbol{\epsilon}$ of robot r_n always contains the most up-to-date ϵ_r values for its peers that r_n has seen within the time window k_t.

At each time step, robot r decides λ_r^{app} by randomly selecting one ϵ_{r_j} entry from its matrix. In other words,

$$\lambda_r^{\text{app}} = \epsilon_{x \sim\ U(r_1, r_m)} \in \boldsymbol{\epsilon}, \tag{5}$$

such that opinion λ^{app} of absolute density is the result of a voter decision model.

Mean decision model (MEAN) In this approach, robots also share their individual inference outputs ϵ_r using matrices $\boldsymbol{\epsilon}$ as defined in Eq. 4.

At each time step, robot r decides λ_r^{app} by averaging the ϵ_{r_j} entries in its matrix. In other words,

$$\lambda_r^{\text{app}} = \overline{\epsilon_{r_j}} \in \boldsymbol{\epsilon}, \tag{6}$$

such that opinion λ^{app} of absolute density is the result of a mean decision model.

2.3.4 Stigmergy approach (STIG): Process B

In this approach, robots influence the collective opinion via the inputs of Process A—the number of objects detected (b_t)—not the outputs of Process A.

This approach uses stigmergic (indirect) communication, through artificial pheromones—i.e., cues left in the environment that are observable by the robots within a certain range[2]. Robots use the pheromones deposited in the environment to reach a consensus about the apparent density in the arena.

When a robot detects an object within short-range radius ρ_1, it counts the object and deposits pheromone at the object location. Robots can detect and differentiate pheromone sources within long-range radius ρ_2, which allows them to count objects already found by their peers in a much larger vicinity than that in which they can detect objects directly. Each robot counts sensed pheromone sources the same as sensed objects when b_t (see Eq. 3 of Process A). Robots also understand that pheromones are cues that the immediate area has already been explored by another robot, and therefore turn away to search for new unexplored areas.

Each robot can sense a pheromone in any direction, and can sense the relative direction and distance of its source. When a robot senses a pheromone source within long-range radius $\rho_2 - \delta_2$ and in a direction close to that of its heading, it turns away from the source to a randomized direction, according to Algorithm 1.

Algorithm 1: Pheromone reaction

Input:
θ_x // Randomized angle, see Eq. 7
$\overrightarrow{s}$ // Vector from robot to pheromone source
$\overrightarrow{h}$ // Vector of robot heading
t_s // Time since last pheromone reaction

$\theta_{sh} \leftarrow$ angle between $\overrightarrow{s}$ and $\overrightarrow{h}$
while $|\theta_{sh}| \leq |\theta_x|$ ***and*** ($t_s \leq 50$ *or* $t_s \geq 250$) **do**
  robot turns heading in the SIGN(θ_x) direction ;
move forward ;

The angle θ_x is selected randomly once every 200 time steps, with the following uniform distributions:

$$\theta_x \sim \begin{cases} U(\frac{\pi}{6}, \frac{\pi}{2}), & \text{if } \left\lfloor \frac{t}{200} \right\rfloor \text{ is even} \\ U(-\frac{\pi}{2}, -\frac{\pi}{6}), & \text{otherwise} \end{cases}. \tag{7}$$

If a robot is simultaneously within range of multiple pheromone sources, it logs them in a list with an arbitrary order, and reacts to the first pheromone source in its list for that time step. After a robot reacts to a given pheromone source, it continues moving forward until it encounters a different pheromone, an object, or a boundary line.

Recall that robots have already influenced the collective opinion via the inputs of Process A. Therefore, at each time step, robot r simply takes its own inference output ϵ_r as its opinion λ_r^{app} of absolute density. In other words,

$$\lambda_r^{\text{app}} = \epsilon_r. \tag{8}$$

2.3.5 Hierarchical approach (Hier): Process B

In this approach, robots influence the collective opinion via the inputs (not the outputs) of Process A, and only the robot occupying the dynamic leadership position, the MNS-brain robot r, performs Process A.

[2] In real robots, a stigmergic approach could be implemented using communication-enabled smart blocks, such as those in [39].

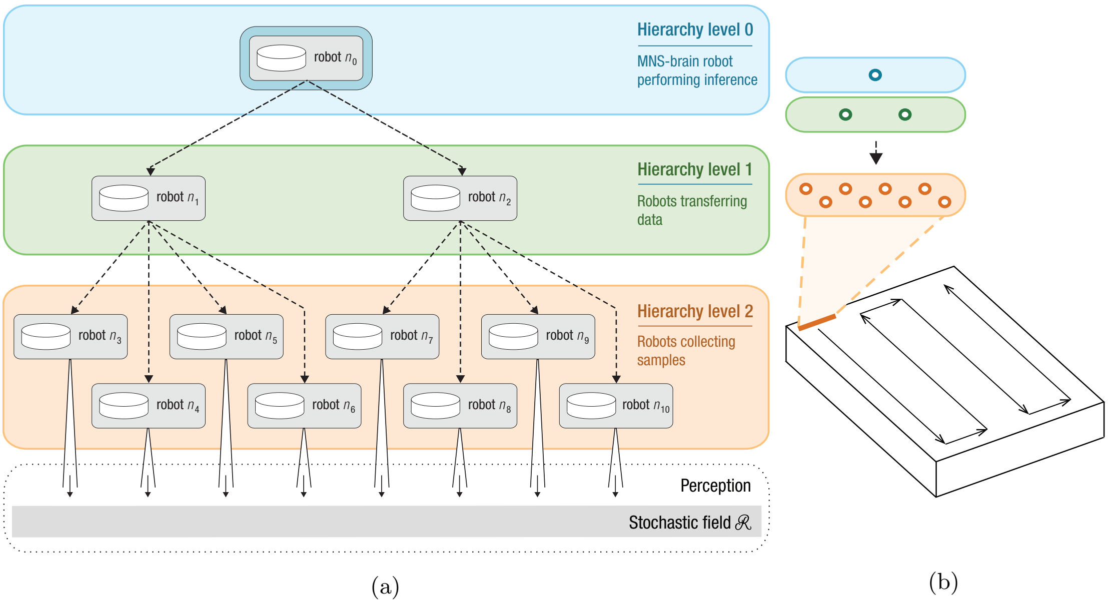


Figure 2: **In the Hier approach, a dynamic ad-hoc hierarchical network is self-organized using local communication and relative positioning.** (a) The roles of robots in the collective perception task according to their hierarchy level. The eight robots in level 2 (orange) collect samples of the stochastic field $\mathcal{R}$, the MNS-brain robot in level 0 (blue) performs inference, and any robots between them (green) transfer data. (b) The reactive boustrophedic motions to sweep the environment. The MNS-brain robot in level 0 (blue) is responsible for determining the reactions and sending motion instructions downstream, the robots in level 2 (orange) are responsible for obstacle avoidance, and any robots between them (green) monitor their children's positions and calculate their motion instructions.

The Hier approach is based on the existing MNS concept, which is a general framework for constructing and reconstructing self-organized hierarchy [4]. Under the MNS framework, robots can self-organize a dynamic ad-hoc control network in which robots temporarily and interchangeably occupy certain positions in a leadership hierarchy, including an MNS-brain (i.e., highest hierarchy level) position [5, 37, 40, 41]. In an MNS control network, each robot communicates only with its direct neighbors, to prevent the type of bottleneck that would occur at the communication hub in a fully centralized system. According to task specifications and system constraints, sensor information can be merged as it is passed upstream, control information can be unmerged as it is passed downstream, and the balance of individual behaviors versus collective behaviors can be actively managed. This flexibility can be used to reduce or eliminate the potential for bottlenecks throughout the hierarchical network.

In this paper, we use the MNS implementation of [5], in which camera-equipped UAVs[3] are responsible for sensing the relative positions and orientations of the ground robots for the purpose of keeping the robot formation together during sweeping. We use the sweeping technique of [40] and apply it to the task of collective perception.

[3]Note that UAVs in the Hier approach in this paper cannot sense objects directly, so do not increase the total sensing range available in the approach. If we were using ground robots that were capable of sensing each others' relative positions and orientations, the UAVs could be removed, and this removal would not have an impact on the collective perception results.

In the motion routine of Process B of the HIER approach, robots self-organize into a roughly linear formation to sweep the environment [for details, see 40]. In the HIER approach, the MNS-brain robot r detects arena boundary lines and reacts to them using a deterministic process. The dimensions of the MNS's collective ground robot sensor range, calculated by the MNS based on the number of ground robots in the formation, are used as parameters in the deterministic process. The MNS reactively sweeps the unknown environment using standard back-and-forth boustrophedic motions [42]. The reactive boustrophedic motions are based on the MNS's knowledge of its own dimensions, in such a way that the collective sensor ranges of the ground robots can cover the environment as completely as possible.

As it sweeps the arena, the MNS-brain robot sends control information downstream, to maintain the formation [for full details, see 43]. Each robot in the MNS receives motion instructions from its parent that include the targeted relative linear velocity $\mathbf{v}$ and angular velocity $\boldsymbol{\omega}$, as well as the current orientation quaternion $\mathbf{q}_t$, following [43]. In order to send motion instructions, each parent in the MNS senses its child's displacement $\mathbf{d}_t$ and relative orientation $\mathbf{q}_t$, determines the new targeted values based on the most recent motion instructions it received from its own parent, and calculates motion instructions for its child as follows [43]:

$$\mathbf{v} = k_1 \left(\frac{\mathbf{d}_{t+1} - \mathbf{d}_t}{||\mathbf{d}_{t+1} - \mathbf{d}_t||} \right), \ \boldsymbol{\omega} = k_2 \cdot ||f(\mathbf{q}_{t+1}^{-1} \times \mathbf{q}_t)||, \tag{9}$$

where k_1 and k_2 are speed constants and function $f(x)$ converts a quaternion to a Euler angle.

In the HIER approach, when a ground robot detects an object within short-range radius ρ_1 and within $\pm\frac{\pi}{3}$ of the heading angle θ_h, it temporarily ignores the motion instructions received from its parent in order to circumvent the object (in a predefined arc trajectory relative to its current position) until the the object is no longer within $\theta_h \pm \frac{\pi}{3}$. Complementarily, if the parent of a ground robot detects that its child is behind another ground robot within $\rho_1 + \delta_1$, then the parent will temporarily ask the child to stop moving, until the two ground robots are no longer within $\rho_1 + \delta_1$ of each other. The child will accept the request as long as it detects an object within short-range radius ρ_1.

In the HIER approach, ground robots essentially act as temporary remote sensors of the MNS-brain robot r. Each child robot sends its sensor readings upstream to the MNS-brain robot r via its parent[4], which calculates one inference output ϵ_r for the whole MNS. Therefore, at each time step, the MNS-brain robot r takes its own inference output ϵ_r as its opinion λ_r^{app} of absolute density, i.e.,

$$\lambda_r^{\text{app}} = \epsilon_r. \tag{10}$$

2.4 Simulation setup

The experiments are conducted in the ARGoS simulator [44], with robot models implemented using an existing plugin [45, 46]. The experiments are conducted with the kinematics of small differential-drive ground robots based on the extended e-puck robot [47–49] and, for the HIER approach, of quad-camera UAVs based on the S-drone quadrotor [50]. For more implementation details of all four approaches and the experiment setup in ARGoS, please see the open-source code repository.[5]

In all approaches and all setups, only ground robots have the capability to directly sense objects. In all setups except scalability, each approach has eight ground robots. All robots in all approaches have the

[4]Implementation note: To reduce the per-step simulation time needed by CPU and GPU solvers, some of the upstream and downstream data transfers within the MNS are simulated as a single-step rather than multi-step process. In this paper, the approaches are assessed according to the simulation steps, not the per-step time. The implementation strategy used negligibly impacts the number of simulation steps needed to collect and pass information (adds a maximum of 1 step), so it does not undermine the analysis of the reported results.

[5]`https://github.com/BlueDiamond07/Collective_perception`

same average linear velocity (7.5 cm/s). The experiments begin when robots start to sweep the arena and end after 50 000 time steps (2 000 s).

2.4.1 Tuning parameters

Regarding the parameters in Eq. 1, in the Mean and Vote approaches, the view area v of robot r is based on the sensor range of onboard object sensing and in the Stig approach v is based on the the sensor range of onboard pheromone sensing. In the Hier approach, v is based on the maximum bounds of the combined sensor ranges of all ground robots. Note that in all approaches, all ground robots have the same sensor range for detecting objects (short-range radius ρ_1). The time window used in the experiments is $k_t^{\mathtt{max}} = 1000$ time steps. In other words, in all approaches a robot's memory of a sensing input lasts for 1 000 time steps (40 s). Parameter P is tuned separately for each approach during a manual testing phase to reduce observable bias in the output of Eq. 1 in each approach. To prevent a reliance on prior knowledge, the tuning phase is not adjusted to specific density conditions. Tuning P is intended to compensate for the stochasticity of the motion routines in the decentralized approaches, relative to the deterministic sweep of the MNS approach. In the Hier approach, the motion trajectory of the MNS-brain robot is deterministic, so $P = 1$. For each fully decentralized approach, we have tuned P during a trial-and-error testing phase to result in the highest observable performance from Eq. 1 for each approach respectively. After the empirical tuning phase, we have set P as follows: for the voter model $P = 0.48$, for the mean model $P = 0.55$, and for stigmergy $P = 1$. Note that P could in principle be optimized in all approaches (see Sec. 4 for discussion).

2.4.2 Variations

The experiment variations are as follows. In the basic experiments, we vary the stochastic fields $\mathcal{R}$ by distributing 50, 100, 200, or 300 small objects ($5 \times 5 \times 5\,\mathrm{cm}^3$ cubes) uniformly randomly in a square $6 \times 6\,\mathrm{m}^2$ arena, but with a minimum distance of 20 cm between object centerpoints such that robots can always pass between them.[6] The mean true densities for these fields are $\lambda^{\mathrm{true}} = 1.3\bar{8}$, $2.\bar{7}$, $5.\bar{5}$, and $8.\bar{3}$ objects/m^2, respectively. For accuracy under time-invariant $\mathcal{R}$, we test three variations with different true mean densities λ^{true}.

Basic uniform environments can provide a performance benchmark, but it is also important to test stochastic fields $\mathcal{R}$ in which the data has significantly nonuniform spatiotemporal features, such as spatial heterogeneity (high variance between a few sub-regions). We therefore run two variations of the basic experiments: temporal variation and spatial variation.

For temporal variation, we test accuracy under time-varying $\mathcal{R}$, with both minor and major density changes that occur both frequently and infrequently. Specifically, we test fields that fluctuate between $\lambda^{\mathrm{true}} = 1.3\bar{8}$ and either $2.\bar{7}$ or $8.\bar{3}$ objects/m^2, over two different rates of change (either 40 s or 400 s), for a total of four possible variations.

For spatial variation, we test accuracy under spatially nonuniform $\mathcal{R}$ with high variance between sub-regions. Specifically, we test fields in which 100 objects are distributed according to different bivariate unimodal and bimodal skew-normal probability distributions.[7] In the environments generated from these

[6] In all variations, the objects are rotated in 2D around their center points according to angles defined uniformly randomly.

[7] Generated by defining the standard skew-normal shape parameter α uniformly randomly for x and y (separately) in the unimodal and bimodal distributions, in addition to, for the bimodal distributions, defining the location parameter ξ across the arena dimensions uniformly randomly for x and y (separately). For the bimodal distributions, α and ξ are defined separately for each peak and each ξ is constrained to half of the arena. Note that the minimum distance between objects is maintained in these setups.

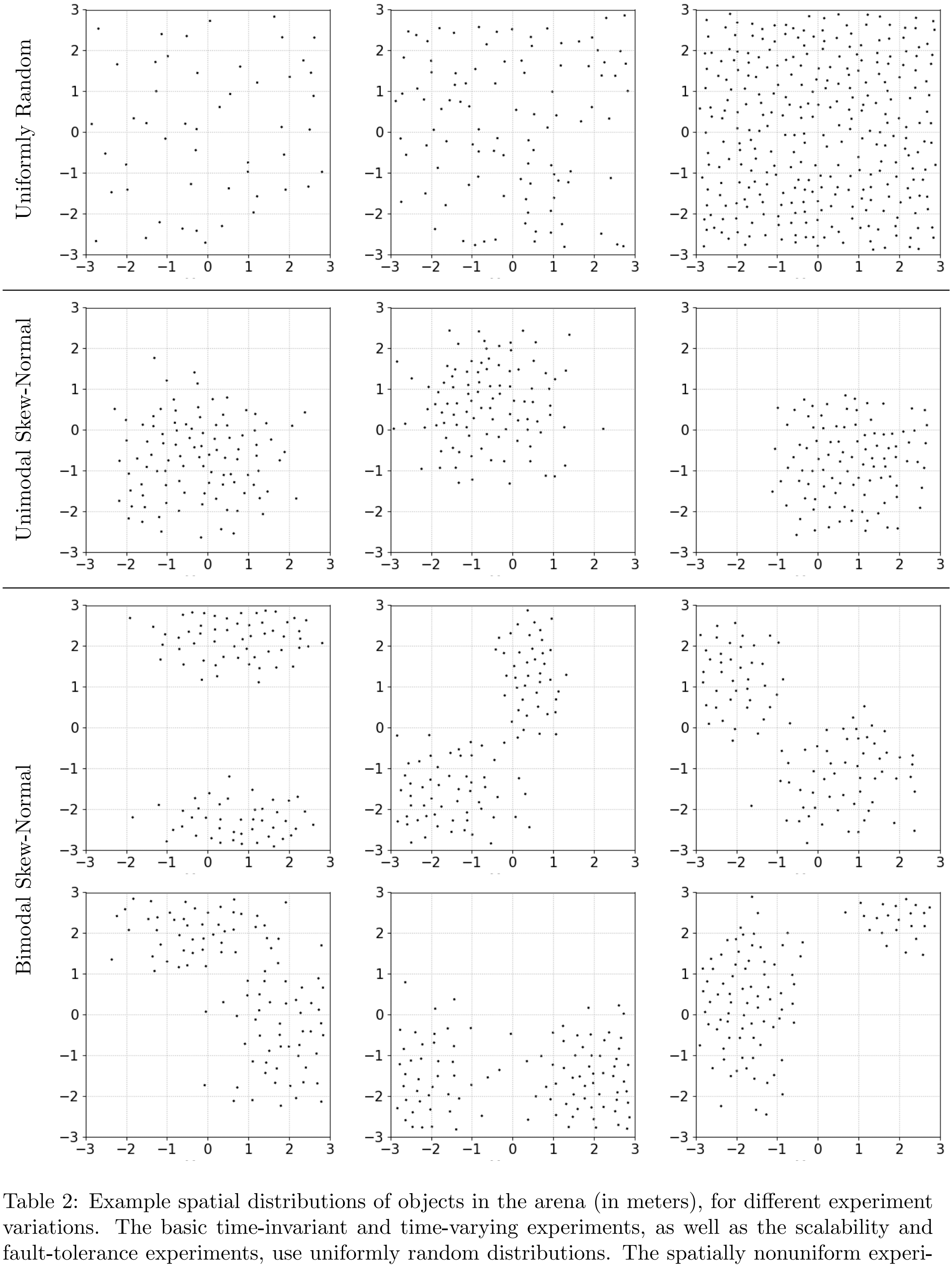


Table 2: Example spatial distributions of objects in the arena (in meters), for different experiment variations. The basic time-invariant and time-varying experiments, as well as the scalability and fault-tolerance experiments, use uniformly random distributions. The spatially nonuniform experiments use bivariate skew-normal distributions (both unimodal and bimodal).

distributions, the mean density of the whole arena is $\overline{\lambda^{\text{true}}} = 2.\bar{7}$ objects/m^2, but the distribution of λ^{true} is of course nonuniform and large portions of the arenas do not contain any objects (see Table 2).

When testing scalability and fault tolerance, all fields are uniform and time-invariant with $\lambda^{\text{true}} = 2.\bar{7}$ (i.e., 100 objects in a 6 x 6 m^2 arena). In the experiments testing scalability, we vary the number of ground robots (4, 8, 12, 16, 20, or 24 ground robots). In the experiments testing fault tolerance, we vary the percentage of ground robots that arbitrarily fail (0%, 25%, 50%, or 75%). Failed ground robots continue their motion routines, communication, and calculations as normal, but experience sensor failure such that they cannot directly count objects—i.e., in Eq. 3 they always add $n = 0$ to $b_t = b_t + n$ as the number of objects detected. (Note that, in the Stig approach, this implies that a failed robot never deposits an artificial pheromone, but can still detect pheromones left by other robots.)

We conduct 10 runs per variation, except for the spatially nonuniform $\mathcal{R}$ experiments with bimodal skew-normal distributions, for which we conduct 20 runs because of the increased complexity in the environment.

2.5 Analysis

Recall that λ^{true} is the true global density of the environment and that robots produce the values ϵ_r (i.e., the individual inference that robot r makes about the density in its own field of view) and λ_r^{app} (i.e., the opinion of robot r on the apparent density in the whole arena, based on the collective influence of all robots). We report the experiment data in plots, in tables in the Supplementary Materials, and in an open-access data repository.[8] We report all data, but given that the time window for sampling inference in the experiments is $k_t^{\max} = 1000$, we only consider time step 1 000 (40 s) and later when making assessments about performance.

We assess the perception accuracy of the approaches according to the error of the robot opinions on apparent density λ^{app}, with respect to true density λ^{true}. The overall error (i.e., incorporating both bias and variance) is measured by the mean squared error of the opinion λ^{app} of robots r over time, calculated as:

$$\text{MSE}(\lambda_{rt}^{\text{app}}) = \tfrac{1}{nm} \textstyle\sum_{r=1}^{n} \sum_{t=1}^{m} (\lambda_t^{\text{true}} - \lambda_{rt}^{\text{app}})^2. \tag{11}$$

The instantaneous error is measured by $\text{MSE}(\lambda_{rt}^{\text{app}})$, calculated as in Eq. 11, but without the terms related to t.

2.5.1 Spatially nonuniform environments

For the spatially nonuniform environments, error is measured by the root mean squared error of the opinion $\text{RMSE}(\lambda_r^{\text{app}}) = \sqrt{\text{MSE}(\lambda_{rt}^{\text{app}})}$ instead of by MSE, because RMSE is better for assessing performance when the errors are large. The measured error in these experiments will be high because the true density λ^{true} varies substantially across the arena but we measure the error according to the mean density $\overline{\lambda^{\text{true}}}$, not the distribution of λ^{true}. We use the mean density $\overline{\lambda^{\text{true}}}$ for the error calculation because the fully decentralized approaches lack any localization or relative positioning capabilities and therefore cannot spatially coordinate their samples among multiple robots in a way that would enable direct comparison with the λ^{true} distribution.

We make the assumption that, in spatially nonuniform environments, the goal of collective perception will be either to (1) estimate the density distribution λ^{true} in a given instance of an environment, or (2) to estimate the overall probability distribution of λ^{true} in a class of environments. In order to fulfill this goal in practice, additional methods would need to be implemented to aggregate the robots' samples and, e.g., perform probability distribution fitting. These additional layers are out of the scope of this paper, but we

[8] https://doi.org/10.5281/zenodo.7244384

assess the similarity of the opinion distribution of each approach to the actual density distribution in the environment, as an indication of how feasible various distribution estimation methods might be.

We estimate the λ^{true} distribution in the environment by taking a set of 50 000 density measurements λ^{REF} of the reference samples (i.e., the objects) using rectilinear windows R of variable area, where the xy positions of the vertices of R are defined uniformly randomly, but with R having minimum length and width of 1.5 m. To make a more direct comparison between the density distribution of the reference samples λ^{REF} and the distribution of opinions λ_r^{app} of the four approaches, we calculate the RMSE(λ^{REF}) according to the mean density $\overline{\lambda^{\text{true}}}$. All estimates are then sorted according to RMSE, and we calculate the mutual information (MI)[9] between the RMSE(λ_r^{app}) of the four approaches and the reference RMSE(λ^{REF}).

3 Results

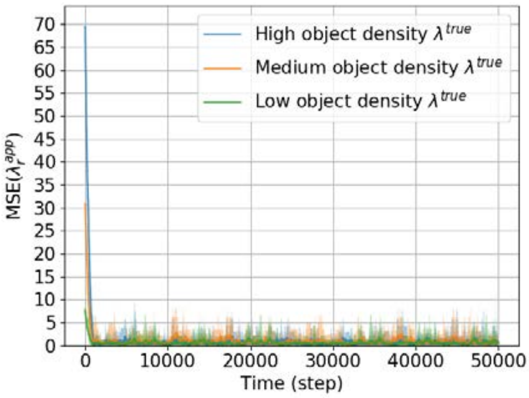
The results show that, under both time-invariant and time-varying stochastic fields $\mathcal{R}$, the Hier approach has higher accuracy, more consistent accuracy, and faster reaction times than the fully decentralized benchmark approaches.

Under all the tested time-invariant fields $\mathcal{R}$ (see Fig. 3), the Hier approach shows minimal error with only minor spikes and stable estimates are reached in less than 50 s. The Mean and Vote approaches are somewhat less accurate, but converge very quickly and rather stably—when error does spike, the spikes are moderately high and last for less than 200 s. The Vote approach is less accurate than the Mean approach and also less consistent (i.e., shows more variance in error across trials). The Stig approach is roughly as accurate and consistent as the Vote approach, but it converges much more slowly and at higher λ^{true} it shows more variance in the mean over time. Across all approaches, time-invariant fields with higher true densities λ^{true} seem to be more challenging than those with lower λ^{true}. This is reasonable, because the objects being detected while perceiving density also create physical obstructions that the robots must avoid or circumnavigate, which implies more interference and therefore more uncertainty originating during the sweeping method $\mathcal{P}$. In the Hier approach, however, the difference in accuracy between lower and higher λ^{true} is extremely small.

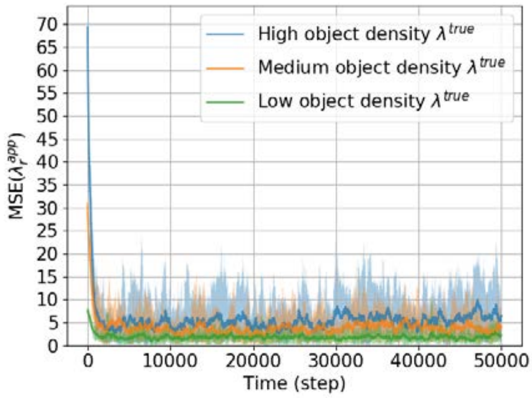
Under time-varying fields $\mathcal{R}$, the Hier approach is the only approach that consistently reaches very low error after all temporal shifts. Under slow fluctuations (see Fig. 4b,d), each approach is able to return to its lowest respective error after a shift to a lower density λ^{true}. In these periods of lowest respective error, the Hier, Mean, and Stig approaches reach a similar minimal error, while the Vote approach converges on a slightly higher error. However, after a shift to a higher density λ^{true}, in all approaches the reaction is slower and error is higher (compared to a shift to lower λ^{true}).

Under time-varying fields, when the fully decentralized approaches are able to converge, all three show noticeably more error than the Hier approach. The Hier approach also reacts much more quickly than the other approaches under all variations of time-varying stochastic fields. The Mean and Vote approaches converge fairly quickly, but the Stig approach converges quite slowly and under major fluctuations cannot converge within 400 s (see Fig. 4d). Under fast fluctuations, these patterns are exacerbated (see Fig. 4a,c). The Mean and Vote approaches have barely enough time to converge under fast, minor fluctuations and cannot converge at all under fast, major fluctuations. The Stig approach never has sufficient time to come close to convergence after a shift to higher λ^{true}. When the fully decentralized approaches do manage to converge under fast fluctuations, their accuracy is the same as it is under slow fluctuations.

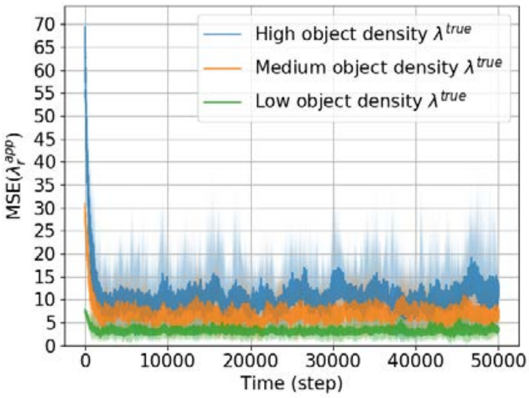
[9] Mutual information between continuous data sets based on entropy estimation from k-nearest neighbors distances [51, 52], using $k = 3$ or 500 neighbors.



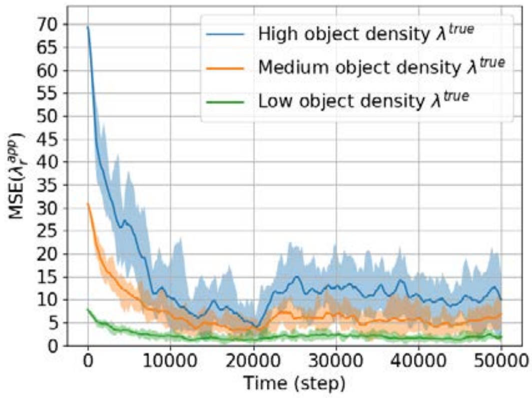
(a) **Hier:** MSE(λ_r^{app})



(b) **Mean:** MSE(λ_r^{app})



(c) **Vote:** MSE(λ_r^{app})



(d) **Stig:** MSE(λ_r^{app})

Figure 3: **Error under time-invariant fields $\mathcal{R}$.** Shaded line plots showing mean, minimum, and maximum of MSE(λ_r^{app}) of all runs, for three variations of true density λ^{true} (objects/m^2): low density $\lambda^{\text{true}} = 2.\bar{7}$, medium density $\lambda^{\text{true}} = 5.\bar{5}$, and high density $\lambda^{\text{true}} = 8.\bar{3}$.

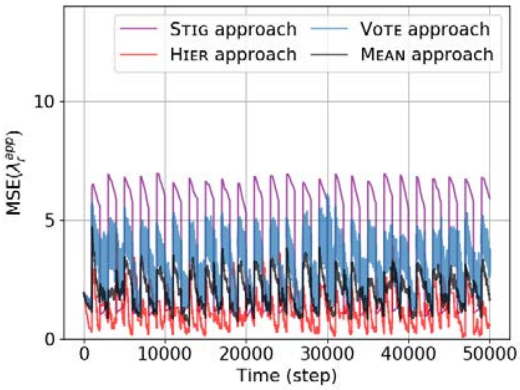
Overall, the Hier approach shows the lowest error (see Tables S1-S5 in the Supplementary Materials for details). Under time-invariant fields (see Fig. 3), mean error in the Hier approach is very minor, almost negligible. The MSE(λ_r^{app}) does spike occasionally in some runs, but generally recovers within 20–40 s. The spikes in error in the Hier approach are much less frequent than in the other approaches and the largest spikes are minor comparatively. Under time-varying fields (see Fig. 4), error in the Hier approach is also overwhelmingly much lower than in the other approaches, under all rates and magnitudes of field fluctuations. Note that, under time-varying fields with major fluctuations, immediately after some of the shifts in the stochastic field, error in the Hier approach spikes more aggressively than in some of the other approaches. However, the Hier approach recovers very quickly after these spikes, and in most cases returns quickly to a lower error than that shown by the other approaches. Overall, we can conclude that the Hier has higher accuracy, has more consistency of accuracy, and reacts accurately and more quickly than the three benchmark approaches.

3.1 Spatially nonuniform environments

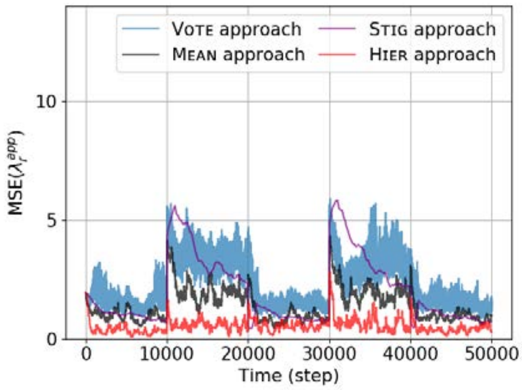
The results show that, under the spatially nonuniform fields $\mathcal{R}$, the distributions of opinions produced by all four approaches have some basic similarity to the true density distributions in terms of shape (see Fig. 5).

For both unimodal and bimodal environments, the mean RMSE(λ_r^{app}) of the Vote and Mean approaches show extremely similar distribution shapes, with the Vote approach having higher error overall. In other words, the Vote approach's distribution curve is effectively a displaced (higher error) duplication of the Mean approach's distribution curve. For both environment types, the Stig approach's distribution curve is much flatter than those of the Vote and Mean approaches, but otherwise is fairly similar to the distribution curve of the Mean approach, both in terms of shape features and overall error. For both environment types, the Hier approach's distribution curve is much steeper than those of the three decentralized approaches, and in the unimodal environments is also quite different from the other three in terms of shape features.

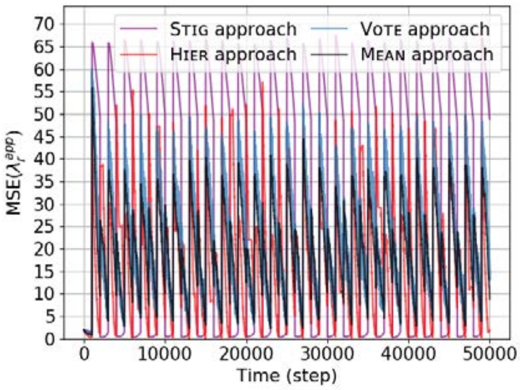
In comparison to the reference samples, all four approaches have much higher overall error, but show



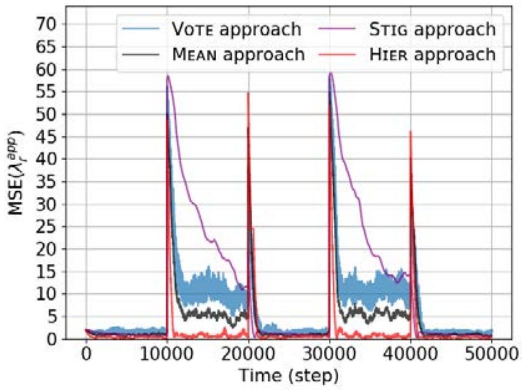
(a) **Fast, minor fluctuations:** field alternating every 40 s between densities that differ slightly: $\lambda^{\text{true}} = 1.3\bar{8}$ or $2.\bar{7}$ objects/m^2.



(b) **Slow, minor fluctuations:** field alternating every 400 s between densities that differ slightly: $\lambda^{\text{true}} = 1.3\bar{8}$ or $2.\bar{7}$ objects/m^2.

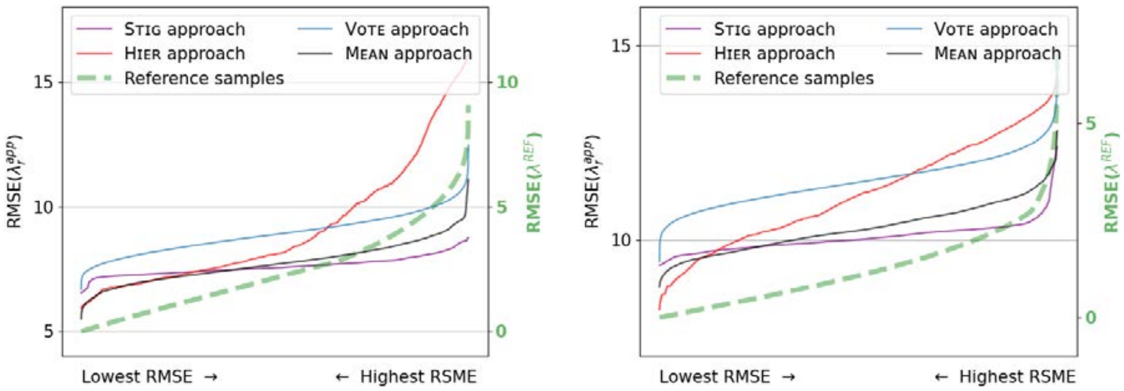


(c) **Fast, major fluctuations:** field alternating every 40 s between densities that differ substantially: $\lambda^{\text{true}} = 1.3\bar{8}$ or $8.\bar{3}$ objects/m^2.



(d) **Slow, major fluctuations:** field alternating every 400 s between densities that differ substantially: $\lambda^{\text{true}} = 1.3\bar{8}$ or $8.\bar{3}$ objects/m^2.

Figure 4: **Error under time-varying fields $\mathcal{R}$.** Mean MSE(λ_r^{app}) of all runs, for four temporal variations of fluctuating true density λ^{true}.



(a) Unimodal skew-normal distributions (b) Bimodal skew-normal distributions

Figure 5: **Error under spatially nonuniform fields $\mathcal{R}$ compared to the reference samples.** Mean RMSE(λ_r^{app}) of the four approaches (left-hand y-axis) compared to the mean RMSE(λ^{REF}) from the reference samples (right-hand y-axis), of all runs, for different fields with spatially nonuniform true density λ^{true}. All RMSE values are calculated according to the mean density $\overline{\lambda^{\mathrm{true}}} = 2.\bar{7}$ (for details of the analysis approach, see Sec. 2.5.1). Estimates are sorted on the x axis according to RMSE. To make visual comparison easier, RMSE(λ^{REF}) has been realigned on the y-axis in relation to RMSE(λ_r^{app}): refer to right-hand and left-hand y-axis labels. For the estimated mutual information (MI), see Table S3 in the supplementary materials.

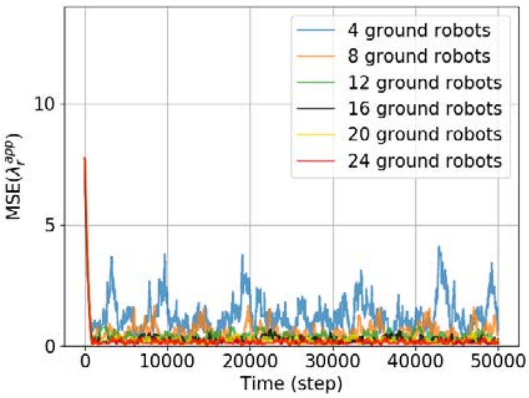
a fair amount of similarity in their distribution curves. For both unimodal and bimodal environments, the three decentralized approaches have approximately the same similarity between their mean RMSE(λ_r^{app}) distributions and the mean RMSE(λ^{REF}) distribution: all three have (1) a sharp increase at the lowest end that the reference distribution does not have at all, (2) a sharp increase at the highest end that is much more gradual in the reference distribution, and (3) a gradual increase in the middle that is flatter than the increase of the reference distribution. Also, in all three approaches, the distance between the lowest and highest values of each respective distribution is noticeably smaller than that of the reference distribution. Of the three, the STIG approach's distribution is the least similar to the reference distribution in terms of rate of increase and overall shape. In comparison to the three decentralized approaches, the HIER approach's distribution in both environment types is more similar to the reference distribution in terms of the difference between the lowest and highest values. The HIER approach's distribution also seems to potentially be more similar in terms of overall shape, but this is difficult to assess from the information available, so overall we consider this finding to be inconclusive.

The estimated mutual information between the mean RMSE(λ_r^{app}) distribution and the mean RMSE(λ^{REF}) distribution is also very similar for all approaches (see Table S3 in the supplementary materials). Overall, the HIER approach is slightly worse in terms of mutual information than the other three, while the STIG approach is slightly better. The STIG approach therefore has the highest dependency (based on mutual information), but the least curve similarity.

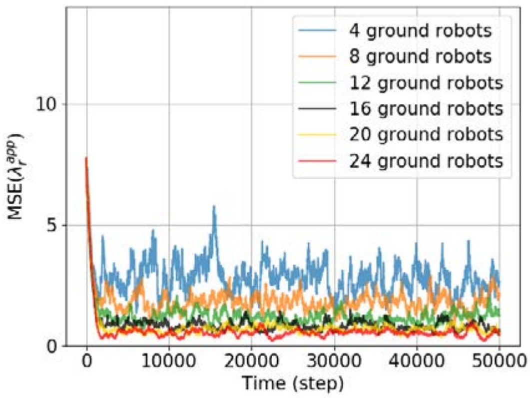
Overall, the accuracy differences between the opinion distribution curves of the four approaches are inconclusive, but all four approaches display some dependency and rough curve similarity with the reference distribution.

3.2 Scalability

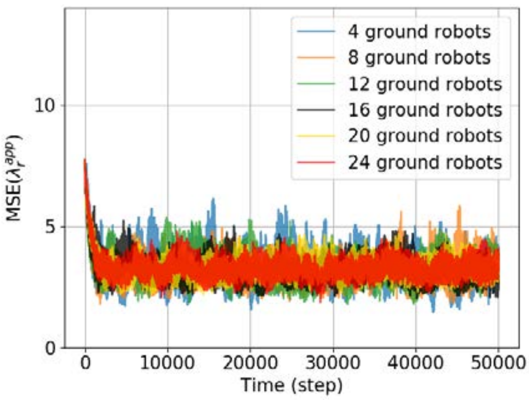
In the scalability setup, in the tested group sizes, none of the approaches show a decrease in accuracy as the group size gets larger (see Fig. 6), implying that the threshold at which inter-robot interference could negatively affect accuracy has not been reached.[10] Rather, we see moderate decreases in performance in some approaches as the group size gets smaller.



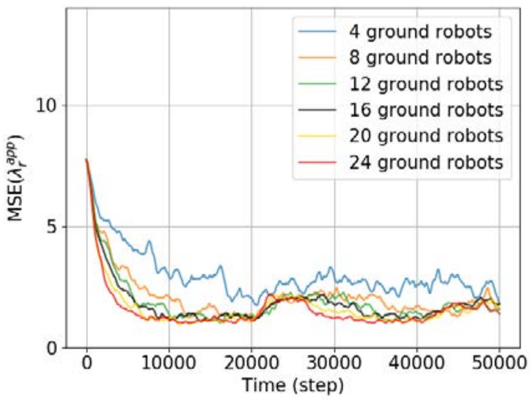
(a) **Hier:** MSE(λ_r^{app})



(b) **Mean:** MSE(λ_r^{app})



(c) **Vote:** MSE(λ_r^{app})



(d) **Stig:** MSE(λ_r^{app})

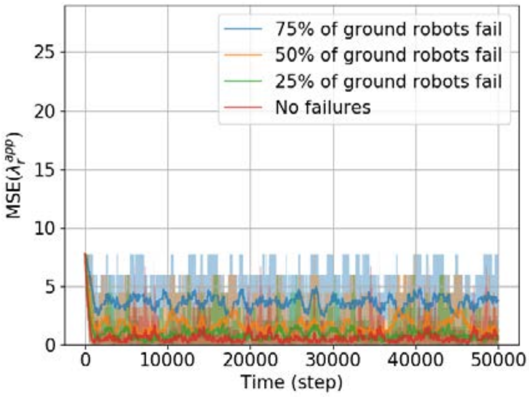
Figure 6: **Error when testing scalability.** Mean MSE(λ_r^{app}) of all runs, for six variations of group size: 4, 8, 12, 16, 20, or 24.

The accuracy difference between group sizes in the Mean approach is the most noticeable, but is still relatively minor—however, its accuracy is also much less consistent in the smaller group sizes, displaying larger and more frequent spikes. The accuracy difference between group sizes in the Hier and Stig approaches is extremely minor, and the Vote approach shows no difference between group sizes. Overall, all four approaches show good scalability of accuracy, because accuracy stays the same or improves as the group size increases. The Vote approach also shows quite good resiliency to smaller group sizes. However, it is important to note that although the accuracy of the Vote approach does not worsen with smaller groups,

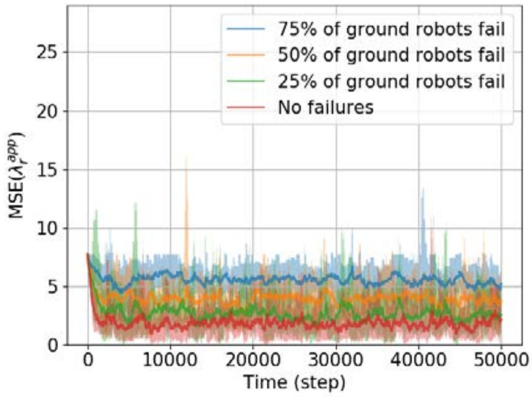
[10] Note that this also implies that none of the approaches, including the Hier approach, display a bottleneck at these sizes. For more discussion of bottlenecks in the MNS approach, please see [43].

it shows more error in all group sizes than the HIER or STIG approaches in their worst group size.

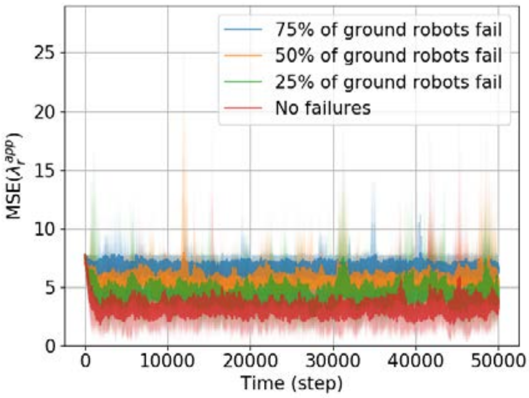
Overall, the HIER approach shows less error than all other approaches for all tested group sizes, and shows extremely minor error (almost no error) in the largest group size (24 ground robots).



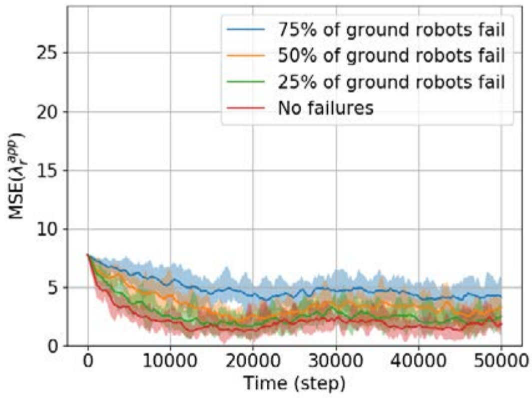
(a) **HIER:** MSE(λ_r^{app})



(b) **MEAN:** MSE(λ_r^{app})



(c) **VOTE:** MSE(λ_r^{app})



(d) **STIG:** MSE(λ_r^{app})

Figure 7: **Error when testing fault tolerance.** Shaded line plots showing mean, minimum, and maximum of MSE(λ_r^{app}) of all runs, for four variations of arbitrary failures: 0% (0 out of 8), 25% (2 out of 8), 50% (4 out of 8), and 75% (6 out of 8) ground robots failing.

3.3 Fault tolerance

In the fault tolerance setup, all four approaches show a noticeable decrease in accuracy as a greater percentage of robots fail. This is expected, because the fault tolerance setup is quite challenging—failed robots continue to move and communicate, but always record that they have directly detected zero objects, and therefore introduce extra bias during sampling method $\mathcal{G}$.

Some of the fault tolerance variants can be considered to have matching respective scalability variants—e.g., the fault tolerance condition of 50% failure leaves the swarm with four failing and four correctly working ground robots, which is a match to the scalability condition of four ground robots. When comparing accuracy under 50% failure to the matching scalability variant, the HIER and MEAN approaches show only slightly more error, while the Vote approach shows a more noticeable increase in error. The Stig approach, by contrast, shows no noticeable difference. All four approaches show a similarly substantial increase in error from 50% failure to 75% failure. Under 75% failure (the highest failure rate), the Hier approach shows noticeably less error than the Mean and Vote approaches. Under 75% failure, the Hier approach shows slightly less error than Stig approach, but with less consistency (noticeably higher spikes).

Overall, the Stig approach shows the least increase in error from no failure to 75% failure, with the other approaches showing similar amounts of increase. However, the Hier approach has a lower mean error than the Stig approach under no, 25%, and 50% failure and a comparable (but still slightly lower) mean error under 75% failure. The Mean and Vote approaches show more error than the other two approaches under all failure rates. Therefore, the Hier approach shows the least overall error in the fault tolerance setups collectively.

Overall, all four approaches are roughly comparable in terms of fault tolerance of accuracy, with the Hier approach having the least overall error and the Stig approach showing the most consistency in its tolerance.

3.4 Summary

All experiment results can be summarized as follows: (i) the Hier approach shows higher accuracy, shows more consistency, and reacts accurately more quickly than the other three approaches; (ii) the Vote approach shows the least change in accuracy under changes in group size, but the Hier approach shows the highest overall accuracy under all group sizes; and (iii) the Stig approach shows the least change in accuracy under changes in failure rates, but the Hier approach shows the highest overall accuracy under all failure rates.

4 Discussion

Based on the assertions made by this paper, in conditions without robot failures, the Hier approach should be able to conduct collective perception with very low uncertainty. In the Hier approach, uncertainty originating during the sweeping method $\mathcal{P}$ is expected to be greatly reduced, because $\mathcal{P}$ is a deterministic method. Likewise, the uncertainty that can originate at the sampling method $\mathcal{T}$ is expected to be greatly reduced, because the deterministic $\mathcal{P}$ is also able to provide nearly optimal sampling distribution (i.e., very evenly distributed sampling sites and a sampling ratio approaching 1, see Fig. 1)[11]. Also, because the MNS network is a connected graph and all sensor information is fused at one node in that graph, no uncertainty should originate during decision sampling $\mathcal{G}$ and uncertainty that can originate during inference ψ, beyond the uncertainty that is inherent without prior knowledge of the stochastic field $\mathcal{R}$, is expected to be minimal.

Where the results of the four approaches can be directly compared with the overall true density λ^{true}, the empirical results align well with these expectations (e.g., see the low error in Fig. 3a). In terms of accuracy, the Hier approach indeed outperforms the others under time-varying and time-invariant stochastic fields. These results therefore support the assertion that self-organized hierarchy can improve perceptual accuracy by integrating aspects of centralized control that reduce uncertainty.

Under the spatially nonuniform environments, where the results are compared to a reference set of density measurements, the opinion distributions of all four approaches display similarity to the reference distribution, with the performance differences between the four approaches being inconclusive. Overall, the empirical results indicate that distribution estimation (either of the density distribution λ^{true} in a given

[11] See [40] for demonstration of uniform and complete spatial coverage using the MNS.

instance of an environment, or of the overall probability distribution of λ^{true} in a class of environments) would in principle be feasible in any of the four approaches, and further study would be required to assess the performance differences between them. However, in practice, fully decentralized approaches to distribution estimation (e.g., probability distribution fitting) would give each robot access either to only a subset of the total samples or to an asynchronously updated opinion on the overall distribution, introducing an extra source of uncertainty. By contrast, the information that would be available for distribution estimation in the HIER approach would be the same as that shown in Fig. 5, without additional sources of uncertainty.

Also based on the assertions of this paper, in setups testing scalability and fault tolerance, the HIER approach should be able to conduct collective perception without an outsized increase in error, as compared to the fully decentralized approaches. Indeed, error in the HIER approach does not increase beyond a level comparable to any of the other approaches, in all scalability and fault tolerance setups—and, in most of these variants, the error of the HIER approach is lower than in all other approaches. Therefore, the empirical results again align with expectations: the scalability and fault tolerance of the HIER approach can be considered commensurate with the other approaches. The results therefore support the assertion that self-organized hierarchy, despite introducing some aspects of centralized control, can maintain the beneficial aspects of decentralized control.

Poor performance of VOTE approach Although the voter decision model has been shown to be relatively accurate with discrete best-of-n decision-making (e.g., choosing one of a few color options), this is because it works well at accurately converging on the most common opinion in a group. When dealing with high-variance samples of a continuous stochastic field, it has no mechanisms to compensate for bias in the sampling nor reduce the variance—it essentially shuffles several highly biased opinions among group members, so the distribution of opinions before and after the voter decision model process is statistically indistinguishable. This notion is confirmed empirically by the similarity between the $\text{MSE}(\epsilon_{rt})$ and $\text{MSE}(\lambda_{rt}^{\text{app}})$, i.e., the collective error in robot opinions before and after the voter decision model process (see Tables S1-S5 in the Supplementary Materials). The VOTE approach results therefore represent roughly what a single robot would perceive on its own, which explains the poor accuracy.

Tuning parameters Tuning parameters are known to be difficult to design, whether using simulated or real testing. Here, P is tuned during an initial manual testing phase. Results should be interpreted with the understanding that parameter P could in principle be optimized further in every approach, but that this optimization could improve only the mean bias of the opinions, not the variance (refer to Eq. 1). It also needs to be acknowledged that optimization of P would require prior knowledge of the stochastic field $\mathcal{R}$. For example, if P was optimized to a very high density field $\mathcal{R}$, the performance under low density would presumably suffer, and vice versa.

4.1 Future work

Future work on collective perception should study the robustness of sampling methods under other types of challenging conditions, such as environments where it is difficult for robots to have good coverage over a spatial area due to large obstacles or boundary irregularity.

More advanced inference methods could be investigated, such as weighting samples by inclusion probability. However, most of these inference methods require some prior knowledge of the stochastic field [35], which might not be suitable for deployment-ready self-organized robot systems. General optimized performance could also be investigated, including for situations in which it is not possible to have prior knowledge of the stochastic field. Optimization without prior knowledge of the field would likely need to be based on online adaptation to or learning of the spatial information being sampled, which could in principle be done in any of the approaches, but would certainly be easier to design in the Hier approach than in the fully decentralized approaches.

It would also be useful to study other types of failures, such as motion control errors (e.g., robots get stuck in corners), odometry errors (e.g., robots believe they have travelled far less distance than they actually have), other sensor errors (e.g., robots believe they are always detecting an object), or full robot shutdown, as well as the timing of such failures and other factors that could exacerbate their impact.

Improvements to the Hier approach Although it might be simple to design and implement system-wide adaptivity in the Hier approach using the MNS, we have not added such behaviors here, because it is comparatively much more difficult to implement such adaptivity in the fully decentralized approaches. To make as fair a comparison as possible, we limited the capabilities of the Hier approach to be more similar to the abilities of the fully decentralized approaches. For instance, the fault tolerance results of the Hier approach reflect the lack of adaptability implemented currently—e.g., the MNS-brain robot believes its indirect field of view to be that of eight fully functioning robots, even though it only has two fully functioning robots remaining. Minor additions to the approach could be made to allow parents to detect malfunctioning sensor readings from children and substantially improve the fault tolerance performance. This would be relatively straightforward to implement and design in the Hier approach (as compared to fully decentralized approaches), because of the aspects of centralization that are integrated into self-organization when using the MNS framework. More broadly, advanced behaviors are much simpler to implement using the self-organized hierarchy capabilities of the MNS than when using strictly decentralized approaches, so the addition of even more advanced adaptability such as self-awareness can be considered. The self-reconfiguration capabilities of the MNS [5] would also make it fairly straightforward to implement additional behaviors that help keep the robot formation together when sweeping challenging environments, such as changing the formation shape or temporarily splitting into multiple smaller formations that sweep independently until they have the opportunity to re-merge.

5 Conclusion

We have identified the sources of uncertainty that are present in the collective perception problem, especially when perceiving an absolute condition without prior knowledge, and detailed why this uncertainty is reduced by using a self-organized hierarchy. We have supported this assertion empirically by showing that a proof-of-concept self-organized hierarchy approach (based on the MNS framework) is generally more accurate, more consistent, and faster than fully decentralized benchmark approaches. We have also shown that the self-organized hierarchy approach to collective perception, besides producing more accurate estimates of spatial data, does not suffer substantial scalability or fault tolerance disadvantages compared to fully decentralized benchmark approaches. We have also tested spatially nonuniform environments with high heterogeneity and shown that the self-organized hierarchy approach is comparable to the other approaches in terms of accuracy compared to reference samples. Therefore, the comparative ease of designing system-wide behaviors under self-organized hierarchy can be taken advantage of, without a reduction in the performance benefits that are often associated with swarm robotics approaches.

Acknowledgments

General

The authors would like to thank Weixu Zhu and Michael Allwright for their advice regarding external code libraries used in the experiments.

Author Contributions

All authors contributed to conceiving the study. A. Jamshidpey conducted the experiments and collected the data. A. Jamshidpey and M. K. Heinrich led the experiment design and analysis. M. K. Heinrich led the writing of the manuscript and all authors read and approved the final manuscript.

Funding

This work was supported by the Program of Concerted Research Actions (ARC) of the Université libre de Bruxelles, the Natural Sciences and Engineering Research Council of Canada under Discovery Grants, and the Ontario Trillium Scholarship Program through the University of Ottawa and the Government of Ontario, Canada. Marco Dorigo and Mary Katherine Heinrich acknowledge support from the Belgian F.R.S.-FNRS, of which they are a Research Director and a Postdoctoral Researcher respectively.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this article.

Data Availability

The code used in the experiments is available open-source on GitHub: `https://github.com/BlueDiamond07/Collective_perception`. All experiment data collected during the study are available open-access on Zenodo: `https://doi.org/10.5281/zenodo.7908113`.

Supplementary Materials

Tables S1-S5. Tables of experiment results.

References

[1] Y. Khaluf, P. Simoens, and H. Hamann, "The neglected pieces of designing collective decision-making processes," *Frontiers in Robotics and AI*, vol. 6, p. 16, 2019.

[2] M. Dorigo, G. Theraulaz, and V. Trianni, "Reflections on the future of swarm robotics," *Science Robotics*, vol. 5, no. 49, eabe4385, 2020.

[3] M. Dorigo, G. Theraulaz, and V. Trianni, "Swarm robotics: Past, present, and future [point of view]," *Proceedings of the IEEE*, vol. 109, no. 7, pp. 1152–1165, 2021.

[4] N. Mathews, A. L. Christensen, R. O'Grady, F. Mondada, and M. Dorigo, "Mergeable nervous systems for robots," *Nature communications*, vol. 8, no. 439, 2017.

[5] W. Zhu, M. Allwright, M. K. Heinrich, S. Oğuz, A. L. Christensen, and M. Dorigo, "Formation control of UAVs and mobile robots using self-organized communication topologies," in *International conference on swarm intelligence*, Springer, 2020, pp. 306–314.

[6] G. Valentini, E. Ferrante, and M. Dorigo, "The best-of-n problem in robot swarms: Formalization, state of the art, and novel perspectives," *Frontiers in Robotics and AI*, vol. 4, p. 9, 2017.

[7] G. Valentini, H. Hamann, M. Dorigo, *et al.*, "Self-organized collective decision making: The weighted voter model.," in *AAMAS*, 2014, pp. 45–52.

[8] G. Valentini, D. Brambilla, H. Hamann, and M. Dorigo, "Collective perception of environmental features in a robot swarm," in *International Conference on Swarm Intelligence*, Springer, 2016, pp. 65–76.

[9] M. K. Heinrich, M. Wahby, M. Dorigo, and H. Hamann, "Swarm robotics," in *Cognitive robotics*, A. Cangelosi and M. Asada, Eds., MIT Press, 2022.

[10] Z. Yan, N. Jouandeau, and A. A. Cherif, "A survey and analysis of multi-robot coordination," *International Journal of Advanced Robotic Systems*, vol. 10, no. 12, p. 399, 2013.

[11] S. Sun, H. Lin, J. Ma, and X. Li, "Multi-sensor distributed fusion estimation with applications in networked systems: A review paper," *Information Fusion*, vol. 38, pp. 122–134, 2017.

[12] Y. Rizk, M. Awad, and E. W. Tunstel, "Cooperative heterogeneous multi-robot systems: A survey," *ACM Computing Surveys (CSUR)*, vol. 52, no. 2, pp. 1–31, 2019.

[13] B. Li, Y. Xian, D. Zhang, J. Su, X. Hu, and W. Guo, "Multi-sensor image fusion: A survey of the state of the art," *Journal of Computer and Communications*, vol. 9, no. 6, pp. 73–108, 2021.

[14] V. Strobel, E. Castelló Ferrer, and M. Dorigo, "Managing byzantine robots via blockchain technology in a swarm robotics collective decision making scenario," in *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems*, M. Dastani, G. Sukthankar, E. André, and S. Koenig, Eds., ser. AAMAS '18, Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2018, pp. 541–549.

[15] J. T. Ebert, M. Gauci, and R. Nagpal, "Multi-feature collective decision making in robot swarms," in *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, 2018, pp. 1711–1719.

[16] Q. Shan and S. Mostaghim, "Collective decision making in swarm robotics with distributed bayesian hypothesis testing," in *International Conference on Swarm Intelligence*, Springer, 2020, pp. 55–67.

[17] P. Bartashevich and S. Mostaghim, "Benchmarking collective perception: New task difficulty metrics for collective decision-making," in *EPIA Conference on Artificial Intelligence*, Springer, 2019, pp. 699–711.

[18] Q. Shan and S. Mostaghim, “Discrete collective estimation in swarm robotics with distributed bayesian belief sharing,” *Swarm Intelligence*, vol. 15, no. 4, pp. 377–402, 2021.

[19] J. Prasetyo, G. De Masi, P. Ranjan, and E. Ferrante, “The best-of-n problem with dynamic site qualities: Achieving adaptability with stubborn individuals,” in *International Conference on Swarm Intelligence*, Springer, 2018, pp. 239–251.

[20] J. Prasetyo, G. De Masi, and E. Ferrante, “Collective decision making in dynamic environments,” *Swarm Intelligence*, vol. 13, no. 3, pp. 217–243, 2019.

[21] Y. Khaluf, “Edge detection in static and dynamic environments using robot swarms,” in *2017 IEEE 11th International Conference on Self-Adaptive and Self-Organizing Systems (SASO)*, IEEE, 2017, pp. 81–90.

[22] M. Wahby, J. Petzold, C. Eschke, T. Schmickl, and H. Hamann, “Collective change detection: Adaptivity to dynamic swarm densities and light conditions in robot swarms,” in *ALIFE 2019: The 2019 Conference on Artificial Life*, MIT Press, 2019, pp. 642–649.

[23] Y. Khaluf, M. Allwright, I. Rausch, P. Simoens, and M. Dorigo, “Construction task allocation through the collective perception of a dynamic environment,” in *International Conference on Swarm Intelligence*, Springer, 2020, pp. 82–95.

[24] J. Capitan, M. T. Spaan, L. Merino, and A. Ollero, “Decentralized multi-robot cooperation with auctioned pomdps,” *The International Journal of Robotics Research*, vol. 32, no. 6, pp. 650–671, 2013.

[25] M. Mazo, A. Speranzon, K. H. Johansson, and X. Hu, “Multi-robot tracking of a moving object using directional sensors,” in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, IEEE, vol. 2, 2004, pp. 1103–1108.

[26] F. M. Mirzaei, A. I. Mourikis, and S. I. Roumeliotis, “On the performance of multi-robot target tracking,” in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, IEEE, 2007, pp. 3482–3489.

[27] T. Rodrigues, M. Duarte, S. Oliveira, and A. L. Christensen, “Beyond onboard sensors in robotic swarms—local collective sensing through situated communication,” in *ICAART (2)*, 2015, pp. 111–118.

[28] A. W. Stroupe, M. C. Martin, and T. Balch, “Distributed sensor fusion for object position estimation by multi-robot systems,” in *Proceedings 2001 ICRA. IEEE international conference on robotics and automation (Cat. No. 01CH37164)*, IEEE, vol. 2, 2001, pp. 1092–1098.

[29] T. Sasaoka, I. Kimoto, Y. Kishimoto, K. Takaba, and H. Nakashima, “Multi-robot slam via information fusion extended kalman filters,” *IFAC-PapersOnLine*, vol. 49, no. 22, pp. 303–308, 2016.

[30] V. Zadorozhny and M. Lewis, “Information fusion based on collective intelligence for multi-robot search and rescue missions,” in *2013 IEEE 14th International Conference on Mobile Data Management*, IEEE, vol. 1, 2013, pp. 275–278.

[31] S. Czarnetzki and C. Rohde, "Handling heterogeneous information sources for multi-robot sensor fusion," in *2010 IEEE Conference on Multisensor Fusion and Integration*, IEEE, 2010, pp. 133–138.

[32] M. Otte, "Collective cognition and sensing in robotic swarms via an emergent group-mind," in *International Symposium on Experimental Robotics*, Springer, 2016, pp. 829–840.

[33] S. Kornienko, O. Kornienko, C. Constantinescu, M. Pradier, and P. Levi, "Cognitive micro-agents: Individual and collective perception in microrobotic swarm," in *Proc. of the IJCAI-05 Workshop on Agents in real-time and dynamic environments, Edinburgh, UK*, 2005, pp. 33–42.

[34] A. Giusti, J. Nagi, L. Gambardella, and G. A. Di Caro, "Cooperative sensing and recognition by a swarm of mobile robots," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2012, pp. 551–558.

[35] J.-F. Wang, A. Stein, B.-B. Gao, and Y. Ge, "A review of spatial sampling," *Spatial Statistics*, vol. 2, pp. 1–14, 2012.

[36] E. R. Hunt, S. Jones, and S. Hauert, "Testing the limits of pheromone stigmergy in high-density robot swarms," *Royal Society open science*, vol. 6, no. 11, p. 190 225, 2019.

[37] A. Jamshidpey, M. Wahby, M. K. Heinrich, M. Allwright, W. Zhu, and M. Dorigo, "Centralization vs. decentralization in multi-robot coverage: Ground robots under UAV supervision," IRIDIA, Université Libre de Bruxelles, Brussels, Belgium, Tech. Rep. TR/IRIDIA/2021-008, 2021.

[38] F. Comets, S. Popov, G. M. Schütz, and M. Vachkovskaia, "Billiards in a general domain with random reflections," *Archive for rational mechanics and analysis*, vol. 191, no. 3, pp. 497–537, 2009.

[39] M. Allwright, W. Zhu, and M. Dorigo, "An open-source multi-robot construction system," *HardwareX*, vol. 5, e00050, 2019.

[40] A. Jamshidpey, W. Zhu, M. Wahby, M. Allwright, M. K. Heinrich, and M. Dorigo, "Multi-robot coverage using self-organized networks for central coordination," in *Swarm Intelligence – Proceedings of ANTS 2020 – Twelfth International Conference*, ser. Lecture Notes in Computer Science, vol. 12421, Berlin, Germany: Springer, 2020, pp. 216–228.

[41] Y. Zhang *et al.*, "Self-reconfigurable hierarchical frameworks for bearing-based formation control of robot swarms," IRIDIA, Université Libre de Bruxelles, Brussels, Belgium, Tech. Rep. TR/IRIDIA/2021-004, 2021.

[42] H. Choset and P. Pignon, "Coverage path planning: The boustrophedon cellular decomposition," in *Field and service robotics*, Springer, 1998, pp. 203–209.

[43] W. Zhu, M. Allwright, M. K. Heinrich, S. Oğuz, A. L. Christensen, and M. Dorigo, "Formation control of UAVs and mobile robots using self-organized communication topologies," in *Swarm Intelligence – Proceedings of ANTS 2020 – Twelfth International Conference*, ser. Lecture Notes in Computer Science, vol. 12421, Berlin, Germany: Springer, 2020, pp. 306–314.

[44] C. Pinciroli *et al.*, "ARGoS: A modular, parallel, multi-engine simulator for multi-robot systems," *Swarm intelligence*, vol. 6, no. 4, pp. 271–295, 2012.

[45] M. Allwright, N. Bhalla, C. Pinciroli, and M. Dorigo, "ARGoS plug-ins for experiments in autonomous construction," IRIDIA, Université Libre de Bruxelles, Brussels, Belgium, Tech. Rep. TR/IRIDIA/2018-007, 2018.

[46] M. Allwright, N. Bhalla, C. Pinciroli, and M. Dorigo, "Simulating multi-robot construction in ARGoS," in *Swarm Intelligence – Proceedings of ANTS 2018 – Eleventh International Conference*, ser. Lecture Notes in Computer Science, vol. 11172, Berlin, Germany: Springer, 2018, pp. 188–200.

[47] F. Mondada *et al.*, "The e-puck, a robot designed for education in engineering," in *Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions*, IPCB: Instituto Politécnico de Castelo Branco, vol. 1, 2009, pp. 59–65.

[48] Á. Gutiérrez, A. Campo, M. Dorigo, J. Donate, F. Monasterio-Huelin, and L. Magdalena, "Open e-puck range & bearing miniaturized board for local communication in swarm robotics," in *IEEE International Conference on Robotics and Automation*, IEEE, 2009, pp. 3111–3116.

[49] A. G. Millard *et al.*, "The Pi-puck extension board: A Raspberry Pi interface for the e-puck robot platform," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2017, pp. 741–748.

[50] S. Oguz *et al.*, "S-drone: An open-source quadrotor for experimentation in swarm robotics," IRIDIA, Université Libre de Bruxelles, Brussels, Belgium, Tech. Rep. TR/IRIDIA/2022-010, 2022.

[51] A. Kraskov, H. Stögbauer, and P. Grassberger, "Erratum: Estimating mutual information [phys. rev. e 69, 066138 (2004)]," *Physical Review E*, vol. 83, no. 1, p. 019 903, 2011.

[52] B. C. Ross, "Mutual information between discrete and continuous data sets," *PloS one*, vol. 9, no. 2, e87357, 2014.

Supplementary Materials

Tables S1-S5. Tables of experiment results.

Recall that, in the Stig and Hier approaches, robots influence the collective opinion before the inference of ϵ_r, so $\epsilon_{rt} = \lambda_{rt}^{\text{app}}$. Therefore, MSE($\epsilon_{rt}$) is redundant in the Stig and Hier approaches and is not reported in the tables.

Table S1: **Time-invariant field $\mathcal{R}$.** MSE(ϵ_{rt}) and MSE($\lambda_{rt}^{\text{app}}$) of all runs.

Approach	λ^{true} (objects/m^2)	MSE(ϵ_{rt})	MSE($\lambda_{rt}^{\text{app}}$)
Hier	$2.\bar{7}$	-	0.5857
	$5.\bar{5}$	-	1.0567
	$8.\bar{3}$	-	1.2053
Vote	$2.\bar{7}$	3.2392	3.2252
	$5.\bar{5}$	7.0243	7.0108
	$8.\bar{3}$	10.6300	10.6169
Mean	$2.\bar{7}$	3.6938	1.8346
	$5.\bar{5}$	7.1599	3.9410
	$8.\bar{3}$	9.7224	5.6553
Stig	$2.\bar{7}$	-	2.0944
	$5.\bar{5}$	-	6.8743
	$8.\bar{3}$	-	13.5151

Table S2: **Time-varying field $\mathcal{R}$.** MSE(ϵ_{rt}) and MSE($\lambda_{rt}^{\text{app}}$) of all runs.

Approach	λ^{true} (objects/m^2)	interval (s)	MSE(ϵ_{rt})	MSE($\lambda_{rt}^{\text{app}}$)
HIER	$1.3\bar{8}$, $2.\bar{7}$	40 s	-	1.2250
		400 s	-	0.5247
	$1.3\bar{8}$, $8.\bar{3}$	40 s	-	18.0003
		400 s	-	1.9763
VOTE	$1.3\bar{8}$, $2.\bar{7}$	40 s	2.9333	3.0351
		400 s	2.2762	2.3249
	$1.3\bar{8}$, $8.\bar{3}$	40 s	19.7566	22.2476
		400 s	6.3037	6.9537
MEAN	$1.3\bar{8}$, $2.\bar{7}$	40 s	3.3630	2.0121
		400 s	2.6962	1.3982
	$1.3\bar{8}$, $8.\bar{3}$	40 s	20.5309	19.5730
		400 s	6.2645	4.5715
STIG	$1.3\bar{8}$, $2.\bar{7}$	40 s	-	3.7937
		400 s	-	2.0009
	$1.3\bar{8}$, $8.\bar{3}$	40 s	-	29.9246
		400 s	-	11.4142

Table S3: **Spatially nonuniform field $\mathcal{R}$.** Mutual information (MI) between mean RMSE(λ_r^{app}) of all runs and mean RMSE(λ^{REF}). (For MI, higher values indicate higher dependency.)

Approach	Distribution type	MI (when $k = 3$)	MI (when $k = 500$)
HIER	unimodal	7.3004	4.0194
	bimodal	7.8164	4.0796
VOTE	unimodal	8.2576	4.3127
	bimodal	8.4299	4.2653
MEAN	unimodal	8.3757	4.2834
	bimodal	8.6759	4.2706
STIG	unimodal	8.8966	4.2369
	bimodal	8.8704	4.1572

Table S4: **Scalability.** MSE(ϵ_{rt}) and MSE($\lambda_{rt}^{\text{app}}$) of all runs.

Approach	# of ground robots	MSE(ϵ_{rt})	MSE($\lambda_{rt}^{\text{app}}$)
HIER	4	-	1.2847
	8	-	0.5857
	12	-	0.4030
	16	-	0.2741
	20	-	0.2580
	24	-	0.2006
VOTE	4	3.3315	3.3373
	8	3.2392	3.2252
	12	3.2694	3.2414
	16	3.2422	3.2570
	20	3.3064	3.2931
	24	3.2578	3.2818
MEAN	4	3.8103	2.8065
	8	3.6938	1.8346
	12	3.7828	1.2344
	16	3.7317	0.8906
	20	3.7679	0.7414
	24	3.7164	0.5966
STIG	4	-	2.9956
	8	-	2.0944
	12	-	1.8724
	16	-	1.7994
	20	-	1.6289
	24	-	1.5476

Table S5: **Fault tolerance.** MSE(ϵ_{rt}) and MSE($\lambda_{rt}^{\text{app}}$) of all runs.

Approach	ground robot failure rate	MSE(ϵ_{rt})	MSE($\lambda_{rt}^{\text{app}}$)
HIER	no failure	-	0.5857
	25%	-	0.8775
	50%	-	1.6370
	75%	-	3.6568
VOTE	no failure	3.2392	3.2252
	25%	4.4482	4.4156
	50%	5.4810	5.5005
	75%	6.6035	6.6078
MEAN	no failure	3.6938	1.8346
	25%	4.8110	2.6813
	50%	5.6990	3.9270
	75%	6.7214	5.5371
STIG	no failure	-	2.0944
	25%	-	2.6528
	50%	-	3.5195
	75%	-	4.8546