

Decentralized, Adaptive Control for Coverage with Networked Robots

Mac Schwager^{*}, Jean-Jacques Slotine[†], and Daniela Rus^{*}

^{*}Computer Science and Artificial Intelligence Lab
MIT, Cambridge, MA, 02139

Email: schwager@mit.edu, rus@csail.mit.edu

[†]Nonlinear Systems Lab
MIT, Cambridge, MA, 02139

Email: jjs@mit.edu

Abstract—A decentralized, adaptive control law is presented to drive a network of mobile robots to a near-optimal sensing configuration. The control law is adaptive in that it integrates sensor measurements to provide a converging estimate of the distribution of sensory information in the environment. It is decentralized in that it requires only information local to each robot. A Lyapunov-type proof is used to show that the control law causes the network to converge to a near-optimal sensing configuration, and the controller is demonstrated in numerical simulations. This technique suggests a broader application of adaptive control methodologies to decentralized control problems in unknown dynamical environments.

I. INTRODUCTION

In this paper we present a control strategy which is both adaptive and decentralized, thereby combining two of the defining qualities of biological swarming, flocking, and herding systems. More importantly, the adaptive, decentralized control law has provable stability and convergence properties, which are summarized in the main result. This work describes one example of the successful combination of these two disciplines, and, it is hoped, will provide a method that can be applied to other problems requiring control with local information in uncertain dynamical environments.

The specific problem we address is coverage control for mobile sensor networks. We consider controlling a group of mobile robots to monitor some quantity of interest over an area. Our solution to this problem would be useful in controlling teams of robots to carry out a number of tasks including search and rescue missions, environmental monitoring (e.g. for forest fires), automatic surveillance of rooms, buildings, or towns, or simulating collaborative predatory behavior. Virtually any application in which a group of automated mobile agents is required to monitor an area could benefit from the proposed control law. We are constrained to use only local sensory and locational information available to each robot. Our control law causes the robots to spread over an area in such a way that their density in different regions of the area is directly related to the sensory importance of those regions. Thus areas of greater importance receive more concentrated coverage than areas that are less important.

This work was supported in part by the MURI SWARM grant and NSF grant number IIS 0513755.

What is more, the robots do not know before hand what are the regions of sensory interest, but acquire this information dynamically.

A. Relation to Previous Work

There has been considerable activity in the controls and robotics community in studying decentralized control as a metaphor for natural and engineered swarming, flocking, and herding systems. A far from exhaustive selection of such works can be found in [1]–[5], where consensus problems are treated, [6], which deals with the synchronization of oscillators, [7], [8], which deals with stability, and [9]–[11], which considers abstractions for representing decentralized dynamical systems. These works are formulated assuming local information to some degree, however, models of the dynamics of the robots and their surroundings are usually assumed to be exact. Conversely the discipline of adaptive control [12]–[14] focuses on controlling systems whose dynamics are not known to the controller. Controllers collect information as the system evolves, identifying system dynamics simultaneously while controlling the system. This work combines these two disciplines to produce a controller that is provably stable, uses only local information, and has limited prior knowledge of the sensing environment.

Various strategies have been introduced to address the specific problem of coverage control for mobile sensor networks, and our work builds on several important results in this category. In [15], mobile sensing agents are controlled using potential functions for inter-agent interactions. Stability results are derived, but the optimality of the network configuration is not addressed. Similarly, in [16] an algorithm is proposed that allows for agents to concentrate in areas of high event density while maintaining area coverage constraints. The algorithm is proved to maintain sensor coverage for a limited case without addressing optimality. Most relevant to this paper is a body of results reported in [17]–[19]. In this work, Cortés et al derived decentralized control laws for positioning mobile sensor networks optimally with respect to a known event probability density. This approach is advantageous because it guarantees that the network (locally) minimizes a cost function relevant to the coverage problem. However, the control strategy requires that each agent have a complete

foreknowledge of the event probability density, thus it is not reactive to the sensed environment. In our previous work, [20], we removed the constraint on the global knowledge of the event density. We presented a control law based on a linear estimation of a sensory function and showed that the controller performed robustly in simulation and in hardware experiments. However, analytical stability guarantees were not obtained.

In this work, a new control law is proposed which implements an estimation of the sensory distribution by integrating measurements gathered throughout a robot's trajectory into a parametric model of the sensory distribution. We use the locational optimization framework introduced in [18], and provide a control law that does not require foreknowledge of the distribution of sensory information in the environment, in the spirit of [20]. Using a Lyapunov-like proof, we show that the control law causes the network to converge to a near-optimal sensing configuration given that the actual sensory distribution is adequately represented by a parametric model. Furthermore, we require that each agent can sense the value of the sensory distribution at its position—a relaxation of the conditions in [20], in which both the value and gradient were required. We provide some background on the results of locational optimization in Section II. We present the controller and prove its stability in Section III. The results of numerical simulations are described in IV.

II. LOCATIONAL OPTIMIZATION BACKGROUND

In this section, we state the basic definitions and results from locational optimization that will be useful in this work. More thorough discussions can be found in [18], [20].

Let there be n robots in a known, convex polytope $Q \subset \mathbb{R}^N$. An arbitrary point in Q is denoted q , the position of the i^{th} robot is denoted p_i , and the set of all robot positions is denoted $P = \{p_1, \dots, p_n\}$. Let $\mathcal{V} = \{V_1, \dots, V_n\}$ be the Voronoi partition of Q , for which the robot positions are the generator points. Specifically,

$$V_i = \{q \in Q \mid \|q - p_i\| \leq \|q - p_j\|, \forall j \neq i\}.$$

Define the sensory function, $\phi(q)$, as a scalar function, $\phi : Q \mapsto \mathbb{R}_+$. The function $\phi(q)$ is not known by the robots in the network, but the robots are equipped with sensors from which a measurement of $\phi(p_i)$ can be derived at the robot's position p_i .

Let the *unreliability* of the sensor measurement be defined by a quadratic function $f(\|q - p_i\|) = \frac{1}{2}\|q - p_i\|^2$. Specifically, $f(\|q - p_i\|)$ describes how unreliable is the measurement of the information at q by a sensor at p_i (henceforth, $\|\cdot\|$ is used to denote the ℓ^2 -norm). This form of $f(\|q - p_i\|)$ is physically appealing since it is reasonable that sensing will become more unreliable farther from the sensor.

We can formulate the cost incurred by the network sensing over the region Q as

$$\mathcal{H}(P) = \sum_{i=1}^n \int_{V_i} \frac{1}{2} \|q - p_i\|^2 \phi(q) dq. \quad (1)$$

Notice that unreliable sensing is expensive and high values of $\phi(q)$ are also expensive. An optimal network configuration corresponds to a set of robot positions P that minimize (1).

Next we define three properties analogous to mass-moments of rigid bodies. The mass of V_i is defined as

$$M_{V_i} = \int_{V_i} \phi(q) dq, \quad (2)$$

the first mass-moment (not normalized) is defined as

$$L_{V_i} = \int_{V_i} q \phi(q) dq \quad (3)$$

and the centroid of V_i is defined as

$$C_{V_i} = \frac{L_{V_i}}{M_{V_i}}, \quad (4)$$

Note that $\phi(q)$ strictly positive imply both $M_{V_i} > 0 \forall V_i \neq \{\emptyset\}$ and $C_{V_i} \in V_i \setminus \partial V_i$ (C_{V_i} is in the interior of V_i). Thus M_{V_i} and C_{V_i} have properties intrinsic to physical masses and centroids. A standard result in locational optimization is that

$$\frac{\partial \mathcal{H}}{\partial p_i} = -M_{V_i}(C_{V_i} - p_i). \quad (5)$$

Equation (5) implies that local minima of \mathcal{H} correspond to the configurations such that $p_i = C_{V_i} \forall i$, that is, each agent is located at the centroid of its Voronoi region. Thus, the optimal coverage task is to drive the group of robots to what is called a centroidal Voronoi configuration.

III. DECENTRALIZED ADAPTIVE CONTROL

We will design a control law with an intuitive interpretation and prove that it causes the network to converge to a near-centroidal Voronoi configuration. The control law will integrate sensory measurements available to each robot to form an on-line approximation of the centroids of its Voronoi regions.

Let the dynamics of each robot be modeled by the first-order equation

$$\dot{p}_i = u_i, \quad (6)$$

where u_i is the control input. Since this work is primarily concerned with the application of adaptive control to the decentralized coverage problem, simple dynamics were chosen so as not to obscure the result. More complicated dynamics can be accommodated.

Assume, furthermore, that the sensory function $\phi(q)$ can be parameterized as an unknown linear combination of a set of known basis functions. This requirement is formalized in the following two assumptions.

Assumption 1 (Matching Conditions): $\exists a \in \mathbb{R}_+^m$ and $\mathcal{K} : Q \mapsto \mathbb{R}_+^m$, such that

$$\phi(q) = \mathcal{K}^T(q)a, \quad (7)$$

where the vector of basis functions \mathcal{K} is available to each agent, but the parameter vector a is unknown.

Assumption 2 (Lower Bound):

$$a(j) \geq \beta \quad \forall j = 1, \dots, m, \quad (8)$$

where $a(j)$ denotes the j^{th} element of the vector a , and $\beta > 0$ is a known real bound. This assumption implies a lower bound on $\phi(q)$ over Q .

We now introduce a number of definitions which will be important in stating and proving the main stability result. Let $\hat{a}_i(t)$ be robot i 's approximation of the parameter vector. Naturally, $\hat{\phi}_i = \mathcal{K}^T(q)\hat{a}_i$ is robot i 's approximation of $\phi(q)$. Define the mass moment approximations

$$\hat{M}_{V_i} = \int_{V_i} \hat{\phi}_i dq, \quad (9)$$

$$\hat{L}_{V_i} = \int_{V_i} q\hat{\phi}_i dq, \text{ and} \quad (10)$$

$$\hat{C}_{V_i} = \frac{\hat{L}_{V_i}}{\hat{M}_{V_i}}, \quad (11)$$

which are analogous to (2), (3), and (4) respectively. Next, define the parameter error

$$\tilde{a}_i = \hat{a}_i - a_i, \quad (12)$$

and the mass moment errors

$$\tilde{M}_{V_i} = \int_{V_i} \mathcal{K}^T(q)\tilde{a}_i dq = \hat{M}_{V_i} - M_{V_i}, \quad (13)$$

$$\tilde{L}_{V_i} = \int_{V_i} q\mathcal{K}^T(q)\tilde{a}_i dq = \hat{L}_{V_i} - L_{V_i}. \quad (14)$$

However, notice that $\tilde{C}_{V_i} = \tilde{L}_{V_i}/\tilde{M}_{V_i} \neq \hat{C}_{V_i} - C_{V_i}$. From (11), (13), and (14) we find that

$$L_{V_i} = M_{V_i}\hat{C}_{V_i} + \tilde{M}_{V_i}(\hat{C}_{V_i} - \tilde{C}_{V_i}). \quad (15)$$

This property will be useful in what follows. We also define two error vectors, the *actual* error, $e_i = C_{V_i} - p_i$, and the *estimated* error $\hat{e}_i = \hat{C}_{V_i} - p_i$. Finally, in order to compress the notation somewhat, we use the shorthand $\mathcal{K}_i = \mathcal{K}(p_i(t))$ for the value of the basis function vector at the position of robot i , and $\phi_i = \phi(p_i(t))$ for the value of ϕ at the position of robot i . As stated previously, robot i can measure ϕ_i with its sensors.

We now propose to use the control law

$$u_i = k(\hat{C}_{V_i} - p_i), \quad (16)$$

where $k \in \mathbb{R}_+$ is a proportional control gain. The parameters \hat{a}_i used to calculate \hat{C}_{V_i} are adjusted according to a set of adaptation laws which are introduced below.

Firstly, we define two quantities, $\Lambda_i(t)$ and $s_i(t)$, according to the differential equations (with zero initial conditions)

$$\dot{\Lambda}_i = -\alpha\Lambda_i + \mathcal{K}_i\mathcal{K}_i^T, \quad (17)$$

and

$$\dot{s}_i = -\alpha s_i + \mathcal{K}_i\phi_i. \quad (18)$$

These quantities can be interpreted as integrals of the measurements taken by the agent i throughout its trajectory. The time constant α determines how much to discount old measurements $\phi_i(\tau)$, $0 \leq \tau \leq t$, and can be considered a "forgetting factor."

The adaptation law for \hat{a}_i is now defined as

$$b_i = - \int_{V_i} \mathcal{K}(q)(q - \hat{C}_{V_i})^T dq \hat{e}_i - \gamma(\Lambda_i \hat{a}_i - s_i), \quad (19)$$

$$\dot{\hat{a}}_i = \Gamma(b_i - I_{\beta_i} b_i) \quad (20)$$

where $\Gamma \in \mathbb{R}^{m \times m}$ is a diagonal, positive definite adaptation gain matrix, and $\gamma \in \mathbb{R}_+$ is an adaptation gain scalar. The term b_i is used as an intermediary placeholder to simplify the expression. The diagonal matrix I_{β_i} is defined element-wise as

$$I_{\beta_i}(j) = \begin{cases} 0 & \text{for } \hat{a}_i(j) > \beta \\ 0 & \text{for } \hat{a}_i(j) = \beta \text{ and } b_i(j) \geq 0 \\ 1 & \text{otherwise} \end{cases}, \quad (21)$$

where (j) denotes the j^{th} element for a vector and the j^{th} diagonal element for a matrix. Equations (20) and (21) implement a projection operation [14], [21] that prevents any element of \hat{a}_i from dropping below the lower bound β . This is done by forcing $\dot{\hat{a}}_i(j) = 0$ whenever $\hat{a}_i(j) = \beta$ and $b_i(j) < 0$. The projection is desirable for two reasons: 1) because the control law has a singularity at $\hat{a}_i = 0$, and 2) because we know from Assumption 2 that the true parameters are lower bounded by β .

The main result of this work is now stated as follows.

Theorem 1 (Convergence Theorem): Under Assumptions 1 and 2, for the system of agents with dynamics (6) and the control law (16), with initial conditions $\Lambda_i = 0$, $s_i = 0$, $\forall i$, and $\hat{a}_i(j) \geq \beta \forall i, j$,

$$i) \quad \lim_{t \rightarrow \infty} \hat{e}_i(t) = 0 \quad \forall i = 1, \dots, n \quad (22)$$

$$ii) \quad \lim_{t \rightarrow \infty} \mathcal{K}_i^T(\tau)\tilde{a}_i(t) = 0 \quad \forall 0 \leq \tau \leq t \text{ and } \forall i = 1, \dots, n. \quad (23)$$

Proof: We will define a lower-bounded function and show that it is non-increasing along the trajectories of the system, and that its time derivative is uniformly continuous. Theorem 1 is then an implication of Barbalat's lemma.

Let

$$V = \mathcal{H} + \sum_{i=1}^n \frac{1}{2} \tilde{a}_i^T k \Gamma^{-1} \tilde{a}_i. \quad (24)$$

Taking the time derivative of V along the trajectories of the system gives

$$\dot{V} = \sum_{i=1}^n \left[\frac{\partial \mathcal{H}^T}{\partial p_i} \dot{p}_i + \tilde{a}_i^T k \Gamma^{-1} \dot{\tilde{a}}_i \right],$$

and substituting from (5) and noticing that $\dot{\tilde{a}}_i = \dot{\hat{a}}_i$ yields

$$\dot{V} = \sum_{i=1}^n \left[(M_{V_i} p_i - M_{V_i} C_{V_i})^T \dot{p}_i + \tilde{a}_i^T k \Gamma^{-1} \dot{\hat{a}}_i \right].$$

Now we make use of the property in (15) to write

$$\dot{V} = \sum_{i=1}^n \left[(M_{V_i} p_i - M_{V_i} \hat{C}_{V_i})^T \dot{p}_i + \tilde{M}_{V_i} (\tilde{C}_{V_i} - \hat{C}_{V_i})^T \dot{p}_i + \tilde{a}_i^T k \Gamma^{-1} \dot{\hat{a}}_i \right].$$

Substituting for \dot{p}_i with (6) and (16), and simplifying with $\tilde{L}_{V_i} = \tilde{M}_{V_i}\tilde{C}_{V_i}$ leads to

$$\dot{V} = \sum_{i=1}^n [-M_{V_i}(\hat{C}_{V_i} - p_i)^T k(\hat{C}_{V_i} - p_i) + (\tilde{L}_{V_i} - \tilde{M}_{V_i}\hat{C}_{V_i})^T k(\hat{C}_{V_i} - p_i) + \tilde{a}_i^T k\Gamma^{-1}\hat{a}_i].$$

Simplifying further with $\hat{e}_i = \hat{C}_{V_i} - p_i$ and expanding \tilde{L} and \tilde{M} with (14) and (13) gives

$$\dot{V} = \sum_{i=1}^n [-M_{V_i}\hat{e}_i^T k\hat{e}_i + (\int_{V_i} \tilde{a}_i^T \mathcal{K} q^T dq - \int_{V_i} \tilde{a}_i^T \mathcal{K} \hat{C}_{V_i}^T dq) k\hat{e}_i + \tilde{a}_i^T k\Gamma^{-1}\hat{a}_i].$$

Collecting terms under the integral, and noticing the crucial fact that \tilde{a}_i is not a function of q we get

$$\dot{V} = \sum_{i=1}^n [-M_{V_i}\hat{e}_i^T k\hat{e}_i + \tilde{a}_i^T k \int_{V_i} \mathcal{K}(q - \hat{C}_{V_i})^T dq \hat{e}_i + \tilde{a}_i^T k\Gamma^{-1}\hat{a}_i].$$

Now we substitute for \hat{a}_i with (20) to get

$$\dot{V} = - \sum_{i=1}^n [M_{V_i}\hat{e}_i^T k\hat{e}_i + \tilde{a}_i^T k\gamma(\Lambda_i\hat{a}_i - s_i) + \tilde{a}_i^T kI_{\beta_i}b_i], \quad (25)$$

where the first term in the adaptive law (19) cancels the integral term. Notice from (17) and (18) that, with zero initial conditions,

$$\Lambda_i(t) = \int_0^t e^{-\alpha(t-\tau)} \mathcal{K}_i(\tau) \mathcal{K}_i^T(\tau) d\tau$$

and

$$s_i(t) = \int_0^t e^{-\alpha(t-\tau)} \mathcal{K}_i(\tau) \phi_i(\tau) d\tau.$$

Substituting these into (25), combining terms under the integral, and noting that $\hat{a}_i(t)$ is not a function of τ leads to

$$\dot{V} = - \sum_{i=1}^n [M_{V_i}\hat{e}_i^T k\hat{e}_i + \tilde{a}_i^T kI_{\beta_i}b_i + \tilde{a}_i^T k\gamma \int_0^t e^{-\alpha(t-\tau)} \mathcal{K}_i(\tau) (\mathcal{K}_i^T(\tau)\hat{a}_i(t) - \phi_i(\tau)) d\tau].$$

Using (7) and bringing the $\tilde{a}_i(t)$ inside the integral in the second term gives

$$\dot{V} = - \sum_{i=1}^n [M_{V_i}\hat{e}_i^T k\hat{e}_i + \tilde{a}_i^T kI_{\beta_i}b_i + k\gamma \int_0^t e^{-\alpha(t-\tau)} (\mathcal{K}_i^T(\tau)\tilde{a}_i(t))^2 d\tau]. \quad (26)$$

Inside the sum, the first and third terms are clearly non-negative. We focus momentarily on the second. Expanding it as a sum of scalar terms, we see that the j^{th} scalar term is of the form

$$k\tilde{a}_i(j)I_{\beta_i}(j)b_i(j). \quad (27)$$

From (21), if $\hat{a}_i(j) > \beta$, or $\hat{a}_i(j) = \beta$ and $b_i(j) \geq 0$, then $I_{\beta_i}(j) = 0$ and the term vanishes. Now, in the case $\hat{a}_i(j) = \beta$ and $b_i(j) < 0$, we have $\tilde{a}_i(j) = \hat{a}_i(j) - a(j) \leq 0$ (from Assumption 2). Furthermore, $I_{\beta_i}(j) = 1$ and $b_i(j) < 0$ implies that the term is non-negative. In all cases, then, each term of the form (27) is non-negative, and all three terms inside the sum in (26) are non-negative. Thus \dot{V} is negative semi-definite.

Also, the facts that u_i is continuous $\forall i$, V has continuous first partial derivatives, V is radially unbounded, and $\dot{V} \leq 0$ imply that \dot{V} is uniformly continuous, therefore, by Barbalat's lemma $\lim_{t \rightarrow \infty} \dot{V} = 0$, which directly implies (22) from Theorem 1, and

$$\lim_{t \rightarrow \infty} \int_0^t e^{-\alpha(t-\tau)} (\mathcal{K}_i^T(\tau)\tilde{a}_i(t))^2 d\tau = 0 \quad (28)$$

$\forall i = 1, \dots, n.$

Now notice that the integrand in (28) is non-negative, therefore it must converge to zero for all τ . Finally, $e^{-\alpha(t-\tau)} > 0 \forall 0 \leq \tau \leq t$ implies (23) from Theorem 1. ■

Remark 1: The term $\mathcal{K}_i^T(\tau)\tilde{a}_i(t)$ in (23) can be interpreted as the difference between any *previously measured* value of $\phi(q)$ and the *current* estimate, $\hat{\phi}_i(q)$. Informally, this assertion from Theorem 1 states that the estimate $\hat{\phi}_i(q)$ will converge asymptotically to all previously measured values of $\phi(q)$. This does not, however, imply that $\hat{\phi}_i(q) \rightarrow \phi(q) \forall q \in Q$, as this would require an extra persistent excitation condition.

Remark 2: The tunable parameters k , Γ , γ , and α can potentially have different values for each agent without affecting the stability proof. The less general from used here was chosen for clarity.

A. Practical Algorithm

A practical method for implementing the proposed control law on a network of robots is detailed in Algorithm 1. Notice that the control law in (16) and adaptation law in (20) both require the computation of integrals over V_i , thus robot i must be able to calculate continuously its Voronoi region. Several algorithms exist for computing V_i in a distributed fashion, for example those given in [18], [22].

Algorithm 1 Adaptive Coverage Control Algorithm

Require: Each robot can compute its Voronoi region

Require: $\phi(q)$ can be parameterized as in (7)

Require: $a(j)$ are lower bounded as in (8)

Initialize Λ_i , s_i to zero, and $\hat{a}_i(j)$ to β

loop

 Compute the robot's Voronoi region

 Compute \hat{C}_{V_i} according to (11)

 Update \hat{a}_i according to (20)

 Update Λ_i and s_i according to (17) and (18)

 Apply control input $u_i = -k(\hat{C}_{V_i} - p_i)$

end loop

Algorithm 1 is decentralized, fully distributed, and requires minimal communication between neighboring robots.

It can be used on teams of large robots, on teams of small robots such as [23], or on mobile sensor network nodes with limited computation and storage capabilities such as the mobile Mica Motes described by [24].

IV. NUMERICAL SIMULATIONS

A. Implementation

Simulations were carried out in a Matlab environment. The dynamics in (6) with the control law in (16), and the adaptation laws in (20), (17), and (18) for a group of $n = 20$ robots were modeled as a system of coupled differential equations. The Matlab numerical solver ode45 was used to integrate the equations of motion of the group of robots. The region Q was taken to be the unit square. The sensory function, $\phi(q)$, was parameterized as a Gaussian network with 9 Gaussians and a constant offset. In particular, for $\mathcal{K} = [1 \ \mathcal{K}(2) \ \cdots \ \mathcal{K}(10)]^T$, each component, $\mathcal{K}(j)$ for $2 \leq j \leq 10$, was implemented as

$$\mathcal{K}(j) = \frac{1}{\sigma_j \sqrt{2\pi}} e^{-\frac{(q-\mu_j)^2}{2\sigma_j^2}}, \quad (29)$$

where $\sigma_j = .18$. The unit square was divided into an even 3×3 grid and each μ_j was chosen so that one of the 9 Gaussian was centered at the middle of each grid square. The parameters were chosen as $a = [\beta \ 1000 \ \beta \ \cdots \ \beta \ 1000]^T$, with $\beta = .1$ so that only the lower left and upper right Gaussians contributed significantly to the value of $\phi(q)$, producing a bimodal distribution.

The robots in the network were started from random initial positions. Each robot used a copy of the Gaussian network described above for $\mathcal{K}(q)$. The estimated parameters \hat{a}_i for each robot were started at a value of β , and Λ_i and s_i were each started at zero. The gains used by the robots were $k = 3$, $\Gamma = 10^{-5} I_{10}$, $\gamma = 2 \times 10^7$ and $\alpha = 1$. In practice, the first integral term in the adaptive law (19) seems to have very little effect on the performance of the controller. Choosing Γ small and γ correspondingly large puts more weight on the second term, which is responsible for integrating measurements of $\phi(p_i)$ into the parameters. The spatial integrals in (9), (10), and (19) required for the control law were computed by discretizing each Voronoi region V_i into a 7×7 grid and summing contributions of the integrand over the grid. Voronoi regions were computed in a centralized fashion using the built-in Matlab function, although equivalent performance is observed with a custom decentralized Voronoi algorithm. The Matlab Voronoi command was used only for computational speed.

B. Results

Figure 1 shows the positions of the robots in the network over the course of a simulation run. The initial configuration of the network is shown in Figure 1(a), the trajectories of the agents (dashed lines) in Figure 1(b), and the final configuration in Figure 1(c). The centers of the two contributing Gaussian functions are marked with \times 's. The somewhat jagged trajectories are caused by the discrete nature of the spatial integration procedure required by the control law. The

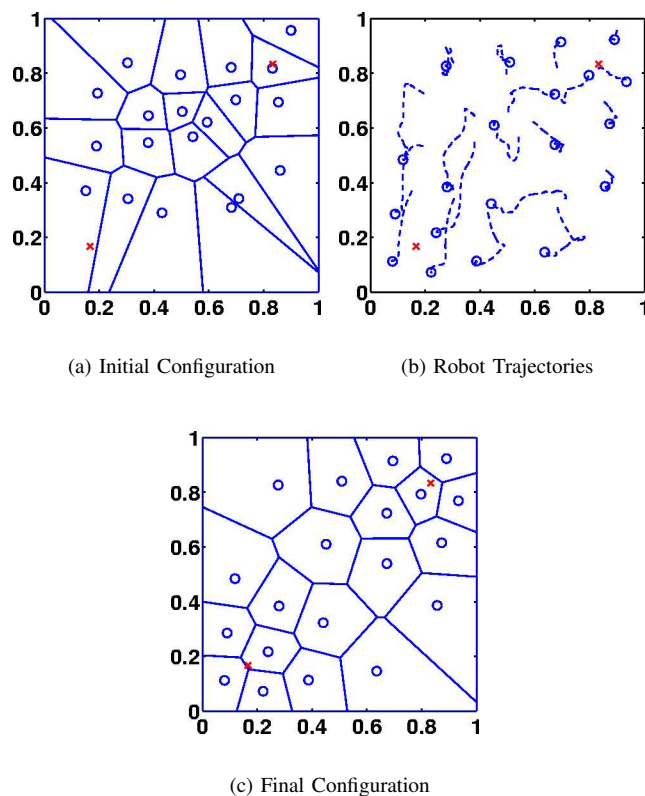


Fig. 1. The initial configuration of the network is shown in 1(a), the trajectories of the robots (dashed lines) in 1(b), and the final configuration in 1(c). The Gaussian centers of $\phi(q)$ are marked by the red \times 's.

jaggedness would be eliminated and overall computing speed would be greatly enhanced if the integral could be computed analytically as in [20]. Nonetheless, the performance of the control scheme is clearly demonstrated in the simulation.

In figure 2 the first assertion (22) of Theorem 1 is demonstrated for the same network of robots. The norm of the estimated error averaged over all the robots is shown to converge asymptotically to zero. This implies that the robots move to the estimated centroid of their Voronoi regions. We call such a configuration near-optimal. The jagged time history is again attributed to the discrete spatial integral computation which is a significant computational bottleneck.

The second assertion (23) of Theorem 1 is demonstrated in figure 3. The plot shows the integral in (28) as a function of time averaged over all the robots in the network converging asymptotically to zero. This implies that the parameters adjust in such a way that the estimate, $\hat{\phi}_i$, matches all previously measured values of $\phi(q)$. As stated previously, this does not imply that $\hat{a}_i \rightarrow a$.

V. CONCLUSION

In this work we proposed an adaptive, decentralized controller to drive a network of robots to a near-optimal sensing configuration. The controller was proven to cause the robots to move to the estimated centroids of their Voronoi regions, while also causing their estimate of the sensory

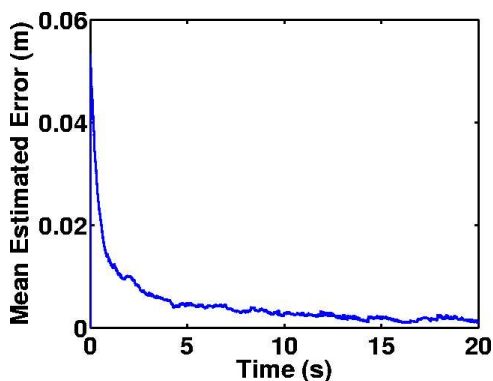


Fig. 2. The norm of the estimated error averaged over all the robots in the network is shown as a function of time for a network of 20 robots. The plot demonstrates the first assertion of Theorem 1.

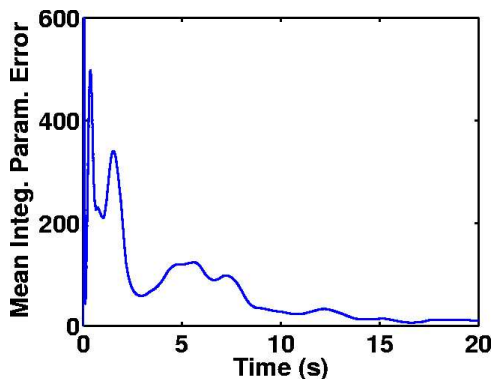


Fig. 3. The integrated $\phi(q)$ error averaged over all the robots is shown for the same network of 20 robots. The plot demonstrates the second assertion of Theorem 1.

distribution to improve over time until all previous sensor measurements fit the estimated sensory distribution. The control law was demonstrated in numerical simulations of a group of 20 robots sensing over an area with a bimodal Gaussian distribution of sensory information.

This work can be expanded along several avenues. One which is especially promising is to have robots share parameters with their neighbors through an agreement algorithm, along the lines of [1]. Such an algorithm will allow each robot to make use of the measurements of each other robot, improving the speed and robustness of the overall function approximation. Furthermore, we expect that the technique used in this paper will find broader application beyond the problem chosen here. For example, this control law can be directly applied as an algorithm for adaptive vector quantization. We hope that the approach will yield fruitful combinations of adaptive control and decentralized control to produce engineered agents that can cooperate with one another while gathering information from their environment to proceed toward a common goal.

REFERENCES

- [1] W. Wang and J. J. E. Slotine, "A theoretical study of different leader roles in networks," *IEEE Transactions on Automatic Control*, vol. 51, no. 7, pp. 1156–1161, July 2006.
- [2] T. Vicsek, A. Czirok, E. Ben-Jacob, I. Cohen, and O. Shochet, "Novel type of phase transition in a system of self-driven particles," *Physical Review Letters*, vol. 75, no. 6, pp. 1226–1229, August 1995.
- [3] V. D. Blondel, J. M. Hendrickx, A. Olshevsky, and J. N. Tsitsiklis, "Convergence in multiagent coordination, consensus, and flocking," in *Proceedings of the Joint IEEE Conference on Decision and Control and European Control Conference*, Seville, Spain, December 2005.
- [4] R. Olfati-Saber and R. R. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, September 2004.
- [5] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, June 2003.
- [6] W. Wang and J. J. E. Slotine, "On partial contraction analysis for coupled nonlinear oscillators," *Biological Cybernetics*, vol. 23, no. 1, pp. 38–53, December 2004.
- [7] V. Gazi and K. M. Passino, "Stability analysis of swarms," *IEEE Transaction on Automatic Control*, vol. 48, no. 4, pp. 692–697, April 2003.
- [8] H. G. Tanner, G. J. Pappas, and R. V. Kumar, "Leader-to-formation stability," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 3, pp. 443–455, June 2004.
- [9] C. Belta and V. Kumar, "Abstraction and control for groups of robots," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 5, pp. 865–875, October 2004.
- [10] G. F. Trecate, A. Buffa, and M. Gati, "Analysis of coordination in multi-agent systems through partial difference equations," *IEEE Transactions on Automatic Control*, vol. 51, no. 6, pp. 1058–1063, June 2006.
- [11] J. Lygeros, K. Johansson, S. N. Simic, J. Zhang, and S. S. Sastry, "Dynamical properties of hybrid automata," *IEEE Transactions on Automatic Control*, vol. 48, no. 1, pp. 2–17, January 2003.
- [12] J.-J. E. Slotine and W. Li, *Applied Nonlinear Control*. Upper Saddle River, NJ: Prentice-Hall, 1991.
- [13] K. S. Narendra and A. M. Annaswamy, *Stable Adaptive Systems*. Englewood Cliffs, NJ: Prentice-Hall, 1989.
- [14] P. A. Ioannou and J. Sun, *Robust Adaptive Control*. Englewood Cliffs, NJ: Prentice-Hall, 1996.
- [15] P. Ogren, E. Fiorelli, and N. E. Leonard, "Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment," *IEEE Transactions on Automatic Control*, vol. 49, no. 8, pp. 1292–1302, August 2004.
- [16] Z. Butler and D. Rus, "Controlling mobile sensors for monitoring events with coverage constraints," in *Proceedings of IEEE International Conference of Robotics and Automation*, New Orleans, LA, April 2004, pp. 1563–1573.
- [17] F. Bullo and J. Cortés, "Adaptive and distributed coordination algorithms for mobile sensing networks," in *Cooperative Control. (Proceedings of the 2003 Block Island Workshop on Cooperative Control)*, ser. Lecture Notes in Control and Information Sciences, V. Kumar, N. E. Leonard, and A. S. Morse, Eds. New York: Springer Verlag, 2005, vol. 309, pp. 43–62.
- [18] J. Cortés, S. Martínez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, pp. 243–255, April 2004.
- [19] J. Cortés, S. Martínez, and F. Bullo, "Spatially-distributed coverage optimization and control with limited-range interactions," *ESIAM: Control, Optimisation and Calculus of Variations*, vol. 11, pp. 691–719, 2005.
- [20] M. Schwager, J. McLurkin, and D. Rus, "Distributed coverage control with sensory feedback for networked robots," in *Proceedings of Robotics: Science and Systems*, Philadelphia, PA, August 2006.
- [21] J. Slotine and J. Coetsee, "Adaptive sliding controller synthesis for nonlinear systems," *International Journal of Control*, vol. 43, no. 4, 1986.
- [22] Q. Li and D. Rus, "Navigation protocols in sensor networks," *ACM Transactions on Sensor Networks*, vol. 1, no. 1, pp. 3–35, Aug. 2005.
- [23] J. McLurkin, "Stupid robot tricks: A behavior-based distributed algorithm library for programming swarms of robots," Master's thesis, MIT, 2004.
- [24] G. T. Sibley, M. H. Rahimi, and G. S. Sukhatme, "Robomote: A tiny mobile robot platform for large-scale sensor networks," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2002.