The International Journal of Robotics Research

http://ijr.sagepub.com/

Distributed robotic sensor networks: An information-theoretic approach

Brian J Julian, Michael Angermann, Mac Schwager and Daniela Rus

The International Journal of Robotics Research 2012 31: 1134 originally published online 6 August 2012
DOI: 10.1177/0278364912452675

The International Journal of
Robotics Research
31(10) 1134–1154
© The Author(s) 2012
Reprints and permission:
sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/0278364912452675
ijr.sagepub.com

Distributed robotic sensor networks: An information-theoretic approach

Brian J Julian[1,2], Michael Angermann[3], Mac Schwager[4] and Daniela Rus[1]

Abstract

In this paper we present an information-theoretic approach to distributively control multiple robots equipped with sensors to infer the state of an environment. The robots iteratively estimate the environment state using a sequential Bayesian filter, while continuously moving along the gradient of mutual information to maximize the informativeness of the observations provided by their sensors. The gradient-based controller is proven to be convergent between observations and, in its most general form, locally optimal. However, the computational complexity of the general form is shown to be intractable, and thus non-parametric methods are incorporated to allow the controller to scale with respect to the number of robots. For decentralized operation, both the sequential Bayesian filter and the gradient-based controller use a novel consensus-based algorithm to approximate the robots' joint measurement probabilities, even when the network diameter, the maximum in/out degree, and the number of robots are unknown. The approach is validated in two separate hardware experiments each using five quadrotor flying robots, and scalability is emphasized in simulations using 100 *robots.*

Keywords

1. Introduction

We consider the task of using a robotic sensor network to infer the state of an environment, for example to collect military intelligence, gather scientific data, or monitor ecological events of interest (Figure 1). Our goal is to enable the robots to efficiently, robustly, and provably learn their environment and reason where to make future sensor measurements. To accomplish this goal, we propose an approach that uses a three-step loop: (i) the robots form a joint observation from measurements taken by all robots' sensors; (ii) the robots apply a sequential Bayesian filter to update their belief about the state of the environment given the observation; and (iii) the robots move to maximize the informativeness of their next observation. Our approach explicitly accounts for the limited computational resources of the robots, the finite bandwidth of their communication network, and the decentralized nature of their computation. This is a significant improvement over current methods which typically make ideal assumptions about one or more of these components. The control strategy also dynamically adapts to changing network connectivity and is scalable with respect to the number of robots, enabling robot teams of arbitrary size. We present the results of both indoor and outdoor experiments using five quadrotor flying robots to demonstrate the effectiveness of our approach. We then present the results of a simulated network of 100 robots to show scalability.

The key to our solution is in the use of non-parametric,[1] sample-based representations of the probability distributions present in the system. Computations on these non-parametric representations are performed using a novel consensus-based algorithm. This approach leads to scalability with respect to the number of robots, and allows for completely decentralized computation under few probabilistic assumptions. Specifically, we do not assume that any probability distribution can be accurately represented by a Gaussian distribution. Assuming Gaussianity significantly simplifies many aspects of the system. However, Gaussian distributions often do not adequately represent the characteristics of realistic environments and sensors, and

[1]Computer Science and Artificial Intelligence Lab, MIT, Cambridge, MA, USA
[2]MIT Lincoln Laboratory, Lexington, MA, USA
[3]Institute of Communications and Navigation, DLR Oberpfaffenhofen, Wessling, Germany
[4]Department of Mechanical Engineering, Boston University, Boston, MA, USA

Corresponding author:
Brian J Julian, Computer Science and Artificial Intelligence Lab, MIT, Cambridge, MA 02139, USA.
Email: bjulian@mit.edu

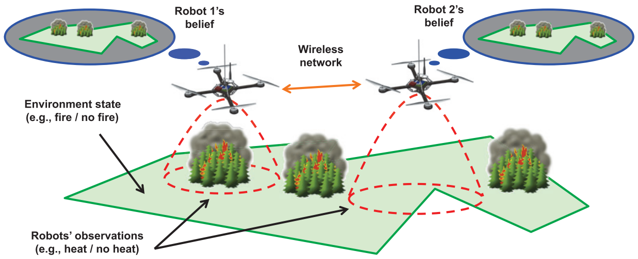
Robot 1's belief
Robot 2's belief
Wireless network
Environment state (e.g., fire / no fire)
Robots' observations (e.g., heat / no heat)

Fig. 1. An example application of using two robots equipped with sensors to monitor the state of a forest fire. The robot's observations formed from sensor measurements provide information on whether or not heat exists within the field of view (red dashed circles), and the robots share this information on the wireless network (orange arrow) to improve the quality of the other robots' beliefs (blue circles).

may result in misleading inferences and poor controller performance. Instead, we approximate the robots' beliefs and likely observations using sample sets that are distributively formed and provably unbiased (Julian et al., 2012b). These non-parametric sampled distributions are used both for the sequential Bayesian filter to update the robots' beliefs, and for the information seeking controller to move the robots and orient their sensors.

The robots are controlled to seek informative observations by moving along the gradient of mutual information at each time step. Mutual information is a quantity from information theory (Cover and Thomas, 1991) that predicts how much a new observation will increase the certainty of the robots' beliefs about the environment state. Thus, by moving along the mutual information gradient, the robots maximally increase the informativeness of their next observation. However, the computation of the mutual information gradient and the sequential Bayesian filter requires global knowledge that is not readily available in a decentralized setting. To overcome this requirement, we use a consensus-based algorithm specifically designed to successively approximate the required global quantities with local estimates. These approximations provably converge to the desired global quantities as the number of the consensus rounds grows or as the network graph becomes complete (i.e. fully connected). Convergence is guaranteed for robots without any knowledge of the number of the robots in the network, nor any knowledge about the corresponding graph's topology (e.g. maximum in/out degree). In addition, the consensus-based algorithm preserves an appropriate probabilistic weighting even if it is stopped before the approximations have fully converged.

For our hardware experiments, the computations for the distributed inference and coordination were done in real-time and were driven by simulated measurements as if they were received from downward looking sensors attached to the robots. To emphasize the performance of our consensus-based algorithm, the robots were not given any global knowledge of the network topology, for which an ideal disk model was used to restrict connectivity. Lastly, we show the scalability of our approach by using the software developed for the hardware experiments to simulate a 100 robot system.

The main contributions of this work are as follows.

(i) We formalize the problem of inferring the state of an environment using multiple robots equipped with sensors, and apply the gradient of mutual information to control the robots to make informative measurements.
(ii) We propose non-parametric methods for representing the robots' beliefs and likely observations to enable distributed inference and coordination.
(iii) We present a novel consensus-based algorithm for approximating the robots' joint measurement probabilities and prove that the approximations converge to the true values as the number of the consensus rounds grows or as the network becomes complete.
(iv) We implement the approach using five quadrotor robots, and provide the results of experiments in both indoor and outdoor environments. We also give the results of numerical simulations using 100 robots.

1.1. *Our preliminary works*

This paper is a culmination of two preliminary works (Julian et al., 2011, 2012a), yet it distinguishes itself from these previous publications as follows.

(i) We present a more rigorous formulation of the multi-robot information acquisition problem, explicitly stating key assumptions and illustrating critical aspects of the system.
(ii) We provide an extended literature search to guide the reader along the sequence of prior works that have inspired us. As a result, our discussions emphasize connecting these existing methods and our approach.
(iii) We have developed a fully decentralized system and have carried out new experiments. We also discuss in detail the implementation aspects of these experiments to assist the readers in replicating our results.

1.2. *Prior works*

Bayesian approaches for estimation have a rich history in robotics, and mutual information has recently emerged as a powerful tool for controlling robots to improve the quality of the Bayesian estimation, particularly in multi-robot systems. The early work of Cameron and Durrant-Whyte (1990) used mutual information as a metric for sensor placement without explicitly considering mobility of the sensors. Later Grocholsky et al. proposed controlling multiple robot platforms near an object of interest so as to increase mutual information in tracking applications (Grocholsky, 2002; Grocholsky et al., 2003). Bourgault et al. (2002) also used a similar method for exploring and mapping uncertain environments. In addition, the difficult problem of planning paths through an environment to optimize mutual information has been investigated recently (Singh et al., 2007; Ny and Pappas, 2009; Choi and How, 2010). Even though the use of mutual information to formulate robot controllers follows a long lineage of information theoretic approaches, our work in collaboration with Schwager et al. demonstrates the first results on using the analytically derived expression for the gradient of mutual information (Julian et al. 2011; Schwager et al. 2011a).

In a multi-robot context, the main challenges in using mutual information as a control metric are computational complexity and network communication constraints. The complexity of computing mutual information and its gradient is exponential with respect to the number of robots, and thus is intractable in realistic applications using a large multi-robot team. Furthermore, the computation of mutual information requires that every robot has current knowledge of every other robot's position and sensor measurements. Thus, many of the prior mutual information methods are restricted to small groups of robots with all-to-all communication infrastructure. To relax this communication requirement, Olfati-Saber et al. (2005) developed a consensus algorithm for Bayesian distributed hypothesis testing in a static sensor network. In our work, we use a consensus-based algorithm inspired by Olfati-Saber et al. (2005) to compute the joint measurement probabilities needed for the mutual information-based controller and the Bayesian filter calculations. This consensus approach is similar to the use of hyperparameters (Fraser et al., 2012); however, our approach enables the early termination of the consensus-based algorithm without the risk of 'double-counting' any single observation, even when the maximum in/out degree and the number of robots are unknown. We then overcome the problem of scalability by judiciously sampling from the complete set of joint measurement probabilities to compute an approximate mutual information gradient.

In addition to affecting their sensing, the positioning of the robots influences the communication properties of the system. Hence, Krause et al. (2008) formalized the task of balancing the informativeness of sensor placement with the need to communicate efficiently. This task was shown to be an 'NP-hard tradeoff', which motivated the development of a polynomial-time, data-driven approximate algorithm for choosing sensor positions. Closely related is the work of Zavlanos and Pappas (2008), which describes a connectivity controller that enables the robots to remove communication links with network neighbors while maintaining global connectedness. The controller uses local knowledge of the network to estimate its topology, rendering the algorithm distributed among robots. Although in our work we analyze the performance of our algorithms assuming global connectedness (but not completeness), we note that our approach naturally accommodates the splitting and merging of network subgraphs by correctly fusing the beliefs of the involved robots. This property also facilitates the use of communication schemes that lack the notion of active connectivity maintenance. For example, we have previously presented how state estimates can be efficiently exchanged within a robot team by broadcasting estimates of nearby robots more frequently than distant ones (Julian et al., 2009).

Concerning state estimation of an environment, the body of work addressing Bayesian estimation methods based on a family of Kalman filters, which are Bayesian filters for Gaussian systems, have commonly been used. For example, Lynch et al. (2008) proposed a distributed Kalman filtering approach in which the robots use consensus algorithms to share information while controlling their positions to decrease the error variance of the state estimate. In addition, Cortés (2009) developed a distributed filtering algorithm based on the Kalman filter for estimating environmental fields. The algorithm also estimated the gradient of the field, which is then used for multi-robot control. There have been similar Kalman filter approaches for tracking multiple targets, such as that of Chung et al. (2004).

The use of non-parametric filters have become popular in robotics as the platforms become more computationally capable. In an early work, Engelson and McDermott (1992) used a sequential Monte Carlo method to construct a mapper robust enough to address the kidnapped robot problem. Since then, non-parametric algorithms have become commonplace in localization (Borenstein et al., 1997), simultaneous localization and mapping (SLAM) (Montemerlo et al., 2002), and target tracking (Schulz et al., 2001). Fox et al. (2000) applied these algorithms to multiple collaborating robots using a sample-based version of Markov localization. Similar to our work are the recent efforts of Hoffmann and Tomlin (2010), which proposed a sequential Monte Carlo method to propagate a Bayesian estimate, then used greedy and pair-wise approximations to calculate mutual information. In addition, belief propagation (Pearl, 1988) has seen non-parametric extensions (Ihler et al., 2005), which use Gaussian mixtures to solve graphical inference problems.

Although we do not explicitly discuss robot localization (i.e. each robot inferring its own configuration within a global frame) and map alignment, the problem of distributed inference is closely related to multi-robot SLAM. Most relevant is the work of Leung et al. (2012), which proposed a decentralized SLAM approach able to obtain 'centralized-equivalent' solutions on non-complete communication graphs. This approach is sufficiently generic to employ a wide range of Bayesian filtering methods, contrary to prior works specifically using extended Kalman filters (EKFs) (Nettleton et al., 2000), sparse extended information filters (SEIFs) (Thrun and Liu, 2005), or particle filters (Howard, 2006). For our work, incorporating the robots' configurations and map offsets into the robots' beliefs has direct benefits for multi-robot SLAM exploration: a domain we are currently investigating. For a more in-depth discussion concerning SLAM and multi-robot SLAM, please see Thrun et al. (2005).

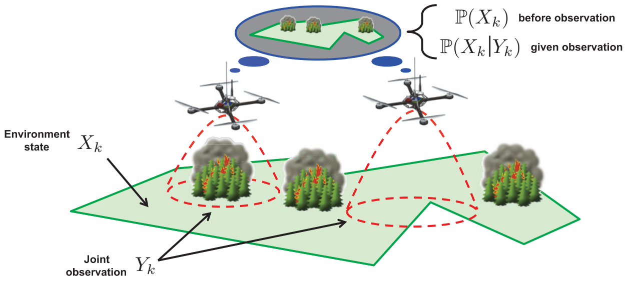
$\mathbb{P}(X_k)$ before observation
$\mathbb{P}(X_k|Y_k)$ given observation
Environment state X_k
Joint observation Y_k

Fig. 2. The representation of a robot system within a probabilistic framework. The robots observe the state of the environment using sensors of finite footprints. The joint measurement probabilities describe the accuracy of the continuous-time joint observations, which through Bayes' rule provide the relationship between the system's prior, $\mathbb{P}(X_k)$, and posterior, $\mathbb{P}(X_k|Y_k)$, distributions.

1.3. Paper outline

This paper is organized as follows. We formalize the multi-robot inference and coordination problem in Section 2, then derive the gradient-based controller with non-parametric extensions in Section 3. In Section 4, we present the communication model and the consensus-based algorithm used to distributively approximate the joint measurement probabilities. In Section 5, we discuss the results from our two hardware experiments and one numerical simulation, then conclude the paper in Section 6. Proofs for all theorems and descriptions for all notation are found in Appendices A and B, respectively.

2. Problem formulation

We motivate our approach with an information-theoretic justification of a utility function, then develop the problem formally for a single robot followed by the centralized multi-robot case with an ideal network.

2.1. Information and utility

We wish to infer the state of an environment from measurements obtained by a number of robots equipped with sensors (see Figure 2). Ideally, we would represent the potentially time-varying state in a continuous manner. However, the robots' inference calculations happen at discrete times, and for this paper we assume that all robots perform these calculations synchronously at a constant rate of $1/T_s$. Thus, at time $t = kT_s$, where k denotes the discrete time step, we model the environment state as a discrete-time random variable, X_k, that takes values from an alphabet, $\mathcal{X}$.

Our goal is to enable the inference calculations necessary for collectively estimating the environment state and reducing uncertainty in the system. Each robot forms an observation from sensor measurements influenced by noise and other effects. We consider the observations of all robots together as a single joint observation, which we model as a discrete-time random variable, Y_k, that takes values from an alphabet, $\mathcal{Y}$. The relationship between the true state and the noisy observation is described by joint measurement probabilities, $\mathbb{P}(Y_k|X_k)$. Sensing may be interpreted as using a noisy channel, and since the sensors are attached to the robots, the joint measurement probabilities are dependent on the position of the robots and the orientation of their sensors. From Bayes' rule, we can use a joint observation and the system's prior distribution, $\mathbb{P}(X_k)$, to compute the system's posterior distribution,

$$\mathbb{P}(X_k|Y_k) = \frac{\mathbb{P}(X_k)\,\mathbb{P}(Y_k|X_k)}{\int_{x\in\mathcal{X}} \mathbb{P}(X_k = x)\,\mathbb{P}(Y_k|X_k = x)\,dx}. \tag{1}$$

Since our objective is to best infer the environment state, we are motivated to move the robots and their sensors into a configuration that minimizes the expected uncertainty of the inference after receiving the next joint observation. Our optimization objective is equivalent to minimizing conditional entropy,

$$H(X_k|Y_k) = H(X_k) - I(X_k, Y_k),$$

where $H(X_k)$ is the entropy of the environment state and $I(X_k, Y_k)$ is the mutual information between the environment state and the joint observation. Since the entropy of the environment state before receiving an observation is independent of the configuration of the robots, minimizing the conditional entropy is equivalent to maximizing mutual information. Hence, we define the utility function for the system to be

$$\begin{aligned} U_k &:= I(X_k, Y_k) \\ &= \int_{y\in\mathcal{Y}} \int_{x\in\mathcal{X}} \mathbb{P}(Y_k = y|X_k = x)\,\mathbb{P}(X_k = x) \\ &\quad \times \log\left(\frac{\mathbb{P}(X_k = x|Y_k = y)}{\mathbb{P}(X_k = x)}\right) dx\,dy, \end{aligned} \tag{2}$$

where $\log(\cdot)$ represents the natural logarithm. Figure 3 illustrates the concept of mutual information and how it relates to the movement of the robots. We are particularly interested in a class of controllers that use a gradient ascent approach with respect to the utility function (2), leading to the following theorem.

Theorem 1 (Gradient of mutual information)**.** *The gradient of the utility function* (2) *with respect to a single robot's configuration,* $c_t^{[i]}$*, at continuous time* $t \in [kT_s, (k+1)T_s)$ *is given by*

$$\begin{aligned} \frac{\partial U_k}{\partial c_t^{[i]}} &= \int_{y\in\mathcal{Y}} \int_{x\in\mathcal{X}} \frac{\partial \mathbb{P}(Y_k = y|X_k = x)}{\partial c_t^{[i]}} \mathbb{P}(X_k = x) \\ &\quad \times \log\left(\frac{\mathbb{P}(X_k = x|Y_k = y)}{\mathbb{P}(X_k = x)}\right) dx\,dy. \end{aligned} \tag{3}$$

Proof. Please refer to Appendix A for a proof. □

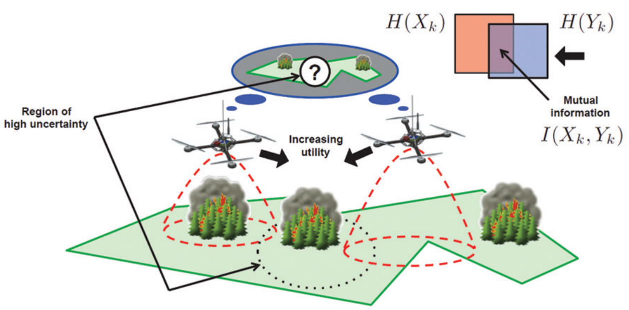

Fig. 3. Robots moving their sensors' field of view towards a region of the environment that corresponds to high uncertainty with respect to the inference. The movement happens in a direction of increasing mutual information (i.e. utility) since this direction corresponds to a decrease in expected uncertainty of the inference given the next joint observation. Mutual information can also be visualized as the overlap (i.e. relevance) of the entropy of the joint observation, $H(Y_k)$, and the entropy of the environment state, $H(X_k)$.

2.2. *Single-robot case*

Consider a single robot, denoted i, that has some belief of the environment state, which is represented by its prior distribution, $\mathbb{P}^{[i]}(X_k)$. We model the corresponding robot's observation as a random variable, $Y_k^{[i]}$, that takes values from an alphabet, $\mathcal{Y}^{[i]}$, and is characterized by sensor measurement probabilities, $\mathbb{P}(Y_k^{[i]}|X_k)$. We assume that the measurement probabilities are known a priori by the robot or, in other words, the robot's sensors are calibrated. Note that included in the calibration is a mapping of how these probabilities change as the robot moves or as its sensors are reoriented.

From a received observation, $y_k^{[i]} \in \mathcal{Y}^{[i]}$, the robot is able to compute from (1) its posterior distribution,

$$\mathbb{P}(X_k|Y_k^{[i]}) = \frac{\mathbb{P}^{[i]}(X_k)\,\mathbb{P}(Y_k^{[i]}|X_k)}{\int_{x\in\mathcal{X}} \mathbb{P}^{[i]}(X_k = x)\,\mathbb{P}(Y_k^{[i]}|X_k = x)\; dx}, \quad (4)$$

which is used in conjunction with its state transition distribution, $\mathbb{P}^{[i]}(X_{k+1}|X_k)$, to form at time $t = (k+1)\,T_s$ its new prior distribution,

$$\mathbb{P}^{[i]}(X_{k+1}) = \frac{\int_{x\in\mathcal{X}} \mathbb{P}^{[i]}(X_{k+1}|X_k{=}x)\mathbb{P}(X_k{=}x|Y_k^{[i]}{=}y_k^{[i]})}{\int_{x'\in\mathcal{X}}\int_{x\in\mathcal{X}} \mathbb{P}^{[i]}(X_{k+1}{=}x'|X_k{=}x)\mathbb{P}(X_k{=}x|Y_k^{[i]}{=}y_k^{[i]})\, dx\, dx'}, \quad (5)$$

Equations (4) and (5) form the well-known duet of prediction and update in sequential Bayesian estimation.

2.3. *Centralized multi-robot case with an ideal network*

Given a centralized system with an ideal network (i.e. complete with infinite bandwidth and no latency), the multi-robot case with n_r robots is a simple extension of the single robot case with a common prior, $\mathbb{P}^{[i]}(X_k) = \mathbb{P}(X_k)$, and state transition distribution, $\mathbb{P}^{[i]}(X_{k+1}|X_k) = \mathbb{P}(X_{k+1}|X_k)$, for all robots $i \in \{1, \ldots, n_r\}$. Let the system be synchronous in that the robots' observations are simultaneously received at a sampling rate of $1/T_s$. We model the joint observation as an n_r-tuple random variable, $Y_k = (Y_k^{[1]}, \ldots, Y_k^{[n_r]})$, that takes values from the Cartesian product of all of the robots' observation alphabets, $\mathcal{Y} = \prod_{i=1}^{n_r} \mathcal{Y}^{[i]}$.

We assume that the noise on the observations are uncorrelated between robots or, in other words, that the robots' observations are conditionally independent. This assumption gives joint measurement probabilities of

$$\mathbb{P}(Y_k|X_k) = \prod_{i=1}^{n_r} \mathbb{P}(Y_k^{[i]}|X_k). \quad (6)$$

Since the sensors of any two robots are physically detached from each other, we can expect that correlated noise is the result of environmental influences. The more these influences are accounted for within the environment state, the more accurate the assumption of conditional independence becomes.

Thus, the posterior distribution prediction from Equation (4) becomes

$$\mathbb{P}(X_k|Y_k) = \frac{\mathbb{P}(X_k)\prod_{i=1}^{n_r}\mathbb{P}(Y_k^{[i]}|X_k)}{\int_{x\in\mathcal{X}} \mathbb{P}(X_k = x)\prod_{i=1}^{n_r}\mathbb{P}(Y_k^{[i]}|X_k = x)\; dx}, \quad (7)$$

and the prior distribution update from Equation (5) becomes

$$\mathbb{P}(X_{k+1}) = \frac{\int_{x\in\mathcal{X}} \mathbb{P}(X_{k+1}|X_k = x)\,\mathbb{P}(X_k = x|Y_k = y_k)}{\int_{x'\in\mathcal{X}}\int_{x\in\mathcal{X}} \mathbb{P}(X_{k+1} = x'|X_k = x)\,\mathbb{P}(X_k = x|Y_k = y_k)\; dx\, dx'},$$

where $y_k = (y_k^{[1]}, \ldots, y_k^{[n_r]}) \in \mathcal{Y}$ is the value of the received joint observation. Note that there is one common prior and posterior distribution for the centralized system, as illustrated in Figure 4. For the decentralized system, we commonly use the notation $\mathbb{P}^{[i]}$ to represent distributions for a particular robot. The notable exception concerns the robot measurement probabilities, $\mathbb{P}(Y_k^{[i]}|X_k)$, where writing $\mathbb{P}^{[i]}(Y_k^{[i]}|X_k)$ is somewhat redundant and thus the extra superscript is omitted for clarity.

3. Distributed inference and coordination

We begin by presenting the high-level architecture of the distributed inference and coordination algorithm in Algorithm 1 accompanied by an example timeline in Figure 5, then discuss the low-level technical details of its implementation.

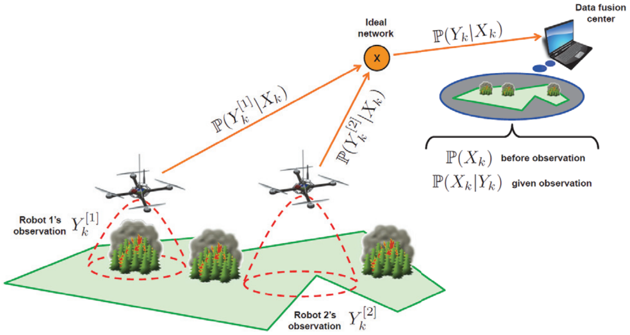
Ideal network
Data fusion center
$\mathbb{P}(Y_k|X_k)$
$\mathbb{P}(Y_k^{[1]}|X_k)$
$\mathbb{P}(Y_k^{[2]}|X_k)$
$\mathbb{P}(X_k)$ before observation
$\mathbb{P}(X_k|Y_k)$ given observation
Robot 1's observation $Y_k^{[1]}$
Robot 2's observation $Y_k^{[2]}$

Fig. 4. A centralized multi-robot system with an ideal network. The robots synchronously receive an observation then transmit the corresponding measurement probabilities over the network to a data fusion center. The joint measurement probabilities are then used with the system's prior distribution to form the posterior distribution.

Algorithm 1 Distributed_Inference_and_Coordination()

```
Require: The ith robot knows its configuration, its measurement probabilities, and the extent of the environment.
Initialize the weighted environment state sample set from Section 3.2.
loop
    Distributively approximate the sampled joint measurement probabilities using Belief_Consensus(sampled) from Section 4.3.
    Apply controller from Corollary 1 in Section 3.3.
    Distributively approximate the joint measurement probabilities using Belief_Consensus(observed) from Section 4.3.
    Update weighted environment state sample set using Sequential_Importance_Resampling( ) from Section 3.4.
end loop
```

3.1. Gradient-based control

Let the ith robot of configuration $c_t^{[i]}$ move in a configuration space, $\mathcal{C}^{[i]} \subset \mathbb{R}^{r_c^{[i]}} \times \mathbb{S}^{s_c^{[i]}}$, where $\mathbb{R}^{r_c^{[i]}}$ and $\mathbb{S}^{s_c^{[i]}}$ represent the $r_c^{[i]}$-dimensional Euclidean space and the $s_c^{[i]}$-dimensional sphere, respectively. This space describes both the position of the robot and the orientation of its sensors, and does not need to be the same space as the environment, denoted $\mathcal{Q} \subset \mathbb{R}^{r_q} \times \mathbb{S}^{s_q}$. For example, if we have a planar environment within $\mathbb{R}^2$, we could have a ground robot with an omnidirectional sensor moving in $\mathbb{R}^2$ or a flying robot with a gimballed sensor moving in $\mathbb{R}^3 \times \mathbb{S}^3$. The Cartesian product of the configuration spaces, $\mathcal{C} = \prod_{i=1}^{n_r} \mathcal{C}^{[i]}$, represents the configuration space for the system of robots, from which the n_r-tuple $c_t = (c_t^{[1]}, \ldots, c_t^{[n_r]})$ denotes the system's configuration at continuous time $t \geq 0$.

At any given time, the robot can choose a control action, $u_t^{[i]}$, from a control space, $\mathcal{U}_t^{[i]} \subset \mathbb{R}^{r_c^{[i]}} \times \mathbb{S}^{s_c^{[i]}}$. We model the robot as having continuous-time integrator dynamics,

$$\frac{dc_t^{[i]}}{dt} = u_t^{[i]}, \tag{8}$$

which is a common assumption in the multi-robot coordination literature Schwager et al. (2011b). In our applications using the quadrotor flying robot platform, we found that generating position commands at a relatively slow rate (e.g., 1 Hz) and feeding these inputs into a relatively fast (e.g., 40 Hz) low-level position controller sufficiently approximates the integrator dynamics assumption (8).

Consider the control objective from Section 2.1. We wish to move the robot system into a configuration that minimizes the expected uncertainty of the environment state after receiving the next joint observation. With respect to the utility function (2), our objective is equivalent to solving the constrained optimization problem $\max_{c \in \mathcal{C}} U_k$. One solution approach is to have the each robot calculate from Equation (3) the partial derivative of the utility function with respect to that robot's configuration,

$$\begin{aligned} \frac{\partial U_k}{\partial c_t^{[i]}} = & \int_{y \in \mathcal{Y}} \int_{x \in \mathcal{X}} \frac{\partial \mathbb{P}(\, Y_k^{[i]} = y^{[i]} | X_k = x)}{\partial c_t^{[i]}} \\ & \times \mathbb{P}(X_k = x) \prod_{i' \neq i} \mathbb{P}(\, Y_k^{[i']} = y^{[i']} | X_k = x) \\ & \times \log \left(\frac{\mathbb{P}(X_k = x | Y_k = y)}{\mathbb{P}(X_k = x)} \right) dx\, dy, \end{aligned} \tag{9}$$

then continuously move in a valid direction of increasing utility. Note that receiving an observation may induce instantaneous discontinuities in the utility gradient even if the gradient is continuous on the configuration space for every time step. Owing to this property, we use the phrase 'convergent between observations' to describe the limit of a state assuming that no further Bayesian filter updates are performed. We considered this to be a useful property since it allows the robots to improve their configuration based on the information at hand (i.e. prior to receiving the next observation). The following theorem describes a gradient-based controller that is convergent between observations and, in its most general form, is locally optimal.

Theorem 2 (Convergence and local optimality)**.** *Let robots having dynamics* (8) *move in the same configuration space and sense a bounded environment that is a subset of the configuration space. Consider the class of systems where for all robots, the change in measurement probabilities with respect to the robot's configuration is continuous on the robot's configuration space and equal to zero for all configurations greater than a certain distance away from the environment (e.g. sensors of limited range). Then for a positive scalar $\gamma^{[i]}$, the controller*

$$u_t^{[i]} = \gamma^{[i]} \frac{\partial U_k}{\partial c_t^{[i]}} \tag{10}$$

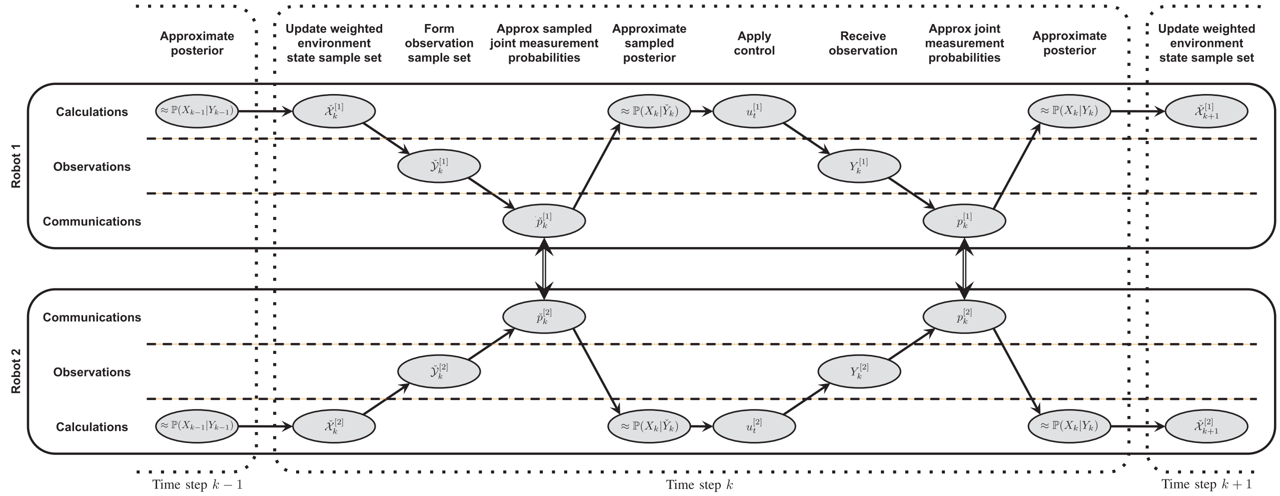
Approximate posterior
Update weighted environment state sample set
Form observation sample set
Approx sampled joint measurement probabilities
Approximate sampled posterior
Apply control
Receive observation
Approx joint measurement probabilities
Approximate posterior
Update weighted environment state sample set
Robot 1
Calculations
Observations
Communications
Robot 2
Communications
Observations
Calculations
$\approx \mathbb{P}(X_{k-1}|Y_{k-1})$
$\check{\mathcal{X}}_k^{[1]}$
$\check{\mathcal{Y}}_k^{[1]}$
$\check{p}_k^{[1]}$
$\approx \mathbb{P}(X_k|\check{Y}_k)$
$u_t^{[1]}$
$Y_k^{[1]}$
$p_k^{[1]}$
$\approx \mathbb{P}(X_k|Y_k)$
$\check{\mathcal{X}}_{k+1}^{[1]}$
$\check{p}_k^{[2]}$
$\check{\mathcal{Y}}_k^{[2]}$
$\approx \mathbb{P}(X_{k-1}|Y_{k-1})$
$\check{\mathcal{X}}_k^{[2]}$
$\approx \mathbb{P}(X_k|\check{Y}_k)$
$u_t^{[2]}$
$Y_k^{[2]}$
$p_k^{[2]}$
$\approx \mathbb{P}(X_k|Y_k)$
$\check{\mathcal{X}}_{k+1}^{[2]}$
Time step $k-1$
Time step k
Time step $k+1$

Fig. 5. This timeline shows the order of events for a two-robot distributed system. Starting with a weighted environment state sample set, each robot approximates the sampled joint measurement probabilities, applies its control, receives an observation, approximates the joint measurement probabilities, then forms the weighted environment state sample set for the next time step.

is convergent to zero between observations for all robots. In addition, an equilibrium system configuration, $c_* = (c_*^{[1]}, \ldots, c_*^{[n_r]})$, defined by

$$\left.\frac{\partial U_k}{\partial c_t^{[i]}}\right|_{c_t^{[i]}=c_*^{[i]}} = 0, \quad \forall i \in \{1, \ldots, n_r\}$$

is Lyapunov stable if and only if it is locally optimal with respect to maximizing the utility function (2).

Proof. Please refer to Appendix A for a proof. □

Remark 1 (Required local and global knowledge). For all controllers presented in this paper, we assume that each robot has knowledge of: (i) its configuration, $c_t^{[i]}$; (ii) its measurement probabilities, $\mathbb{P}(Y_k^{[i]}|X_k)$; and (iii) the extent of the environment, $\mathcal{Q}$. However, the gradient-based controller (10) also requires that each robot has global knowledge of (i) the system's prior, $\mathbb{P}(X_k)$; (ii) the system's configuration, c_t; and (iii) the joint measurement probabilities, $\mathbb{P}(Y_k|X_k)$. Thus, the controller is not distributed among the robots.

Remark 2 (Intractability of the general form). Consider both X_k and Y_k to be discrete-valued random variables, whose alphabet are sizes of $|\mathcal{X}|$ and at most $\max_i |\mathcal{Y}^{[i]}|^{n_r}$, respectively. To calculate and store all possible instantiations from the posterior calculation (7), the robots require $\max_i O(n_r|\mathcal{X}||\mathcal{Y}^{[i]}|^{n_r})$ time and $\max_i O(|\mathcal{X}||\mathcal{Y}^{[i]}|^{n_r})$ memory. Finally, the utility gradient (9) requires an additional $\max_i O(n_r|\mathcal{X}||\mathcal{Y}^{[i]}|^{n_r})$ time. Since the complexity of the gradient-based controller (10) is exponential with respect to number of robots, n_r, it is not scalable.

3.2. Non-parametric implementation

Let each robot maintain a weighted environment state sample set,

$$\check{\mathcal{X}}_k^{[i]} = \left\{(\check{x}_k^{[i,j]}, \check{w}_k^{[i,j]}) : j \in \{1, \ldots, n_x\}\right\},$$

of size n_x, where each sample, $\check{x}_k^{[i,j]} \in \mathcal{X}$, has a corresponding weight,[2] $\check{w}_k^{[i,j]} \in (0, 1]$. Each sample is a candidate instantiation of the environment state, and the pairing of the samples and their corresponding weights represents a non-parametric approximation of the robot's belief of the environment state.

Using this set, samples of likely observations for each robot are formed. Let each robot create a temporary unweighted environment state sample set by drawing n_y samples from the weighted sample set with probabilities proportional to the corresponding weights. Note that the drawn samples represent equally likely state instantiations (they are formed in a method analogous to the importance sampling step for particle filters (Thrun et al., 2005)). A robot's observation sample set,

$$\check{\mathcal{Y}}_k^{[i]} = \left\{\check{y}_k^{[i,\ell]} : \ell \in \{1, \ldots, n_y\}\right\},$$

is then formed by drawing one observation sample for each entry in the temporary state sample set using the robot's measurement probabilities. The corresponding sampled measurement probabilities become

$$\mathbb{P}(\check{Y}_k^{[i]}|X_k) = \frac{\mathbb{P}(Y_k^{[i]}|X_k)}{\sum_{\ell=1}^{n_y} \mathbb{P}(Y_k^{[i]} = \check{y}_k^{[i,\ell]}|X_k)},$$

where $\check{Y}_k^{[i]}$ is a random variable that takes values from $\check{\mathcal{Y}}_k^{[i]}$. The sampling methodology is illustrated in Figure 6.

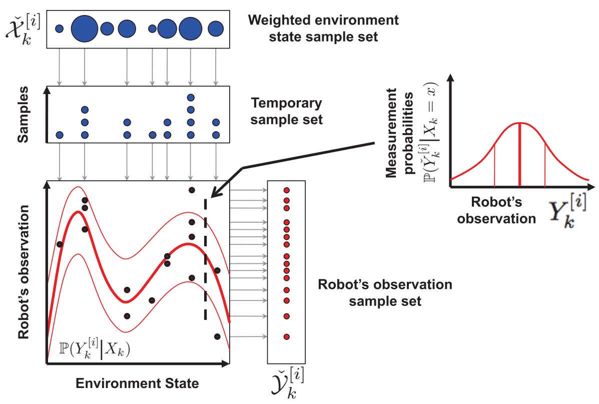
$\check{\mathcal{X}}_k^{[i]}$
Weighted environment state sample set
Samples
Temporary sample set
Measurement probabilities $\mathbb{P}(Y_k^{[i]}|X_k = x)$
Robot's observation $Y_k^{[i]}$
Robot's observation
Robot's observation sample set
$\mathbb{P}(Y_k^{[i]}|X_k)$
Environment State
$\check{\mathcal{Y}}_k^{[i]}$

Fig. 6. The sampling methodology for creating the robot's observation sample set. Samples are drawn from the weighted environment state sample set to form a temporary unweighted sample set. Using this set, samples of likely local observations are then drawn from the measurement probabilities, which will be used to form the joint observation sample set. Note that the measurement probabilities do not need to be Gaussian.

We then define the joint observation sample set, $\check{\mathcal{Y}}_k$, as the unweighted set of n_r-tuples formed from the robots' observation samples having equal indices. More formally, we have that

$$\check{\mathcal{Y}}_k = \left\{ \check{y}_k^{[\ell]} = (\check{y}_k^{[1,\ell]}, \ldots, \check{y}_k^{[n_r,\ell]}) : \check{y}_k^{[i,\ell]} \in \check{\mathcal{Y}}_k^{[i]} \right\}.$$

Note that a generic formulation of a joint observation sample set would be the Cartesian product of all of the robots' observation sample sets, $\prod_{i=1}^{n_r} \check{\mathcal{Y}}_k^{[i]}$, which in size scales exponentially with respect to the number of robots. Here we use the fact that a robot's observation sample set is both unweighted (i.e. all samples are equally likely) and conditionally independent to form an unbiased joint observation sample set of constant size with respect to the number of robots. In other words, each robot independently draws its own observation samples using its local measurement probabilities, and due to conditional independence the concatenation of these samples across all robots is equivalent to a sample set formed by using the system's joint measurement probabilities.

3.3. Distributed controller

We show in Section 4.3 that by using a consensus-based algorithm, each robot can distributively approximate the sampled joint measurement probabilities, $\mathbb{P}(\check{Y}_k|X_k)$, where $\check{Y}_k$ is a random variable that takes values from $\check{\mathcal{Y}}_k$. Let the matrix $\check{p}_k^{[i]}$ denote these approximations, where an approximation to the posterior calculation (7) becomes

$$\mathbb{P}(X_k = \check{x}_k^{[i,j]} | \check{Y}_k = \check{y}_k^{[\ell]}) \approx \frac{\check{w}_k^{[i,j]} [\check{p}_k^{[i]}]_{j\ell}}{\sum_{j'=1}^{n_x} \check{w}_k^{[i,j']} [\check{p}_k^{[i]}]_{j'\ell}}$$

for all $j \in \{1, \ldots, n_x\}$ and $\ell \in \{1, \ldots, n_y\}$, with $[\cdot]_{j\ell}$ denoting the matrix entry (j, ℓ).

By incorporating the weighted environment state sample set, the joint observation sample set, and the sampled joint measurement probability approximations into Equation (2), we define

$$\begin{aligned} \check{U}_k^{[i]} &:= \sum_{\ell=1}^{n_y} \sum_{j=1}^{n_x} \mathbb{P}(\check{Y}_k^{[i]} = \check{y}_k^{[i,\ell]} | X_k = \check{x}_k^{[i,j]}) \\ &\times \frac{\check{w}_k^{[i,j]} [\check{p}_k^{[i]}]_{j\ell}}{\mathbb{P}(\check{Y}_k^{[i]} = \check{y}_k^{[i,\ell]} | X_k = \check{x}_k^{[i,j]})} \Bigg|_{t=kT_s} \\ &\times \log \left(\frac{[\check{p}_k^{[i]}]_{j\ell}}{\sum_{j'=1}^{n_x} \check{w}_k^{[i,j']} [\check{p}_k^{[i]}]_{j'\ell}} \right) \end{aligned} \qquad (11)$$

to be the ith robot's approximation of the utility function given its measurement probabilities at time $t = kT_s$. Taking the partial derivative of Equation (11) with respect to the robot's configuration, we have that

$$\begin{aligned} \frac{\partial \check{U}_k^{[i]}}{\partial c_t^{[i]}} &= \sum_{\ell=1}^{n_y} \sum_{j=1}^{n_x} \frac{\partial \mathbb{P}(\check{Y}_k^{[i]} = \check{y}_k^{[i,\ell]} | X_k = \check{x}_k^{[i,j]})}{\partial c_t^{[i]}} \\ &\times \frac{\check{w}_k^{[i,j]} [\check{p}_k^{[i]}]_{j\ell}}{\mathbb{P}(\check{Y}_k^{[i]} = \check{y}_k^{[i,\ell]} | X_k = \check{x}_k^{[i,j]})} \Bigg|_{t=kT_s} \\ &\times \log \left(\frac{[\check{p}_k^{[i]}]_{j\ell}}{\sum_{j'=1}^{n_x} \check{w}_k^{[i,j']} [\check{p}_k^{[i]}]_{j'\ell}} \right), \end{aligned} \qquad (12)$$

which is a distributed approximation of the gradient of mutual information (3). Multiplying this result by the positive scalar control gain $\gamma^{[i]}$ results in a gradient-based controller that is distributed among the robots and uses the non-parametric representation of the robots' beliefs. Note that the dependency on the approximated joint measurement probabilities or the environment state sample weights does not preclude the use of LaSalle's invariance principle to prove convergence, leading to the following.

Corollary 1 (Convergence of gradient-based controller)**.** *With the assumptions stated in Theorem 2, the gradient-based controller*

$$u_t^{[i]} = \gamma^{[i]} \frac{\partial \check{U}_k^{[i]}}{\partial c_t^{[i]}} \qquad (13)$$

is convergent to zero for all robots between the consensus updates of the joint measurement probability approximations.

Proof. The proof directly follows the convergence proof for Theorem 2, using the Lypanov-type function candidate

$$V_k = -\sum_{i=1}^{n_r} \check{U}_k^{[i]}.$$

□

Remark 3 (Distributed among robots). Compared with its general form (10), we note that the gradient-based controller (13) does not require that each robot has global knowledge of (i) the system's prior, $\mathbb{P}(X_k)$; (ii) the system's configuration, c_t; and (iii) the joint measurement probabilities, $\mathbb{P}(Y_k|X_k)$. Thus, the controller is distributed among the robots.

Remark 4 (Loss of local optimality). Since the distributed controller (13) incorporates approximations for the robot priors and the joint measurement probabilities, an equilibrium system configuration, $c_* = (c_*^{[1]}, \ldots, c_*^{[n_r]})$, defined by

$$\left.\frac{\partial \check{U}_k^{[i]}}{\partial c_t^{[i]}}\right|_{c_t^{[i]}=c_*^{[i]}} = 0, \quad \forall i \in \{1, \ldots, n_r\}$$

is not guaranteed to be locally optimal with respect to maximizing the utility function (2).

Remark 5 (Computational tractability). The utility gradient approximation (12) requires $O(n_x n_y)$ time and $O(n_y)$ memory, where the memory requirement is due to precalculating the summation in the logarithm function for all joint observation samples. Hence, the distributed controller (13) scales linearly with respect to the sizes of the environment state and joint observation sample sets. Moreover, computational complexity remains constant with respect to the number of robots.

Remark 6 (Details of the measurement probabilities). For the approximate gradient of mutual information (12) to be well defined, we must have $\mathbb{P}(\check{Y}_k^{[i]}|X_k) \in (0, 1)$ for all $x \in \mathcal{X}$, $y^{[i]} \in \mathcal{Y}^{[i]}$, and $c_t^{[i]} \in \mathcal{C}^{[i]}$. This assumption is equivalent to saying that all robots' observations have some finite, nonzero amount of uncertainty. In addition, note that $\mathbb{P}(\check{Y}_k^{[i]}|X_k)$ in the denominator of (12) is evaluated at time $t = kT_s$ since the terms that make up $\check{p}_k^{[i]}$ are formed at the beginning of the time step (see Figure 5).

3.4. *Sequential importance resampling*

By following the approximate gradient of mutual information, the robots better position themselves for the next joint observation, $y_k \in \mathcal{Y}$. Once received, an approximation for the joint measurement probabilities, $\mathbb{P}(Y_k = y_k|X_k)$, is distributively calculated by again using a consensus-based algorithm. Let the column vector $p_k^{[i]}$ denote these approximations, where the approximation to the posterior calculation (7) now becomes

$$\mathbb{P}(X_k = \check{x}_k^{[i,j]}|Y_k = y_k) \approx \frac{\check{w}_k^{[i,j]}[p_k^{[i]}]_j}{\sum_{j'=1}^{n_x} \check{w}_k^{[i,j']}[p_k^{[i]}]_{j'}} \tag{14}$$

for all $j \in \{1, \ldots, n_x\}$, with $[\cdot]_j$ denoting the jth row entry. Thus, each robot forms its weighted environment state sample set for the upcoming time step $k+1$ by drawing from its state transition distribution, $\mathbb{P}^{[i]}(X_{k+1}|X_k)$, calculating the corresponding weights from Equation (14), and applying an appropriate resampling technique. The process of updating the weighted sample set is a well-known sequential Monte Carlo method called sequential importance resampling (Algorithm 2).

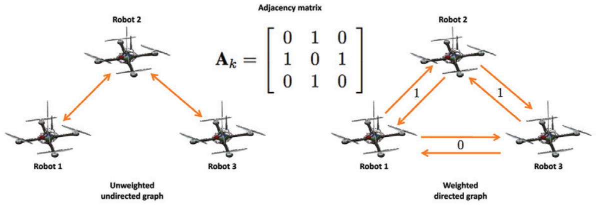
Adjacency matrix
$\mathbf{A}_k = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$
Robot 2
Robot 1
Robot 3
Unweighted undirected graph
Robot 2
1
1
0
Robot 1
Robot 3
Weighted directed graph

Fig. 7. An example three-robot communication graph with a corresponding adjacency matrix. Note that the unweighted undirected graph and the weighted directed graph are equivalent.

Algorithm 2 Sequential_Importance_Resampling()

```
$\check{\mathcal{X}}_{k+1}^{[i]} \leftarrow \emptyset$.
**for** $j = 1$ to $n_x$ **do**
    Sample $\check{x}_{k+1}^{[i,j]} \sim \mathbb{P}^{[i]}(X_{k+1}|X_k = \check{x}_k^{[i,j]})$.
    $\check{w}_{k+1}^{[i,j]} \leftarrow \frac{\check{w}_k^{[i,j]}[p_k^{[i]}]_j}{\sum_{j'=1}^{n_x} \check{w}_k^{[i,j']}[p_k^{[i]}]_{j'}}$.
    $\check{\mathcal{X}}_{k+1}^{[i]} \leftarrow \check{\mathcal{X}}_{k+1}^{[i]} \cup \{(\check{x}_{k+1}^{[i,j]}, \check{w}_{k+1}^{[i,j]})\}$.
**end for**
Apply appropriate resampling technique.
**return** $\check{\mathcal{X}}_{k+1}^{[i]}$.
```

4. Decentralized system

We formalize a communication model and present a consensus-based algorithm that is derived from a general form of averaging consensus algorithms (Olfati-Saber et al., 2005). This algorithm guarantees that all robots' distributed approximations of the joint measurement probabilities converge to the true values when the number of robots, the maximum in/out degree, and the network diameter are unknown. Note that other consensus approaches, including using the network's Metropolis–Hastings weights (Xiao et al., 2007), are also applicable.

4.1. *Communication model*

Between observations, let the robots simultaneously transmit and receive communication data at a much shorter time interval, $T_c \ll T_s$, according to an undirected communication graph, $\mathcal{G}_k$. The pair $\mathcal{G}_k = (\mathcal{V}, \mathcal{E}_k)$ consists of a vertex set, $\mathcal{V} = \{1, \ldots, n_r\}$, and an unordered edge set, $\mathcal{E}_k \subset \mathcal{V} \times \mathcal{V}$. The corresponding symmetric unweighted adjacency matrix, $\mathbf{A}_k \in \{0, 1\}^{n_r \times n_r}$, is of the form

$$[\mathbf{A}_k]_{iv} = \begin{cases} 1, \text{ if } (i, v) \in \mathcal{E}_k \\ 0, \text{ otherwise} \end{cases}.$$

Figure 7 illustrates the adjacency matrix of an example undirected graph and its equivalent weighted directed graph. We also denote $\mathcal{N}_k^{[i]}$ as the set of neighbors of the ith robot, who has an in/out degree of $|\mathcal{N}_k^{[i]}| = \sum_{v=1}^{n_r} [\mathbf{A}_k]_{iv}$.

Given the volatile nature of mobile networks, we expect $\mathcal{G}_k$ to be incomplete, time-varying, and stochastic. The algorithms presented in this paper work in practice even when properties of $\mathcal{G}_k$ cannot be formalized. However, to allow for meaningful analysis from a theoretical perspective, we assume that $\mathcal{G}_k$ remains connected and is time-invariant between observations. Connectivity allows the system to be analyzed as a single unit instead of separate independent subsystems. The property of graph time-invariance between observations is more strict, however, this assumption is used to formalize the convergence of our consensus-based algorithm. Thus, $\mathcal{G}_k$ is modeled as a discrete-time dynamic system with time period of T_s, hence the subscript $\sqcup_k$.

4.2. *Discovery of maximum in/out degree using FloodMax*

The FloodMax algorithm is a well-studied distributed algorithm used in leader elect problems (Lynch, 1997). Traditionally implemented, each robot would transmit the maximum unique identifier (UID) it received up to the given communication round[3]. After diam($\mathcal{G}_k$) communication rounds, where diam($\cdot$) represents the diameter of a graph, all robots would then know the maximum UID in the network. To solve the leader elect problem, the robot whose own UID matches the maximum UID of the network would declare itself the leader.

For distributed inference and coordination, the robots do not need to select a leader, but instead need to discover the network's maximum in/out degree, Δ_k. Moreover, we assume that the robots only know characteristics that describe their local neighborhood (e.g. number of neighbors). In other words, the robots do not know characteristics describing the overall network topology, such as the number of robots and the network diameter. This restriction implies that the robots may never identify that the maximum in/out degree has been found. Regardless, the robots can still reach an agreement during consensus by using in parallel the FloodMax algorithm described in the following.

Lemma 1 (Maximum in/out degree discovery)**.** *Consider the following discrete-time time-invariant dynamical system,*

$$\delta_{k'+1}^{[i]} = \max\left\{\{\delta_{k'}^{[i]}\} \cup \{\delta_{k'}^{[v]} : v \in \mathcal{N}_k^{[i]}\}\right\}, \qquad (15)$$

where for each robot $\delta_{k'}^{[i]}$ is initialized to the robot's number of neighbors plus one. Then for all robots after diam($\mathcal{G}_k$) *communication rounds, $\delta_{k'}^{[i]}$ is equal to the network's maximum in/out degree plus one.*

Proof. The proof is a simple extension of the proof for Theorem 4.1 of Lynch (1997). □

Remark 7 (Shorthand notation of communication round states). We will commonly use the notation $\sqcup_{k'}$ as shorthand for $\sqcup_{k,k'}$ since explicitly including the time step does not contribute any useful information. The values of such states are not carried over between time steps.

4.3. *General consensus in networks of unknown topology*

Consider a system of robots running a discrete-time consensus algorithm (Olfati-Saber et al., 2005) of the form

$$\psi_{k'+1}^{[i]} = \psi_{k'}^{[i]} + \epsilon_k \sum_{v \in N_k^{[i]}} (\psi_{k'}^{[v]} - \psi_{k'}^{[i]}), \qquad (16)$$

where $0 < \epsilon_k < 1/\Delta_k$ guarantees that for all robots, $\psi_{k'}^{[i]}$ exponentially converges to the average initial state of all robots, $\sum_{i=1}^{n_r} \psi_0^{[i]}/n_r$. For the robots to select a valid ϵ_k, they need to know either the maximum in/out degree of the network or the number of robots (since $1/n_r < 1/\Delta_k$). Since we are assuming that neither parameter is known, the consensus algorithm is modified to use in parallel the FloodMax algorithm (15). As a result, convergence to the average initial state is preserved as described by the following, and the process of evolving $\delta_{k'}^{[i]}$, $\psi_{k'}^{[i]}$, and $\pi_{k'}^{[i]}$ in parallel over n_π communication rounds will be summarized in Algorithm 3.

Theorem 3 (Convergence of the consensus algorithm)**.** *Consider the following discrete-time time-invariant dynamical system,*

$$\psi_{k'+1}^{[i]} = \frac{\delta_{k'+1}^{[i]} - \delta_{k'}^{[i]}}{\delta_{k'+1}^{[i]}} \psi_0^{[i]} + \frac{\delta_{k'}^{[i]} - |\mathcal{N}_k^{[i]}|}{\delta_{k'+1}^{[i]}} \psi_{k'}^{[i]} + \frac{1}{\delta_{k'+1}^{[i]}} \sum_{v \in \mathcal{N}_k^{[i]}} \psi_{k'}^{[v]}, \qquad (17)$$

and its exponential form,

$$\pi_{k'+1}^{[i]} = (\pi_0^{[i]})^{\frac{\delta_{k'+1}^{[i]} - \delta_{k'}^{[i]}}{\delta_{k'+1}^{[i]}}} (\pi_{k'}^{[i]})^{\frac{\delta_{k'}^{[i]} - |\mathcal{N}_k^{[i]}|}{\delta_{k'+1}^{[i]}}} \times \prod_{v \in \mathcal{N}_k^{[i]}} (\pi_{k'}^{[v]})^{\frac{1}{\delta_{k'+1}^{[i]}}}. \qquad (18)$$

Then for all robots, $\psi_{k'}^{[i]}$ and $\pi_{k'}^{[i]}$ will converge to $\sum_{v=1}^{n_r} \psi_0^{[v]}/n_r$ and $\prod_{v=1}^{n_r} \left(\pi_0^{[v]}\right)^{1/n_r}$, respectively, in the limit as k' tends to infinity.

Proof. Please refer to Appendix A for a proof. □

Remark 8 (Order of consensus state calculations). We require that $\delta_{k'}^{[i]}$ is available when calculating both $\psi_{k'}^{[i]}$ and $\pi_{k'}^{[i]}$. In words, the FloodMax algorithm update (15) is computed prior to the consensus updates (17) and (18) during a given communication round (see Algorithm 3).

Lemma 2 (Convergence on complete network graphs)**.** *For a complete network graph, $\psi_{k'}^{[i]}$ and $\pi_{k'}^{[i]}$ converge for all robots after one communication round.*

Proof. We have for all robots that $\delta_0^{[i]} = \delta_1^{[i]} = (1 + \Delta_k)$ for a complete network graph. Thus, $\psi_1^{[i]}$ and $\pi_1^{[i]}$ are equal to $\sum_{v=1}^{n_r} \psi_0^{[v]}/n_r$ and $\prod_{v=1}^{n_r} \left(\pi_0^{[v]}\right)^{1/n_r}$, respectively. The proof follows by induction on k'. □

4.4. Consensus of the joint measurement probabilities

In an earlier paper, we showed that the structuring and eventual consensus of the joint measurement probabilities relied on the indices of the elements in the environment state alphabet, and as a result this alphabet was assumed to be of finite size describing a discrete-valued random variable (Julian et al., 2011). This assumption in the previous work was partly motivated by the fact that the robots' beliefs were represented in full, and thus the Bayesian prediction and update calculations required some form of quantization. Here we do not need to make this assumption for the inference as our non-parametric methods naturally account for continuous distributions. However, some predefined discrete mapping of the environment state alphabet is needed for the robots to properly average the joint measurement probabilities over the network using the consensus-based algorithm.

For continuous distributions well approximated by certain parametric statistics, the consensus-based algorithm can be readily used to average distribution parameters instead of the joint measurement probabilities themselves. For example, the approximation of a joint Gaussian distribution converges using only two parameters for the consensus-based algorithm, and the number of parameters used to represent a multivariate distribution scales quadratically with respect to the distribution's dimension (Julian et al., 2012b). We also showed that mixtures of Gaussians can be used for 'arbitrary' continuous distributions. Nonetheless, in this paper we do assume for the sake of simplicity a finite-sized environment state alphabet,

$$\mathcal{X} = \left\{x^{[j]} : j \in \{1, \ldots, |\mathcal{X}|\}\right\},$$

but note that this assumption is not necessary in general.

For consensus of the sampled joint measurement probabilities, let $\pi_{k'}^{[i]}$ be a belief matrix[4] representing the unnormalized approximated n_rth root of these probabilities known by the ith robot after k' communication rounds. In addition, let the belief matrix be initialized as

$$[\pi_0^{[i]}]_{j\ell} = \mathbb{P}(\check{Y}_k^{[i]} = \check{y}_k^{[i,\ell]} | X_k = x^{[j]}),$$

for all $j \in \{1, \ldots, |\mathcal{X}|\}$ and $\ell \in \{1, \ldots, n_y\}$. In words, the belief matrix is initialized to the ith robot's conditionally independent contribution to the unnormalized sampled joint measurement probabilities,

$$\frac{\mathbb{P}(\check{Y}_k | X_k)}{\eta} = \prod_{i=1}^{n_r} \mathbb{P}(\check{Y}_k^{[i]} | X_k), \qquad (19)$$

where η is a normalization factor.

By allowing the belief matrix to evolve using Equation (18), we define an approximation for the sampled joint measurement probabilities,

$$[\check{p}_k^{[i]}]_{j\ell} := \frac{[\pi^{[i]}]_{\xi\ell}^{\beta_k^{[i]}}}{\sum_{\ell'=1}^{n_y} [\pi^{[i]}]_{\xi\ell'}^{\beta_k^{[i]}}} \approx \mathbb{P}(\check{Y}_k = \check{y}_k^{[\ell]} | X_k = \check{x}_k^{[i,j]}), \qquad (20)$$

for all $j \in \{1, \ldots, n_x\}$ and $\ell \in \{1, \ldots, n_y\}$, where ξ is such that $\check{x}_k^{[i,j]} = x^{[\xi]}$, $\pi^{[i]}$ is shorthand denoting $\pi_{k'}^{[i]}$ with $k' = n_\pi$, and $\beta_k^{[i]}$ is an exponential factor accounting for the fact that the consensus-based algorithm may terminate before converging. More specifically, $\pi^{[i]}$ can be thought of as a weighted logarithmic summation of $\mathbb{P}(\check{Y}_k^{[v]} | X_k)$ over all $v \in \{1, \ldots, n_r\}$, and $\beta_k^{[i]}$ is the inverse of the largest weight to ensure that no single measurement probability in the right-hand side product of Equation (19) has an exponent of value larger than one. In other words, no observation 'gets counted' more than once.

To calculate the exponential factor $\beta_k^{[i]}$ in parallel with the belief matrix, let the term $\psi_{k'}^{[i]}$ evolve by using Equation (17) and be initialized to $\mathbf{e}_i$, where $\mathbf{e}_i$ is the standard basis pointing in the ith direction in $\mathbb{R}^{n_r}$. From the discussion above, we have that

$$\beta_k^{[i]} = \|\psi^{[i]}\|_\infty^{-1},$$

where $\psi^{[i]}$ is shorthand denoting $\psi_{k'}^{[i]}$ with $k' = n_\pi$. Note that $\psi_0^{[i]}$ does not need to be of size n_r, which would violate the assumption that the number of robots is unknown. Instead, each robot maintains an indexed vector initially of size one, then augments this vector when unknown indices are received during the consensus round.

From Theorem 3, we have for all robots that $\beta_k^{[i]}$ converges to n_r in the limit as n_π tends to infinity, or after one communication round if the network is complete (Lemma 2). This property is required for the convergence of the approximations to the true sampled joint measurement probabilities, which will be discussed in Theorem 4. Nonetheless, we showed that the robots can use consensus of the belief matrix (as will be summarized in Algorithm 3) to enable the gradient-based distributed controller from Corollary 1, which continuously moves the robots to improve the informativeness of the next joint observation.

Remark 9 (Size of the belief matrix). Owing to the construction of the joint observation sample set, the column size of the belief matrix is reduced from exponential to constant with respect to the number of robots. However, the row dimension is linear with respect to the size of the environment state alphabet, which can be misleading since this alphabet size is usually exponential with respect to other quantities. For example, the size of an alphabet for an environment that is partitioned into cells scales exponentially with respect to the number of cells. Thus, a system designer will need to consider both lossless and lossy compression techniques specific for the application at hand. One such lossless technique is implemented for the experiments and simulations in Section 5.

Once an observation is received, a second consensus round is performed to update the weighted environment state sample set. Let $\pi_{k'}^{[i]}$ now be a belief vector representing the normalized approximated n_rth root of the observed joint measurement probabilities, $\mathbb{P}(Y_k = y_k|X_k)$, known by the ith robot after k' communication rounds. In addition, let the belief vector for all $j \in \{1, \ldots, |\mathcal{X}|\}$ be initialized as

$$[\pi_0^{[i]}]_j = \mathbb{P}(Y_k^{[i]} = y_k^{[i]}|X_k = x^{[j]}) .$$

After n_π communication rounds, the approximation for the observed joint measurement probabilities is given by

$$[p_k^{[i]}]_j := [\pi^{[i]}]_\xi^{\beta_k^{[i]}} \approx \mathbb{P}(Y_k = y_k|X_k = \check{x}_k^{[i,j]}) \qquad (21)$$

for all $j \in \{1, \ldots, n_x\}$. The process of forming the approximation is also summarized in Algorithm 3, and the result is used to update the weighted environment state sample set for the next time step $k + 1$.

Algorithm 3 Belief_Consensus(*type*)

```
$\delta_0^{[i]} \leftarrow \mathbf{e}_i$.
$\psi_0^{[i]} \leftarrow (|\mathcal{N}_k^{[i]}| + 1)$
if type is sampled then
    $[\pi_0^{[i]}]_{j\ell} \leftarrow \mathbb{P}(\check{Y}^{[i]} = \check{y}^{[i,\ell]}|X = x^{[j]}), \quad \forall j, \ell.$
else if type is observed then
    $[\pi_0^{[i]}]_j \leftarrow \mathbb{P}(Y^{[i]} = y_k^{[i]}|X = x^{[j]}), \quad \forall j.$
end if
for $k' = 1$ to $n_\pi$ do
    $\delta_{k'}^{[i]} \leftarrow \max\left\{\{\delta_{k'-1}^{[i]}\} \cup \{\delta_{k'-1}^{[v]} : v \in \mathcal{N}_k^{[i]}\}\right\}$.
    Update $\psi_{k'}^{[i]}$ and $\pi_{k'}^{[i]}$ using (17) and (18), respectively.
end for
$\beta_k^{[i]} \leftarrow \|\psi^{[i]}\|_\infty^{-1}$.
if type is sampled then
    return $\check{p}_k^{[i]}$ from (20)
else if type is observed then
    return $p_k^{[i]}$ from (21)
end if
```

Lastly, we prove that both distributed approximations, $\check{p}_k^{[i]}$ and $p_k^{[i]}$, converge to their corresponding joint measurement probabilities.

Theorem 4. *For all robots, $j \in \{1, \ldots, n_x\}$, and $\ell \in \{1, \ldots, n_y\}$, we have that $[\check{p}_k^{[i]}]_{j\ell}$ and $[p_k^{[i]}]_j$ converge to $\mathbb{P}(\check{Y} = \check{y}^{[\ell]}|X = x^{[\xi]})$ and $\mathbb{P}(Y|X = x^{[\xi]})$, respectively, in the limit as n_π tends to infinity, where ξ is again such that $\check{x}^{[i,j]} = x^{[\xi]}$. In addition, this convergence happens after one communication round for a complete network graph.*

Proof. Please refer to Appendix A for a proof. □

Fig. 8. This snapshot of an outdoor hardware experiment shows the autonomous deployment of five quadrotor flying robots. Even though the robots were fully autonomous, a safety pilot was assigned to each robot to allow for manual overrides in the event of an emergency.

5. Experiments and simulations

We verified our approach with a small-scale indoor experiment and a large-scale outdoor experiment using five quadrotor flying robots. We then used the developed inference and coordination software to simulate a system of 100 robots.

5.1. Setup for the hardware experiments

We are working towards a multi-robot system that can rapidly assess the state of disaster-affected environments. In these cases the state can represent a wide spectrum of relevant information, ranging from the presence of fires and harmful substances to the structural integrity of buildings. Motivated by this goal, the task for the hardware experiments was to infer the state of a bounded, planar environment by deploying $n_r = 5$ Ascending Technologies Hummingbird quadrotor flying robots (Gurdan et al., 2007) belonging to the class of systems described in Theorem 2. Five heterogeneous sensors were simulated with measurement noise that was proportional to the field of view, meaning that sensors of larger footprints produced noisier observations. Sample sizes of $n_x = n_y = 500$ were selected for each experiment, and the consensus-based algorithm (Algorithm 3) of consensus round size $n_\pi = 3$ was implemented on the undirected network graph using an ideal disk model to determine connectivity.

For each environment, we defined $\mathcal{W}$ to be an n_w cell partition,[5] where for each cell, $\mathcal{W}^{[m]}$, the state was modeled as a random variable, $X_k^{[m]}$, that took values from an alphabet, $\mathcal{X}^{[m]}$. Thus, for the environment state, we had an n_w-tuple random variable, $X_k = (X_k^{[1]}, \ldots, X_k^{[n_w]})$, that took values from the alphabet $\mathcal{X} = \prod_{m=1}^{n_w} \mathcal{X}^{[m]}$. A first-order Markov model was used for the robots' state transition distributions, where a uniform probability represented the likelihood that the state of an environment discretization cell transitioned to any other state. Finally, we modeled the robot's observation as an n_w-tuple random variable, $Y_k^{[i]} = (Y_k^{[i,1]}, \ldots, Y_k^{[i,n_w]})$, that took values from an alphabet, $\mathcal{Y}^{[i]} = \prod_{m=1}^{n_w} \mathcal{Y}^{[i,m]}$.

Revisiting Remark 9, the belief matrix was compressed in a lossless manner by assuming conditional independence between environment discretization cells for the measurement probabilities. This assumption resulted in the cell measurement probabilities, $\mathbb{P}(Y_k^{[i,m]}|X_k)$, being dependent only on the state of the corresponding environment discretization cell, $X_k^{[m]}$, and conditionally independent from all other $Y_k^{[i,m']}$ with $m' \neq m$, such that

$$\mathbb{P}(Y_k^{[i]}|X_k) = \prod_{m=1}^{n_w} \mathbb{P}(Y_k^{[i,m]}|X_k^{[m]}) \,. \qquad (22)$$

In words, the robot's observation was composed of n_w conditionally independent observation elements, where each element concerned a specific environment discretization cell. For the experiments, the robots had maximum cell measurement probabilities of $\{0.95, 0.9, 0.85, 0.8, 0.75\}$, which decreased quadratically (e.g. power decay of light) to 0.5 at the edge of the robot's field of view.

Remark 10 (Using simulated sensors). We note that the experiments are a validation of the inference and coordination algorithm and not of the sensing capabilities of our hardware platforms. The selection of the simulated sensor properties was primarily motivated by their generality, in particular for systems employing passive sensors that measure the intensity of electromagnetic signals radiating from point sources. Nonetheless, we expect qualitatively good controller performance for systems that obtain noisier observations towards the boundaries of their sensors' limited range, even though convergence may not be guaranteed for such systems.

We then define a modified belief matrix $\breve{\pi}_{k'}^{[i]}$ that independently represents each possible element value of the n_w-tuple environment state. More formally, for all $m \in \{1, \ldots, n_w\}$ we assigned a single row of the modified belief matrix to each value that $X_k^{[m]}$ can take. Each column entry was initialized to the corresponding measurement probability from $\mathbb{P}(\check{Y}_k^{[i,m]}|X_k^{[m]})$, where $\check{Y}_k^{[i,m]}$ is the ith element of the n_w-tuple random variable $\check{Y}_k^{[i]}$ that took values from the alphabet $\check{\mathcal{Y}}_k^{[i]} = \prod_{m=1}^{n_w} \check{\mathcal{Y}}_k^{[i,m]}$. Let $\mathcal{M}(x^{[j]})$ denote the set of row indices that are initialized to elements contributing to the right hand side product of Equation (22) with respect to $\mathbb{P}(\check{Y}_k^{[i,m]}|X_k^{[m]} = x^{[j,m]})$. Using Belief_Consensus(*sampled*) to evolve the modified belief matrix, each robot calculated the original belief matrix entries needed for the sampled joint measurement probability approximations (20) from

$$\left[\pi_{k'}^{[i]}\right]_{j\ell} = \prod_{m \in \mathcal{M}(x^{[j]})} \left[\breve{\pi}_{k'}^{[i]}\right]_{m\ell}$$

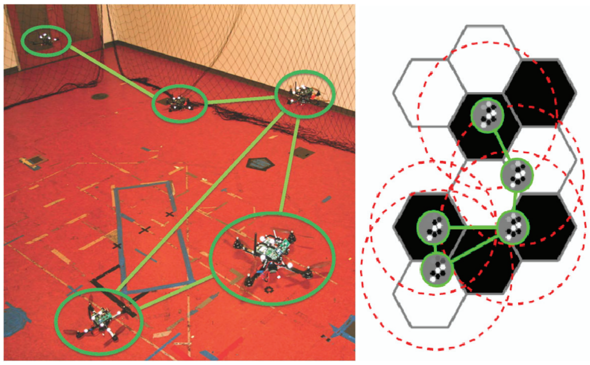

Fig. 9. *Left:* This snapshot of an indoor hardware experiment shows five quadrotor flying robots inferring the state of an environment. The green lines between robots represent network connectivity. *Right:* This schematic illustrates the state of each robot's inference enabled by simulated sensors with their footprints drawn in red dashed circles.

for all $j \in \{1, \ldots, n_x\}$ and $\ell \in \{1, \ldots, n_y\}$. Again, note that this method is a lossless compression as we are perfectly reconstructing $\pi_{k'}^{[i]}$ from $\breve{\pi}_{k'}^{[i]}$. The belief vector was also compressed in the same manner.

Remark 11 (Complexity). We do not assume independence between environment discretization cells, resulting in a state alphabet size that scales exponentially with respect to the number of cells ($\max_m O(|\mathcal{X}^{[m]}|^{n_w})$). One can assume independence to have this size scale linearly with respect to the number of cells ($\max_m O(n_w|\mathcal{X}^{[m]}|)$), which is a common assumption in the robot mapping literature (Thrun et al., 2005). Regardless, the size of the belief matrix remains linear with respect to the number of cells ($\max_m O(n_w n_y|\mathcal{X}^{[m]}|)$) due to the assumption of conditional independence between cells for the measurement probabilities.

5.2. *Indoor experiment with a small environment*

For the indoor experiment, the 10 m long environment (see Figure 9) was discretized into $n_w = 10$ hexagon cells, each being of width 2 m and having a binary static state of either 1 (e.g. fire) or 0 (e.g. no fire). In addition, the random variables representing the robots' observation elements, $Y_k^{[i,m]}$, also took values of 1 or 0. The experiment was conducted in an MIT CSAIL laboratory equipped with a Vicon motion capture system that wirelessly transmitted six degree-of-freedom robot pose information at 40 Hz via XBee. The real-time inference and coordination algorithm (Algorithm 1) ran in distributed fashion on each robot. The onboard 2 GHz single board computer hosted its own independent ROS environment (Quigley et al., 2009) and wirelessly communicated to the other robots via UDP multicast. We were able to achieve a sample period of $T_s \approx 2$ s with this system.

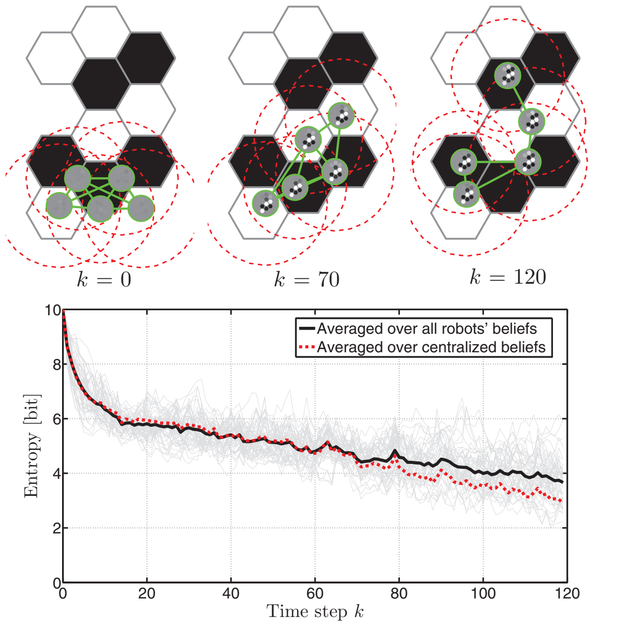

Fig. 10. *Top:* These three illustrations show the beginning, middle, and end configuration of a five-robot experiment over a 10 cell environment, where the state of each cell is either 1 (black) or 0 (white). The robots are represented by the gray circles, within which their prior distributions can be visualized. The green lines represent network connectivity, and the dashed red circles represent the simulated sensors' footprints. *Bottom:* This plot shows the decrease in entropy of the inferences averaged over 10 consecutive runs. In addition, the light grey lines show the entropy of each robot's belief for every run.

The five heterogeneous sensors had sensor radii of $\{2.0, 2.1, 2.2, 2.3, 2.4\}$ m, which we emphasized by setting the hovering height of a robot proportional to its value. In words, robots hovering closer to the environment had more accurate observations, but also had smaller fields of view. For all robots, we used a control policy set of $\mathcal{U}^{[i]} = [-0.1, 0.1]^2$ m/s, control gain of $\gamma^{[i]} = 10$, and an ideal disk network radius of 3 m. In addition, a safety radius of 1 m was enforced between neighboring robots, meaning the gradient projection of $u_t^{[i]}$ would be taken to prevent two directly communicating robots from moving closer than 1 m from each other.

We recorded 10 consecutive runs deploying the five robots from the bottom of the environment, including one robot that started on the environment boundary and another outside. Figure 10 shows the beginning, middle, and end configuration of a typical run, along with a plot showing the decrease in average entropy (i.e. uncertainty) of the robots' inferences compared with a centrally computed one. The centralized inferences considered observations from all robots, and can be interpreted as a baseline. To date, we have over 100 successful indoor runs with various starting positions and algorithm parameters, compared with one unsuccessful run caused by the motion capture system losing track of one robot. Even during this run, the distributed inference and coordination algorithm continued to run properly for the other robots, showing the approach's robustness to individual robot failures.

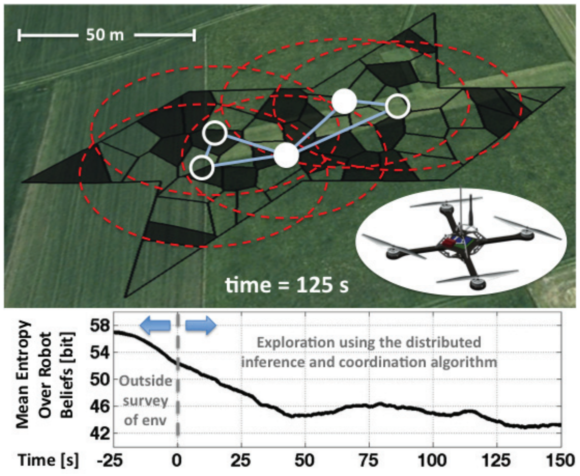

Fig. 11. The deployment of five quadrotor flying robots (white circles) tasked to explore a 150 m wide outdoor environment containing 58 discretized cells of binary state. Exploration by the robotic sensor network (blue lines) is accomplished by a gradient-based distributed controller that continuously moves the robots to minimize the uncertainty of the non-parametric inference. In parallel, the robots can be assigned by a higher-level communication scheme to act as dynamic network relays (white filled circle). The end result is a decrease in average entropy over time, as shown in the lower plot.

5.3. *Outdoor experiment with a large environment*

For the outdoor experiment, a 150 m wide environment (see Figures 11 and 12) was discretized using a Voronoi partitioner into $n_w = 58$ heterogeneous cells. The real-time inference and coordination algorithm (Algorithm 1) for all robots ran at 1 Hz on a single ground workstation. The resulting GPS-based control commands were wirelessly transmitted via XBee to each robot's onboard autopilot. The five heterogeneous sensors were simulated with measurement noise proportional to the sensor radii of $\{30, 32.5, 35, 37.5, 40\}$ m, again meaning that sensors of larger footprints produced noisier observations. Each robot used a control policy set of $\mathcal{U}^{[i]} = [-3, 3]^2$ m/s, control gain of $\gamma^{[i]} = 5$, and an ideal disk network radius of 50 m.

In preparation for the outdoor experiment, reproducible results were recorded from multiple preliminary deployments, producing over 25 minutes of total flight time. This initial effort was to verify the non-parametric methods without any higher-level control except for the manual override capabilities enabled by the Disaster Management Tool

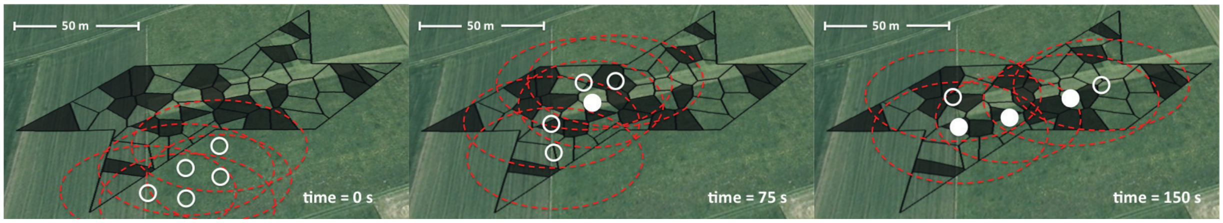

Fig. 12. The time evolution of a constellation of five quadrotor flying robots (white circles) with simulated sensor (red dashed circles). These robots are tasked to explore a 150 m wide outdoor environment containing 58 discretized cells of binary state. *Left:* The experiment starts with all robots hovering at their starting positions. *Middle:* The robots at time = 75 s have begun to explore the environment. In addition, one robot is assigned by a higher-level communication scheme to act as a dynamic network relay (white filled circle), and thus the control actions produced by the distributed controller are overridden for that robot. *Right:* At time = 150 s, even though three robots are assigned as dynamic relays, the distributed controller has driven the system into a configuration that covers a large portion of the cells.

(DMT) developed at the German Aerospace Center (DLR) (Frassl et al., 2010). Once we obtained qualitative validation for our approach, the algorithms were adjusted to handle binary event detection (e.g. fire or no fire) as described for the indoor experiment. In addition, a decentralized communication scheme continuously assigned robots to act as dynamic network relays, overriding the control actions produced by the distributed controller (13). For the experiment, the robots were deployed from outside the environment, and at any given point could have at most 58 bits of uncertainty concerning the inference. The plot in Figure 11 shows the decrease in entropy over the extent of the experiment, even though the higher-level communication scheme at times was overriding the distributed controller.

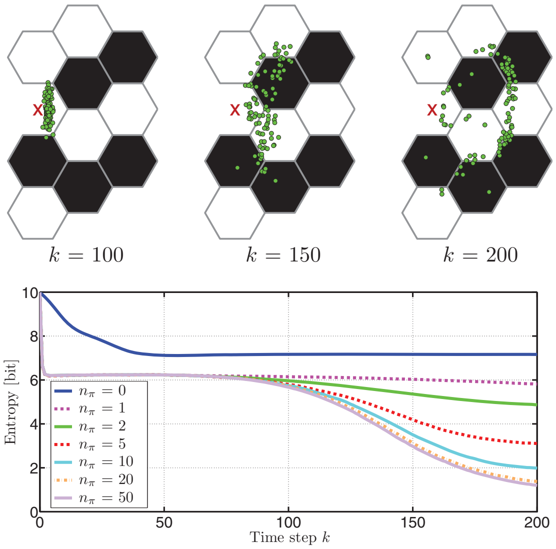

Fig. 13. *Top:* The configuration of the 100 simulated robots at time steps $k = 100$, 150, and 200. All robots (green ○) used a consensus round size of $n_\pi = 10$ and were deployed from a common location (red ×). *Bottom:* This plot shows the entropy of the robots' beliefs for various consensus round sizes averaged over 1000 Monte Carlo simulations.

5.4. *Simulation of a large robot system*

To demonstrate the scalability of our approach with respect to the number of robots, we simulated a $n_r = 100$ robot system using different values for the consensus round size. For each run, the heterogeneous sensors were randomly selected from the sensor set used in the indoor hardware experiments, and the robots were deployed from a single location outside the environment (see Figure 13). To emulate a physically larger environment for the simulation, each robot used an ideal disk network radius of 1.5 m and no safety radius. All other parameters were identical to the indoor experiments.

Using the same software developed for the robot hardware, we verified that the increase in runtime for the simulation scaled appropriately. Figure 13 shows the decrease in the average entropy of the robots' inferences over 1000 Monte Carlo simulations. As expected, larger consensus round sizes resulted in lower overall uncertainty within the system, although there is clearly evidence of diminishing returns. In addition, the simulations highlight the importance of the network topology; even though many more robots are deployed in comparison to the hardware experiments, the propagation of information throughout the system is hindered by the sparsity of the network when using small consensus round sizes. This result raises interesting questions about fundamental limitations that cannot be overcome by simply deploying more robots.

6. Conclusion

We present a suite of novel representations and algorithms for distributed inference and coordination in multi-robot systems. We emphasized how structuring information exchanged on the network can reduce or even negate the influence external factors have on each robot, resulting

Table 1. Common parameters used for the hardware experiments.

Parameter	Symbol	Value
Consensus round size	n_π	3
Environment state alphabet	$\mathcal{X}^{[m]}$	$\{0, 1\}$
Min cell measurement probability	$\min \mathbb{P}(Y_k^{[i,m]} \vert X_k)$	0.5
Max cell measurement probability	$\max \mathbb{P}(Y_k^{[i,m]} \vert X_k)$	$\{0.95, 0.9, 0.85, 0.8, 0.75\}$
Number of robots	n_r	5
Robot's observation alphabet	$\mathcal{Y}^{[i,m]}$	$\{0, 1\}$
Sample set sizes	n_x, n_y	500, 500

Table 2. Parameters used for the indoor hardware experiment.

Parameter	Symbol	Value
Control gain	$\gamma^{[i]}$	10
Control policy	$\mathcal{U}^{[i]}$	$[-0.1, 0.1]^2$ m/s
Ideal disk network radius	–	3.0 m
Number of environment discretization cells	n_w	10
Safety radius	–	1 m
Sample period	T_s	2 s
Sensor radii	–	$\{2.0, 2.1, 2.2, 2.3, 2.4\}$ m
State transition distribution	$\mathbb{P}^{[i]}(X_{k+1}^{[m]} \vert X_k^{[m]})$	0.99, uniform otherwise

Table 3. Parameters used for the outdoor hardware experiment.

Parameter	Symbol	Value
Control gain	$\gamma^{[i]}$	5
Control policy	$\mathcal{U}^{[i]}$	$[-3, 3]^2$ m/s
Ideal disk network radius	–	50 m
Number of environment discretization cells	n_w	58
Safety radius	–	10 m
Sample period	T_s	1 s
Sensor radii	–	$\{30, 32.5, 35, 37.5, 40\}$ m
State transition distribution	$\mathbb{P}^{[i]}(X_{k+1}^{[m]} \vert X_k^{[m]})$	0.95, uniform otherwise

in a scalable and robust approach to information acquisition tasks. In contrast to previous works, we made less restrictive assumptions concerning the type of probability distributions present in the system, or what knowledge the robots have about the topology of the communication network. Nevertheless, the complexity in terms of computation, memory, and network load remains constant with respect to number of robots.

We gave several insights into the information-theoretic framework supporting multi-robot inference and coordination algorithms. We showed that these algorithms are computationally intractable in their general form, and that principled approximations forfeit optimality to reduce overall complexity. In addition, we proved that each robot can distributively approximate the joint measurement probabilities such that they converge to the true values as the size of the consensus rounds grows or as the network becomes complete. This ability enables each robot to maintain its own belief of the environment and intelligently reason how to position its sensors to take future measurements.

We validated our approach by conducting small-scale indoor experiments, large-scale outdoor experiments, and numerical simulations that showed fully autonomous exploration of a predefined area with an arbitrary number of robots. Moreover, we demonstrated how the algorithm automatically adapts to overrides from a higher-level controller. We believe the natural ability to combine low-level autonomy with higher-level cognitive supervision is particularly advantageous and an important step towards fieldable surveillance and exploration systems in the foreseeable future.

Notes

1. We use the term *non-parametric* to convey that we do not assume that the statistics of the involved random variables can be exactly described by particular distributions with finite numbers of parameters.
2. We have for all robots that $\sum_{j\in\{1,\ldots,n_x\}} \check{w}_k^{[i,j]} = 1$.
3. A communication round, denoted k', is defined as a single update using a FloodMax algorithm, a consensus algorithm, or both algorithms if run in parallel.
4. We are using terminology introduced by Pearl in Pearl (1988).
5. The partition $\mathcal{W}$ is defined as a collection of closed connected subsets of $\mathcal{Q}$ satisfying $\prod_{m=1}^{n_w} \mathcal{W}^{[m]} = \mathcal{W}$, $\bigcup_{m=1}^{n_w} \mathcal{W}^{[m]} = \mathcal{Q}$ and $\bigcap_{m=1}^{n_w} \mathrm{int}(\mathcal{W}^{[m]}) = \emptyset$, where int($\cdot$) denotes the subset of interior points.

Funding

This work is sponsored by the Department of the Air Force (Air Force contract number FA8721-05-C-0002). The opinions, interpretations, recommendations, and conclusions are those of the authors and are not necessarily endorsed by the United States Government. This work is also supported in part by ONR (grant number N00014-09-1-1051), NSF (grant number EFRI-0735953), MIT Lincoln Laboratory, the European Commission, the Helmholtz Foundation, the Project SOCIETIES, and the Boeing Company.

Acknowledgements

The authors would like to thank Emilio Frazzoli and Patrick Robertson for their many significant contributions in the areas of consensus algorithms, information theory, and probabilistic methods. The authors would also like to thank Martin Frassl, Michael Lichtenstern, Michael Walter, Ulrich Epple, and Frank Schubert for their assistance with the experiments, as well as the NASA World Wind Project for providing the World Wind technology that has been an essential part for experimental visualization. Lastly, we are indebted to the editor and reviewers for supporting this work and providing profound insights.

References

Borenstein J, Feng L and Everett HR (1997) *Navigating Mobile Robots: Systems and Techniques* (*Data Management Systems Series*). San Mateo, CA: Morgan Kaufmann Publishers, Inc.

Bourgault F, Makarenko A, Williams SB, Grocholsky B and Durrant-Whyte HF (2002) Information based adaptive robotic exploration. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, pp. 540–545.

Bullo F, Cortés J and Martínez S (2009) *Distributed Control of Robotic Networks* (*Applied Mathematical Series*). Princeton, NJ: Princeton University Press.

Cameron A and Durrant-Whyte H (1990) A Bayesian approach to optimal sensor placement. *The International Journal of Robotics Research* 9(5): 70.

Choi HL and How JP (2010) Continuous trajectory planning of mobile sensors for informative forecasting. *Automatica* 46: 1266–1275.

Chung TH, Gupta V, Burdick JW and Murray RM (2004) On a decentralized active sensing strategy using mobile sensor platforms in a network. In *Proceedings of the IEEE Conference on Decision and Control*, vol. 2, pp. 1914–1919.

Cortés J (2009) Distributed kriged Kalman filter for spatial estimation. *IEEE Transactions on Automatic Control* 54: 2816–2827.

Cover TM and Thomas JA (1991) *Elements of Information Theory*. New York: John Wiley & Sons.

Engelson SP and McDermott DV (1992) Error correction in mobile robot map learning. In *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 3, pp. 2555–2560.

Fox D, Burgard W, Kruppa H and Thrun S (2000) A probabilistic approach to collaborative multi-robot localization. *Autonomous Robots*.

Fraser CSR, Bertuccelli LF, Choi HL and How JP (2012) A hyper-parameter consensus method for agreement under uncertainty. *Automatica* 48: 374–380.

Frassl M, Lichtenstern M, Khider M and Angermann M (2010) Developing a system for information management in disaster relief - methodology and requirements. In *Proceedings of the International Community on Information Systems for Crisis Response and Management Conference*.

Grocholsky B (2002) *Information-theoretic Control of Multiple Sensor Platforms*. Ph.D. thesis, University of Sydney.

Grocholsky B, Makarenko A and Durrant-Whyte H (2003) Information-theoretic control of multiple sensor platforms. In *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, pp. 1521–1526.

Gurdan D, Stumpf J, Achtelik M, Doth KM, Hirzinger G and Rus D (2007) Energy-efficient autonomous four-rotor flying robot controlled at 1 kHz. In *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, pp. 361–366.

Hirsche MW and Smale S (1974) *Differential Equations, Dynamical Systems, and Linear Algebra* (*Applied Mathematical Series*). New York: Academic Press, Inc.

Hoffmann GM and Tomlin CJ (2010) Mobile sensor network control using mutual information methods and particle filters. *IEEE Transactions on Automatic Control* 55: 32–47.

Howard A (2006) Multi-robot simultaneous localization and mapping using particle filters. *The International Journal of Robotics Research* 25: 1243–1256.

Ihler AT, Fischer III JW, Moses RL and Willsky AS (2005) Non-parametric belief propagation for self-localization of sensor networks. *IEEE Journal on Selected Areas in Communications* 23: 809–819.

Julian BJ, Angermann M and Rus D (2012a) Non-parametric inference and coordination for distributed robotics. In *Proceedings of the IEEE Conference on Decision and Control*.

Julian BJ, Angermann M, Schwager M and Rus D (2011) A scalable information theoretic approach to distributed robot coordination. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5187–5194.

Julian BJ, Schwager M, Angermann M and Rus D (2009) A location-based algorithm for multi-hopping state estimates within a distributed robot team. In *Proceedings of the International Conference on Field and Service Robotics*.

Julian BJ, Smith SL and Rus D (2012b) Distributed approximation of joint measurement distributions using mixtures of Gaussians. In *Proceedings of the Robotics: Science and Systems Conference*.

Krause A, Singh A and Guestrin C (2008) Near-optimal sensor placements in Gaussian processes: theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research* 9: 235–284.

LaSalle J (1960) Some extensions of Lyapunov's second method. *IRE Transactions on Circuit Theory* 7: 520–527.

Leung K, Barfoot T and Liu H (2012) Decentralized cooperative SLAM for sparsely-communicating robot networks: a centralized-equivalent approach. *Journal of Intelligent and Robotic Systems* 66: 321–342.

Lynch KM, Schwartz IB, Yang P and Freeman RA (2008) Decentralized environmental modeling by mobile sensor networks. *IEEE Transactions on Robotics* 24: 710–724.

Lynch NA (1997) *Distributed Algorithms* (*Data Management Systems Series*). San Mateo, CA: Morgan Kaufmann Publishers, Inc.

Montemerlo M, Thrun S, Koller D and Wegbreit B (2002) FastSLAM: A factored solution to the simultaneous localization and mapping problem. In *Proceedings of the AAAI National Conference on Artificial Intelligence*. AAAI, pp. 593–598.

Nettleton EW, Durrant-Whyte HF, Gibbens PW and Goektogan AH (2000) Multiple platform localisation and map building. *Proceedings of the Society of Photo-Optical Instrumentation Engineers* 4196: 337–347.

Ny JL and Pappas GJ (2009) On trajectory optimization for active sensing in Gaussian process models. In *Proceedings of the Joint IEEE Conference on Decision and Control and Chinese Control Conference*, pp. 6282–6292.

Olfati-Saber R, Franco E, Frazzoli E and Shamma JS (2005) Belief consensus and distributed hypothesis testing in sensor networks. In *Proceedings of the Network Embedded Sensing and Control Workshop*, pp. 169–182.

Palomar DP and Verdú S (2007) Representation of mutual information via input estimates. *IEEE Transactions on Information Theory* 53: 453–470.

Pearl J (1988) *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, CA: Morgan Kaufmann Publishers, Inc.

Quigley M, Gerkey B, Conley K, et al. (2009) ROS: an open-source robot operating system. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 3515–3522.

Schulz D, Burgard W, Fox D and Cremers AB (2001) Tracking multiple moving targets with a mobile robot using particle filters and statistical data association. In *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1650–1670.

Schwager M, Dames P, Rus D and Kumar V (2011a) A multi-robot control policy for information gathering in the presence of unknown hazards. In *Proceedings of the International Symposium on Robotics Research*.

Schwager M, Julian BJ, Angermann M and Rus D (2011b) Eyes in the sky: decentralized control for the deployment of robotic camera networks. *Proceedings of the IEEE* 99: 1541–1561.

Singh A, Krause A, Guestrin C, Kaiser W and Batalin M (2007) Efficient planning of informative paths for multiple robots. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 2204–2211.

Thrun S, Burgard W and Fox D (2005) *Probabilistic Robotics* (*Intelligent Robotics and Autonomous Agents Series*). Cambridge, MA: The MIT Press.

Thrun S and Liu Y (2005) Multi-robot SLAM with sparse extended information filters. In *Robotics Research* (*Springer Tracts in Advanced Robotics*, vol. 15). Berlin: Springer, pp. 254–266.

Xiao L, Boyd S and Kim S-J (2007) Distributed average consensus with least-mean-square deviation. *Journal of Parallel and Distributed Computing* 67: 33–46.

Zavlanos MM and Pappas GJ (2008) Distributed connectivity control of mobile networks. *IEEE Transactions on Robotics* 24: 1416–1428.

Appendix A: Proofs

This section contains proofs for four theorems stated in this paper.

Proof (Theorem 1). This proof was derived in collaboration with Schwager et al. (2011a), and a similar result in the context of channel coding was proved by Palomar and Verdú (2007). Concerning the partial derivative of Equation (2) with respect to $c_t^{[i]}$, we can move the differentiation inside the integrals since it does not depend on $c_t^{[i]}$. Applying the chain rule to the integrand and separating the two resulting terms, we have

$$\begin{aligned}\frac{\partial U_k}{\partial c_t^{[i]}} &= \int_y \int_x \frac{\partial \mathbb{P}(X_k = x | Y_k = y)}{\partial c_t^{[i]}} \mathbb{P}(Y_k = y)\, dx\, dy \\ &+ \int_y \int_x \frac{\partial \mathbb{P}(Y_k = y | X_k = x)}{\partial c_t^{[i]}} \mathbb{P}(X_k = x) \\ &\times \log\left(\frac{\mathbb{P}(X_k = x | Y_k = y)}{\mathbb{P}(X_k = x)}\right) dx\, dy, \end{aligned} \tag{23}$$

where

$$\mathbb{P}(Y_k) = \int_x \mathbb{P}(Y_k | X_k = x)\, \mathbb{P}(X_k = x)\, dx$$

from the law of total probability. We will now show that the first integration term on the right-hand side of (23) is equal to zero. First using the chain rule to take the partial derivative of Equation (1) with respect to $c_t^{[i]}$, we have

$$\begin{aligned}\frac{\partial \mathbb{P}(X_k | Y_k)}{\partial c_t^{[i]}} &= \frac{\partial \mathbb{P}(Y_k | X_k)}{\partial c_t^{[i]}} \frac{\mathbb{P}(X_k)}{\mathbb{P}(Y_k)} \\ &- \frac{\partial \mathbb{P}(Y_k = y)}{\partial c_t^{[i]}} \frac{\mathbb{P}(Y_k | X_k)\, \mathbb{P}(X_k)}{\mathbb{P}(Y_k)^2}. \end{aligned} \tag{24}$$

Substituting Equation (24) back into the first integration term on the right-hand side of Equation (23) and considering the two resulting integrals separately, we have

$$\begin{aligned} &\int_y \int_x \frac{\partial \mathbb{P}(Y_k = y | X_k = x)}{\partial c_t^{[i]}} \mathbb{P}(X_k = x)\, dx\, dy \\ &= \frac{\partial}{\partial c_t^{[i]}} \int_y \int_x \mathbb{P}(Y_k = y | X_k = x)\, \mathbb{P}(X_k = x)\, dx\, dy \\ &= \frac{\partial}{\partial c_t^{[i]}} 1 \\ &= 0 \end{aligned}$$

and

$$
\begin{aligned}
&\int_y\int_x \frac{\partial\mathbb{P}(Y_k=y)}{\partial c_t^{[i]}}\,\frac{\mathbb{P}(Y_k=y|X_k=x)\,\mathbb{P}(X_k=x)}{\mathbb{P}(Y_k=y)}\,dx\,dy\\
&=\int_y \frac{\partial\mathbb{P}(Y_k=y)}{\partial c_t^{[i]}}\int_x \frac{\mathbb{P}(Y_k=y|X_k=x)\,\mathbb{P}(X_k=x)}{\mathbb{P}(Y_k=y)}\,dx\,dy\\
&=\int_y \frac{\partial\mathbb{P}(Y_k=y)}{\partial c_t^{[i]}}\,dy\\
&=\frac{\partial}{\partial c_t^{[i]}}\int_y \mathbb{P}(Y_k=y)\,dy\\
&=\frac{\partial}{\partial c_t^{[i]}}1\\
&=0.
\end{aligned}
$$

□

Proof (Theorem 2). Let

$$
V_k=-U_k=-\int_{y\in\mathcal{Y}}\int_{x\in\mathcal{X}}\mathbb{P}(Y_k=y|X_k=x)\,\mathbb{P}(X_k=x)\times\log\left(\frac{\mathbb{P}(X_k=x|Y_k=y)}{\mathbb{P}(X_k=x)}\right)dx\,dy
$$

be a Lypanov-type function candidate whose partial derivative with respect to $c_t^{[i]}$ is the negative of the discrete-time version of $\partial U_k/\partial c_t^{[i]}$ from Equation (3). The closed-loop dynamics

$$
\frac{dc_t^{[i]}}{dt}=-\gamma^{[i]}\frac{\partial V_k}{\partial c_t^{[i]}}
$$

are autonomous, and since $\partial\mathbb{P}(Y_k^{[i]}|X_k)/\partial c_t^{[i]}$ is continuous on $\mathcal{C}^{[i]}$ for all all robots $i\in\{1,\dots,n_r\}$, $\partial V_k/\partial c_t$ is continuous on $\mathcal{C}$. Therefore, the dynamics are locally Lipschitz, and V_k is continuously differentiable. Taking the Lie derivative of V_k along the trajectories of the system, we have

$$
\mathcal{L}_t V_k=\sum_{i=1}^{n_r}\frac{\partial V_k}{\partial c_t^{[i]}}\frac{dc_t^{[i]}}{dt}=-\sum_{i=1}^{n_r}\gamma^{[i]}\left(\frac{\partial V_k}{\partial c_t^{[i]}}\right)^2\leq 0.
$$

Now consider robots 'far' enough away from $\mathcal{Q}$ such that for some $D>0$ we have $\mathrm{dist}(c_t^{[i]},q)\geq D$ for all $q\in\mathcal{Q}$ and $i\in\{1,\dots,n_r\}$, where $\mathrm{dist}:\mathcal{C}^{[i]}\times\mathcal{Q}\to\mathbb{R}$ is a valid distance function. Then for all robots, we have

$$
\frac{\partial\mathbb{P}(Y_k^{[i]}|X_k)}{\partial c_t^{[i]}}=0,
$$

implying that $dc_t^{[i]}/dt=0$ for all time. Hence, all evolutions of the system are bounded. Finally, consider the set of all $c_*=(c_*^{[1]},\dots,c_*^{[n_r]})\in\mathcal{C}$ such that for all robots we have

$$
\left.\frac{\partial V_k}{\partial c_t^{[i]}}\right|_{c_t^{[i]}=c_*^{[i]}}=0.
$$

This set is invariant since it implies $\partial U_k/\partial c_t^{[i]}=0$ for all $i\in\{1,\dots,n_r\}$. Thus, all conditions of LaSalle's Invariance Principle are satisfied and the trajectories will converge to this invariant set (LaSalle, 1960; Bullo et al., 2009).

In addition, we have that c_* is either a local minimum, maximum, or saddle point for the constrained optimization problem $\max_{c\in\mathcal{C}}U_k$. However, for a gradient system of this type, we know that this configuration is a Lyapunov-stable equilibrium if and only if it is a local maximum, and thus locally optimal (Hirsche and Smale, 1974). □

Proof (Theorem 3). From Lemma 1, we have for all robots and $k'\geq\mathrm{diam}(\mathcal{G}_k)$ that $\delta_{k'}^{[i]}$ is equal to $(1+\Delta_k)$. Substituting $(1+\Delta_k)$ into Equation (17) for all $\delta_{k'}^{[i]}$ and $\delta_{k'+1}^{[i]}$ results in a consensus algorithm that is equivalent to Equation (16) with $\epsilon_k=1/(1+\Delta_k)$. In addition, we know for $k'<\mathrm{diam}(\mathcal{G}_k)$ that the time-varying nonlinear system will not have worse than exponential divergence since all coefficients in the right-hand side of Equation (17) are bounded below and above by 0 and 1, respectively. Thus, since $\psi_{k'}^{[i]}$ in Equation (16) was proven to converge to $\sum_{v=1}^{n_r}\psi_0^{[v]}/n_r$, we have that $\psi_{k'}^{[i]}$ in Equation (17) will do the same if and only if $\sum_{v=1}^{n_r}\psi_{k'}^{[v]}$ equals $\sum_{v=1}^{n_r}\psi_0^{[v]}$ for $k'=\mathrm{diam}(\mathcal{G}_k)$.

For this aim, consider the summation

$$
\begin{aligned}
\textstyle\sum_{i=1}^{n_r}\delta_{k'+1}^{[i]}\psi_{k'+1}^{[i]}&=\textstyle\sum_{i=1}^{n_r}(\delta_{k'+1}^{[i]}-\delta_{k'}^{[i]})\,\psi_0^{[i]}+\\
&\textstyle\sum_{i=1}^{n_r}\delta_{k'}^{[i]}\psi_{k'}^{[i]}+\sum_{i=1}^{n_r}\sum_{v\in\mathcal{N}_k^{[i]}}(\psi_{k'}^{[v]}-\psi_{k'}^{[i]}).
\end{aligned}
$$

In an undirected graph, the last term on the right-hand side of the last equation is equal to zero, and thus we have for all communication rounds that

$$
\begin{aligned}
\sum_{i=1}^{n_r}\delta_{k'+1}^{[i]}\psi_{k'+1}^{[i]}&=\sum_{i=1}^{n_r}(\delta_{k'+1}^{[i]}-\delta_{k'}^{[i]})\,\psi_0^{[i]}+\sum_{i=1}^{n_r}(\delta_{k'}^{[i]}-\delta_{k'-1}^{[i]})\,\psi_0^{[i]}\\
&+\cdots+\sum_{i=1}^{n_r}(\delta_1^{[i]}-\delta_0^{[i]})\,\psi_0^{[i]}+\sum_{i=1}^{n_r}\delta_0^{[i]}\psi_0^{[i]}=\sum_{i=1}^{n_r}\delta_{k'+1}^{[i]}\psi_0^{[i]},
\end{aligned}
$$

implying that $\sum_{v=1}^{n_r}\psi_{k'}^{[v]}$ equals $\sum_{v=1}^{n_r}\psi_0^{[v]}$ when $\delta_{k+1}^{[i]}=\Delta_k$. Thus, we have that $\psi_{k'}^{[i]}$ in Equation (17) converges to $\sum_{v=1}^{n_r}\psi_0^{[v]}/n_r$, which also implies that $\pi_{k'}^{[i]}$ in Equation (18) converges to $\prod_{v=1}^{n_r}(\pi_0^{[v]})^{1/n_r}$. □

Proof (Theorem 4). For all robots, $j\in\{1,\dots,|\mathcal{X}|\}$, and $\ell\in\{1,\dots,n_y\}$, let $[\pi_{k'}^{[i]}]_{j\ell}$ be initialized to

$$
[\pi_0^{[i]}]_{j\ell}=\mathbb{P}(\check{Y}_k^{[i]}=\check{y}^{[i,\ell]}|X_k=x^{[j]})
$$

and evolve using (18). From Theorem 3, we have that

$$
[\pi^{[i]}]_{j\ell}\to\prod_{v=1}^{n_r}\mathbb{P}(\check{Y}_k^{[v]}=\check{y}^{[v,\ell]}|X_k=x^{[j]})^{1/n_r}
$$

and

$$
\beta_k^{[i]}\to n_r
$$

in the limit as n_π tends to infinity. Hence, from Equation (19) we have for all $j \in \{1, \dots, n_x\}$ that

$$[\pi^{[i]}]_{j\ell} \to \eta\, \mathbb{P}(\check{Y} = \check{y}^{[\ell]} | X_k = x_k^{[j]})\,.$$

Lastly, using the definition of $\check{p}_k^{[i]}$ from Equation (20), we have that

$$[\check{p}_k^{[i]}]_{j\ell} \to \mathbb{P}(\check{Y}_k = \check{y}_k^{[\ell]} | X = x^{[\xi]})\,.$$

Convergence after one communication round for a complete network graph is a direct consequence of Lemma 2.

The proof for $p_k^{[i]}$ follows in the same manner with $[\pi_{k'}^{[i]}]_j$ being initialized to $\mathbb{P}(Y_k^{[i]} = y_k^{[i]} | X_k = x^{[j]})$ and converging to $\mathbb{P}(Y_k = y_k | X_k = x^{[j]})$. □

Appendix B: Notation

Tables B1–B12 contains descriptions of the notation used in this paper.

Table B1. Math preliminaries.

Notation	Description
$\mathbb{R}$	Euclidean space
$\mathbb{S}$	Sphere space
$\mathbf{e}_i$	Standard basis pointing in the ith direction
$\int(\cdot)$	Lebesgue integral
$\sum(\cdot)$	Set summation
$\prod(\cdot)$	Set product
$\log(\cdot)$	Natural logarithm
$\max(\cdot)$	Maximum valued element of a set
$\mathrm{dist}(\cdot)$	Distance function
$\mathrm{diam}(\cdot)$	Diameter of a graph
$\mathrm{int}(\cdot)$	Subset of interior points
$\to$	Converges to
$[\cdot]_j$	Column vector entry located in the jth row
$[\cdot]_{jk}$	Matrix entry located in the jth row and kth column

Table B2. Probability theory and information theory.

Notation	Description
$\mathbb{P}(\cdot)$	Probability
$\mathbb{P}(\cdot\|\cdot)$	Conditional probability
$H(\cdot)$	Entropy
$H(\cdot\|\cdot)$	Conditional entropy
$I(\cdot,\cdot)$	Mutual information

Table B3. System parameters.

Notation	Description
n_r	Number of robots
n_π	Number of communication rounds in a consensus round
n_x	Number of samples in the environment state sample set
n_y	Number of samples in the observation sample set
n_w	Number of discretization cells in the environment partition

Table B4. Continuous and discrete time.

Notation	Description
t	Continuous time of the system
k	Discrete time step of the system
k'	Communication round within a consensus round
T_s	Sampling interval of the system
T_c	Time period of the consensus round

Table B5. Environment properties.

Notation	Description
$\mathcal{Q}$	Environment of interest
r_q	Euclidean dimension of the environment
s_q	Sphere dimension of the environment
$\mathcal{W}$	Partition of the environment
$\mathcal{W}^{[m]}$	The mth discretization cell of the environment

Table B6. Robot properties.

c_t	Configuration of the system at time t
$c_t^{[i]}$	Configuration of the ith robot at time t
$\mathcal{C}$	Configuration space of the system
$\mathcal{C}^{[i]}$	Configuration space of the ith robot
$r_c^{[i]}$	Euclidean dimension of the ith robot's configuration space
$s_c^{[i]}$	Sphere dimension of the ith robot's configuration space

Table B7. Random variables for the environment state.

Notation	Description
X_k	Random variable for the environment state at discrete time step k
$X_k^{[m]}$	Random variable for the state of the mth environment discretization cell at discrete time step k

Table B8. Environment state sets.

Notation	Description
$\mathcal{X}$	Alphabet of all environment states
$\mathcal{X}^{[m]}$	Alphabet of all states of the mth environment discretization cell
$\check{\mathcal{X}}_k^{[i]}$	Weighted sample set for the environment state of the ith robot at discrete time step k

Table B9. Random variables for the observations.

Notation	Description
Y_k	Random variable for the joint observation at discrete time step k
$Y_k^{[i]}$	Random variable for the ith robot's observation at discrete time step k
$Y_k^{[i,m]}$	Random variable for the ith robot's observation of the mth environment discretization cell at discrete time step k
$\check{Y}_k$	Random variable for the sampled joint observation at discrete time step k
$\check{Y}_k^{[i]}$	Random variable for the sampled ith robot's observation at discrete time step k
$\check{Y}_k^{[i,m]}$	Random variable for the sampled ith robot's observation of the mth environment discretization cell at discrete time step k

Table B10. Observation sets.

Notation	Description
$\mathcal{Y}$	Alphabet of all joint observations
$\mathcal{Y}^{[i]}$	Alphabet of all ith robot's observations
$\mathcal{Y}^{[i,m]}$	Alphabet of all ith robot's observations of the mth environment discretization cell
$\check{\mathcal{Y}}_k$	Sampled set for the joint observations at discrete time step k
$\check{\mathcal{Y}}_k^{[i]}$	Sample set for the ith robot's observations at discrete time step k
$\check{\mathcal{Y}}_k^{[i,m]}$	Sample set for the ith robot's observations of the mth environment discretization cell at discrete time step k

Table B11. Communication system.

$\mathcal{G}_k$	Undirected communication graph at discrete time step k
$\mathcal{V}$	Vertex set
$\mathcal{E}_k$	Unordered edge set at discrete time step k
$\mathbf{A}_k$	Symmetric unweighted adjacency matrix at discrete time step k
Δ_k	Maximum in/out degree at discrete time step k
$\mathcal{N}_k^{[i]}$	Neighbors of the ith robot at discrete time step k
$\delta_{k'}^{[i]}$	FloodMax state of the ith robot at communication round k'
$\psi_{k'}^{[i]}$	Consensus state of the ith robot at communication round k'
$\pi_{k'}^{[i]}$	Exponential consensus state of the ith robot at communication round k'

Table B12. Inference and coordination.

U_k	Utility function for the system at discrete time step k
$\check{U}_k^{[i]}$	Approximation the utility function for the ith robot at discrete time step k
$\mathcal{U}_t^{[i]}$	Control action for ith robot at continuous time t
$\gamma^{[i]}$	Control gain for the ith robot
V_k	Lyapunov-type function candidate at discrete time step k
$\beta_k^{[i]}$	Exponential factor for joint measurement probabilities for the ith robot at discrete time step k
$p_k^{[i]}$	Distributed approximation of the joint measurement probabilities for the ith robot at discrete time step k
$\check{p}_k^{[i]}$	Distributed approximation of the sampled joint measurement probabilities for the ith robot at discrete time step k
$\pi_{k'}^{[i]}$	Consensus state for the ith robot at communication round k'
$\check{\pi}_{k'}^{[i]}$	Compressed consensus state for the ith robot at communication round k'
$\mathcal{M}(\cdot)$	Product indices for compressed consensus state