A Scalable Information Theoretic Approach to Distributed Robot Coordination

Brian J. Julian*[†], Michael Angermann[‡], Mac Schwager*[§], and Daniela Rus*

Abstract—This paper presents a scalable information theoretic approach to infer the state of an environment by distributively controlling robots equipped with sensors. The robots iteratively estimate the environment state using a recursive Bayesian filter, while continuously moving to improve the quality of the estimate by following the gradient of mutual information. Both the filter and the controller use a novel algorithm for approximating the robots' joint measurement probabilities, which combines consensus (for decentralization) and sampling (for scalability). The approximations are shown to approach the true joint measurement probabilities as the size of the consensus rounds grows or as the network becomes complete. The resulting gradient controller runs in constant time with respect to the number of robots, and linear time with respect to the number of sensor measurements and environment discretization cells, while traditional mutual information methods are exponential in all of these quantities. Furthermore, the controller is proven to be convergent between consensus rounds and, under certain conditions, is locally optimal. The complete distributed inference and coordination algorithm is demonstrated in experiments with five quad-rotor flying robots and simulations with 100 robots.
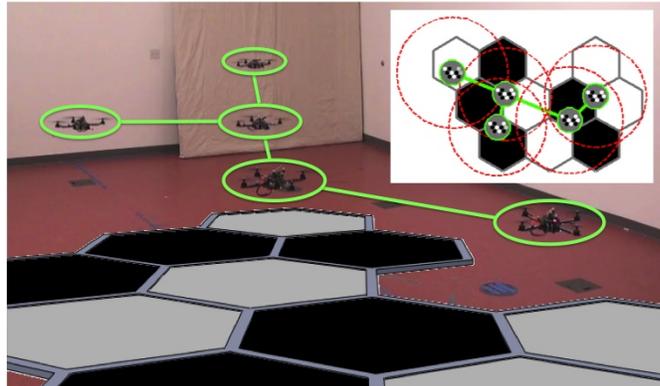
Fig. 1. This snapshot of a hardware experiment shows five quad-rotor flying robots inferring the state of an environment. The hexagon cells overlaying the snapshot illustrate the state and location of the discretized environment. The green lines between robots represent network connectivity, and the embedded schematic illustrates the state of the inference enabled by the simulated sensors drawn in red dashed circles.

I. Introduction

Robots are increasingly acting as our eyes and ears, extending human perception to remotely sense relevant aspects of an environment of interest. Especially when deployed in large numbers, robots are capable of rapidly gathering information by sharing their workload in a distributed fashion. While their operation may in general be more cost efficient than a human workforce, or even be the only option in situations that would otherwise endanger humans, it is highly desirable to use the robots and their sensors as effectively as possible in order to minimize operational cost or maximize the rate of information acquired. For this aim, we describe a novel approach for inferring the state of an environment and distributively controlling a multi-robot system. The approach significantly relaxes networking requirements of current state-of-the-art methods, is computationally efficient, and provably convergent. In the following, we lay out the underlying model and theory, based on an information theoretic foundation, and build a distributed inference and coordination algorithm. We then present empirical results of hardware experiments and numerical simulations.

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. As an early example, Cameron and Durrant-Whyte [2] used mutual information as a metric for sensor placement without explicitly considering mobility of the sensors. Later Grocholsky et al. [6], [7] proposed controlling multiple robot platforms so as to increase the mutual information between the robots' sensors and the position of a target in tracking applications. Bourgault et al. [1] 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 recently investigated, for example in [3], [11], [13]. In a

*Brian J. Julian, Mac Schwager, and Daniela Rus are with the Computer Science and Artificial Intelligence Lab, MIT, Cambridge, MA 02139, USA, bjulian@mit.edu, schwager@seas.upenn.edu, and rus@csail.mit.edu

[†]Brian J. Julian is also with MIT Lincoln Laboratory, 244 Wood Street, Lexington, MA 02420, USA

[‡]Michael Angermann is with the Institute of Communications and Navigation, DLR Oberpfaffenhofen, 82234 Wessling, Germany, michael.angermann@drl.de

[§]Mac Schwager is also with the GRASP Laboratory, University of Pennsylvania, Philadelphia, PA 19106

multi-robot context, the main challenges in using mutual information as a control metric are computational complexity and network communication constraints. Mutual information and its gradient are exponential with respect to the number of robots, sensor measurements, and environment discretization cells, making their computation intractable in realistic applications. Furthermore, the computation of these quantities requires that every robot have current knowledge of every other robot's position and sensor measurements. Thus, most of the mutual information methods mentioned above are restricted to small groups of robots with all-to-all communication.

With respect to Bayesian estimation, methods based on Kalman filters, which are Bayesian filters for linear Gaussian systems, have commonly been used to estimate the state of an environment. For example, Lynch et al. [10] 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 [5] 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 Chung et al. in [4].

Similar to mutual information, traditional Bayesian approaches are not scalable with respect to the number of robots, sensor measurements, and environment discretization cells, but recent work has tried to address this issue. Notably, Hoffmann and Tomlin [8] proposed a particle filter to propagate a Bayesian estimate, and used greedy and pair-wise approximations to calculate mutual information. To relax the requirement for all-to-all communication, Olfati-Saber et al. [12] developed a consensus algorithm for Bayesian distributed hypothesis testing in a static sensor network context. In our work, we use a consensus algorithm inspired by [12] to compute the joint measurement probabilities needed for mutual information and Bayesian filter calculations. We then overcome the problem of scalability by judiciously sampling from the complete set of measurement probabilities to compute an approximate mutual information gradient.

The paper is organized as follows. In Section II we formulate the problem of inferring the state of an environment using a single robot equipped with sensors, then expand this concept to multiple robots. In Section III we formalize a communication model and construct a consensus algorithm that distributively approximates the joint measurement probabilities of the robots' observations given the state of the environment. In Section IV we introduce principled approximations that allow for mutual information calculations that run in constant time with respect to the number of robots, and linear time with respect to the number of sensor measurements and environment discretization cells. In Section V we implement the findings of all previous sections to develop a distributed controller

that is convergent between consensus rounds and, under certain conditions, is locally optimal. Finally, we demonstrate our complete approach in Section VI first through hardware experiments with five quad-rotor flying robots, then through numerical simulations with 100 robots.

Due to space constraints, proofs for all theorems and corollaries are presented in [9].

II. 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 multi-robot case.

A. Utility in Information

We wish to infer the state of an environment from measurements obtained by a number of robots equipped with sensors. We model the potentially time varying state as a continuous-time random variable, $X(t)$, taking values from an alphabet, $\mathcal{X}$. The robots cannot measure $X(t)$ directly but instead receive from their sensors an observation, which is also modeled as a random variable, $Y(t)$, taking values from an alphabet, $\mathcal{Y}$. The relationship between the true state and the noisy observation is described by measurement probabilities, $p_{Y(t)|X(t)}$. The sensors may be interpreted as a noisy channel, and since they are attached to the robots, $p_{Y(t)|X(t)}$ is a function of the robots' configuration, $c(t)$.

From Bayes' Rule, we can use an observation, $y \in \mathcal{Y}$, and the system's prior, $p_{X(t)}$, to compute the posterior,

$$p_{X(t)|Y(t)}\left(x|y\right) = \frac{p_{X(t)}(x)p_{Y(t)|X(t)}(y|x:c)}{\int\limits_{x' \in \mathcal{X}} p_{X(t)}(x')p_{Y(t)|X(t)}(y|x':c)dx'} \quad (1)$$

where c is shorthand representing $c(t)$ and the notation $p_{Y(t)|X(t)}(\cdot : c)$ emphasizes that the measurement probabilities depend on c. Since our objective is to best infer $X(t)$, we are motivated to move the robots into a configuration that minimizes the expected uncertainty of the inference after receiving the next observation. Our optimization objective is equivalent to minimizing conditional entropy,

$$H(X(t)|Y(t)) = H(X(t)) - I(X(t), Y(t))$$

where $H(X(t))$ is the entropy of the inference and $I(X(t), Y(t))$ is the mutual information between the inference and the observation. Since $H(X(t))$ is independent of c, minimizing $H(X(t)|Y(t))$ is equivalent to maximizing $I(X(t), Y(t))$. Thus, we define the utility function for the system to be

$$U := I(X(t), Y(t)) = \int\limits_{y \in \mathcal{Y}} \int\limits_{x \in \mathcal{X}} p_{Y(t)|X(t)}(y|x:c) \times$$
$$p_{X(t)}(x) \log\left(\frac{p_{X(t)|Y(t)}(x|y)}{p_{X(t)}(x)}\right) dx\, dy \quad (2)$$

where log represents the natural logarithm (i.e. unit nat). We are particularly interested in a class of controllers that use

a gradient ascent approach with respect to U, leading to the following theorem.

Theorem 1. *The gradient of the utility function* (2) *with respect to a single robot's configuration, c_i, is given by*

$$\frac{\partial U}{\partial c_i} = \int\limits_{y \in \mathcal{Y}} \int\limits_{x \in \mathcal{X}} \frac{\partial p_{Y(t)|X(t)}(y|x:c)}{\partial c_i} \times$$

$$p_{X(t)}(x) \log \left(\frac{p_{X(t)|Y(t)}(x|y)}{p_{X(t)}(x)} \right) dx \, dy \qquad (3)$$

Proof. *Please refer to [9] for a proof.*

B. Single Robot Case

Consider an environment, $\mathcal{Q} \subset \mathbb{R}^{r_q} \times \mathbb{S}^{s_q}$, where $\mathbb{R}^{r_q}$ and $\mathbb{S}^{s_q}$ represent the r_q-dimensional Euclidean space and the s_q-dimensional sphere, respectively. Although $X(t)$ is continuous-time in general, the digital inference calculations occur at discrete-time instances. Thus, we represent $X(t)$ as a discrete-time random variable, $X[k]$, with samples taken at constant time intervals, kT_s. In order to properly represent the temporal evolution of $X(t)$ and avoid aliasing, we need to apply the Nyquist-Shannon sampling theorem and have the sampling frequency, $f_s = 1/T_s$, be at least $2f_{\max}$, where $f_{\max}$ is the highest temporal frequency component of interest. We then define $\mathcal{W}$ to be an n_w cell partition[1] of $\mathcal{Q}$ where for each cell, $\mathcal{W}_m$, the state is modeled as a random variable, $X_m[k]$, that takes a value from an alphabet, $\mathcal{X}_m$. Analogous to discretizing $X(t)$ in the temporal domain, we need to apply the Nyquist-Shannon theorem in the spatial domain and have sufficient sampling frequencies in all dimensions. Without loss of generality we have $\prod_{m=1}^{n_w} \mathcal{X}_m = \mathcal{X}$.

For both the temporal and spatial domains the maximum frequency component of interest may itself be temporally and spatially varying, but for this paper we assume they are constant over all time and space. In addition, we recognize that $X(t)$ may not be bandlimited, so our approach cannot in general be considered lossless, but instead we are preserving the ability to perfectly reconstruct $X(t)$ within a predefined temporal and spatial frequency band. Finally, note that we are not making any assumptions on whether the random variables are continuous, discrete, or even numerical in value.

Now consider a single robot, denoted i, that has a prior, $p^i_{X[k]}$. Based on the sampling methodology described above, the robot should receive sensor observations at a frequency of at least f_s. Let these observations occur simultaneously with $X[k]$, where the discrete-time random variable $Y_i[k]$ is characterized by $p_{Y_i[k]|X[k]}$. Between observations, the robot is able to compute from (1) the posterior, $p_{X[k]|Y_i[k]}$, and form a new prior based on a state transition distribution, $p^i_{X[k+1]|X[k]}$. Thus, the prior at time $(k+1)T_s$ is given by

$$p^i_{X[k+1]}(x) = p^i_{X[k+1]|X}(x) p_{X|Y_i}(x|y_i) \qquad (4)$$

[1] 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 $\bigcap_{m=1}^{n_w} \text{int}(\mathcal{W}_m) = \emptyset$, where $\text{int}(\cdot)$ denotes the subset of interior points.

for all $x \in \mathcal{X}$ and $y_i \in \mathcal{Y}_i$, where X and Y_i are shorthand representing $X[k]$ and $Y_i[k]$, respectively. Equations (1) and (4) form the well-known duet of *prediction* and *update* in sequential Bayesian estimation.

C. Multi-Robot Case

With respect to a centralized system, the n_r robot case is a simple extension of the single robot case with $p^i_X = p_X$ and $p^i_{X[k+1]|X} = p_{X[k+1]|X}$ for all $i \in \{1, \ldots, n_r\}$. Let the system be synchronous in that observations are simultaneously received at time intervals of T_s. These observations are modeled as an n_r-tuple random variable, $Y = (Y_1, \ldots, Y_{n_r})$, that takes a value from an alphabet, $\mathcal{Y} = \prod_{i=1}^{n_r} \mathcal{Y}_i$. We assume conditional independence of Y_i for all $i \in \{1, \ldots, n_r\}$, giving joint measurement probabilities of

$$p_{Y|X}(y|x : c) = \prod_{i=1}^{n_r} p_{Y_i|X}(y_i|x : c_i) \qquad (5)$$

for all $x \in \mathcal{X}$ and $y = (y_1, \ldots, y_{n_r}) \in \mathcal{Y}$, where $c = (c_1, \ldots, c_{n_r})$ is an n_r-tuple denoting the configuration of all the robots. Thus, the posterior from (1) becomes

$$p_{X|Y}(x|y) = \frac{\prod\limits_{i=1}^{n_r} p_{Y_i|X}(y_i|x:c_i) p_X(x)}{\int\limits_{x' \in \mathcal{X}} \prod\limits_{i=1}^{n_r} p_{Y_i|X}(y_i|x':c_i) p_X(x') dx'} \qquad (6)$$

and the prior update from (4) becomes

$$p_{X[k+1]}(x) = p_{X[k+1]|X}(x) p_{X|Y}(x|y) \qquad (7)$$

Note that there is one system prior and posterior for the centralized system. For the decentralized system, we use superscript $\sqcup^i$ to denote distributions for a particular robot.

III. DECENTRALIZED SYSTEM

We formalize a communication model and introduce a consensus algorithm to distributively approximate the system's joint measurement probabilities, which we use for the controller in Section V-C.

A. Communication Model

We use graph theory to represent communication connectivity among the n_r robots. Between observations, 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, $\mathcal{G}$. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{A})$ consist of a vertex set, $\mathcal{V} = \{1, \ldots, n_r\}$, an unordered edge set, $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$, and a symmetric unweighted adjacency matrix, $\mathbf{A} \in \{0, 1\}^{n_r \times n_r}$, where

$$[\mathbf{A}]_{iv} = a_{iv} = \begin{cases} 1, & \text{if } (i, v) \text{ or } (v, i) \in \mathcal{E} \\ 0, & \text{otherwise} \end{cases} \qquad (8)$$

We denote $\mathcal{N}_i$ as the set of neighbors of robot i, who has an in/out degree of $|\mathcal{N}_i| = \sum_{v=1}^{n_r} a_{iv}$.

3

Given the volatile nature of mobile networks, we expect $\mathcal{G}$ to be incomplete, time-vary, and stochastic. The algorithms presented in this paper work in practice even when properties of $\mathcal{G}$ cannot be formalized. However, to allow for meaningful analysis from a theoretical perspective, we assume that $\mathcal{G}$ 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 rate of convergence of our consensus algorithm. Thus, $\mathcal{G}$ is modeled as a discrete-time dynamic system with time step T_s.

B. Consensus of the Joint Measurement Probabilities

Consider a robot that calculates an expectation involving the joint measurement probabilities, such as (2) or (3). Normally the robot would need to know the current configuration, observation, and sensor channel model for *all* robots. We instead introduce a consensus algorithm that approximates the joint measurement probabilities and in the limit converges to $p_{Y|X}(y|x : c)$. To simplify the formulation, we consider X and Y to be discrete-value random variables. Extensions to the continuous domain will be highlighted in future work.

Let the belief matrix $\pi_{k'}^i[k] \in \{[0,1]\}^{|\mathcal{X}| \times |\mathcal{Y}|}$ be a matrix representation of the approximated n_rth root of the joint measurement probabilities known by robot i at time $kT_s + k'T_c \le (k+1)T_s$, and set $\pi_0^i[k]$ to be

$$\left[\pi_0^i\right]_{j\ell} = p_{Y_i|X}(y_i^\ell|x^j : c_i)$$

for all $x^j \in \mathcal{X}$ and $y_i^\ell \in \mathcal{Y}_i$, where $\pi_{k'}^i$ is shorthand representing $\pi_{k'}^i[k]$, x^j denotes the jth element of $\mathcal{X}$, and y_i^ℓ denotes the ith element of the ℓth tuple of $\mathcal{Y}$. In words, the belief matrix is initialized to robot i's contribution to $p_{Y|X}(y|x : c)$ in (5). Before time $(k+1)T_s$, the robot is able to transmit its belief matrix and receive its neighbors' ones a total of $n_\pi \le \lfloor T_s/T_c \rfloor$ times. Let the evolution of each robot's belief matrix with each communication time step be described by the discrete-time dynamical system

$$\pi_{k'+1}^i = \pi_{k'}^i \prod_{v \in \mathcal{N}_i} \left(\frac{\pi_{k'}^v}{\pi_{k'}^i}\right)^{\frac{1}{1+\Delta_i[k]}} \tag{9}$$

where $\Delta_i[k]$ is any value less than or equal to $n_r - 1$ and greater than or equal to the maximum in/out degree of all robots participating in the consensus of $\pi_{n_\pi}^i$ (see Remark 2).

Theorem 2. *For the communication model described in Section III-A, the belief matrix π^i converges to $\bar{\pi}[k]$ for all $i \in \{1, \ldots, n_r\}$ in the limit as $n_\pi \to \infty$, where*

$$\left[\bar{\pi}\right]_{j\ell}^{n_r} = \prod_{i=1}^{n_r} p_{Y_i|X}\left(y_i^\ell|x^j : c_i\right)\Big|_{t=kT_s} \tag{10}$$

with π^i and $\bar{\pi}$ being shorthand for $\pi_{n_\pi}^i$ and $\bar{\pi}[k]$, respectively.

Proof. *Please refer to [9] for a proof.*

Olfati-Saber et al. showed in [12] that such an algorithm could be used for distributed hypothesis testing. Here we can use the consensus algorithm to, in theory, calculate $p_{Y|X}(y|x : c)$ for all $x \in \mathcal{X}$ and $y \in \mathcal{Y}$. However, since the belief matrix is of size $\max_{im} \mathcal{O}(|\mathcal{Y}_i|^{n_r}|\mathcal{X}_m|^{n_w})$ and convergence may only be an asymptotic result, we use the consensus algorithm to distributively approximate these probabilities. Techniques to reduce the size of the belief matrix are described in Section IV, but here we discuss how to account for the fact that π^i may be used before it actually converges to $\bar{\pi}$.

The proof for Theorem 2 considers (9) in logarithmic form, which at time step k is a linear time-invariant system with state matrix $\mathbf{A}_\beta[k]$, where

$$[\mathbf{A}_\beta]_{im} = \begin{cases} 1 - |\mathcal{N}_i|/(1 + \Delta_i), & \text{if } i = m \\ a_{im}/(1 + \Delta_i), & \text{otherwise} \end{cases}$$

with $\mathbf{A}_\beta$ and Δ_i being shorthand for $\mathbf{A}_\beta[k]$ and $\Delta_i[k]$, respectively. We then define

$$\beta_i[k] = \left(\max_v \left[\mathbf{A}_\beta^{n_\pi}\right]_{iv}\right)^{-1}$$

to be an exponential factor that describes how fast π^i converges to $\bar{\pi}$ as n_π increases. Consider the following approximation for the joint measurement probabilities,

$$\hat{p}_{Y|X}^i(y^\ell|x^j) = [\pi^i]_{j\ell}^{\beta_i} \tag{11}$$

for all $x^j \in \mathcal{X}$ and $y^\ell \in \mathcal{Y}$, where β_i is shorthand representing $\beta_i[k]$, and y^ℓ denotes the ℓth element of $\mathcal{Y}$. The logarithm of $[\pi^i]_{j\ell}$ can be thought of as a weighted mean of the logarithms of $p_{Y_i|X}(y_i^\ell|x^j : c_i)$ for all $i \in \{1, \ldots, n_r\}$. In words, β_i is the inverse of the largest weight, which in the approximation of (5) ensures that no entry in the right hand side product is raised to a power greater than 1. We can also approximate in a purely distributed manner the posterior from (6),

$$\hat{p}_{X|Y}^i(x|y) = \frac{p_X^i(x)\hat{p}_{Y|X}^i(y|x)}{\sum\limits_{x' \in \mathcal{X}} p_X^i(x')\hat{p}_{Y|X}^i(y|x')} \tag{12}$$

and the utility gradient from (3),

$$\frac{\partial \hat{U}}{\partial c_i} = \sum_{y \in \mathcal{Y}} \sum_{x \in \mathcal{X}} \frac{\partial p_{Y_i|X}(y_i|x:c_i)}{\partial c_i} \frac{\hat{p}_{Y|X}^i(y|x)}{p_{Y|X}(y_i|x:c_i)\big|_{t=kT_s}} \times$$
$$p_X^i(x) \log\left(\frac{\hat{p}_{X|Y}^i(x|y)}{p_X^i(x)}\right) \tag{13}$$

For the gradient approximation to be well defined, we must have $p_{Y_i|X}(y_i|x : c_i) \in \{(0,1)\}$ for all $x \in \mathcal{X}$, $y_i \in \mathcal{Y}_i$, and $c_i \in \mathcal{C}_i$. This requirement is equivalent to saying that all observations have some finite amount of uncertainty.

Remark 1. *To build intuition on how β_i affects (11) and (12), we can easily verify that for all $i \in \{1, \ldots, n_r\}$,*

(i) *$\beta_i = 1$ when $n_\pi = 0$, implying that with no communication each robot acts as an isolated system with $\hat{p}_{Y|X}^i(y|x) = p_{Y_i|X}(y_i|x : c_i)$;*

4

(ii) β_i *converges to* n_r *in the limit as* $n_\pi \to \infty$*, implying that with ideal communication the system behaves as if there exists a central fusion center that provides each robot with the true joint measurement probabilities.*

Remark 2. *The posterior approximation does not require knowledge of robot configurations or observations. In fact, the only non-local information needed by a robot is a subsection of the adjacency matrix used to calculate* β_i *and select* Δ_i*. In words, these parameters are dependent only on the entries of* A *that contribute to the consensus averaging of* π^i*. This implies that the system would naturally be able to acquire this information as part of the consensus algorithm.*

Lemma 1. *For a complete network graph with* $n_\pi \geq 1$*,* $\beta_i = n_r$ *and thus* $\pi^i = \bar{\pi}$ *for all* $i \in \{1, \ldots, n_r\}$*.*

IV. PRINCIPLED APPROXIMATIONS FOR SCALABILITY

We formulated a consensus algorithm to distributively approximate the joint measurement probabilities, however, the size of the belief matrix remained exponential with respect to the number of robots, sensor measurements, and environment discretization cells. We now introduce a sampling technique to approximate distributions over likely observations, then consider a special class of robots that allows for the lossless compression of the belief matrix. The resulting size is constant with respect to the number of robots and linear with respect to the number of sensor measurements and environment discretization cells. This reduction enables systems of more than 100 robots when traditional information theoretic methods are limited to 10 or less.

A. State and Observation Sample Sets

Let each robot draw n_x samples of equal weight, $1/n_x$, from p_X^i to generate a state sample set, $\mathbb{X}_i$. These samples form an approximation of p_X^i, from which samples of likely observations for robot i can be drawn using $p_{Y_i|X}$. Let n_y be a multiple of n_x representing the number of drawn sample observations. For each $\tilde{x} \in \mathbb{X}_i[k]$, the robot draws n_y/n_x sample observations from $p_{Y_i|X}(\cdot|\tilde{x} : c_i)$ to generate a temporary observation sample set. The final observation sample set, $\mathbb{Y}_i[k]$, is formed by taking a random permutation of the temporary set, and the corresponding measurement probabilities become

$$p_{\mathbb{Y}_i|X}(\tilde{y}_i|x : c_i) = \frac{p_{Y_i|X}(\tilde{y}_i|x:c_i)}{\sum_{\tilde{y}_i' \in \mathbb{Y}_i} p_{Y_i|X}(\tilde{y}_i'|x:c_i)}$$

for all $x \in \mathcal{X}$ and $\tilde{y}_i \in \mathbb{Y}_i$, where $\mathbb{Y}_i$ is shorthand representing $\mathbb{Y}_i[k]$.

With these sample sets each robot can form a belief matrix, $\tilde{\pi}_{k'}^i[k] \in \{[0,1]\}^{|\mathcal{X}| \times n_y}$, that is of constant size with respect to the number of robots and sensor measurements. Setting $[\tilde{\pi}_0^i]_{j\ell}$ to be $p_{\mathbb{Y}_i|X}(\tilde{y}_i^\ell|x^j : c_i)$ for all $x^j \in \mathcal{X}$ and $\tilde{y}_i^\ell \in \mathbb{Y}_i$, where $\tilde{\pi}_{k'}^i$ is shorthand representing $\tilde{\pi}_{k'}^i[k]$, and $\tilde{y}_i^\ell$ denotes

the ℓth element of $\mathbb{Y}_i$, we can use the consensus algorithm to have $\tilde{\pi}_{n_\pi}^i$ converge to $\prod_{i=1}^{n_r} p_{\mathbb{Y}_i|X}(\tilde{y}_i^\ell|x^j : c_i)|_{t=kT_s}$ in the limit as $n_\pi \to \infty$. Since $\mathbb{Y}_i$ is randomly ordered for all robots, $\tilde{\pi}_{n_\pi}^i$ can be used to form a *sampled* approximation for the joint measurement probabilities,

$$\tilde{p}_{\mathbb{Y}|X}^i(\tilde{y}^\ell|x^j) = [\tilde{\pi}^i]_{j\ell}^{\beta_i} \Big/ \sum_{m=1}^{n_y} [\tilde{\pi}^i]_{jm}^{\beta_i} \tag{14}$$

for all $x^j \in \mathcal{X}$ and $\tilde{y}^\ell \in \mathbb{Y}$, where $\tilde{\pi}^i$ and $\mathbb{Y}$ are shorthand representing $\tilde{\pi}_{n_\pi}^i$ and $\mathbb{Y}[k]$, respectively, and $\tilde{y}^\ell$ denotes the ℓth element of the joint observation sample set $\mathbb{Y}$. Note that it is not necessary for the robots to actually know the entries of $\mathbb{Y}$, but instead what is important is that $\tilde{p}_{\mathbb{Y}|X}^i$ was formed from samples representing Y. In addition, the posterior approximation with respect to the robot's sampled state set becomes

$$\tilde{p}_{\mathbb{X}_i|\mathbb{Y}}^i(\tilde{x}|\tilde{y}) = \frac{\tilde{p}_{\mathbb{Y}|X}^i(\tilde{y}|\tilde{x})}{\sum_{\tilde{x}' \in \mathbb{X}_i} \tilde{p}_{\mathbb{Y}|X}^i(\tilde{y}|\tilde{x}')} \tag{15}$$

for all $\tilde{x} \in \mathbb{X}_i$ and $\tilde{y} \in \mathbb{Y}$, where $\mathbb{X}_i$ is shorthand representing $\mathbb{X}_i[k]$.

B. Scalability with Respect to Environment Size

The above sampling technique reduces the column dimension of the measurement probability matrix from exponential to constant with respect to the number of robots and sensor measurements. Unfortunately, since the priors are being represented in full, $\tilde{\pi}^i$ has a row dimension of $\max_m \mathcal{O}(|\mathcal{X}_m|^{n_w})$. We are currently investigating approximations (e.g. particle filters) to reduce the row dimension, but for this paper we consider a special class of robots that allows for the lossless compression of $\tilde{\pi}^i$.

Let each robot receive an observation that is modeled as an n_w-tuple random variable, $Y_i = (Y_{i1}, \ldots, Y_{in_w})$. In addition, the measurement probabilities of Y_{im} given X are only dependent on X_m and conditionally independent from all other $Y_{im'}$ with $m' \neq m$, such that

$$p_{Y_i|X}(y_i|x : c_i) = \prod_{m=1}^{n_w} p_{Y_{im}|X_m}(y_{im}|x_m : c_i) \tag{16}$$

for all $x \in \mathcal{X}$ and $y_i \in \mathcal{Y}_i$, where x_m and y_{im} are the mth element of the n_w-tuples x and y_i, respectively. In words, each robot's observation is composed of n_w conditionally independent observations, where each observation element concerns a specific environment partition cell. Due to this conditional independence, $\tilde{\pi}^i$ does not need a designated row for every $x \in \mathcal{X}$. Instead, each row can represent a single x_m for all $x_m \in \mathcal{X}_m$ and $m \in \{1, \ldots, n_w\}$. From (16), we can restructure $\tilde{\pi}_{k'}^i$ to form $\breve{\pi}_{k'}^i[k]$ such that

$$\left[\tilde{\pi}_{k'}^i\right]_{j\ell} = \prod_{m \in \mathcal{M}(x^j)} \left[\breve{\pi}_{k'}^i\right]_{m\ell} \tag{17}$$

for all $k' \in \{0, \ldots, n_\pi\}$, where $\breve{\pi}_{k'}^i$ is shorthand representing $\breve{\pi}_{k'}^i[k]$ and $\mathcal{M}(x^j)$ denotes the n_w element set of row indices

of $\breve{\pi}_0^i$ that were set to $p_{Y_{im}|X_j}(y_{im}|x_m : c_i)$ for all $x^j \in \mathcal{X}$, $y_i \in \mathcal{Y}_i$ and $m \in \{1, \ldots, n_w\}$.

Note that this technique is not an approximation, as we can perfectly reconstruct $\tilde{\pi}^i$ from $\breve{\pi}^i$. However, it has a significant impact by reducing the size of the measurement probability matrix from exponential to $\max_m \mathcal{O}(|\mathcal{X}_m|n_w)$ (i.e. linear) with respect to the number of environment discretization cells, which directly affects memory and network bandwidth requirements.

V. DISTRIBUTED COORDINATION

We first formalize a dynamic model for the robots then describe the two consensus rounds performed between observations. Finally, we introduce a distributed controller that, under certain conditions, moves the robots to a locally optimal configuration with respect to the utility function.

A. Dynamic Model

Let a robot of configuration c_i move in a configuration space, $\mathcal{C}_i \subset \mathbb{R}^{r_i} \times \mathbb{S}^{s_i}$. This space describes both the position of the robot and the orientation of its sensors, and does not need to be the same as $\mathcal{Q}$. For example, if we have a planar environment within $\mathbb{R}^2$, we could have ground point robots with omnidirectional sensors moving in $\mathbb{R}^2$ or flying robots with a gimbaled sensor moving in $\mathbb{R}^3 \times \mathbb{S}^3$. Let $\mathcal{C} = \prod_{i=1}^{n_r} \mathcal{C}_i$ denote the configuration space for all robots.

At any given time, the robot can choose a control action $u_i(t) \in \mathcal{U}_i \subset \mathbb{R}^{r_i} \times \mathbb{S}^{s_i}$. We model the robot as having continuous-time integrator dynamics,

$$\frac{dc_i}{dt} = u_i(t) \qquad (18)$$

which is a common assumption in the multi-robot coordination literature. In our applications using quad-rotor flying robots, 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.

B. The Two Consensus Rounds

We introduced the consensus algorithm in Section III-B as consisting of a single consensus round[2] performed between observations. For the complete distributed inference and coordination algorithm, we perform two consensus rounds between observations; the first round is to calculate the posterior based on the received observation, and the second to calculate the utility gradient to generate control inputs (18).

Consider the time of kT_s when each robot receives an observation, y_i^ℓ. Let each robot form a belief vector, $\breve{\pi}_0^{i,y}$, which is the ℓth column of $\breve{\pi}_0^i$. Using the consensus algorithm, each

robot can transmit its belief vector and receive its neighbors' ones a total of $n_\pi = n_\pi^y$ times. Since only a single observation is considered, we have

$$[\pi_{k'}^i]_{j\ell} = \prod_{m \in \mathcal{M}(x^j)} \left[\breve{\pi}_{k'}^{i,y}\right]_m \qquad (19)$$

Thus, the robots can use (11), (12), and (7) to calculate $p_{X[k+1]}^i$. Note that the belief vector is of size $\max_m \mathcal{O}(|\mathcal{X}_m|n_w)$, which is much smaller than a belief matrix representing the sampled observation set. In addition, knowledge of what actual observations other robots received are not needed for the Bayesian estimation since $[\pi^i]_{j\ell}$ naturally converges to $p_{Y|X}(y^\ell|x^j : c)$ for all $x^j \in \mathcal{X}$ in the limit as $n_\pi^y \to \infty$.

Revisiting our control objective from Section II, we wish to move the robots into a configuration that minimizes the expected uncertainty of the inference after receiving the next observation. With respect to the utility function, our objective is equivalent to solving the constrained optimization problem $\max_{c \in \mathcal{C}} U$. One technique to find a locally optimal solution is to have the robots calculate their utility gradient and move in a valid direction of increasing utility. We now use a second consensus round to allow for a distributed approximation of this gradient.

Let each robot form a belief matrix, $\breve{\pi}_0^{i,u} = \breve{\pi}_0^i[k+1]$. Note that since the observation at time kT_s has already been received and the prior updated, the belief matrix represents the one-step look ahead measurement probabilities. Again using the consensus algorithm with $n_\pi = n_\pi^u$, each robot can calculate its *sampled* utility gradient from (13) by

$$\frac{\partial \tilde{U}}{\partial c_i} = \frac{1}{n_x} \sum_{\tilde{y} \in \mathbb{Y}} \sum_{\tilde{x} \in \mathbb{X}_i} \frac{\partial p_{\mathbb{Y}_i|X}(\tilde{y}_i|\tilde{x}:c_i)}{\partial c_i} \times$$
$$\frac{\tilde{p}_{\mathbb{Y}|X}(\tilde{y}|\tilde{x})}{p_{\mathbb{Y}_i|X}(\tilde{y}_i|\tilde{x}:c_i)\big|_{t=kT_s}} \log\left(\tilde{p}_{\mathbb{X}|\mathbb{Y}}^i(\tilde{x}|\tilde{y})n_x\right) \qquad (20)$$

where the measurement probabilities (14) and (15) are approximated using $\breve{\pi}^{i,u}$ with (17).

C. Distributed Controller

We presented a fully distributed algorithm that enables each robot to calculate its utility gradient in $\mathcal{O}(n_x n_y)$ time. In other words, a calculation that normally requires global knowledge and $\max_{im} \mathcal{O}(|\mathcal{Y}_i|^{n_r}|\mathcal{X}_m|^{n_w})$ time can now be performed locally on a non-complete communication graph with computational complexity remaining *constant* as the number of robots grows. Using this algorithm, we can formulate distributed controllers with provable convergence for special classes of systems, such as the following.

Theorem 3. *Consider a system of n_r robots moving in the same configuration space, $\mathcal{C}_i = \mathbb{R}^{r_q} \times \mathbb{S}^{s_q}$, and sensing a bounded environment $\mathcal{Q}$ that is a subset of $\mathcal{C}_i$. Let the robots have dynamics (18) and communications described in Section III-A. Consider the class of systems where for all robots*

$\frac{\partial p_{Y_i|X}}{\partial c_i}$ *is continuous on $\mathcal{C}_i$ and equal to zero for all $c_i' \in \mathcal{C}_i$ such that $d_i(c_i', q) \geq D$, where D is some constant, q is any element in $\mathcal{Q}$, and $d_i : \mathcal{C}_i \times \mathcal{Q} \to \mathbb{R}$ is a valid distance function. Then for the controller $u_i(t) = \gamma_i \frac{\partial \tilde{U}}{\partial c_i}$, where $\gamma_i > 0$, we have that between the consensus updates of $\frac{\partial \tilde{U}}{\partial c_i}$, $u_i(t)$ is convergent to 0 for all $i \in \{1, \dots, n_r\}$.*

Proof. *Please refer to [9] for a proof.*

Remark 3. *In words, the assumption on $\partial p_{Y_i|X}/\partial c_i$ says that the measurement probabilities of the observation do not change when the robot is more than a certain distance away from the environment (e.g. sensors of limited range), and within this distance the change in measurement probabilities are continuous with robot position.*

Corollary 1. *Consider the aforementioned system in the limit as $n_\pi^u \to \infty$, or when $n_\pi^u \geq 1$ with a complete network graph. Let there exist a time after which no observations are received, and let the robots share a common prior. If the robots believe the environment to be static and use the unsampled controller $u_i^*(t) = \gamma_i \frac{\partial U}{\partial c_i}$, where $\gamma_i > 0$, then we have that if the robots converge to a configuration that is Lyapunov stable, then the configuration must also be locally optimal for the constrained optimization problem $\max_{c \in \mathcal{C}} U$.*

Proof. *Please refer to [9] for a proof.*

We now present with Algorithm 1 our complete algorithm for distributed inference and coordination.

Algorithm 1 Distributed Inference and Coordination

Require: Robot i knows its configuration c_i, its distribution $p_{Y_i|X}(y_i|x : c_i)$, and the extent of the environment $\mathcal{Q}$.
  loop
    Receive observation and create $\breve{\pi}_0^{i,y}$ from (17).
    Run consensus (9) with $n_\pi = n_\pi^y$ to form $\breve{\pi}^{i,y}$.
    Run Bayesian estimation from (11), (12), and (7).
    Draw state and observation sample sets to create $\breve{\pi}_0^{i,u}$.
    Run consensus (9) with $n_\pi = n_\pi^u$ to form $\breve{\pi}^{i,u}$.
    Calculate gradient (20) for the controller $u(t) = \gamma_i \frac{\partial \tilde{U}}{\partial c_i}$.
  end loop

VI. EXPERIMENTS AND SIMULATIONS

The objective for our hardware experiments and numerical simulations was to infer the state of a bounded, planar environment by deploying omnidirectional robots belonging to the class of systems described in Theorem 3. The environment (see Figure 2) was discretized into $n_w = 10$ hexagon cells, each being of width 2 m and having a static state of $x_m \in \{0, 1\}$. In addition, the robots' observations took values of $y_{im} \in \{0, 1\}$, and an ideal disk model was used to determine network connectivity.

Our hardware experiments used $n_r = 5$ Ascending Technology Hummingbird quad-rotor flying robots in a laboratory
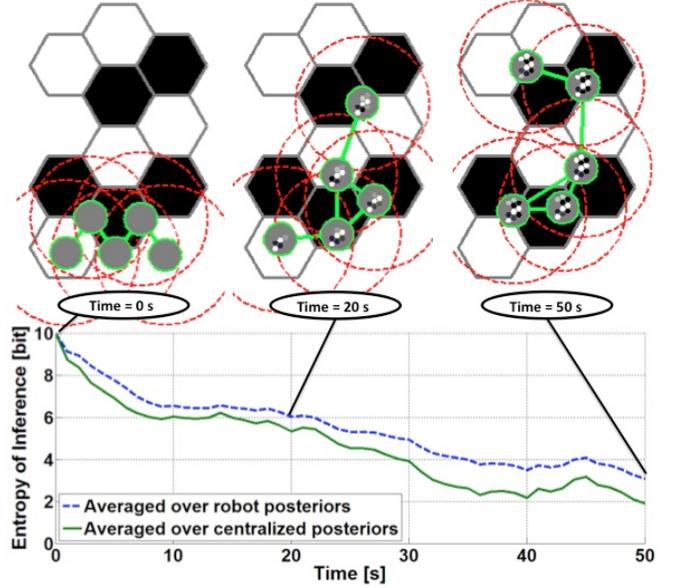

Fig. 2. The three illustrations show the beginning, middle, and end configuration of a five robot experiment over a 10 cell environment, where the state of each cell is either 0 (black) or 1 (white). The robots are represented by the gray circles, within which their prior distributions can be visualized. The green lines represent network connectivity, and the dashed red circles represent simulated sensor ranges. The plot shows the decrease in entropy of the inferences averaged over 10 consecutive runs.

equipped with a Vicon motion capture system. The realtime software for each robot ran in distributed fashion on a single computer and included a low level linear-quadratic regulator position controller that accepted waypoint inputs from (18) and sent low level control commands to the robots via 2.4 Ghz Xbee-Pro wireless modules. Five heterogeneous sensors were simulated with measurement probabilities $p_{Y_{im}|X_m}$ that were a maximum value of $(0.95, 0.90, 0.85, 0.80, 0.75)$ and decreased quadratically (e.g. power decay of light) to 0.5 at a sensor radius of $(2.0, 2.1, 2.2, 2.3, 2.4)$ m. We emphasized these properties by setting the hovering heights proportional to the sensor radii. 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 $\mathcal{U}_i = \{[-0.2 \text{ m/s}, 0.2 \text{ m/s}]\}^2$, control gain of $\gamma_i = 1000$, and a network radius that was continuously adjusted to be the smallest value such that the graph remained connected. In addition, a safety radius of 1 m was enforced between neighboring robots, meaning the gradient projection of $u_i(t)$ would be taken to prevent two directly communicating robots from moving closer than 1 m from each other. Sample sizes of $n_x = 200$ and $n_y = 800$ were used with consensus round sizes of $n_\pi^y = n_\pi^u = 3$, which allowed for a sampling interval of $T_s = 1$ s.

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 2 shows the beginning, middle, and end configura-

tion of a typical 50 s run, along with a plot showing the decrease in average entropy (i.e. uncertainty) of the robots' inferences compared to a centralized one. The centralized inference considered observations from all robots, and can be interpreted as a baseline. On average the entropy of the robots' inferences were within 0.8 bits of the centralized one over the 50 s. To date, we have over 50 successful runs with various starting positions and algorithm parameters, compared to 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.
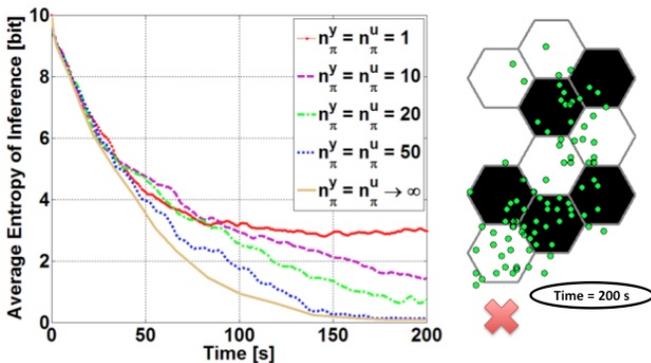

Fig. 3. This plot shows the average entropy of the robots' inferences for the 100 robot simulations. The top four curves illustrate how the consensus round sizes affect overall uncertainty within the system, and the bottom curve is an estimate representing the entropy in the limit as $n_\pi^y, n_\pi^u \to \infty$. The illustration to the right shows the location where all robots were deployed (red $\times$) and the final robot configurations after a typical run (green $\circ$).

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 of n_π^y and n_π^u. For each run, the heterogeneous sensors were randomly selected from the sensor set used in the hardware experiments, and the robots were deployed from a single location outside the environment (see Figure 3). To emulate a physically larger environment for the simulation, no safety radius was used and the network radius was fixed to 2.0 m - roughly half the average value seen to keep the network connected in the hardware experiments.

Using the same software developed for the hardware experiments, we verified that the increase in runtime for the simulation scaled appropriately. Figure 3 shows the decrease in the average entropy of the robots' inferences over 100 Monte-Carlo simulations. As expected, larger consensus round sizes resulted in lower overall uncertainty within the system. 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 gives insight into fundamental limitations that cannot be overcome by simply deploying more robots.

VII. CONCLUSION

We presented an information theoretic approach to distributed robot coordination that scales well with respect to the number of robots, sensor measurements, and environment discretization cells. The resulting controller is proven to be convergent for a special class of systems, and its performance is demonstrated in a five quad-rotor flying robot experiment and 100 robot numerical simulation. Due to the generality of our problem formulation, we are pursuing many extensions to address problems in distributed robotics concerning inference tasks. We have begun investigating more efficient sequential Monte-Carlo methods, and are currently equipping our robots with on-board computers to physically realize the decentralized algorithms.

REFERENCES

[1] F. Bourgault, A. Makarenko, S. B. Williams, B. Grocholsky, and H. F. Durrant-Whyte. Information based adaptive robotic exploration. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, pages 540–545, 2002.

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

[3] H. L. Choi and J. P. How. Continuous trajectory planning of mobile sensors for informative forecasting. *Automatica*, 2011. Accepted.

[4] T. H. Chung, V. Gupta, J. W. Burdick, and R. M. Murray. On a decentralized active sensing strategy using mobile sensor platforms in a network. In *Proceedings of the IEEE Conference on Decision and Control*, volume 2, pages 1914–1919, 2004.

[5] J. Cortés. Distributed kriged kalman filter for spatial estimation. *IEEE Transactions on Automatic Control*, 54(12):2816–2827, 2009.

[6] B. Grocholsky. *Information-theoretic control of multiple sensor platforms*. PhD thesis, University of Sydney, 2002.

[7] B. Grocholsky, A. Makarenko, and H. Durrant-Whyte. Information-theoretic control of multiple sensor platforms. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 1, pages 1521–1526, September 2003.

[8] G. M. Hoffmann and C. J. Tomlin. Mobile sensor network control using mutual information methods and particle filters. *IEEE Transactions on Automatic Control*, 55(1):32–47, January 2010.

[9] B. J. Julian, M. Angermann, M. Schwager, and D. Rus. A scalable information theoretic approach to distributed robot coordination. Technical Report MIT-CSAIL-TR-2011-034, Massachusetts Institute of Technology, July 2011.

[10] K. M. Lynch, I. B. Schwartz, P. Yang, and R. A. Freeman. Decentralized environmental modeling by mobile sensor networks. *IEEE Transactions on Robotics*, 24(3):710–724, June 2008.

[11] J. L. Ny and G. J. Pappas. 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*, pages 6282–6292, December 2009.

[12] R. Olfati-Saber, E. Franco, E. Frazzoli, and J. S. Shamma. Belief consensus and distributed hypothesis testing in sensor networks. In *Proceedings of the Network Embedded Sensing and Control Workshop*, pages 169–182, 2006.

[13] A. Singh, A. Krause, C. Guestrin, W. Kaiser, and M. Batalin. Efficient planning of informative paths for multiple robots. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2007.