Adaptive Control of Multiagent Systems for Finding Peaks of Uncertain Static Fields

In this paper, we design and analyze a class of multiagent systems that locate peaks of uncertain static fields in a distributed and scalable manner. The scalar field of interest is assumed to be generated by a radial basis function network. Our distributed coordination algorithms for multiagent systems build on techniques from adaptive control. Each agent is driven by swarming and gradient ascent efforts based on its own recursively estimated field via locally collected measurements by itself and its neighboring agents. The convergence properties of the proposed multiagent systems are analyzed. We also propose a sampling scheme to facilitate the convergence. We provide simulation results by applying our proposed algorithms to nonholonomic differentially driven mobile robots. The extensive simulation results match well with the predicted behaviors from the convergence analysis and illustrate the usefulness of the proposed coordination and sampling algorithms. [DOI: 10.1115/1.4006369]

Keywords: multiagent systems, cooperative control, adaptive control

1 Introduction

In recent years, due to significant progress in sensing, communication, and embedded-system technologies, many research activities have been focused on the areas of mobile sensor networks and multiagent systems [1–6]. A mobile sensor network usually forms an ad hoc wireless communication network in which each agent shares information with neighboring agents within a short communication range, with limited memory and computational power. In order to achieve a global goal such as exploration, surveillance, and environmental monitoring, mobile sensing agents require distributed coordination to deal with uncertain environments. Although each agent has limited capabilities, as a group, the multiagent system may perform various tasks at a level, which is compatible to a small number of high-end mobile agents.

For multiagent systems with limited resources, having scalable and distributed coordination algorithms using only local information from local neighboring agents is very useful in many aspects [1–6]. For example, such algorithms and their complexity will not change as the number of robotic sensors changes in space and time.

In Ref. [4], decentralized and adaptive control algorithms have been proposed for networks of robots to converge to optimal sensing configurations while simultaneously learning the distribution of sensory information in the environment.

Olaf-Saber [2] developed comprehensive analysis of the flocking algorithm. Such collective swarm behaviors are known to be the outcomes of natural optimization [7]. A flocking algorithm has been used to move mobile sensor networks in groups [5].

Among many problems in coordinating robotic sensors, finding peaks of a scalar field of interest has attracted much attention of control engineers [5,8]. This is due to numerous applications of tracking toxins by robotic sensors in uncertain environments. Such demand also exists in environmental monitoring where a dominant method for measuring environmental variables (e.g., biomass of harmful algal blooms) is still manual sampling followed by laboratory analysis. The detrimental effects of harmful environmental variables can be seen from satellite images in a large scale with a low resolution. For example, Figs. 1(a) and 1(b) show the dead zone created by harmful algal blooms and the oil slick in the Gulf of Mexico, respectively.

The cooperative network of agents that performs adaptive gradient climbing in a spatial environmental field was presented in Ref. [8]. The centralized network can adapt its configuration in response to the sensed environment in order to optimize its gradient climb.

In Ref. [5], distributed learning and control algorithms are proposed to be executed by each agent independently to estimate a scalar field of interest from noisy measurements and to coordinate multiple agents in a distributed manner to discover peaks of the field. In analyzing convergence properties, the estimation error dynamics have been averaged out under sufficient conditions and so only the ODE of the controlled multiagent system dynamics was considered. However, if the parameter estimation error dynamics cannot be averaged out (e.g., deterministic case), the closed-loop dynamics including error dynamics need to be analyzed.

The objective of this paper is to synthesize scalable and distributed coordination algorithms such that the multiagent system estimates the field and locates peaks of the field in a collective manner. Moreover, convergence properties of the proposed coordination algorithms need to be analyzed and the conditions for which the convergence is guaranteed need to be identified.

To this end, in this paper, we design and analyze a class of multiagent systems that locate peaks of static scalar fields in a distributed and scalable manner. Our approach shares the viewpoint of adaptive control. The scalar field of interest is assumed to be generated by a radial basis function network (Sec. 3). We use the flocking algorithm for robotic sensors to make spatially distributed sampling and to maintain communication connectivity (Sec. 4.1). The proposed distributed adaptive control consists of the swarming effort and the gradient ascent motion control based on the recursively estimated field (Sec. 4.2). The associated recursive estimation laws have been developed using gradient-based and recursive least squares (RLS) algorithms. In contrast to Ref. [5], the closed-loop dynamics combining the motion control of the multiagent system and the parameter estimation error dynamics under proposed strategies have been analyzed (Sec. 4.4). A set of
sufficient conditions for which the convergence of the closed-loop multiagent system is achieved has been provided. To facilitate the successful convergence, we provide an additional scalable and distributed sampling strategy that keeps selective past measurements (Sec. 5). We also provide simulation results by applying our proposed algorithms to fully actuated nonholonomic differentially driven mobile robots under different conditions. The extensive simulation study illustrates the effectiveness of the proposed schemes (Sec. 6).

Standard notation will be used throughout the paper. Let $\mathbb{R}$, $\mathbb{R}_{\geq 0}$, $\mathbb{R}_{> 0}$ denote the set of real, non-negative real, and positive real, respectively. The positive definiteness (respectively, semi definiteness) of a matrix $A$ is denoted by $A > 0$ (respectively, $A \succeq 0$). $I_n \in \mathbb{R}^{n \times n}$ denotes the identity matrix of size $n$. $I_k \in \mathbb{R}^{n \times k}$ denotes the column vector of size $n$ whose elements are 1. $|\mathcal{N}|$ denotes the cardinality of the set $\mathcal{N}$. For column vectors $v_i \in \mathbb{R}^n$, $v_i \in \mathbb{R}^b$, and $v_i \in \mathbb{R}^n$, col$(v_a, v_b, v_c) := [v_a \ v_b \ v_c]^T \in \mathbb{R}^{n+b+c}$ stacks all vectors to create one column vector. $\|v\|$ denotes the Euclidean norm (or the vector 2-norm) of a vector $v \in \mathbb{R}^n$, diag $(A, B)$ denotes the (generalized) block diagonal matrix of $A \in \mathbb{R}^{n \times m}$, $B \in \mathbb{R}^{p \times q}$ and is denoted by diag$(A, B) = \begin{bmatrix} A & 0 \\ 0 & B \end{bmatrix} \in \mathbb{R}^{(a+b) \times (m+n)}$. Other notation will be explained in due course.

## 2 Multiagent Systems

In this section, we describe the multiagent system regarding its individual dynamics and limited communication capability in Secs. 2.1 and 2.2, respectively.

### 2.1 Individual Dynamics. We assume that $n$ sensing agents are distributed over the surveillance region $Q \subset \mathbb{R}^2$. $Q$ is assumed to be a convex and compact set. The identity of each agent is indexed by $i \in \{1, 2, \ldots, n\}$. Let $q_i(t) \in Q$ be the location of the sensor attached to agent $i$ at time $t \in \mathbb{R}_{\geq 0}$. Let $g := \text{col} (q_1, q_2, \ldots, q_n) \in \mathbb{R}^n$ be the configuration of the multiagent system.

Consider a collection of multiple agents, each of which is a nonholonomic differentially driven mobile agent as shown in Fig. 2. In this case, the equations of motion for mobile agent $i$ [3,9] may be given by

$$
\begin{bmatrix}
\dot{r}_x(t) \\
\dot{r}_y(t) \\
\dot{\psi}_i(t) \\
\dot{v}_i(t) \\
\dot{\omega}_i(t)
\end{bmatrix} =
\begin{bmatrix}
v_i \cos \psi_i(t) \\
v_i \sin \psi_i(t) \\
\omega_i(t) \\
0 \\
0
\end{bmatrix}
+ \begin{bmatrix}
0 & 0 & 0 & 0 & F_i(t) \\
0 & 0 & 1 & m_i & 0 \\
0 & 0 & 0 & 1 & \tau_i(t)
\end{bmatrix}
$$

---

Fig. 1 (a) Dead zone created by algal blooms in the Gulf of Mexico (NASA). (b) The oil slick as seen from space by NASA’s Terra satellite on May 24, 2010 (Photo courtesy of NASA).

---

Fig. 2 The position of the sensor and states of robot $i$

where $r_i = [r_{x_i}, r_{y_i}]^T$ and $\psi_i$ denote the inertial position and the orientation of agent $i$, respectively. $v_i$ and $\omega_i$ are linear and angular speeds, respectively. $F_i$ and $\tau_i$ denote force and torque inputs, respectively. $m_i$ and $J_i$ are the mass and the moment of inertia, respectively. In this paper, we need to control the sensor location. Let the sensor location be at a point that is on a center line perpendicular to the wheel axis and is $\ell_i$ distance away from the wheel axis, i.e., $\|q_i(t) - r_i(t)\| = \ell_i$ as shown in Fig. 2. The sensor location can be described by

$$q_i(t) = r_i(t) + \ell_i \begin{bmatrix} \cos \psi_i(t) \\ \sin \psi_i(t) \end{bmatrix}$$

By differentiating $q_i(t)$ twice with respect to time $t$, we obtain

$$\ddot{q}_i(t) = -\frac{1}{m_i} \cos \psi_i(t) \sin \psi_i(t) F_i(t) + \frac{1}{J_i} \frac{\dot{\ell}_i}{\ell_i} \cos \psi_i(t) \tau_i(t)$$

Since $B(t)$ is nonsingular as long as $\|q_i(t) - r_i(t)\| = \ell_i \neq 0$, we can perform the output feedback linearization at the sensor location.
\[ q(t) \text{ using the output feedback linearizing control [9] given as follows} \]

\[ F_i(t) \begin{bmatrix} F_i(t) \\ \tau_i(t) \end{bmatrix} = B^{-1}(t) \times (u_i(t) - A(i)) \]  

(4)

With this feedback linearizing control in Eq. (4) being applied to Eq. (3), we obtain that \( \dot{q}_i(t) = u_i(t) \). It can be shown that the zero dynamics are stable [9] and differentiating Eq. (2) shows that \( \omega_i(t) \to 0 \) and \( v_i(t) \to 0 \) as \( \dot{q}_i(t) \to 0 \). There exists a large class of vehicle models [10,11] that can be output feedback linearized with respect to the sensor location to obtain the similar result.

Therefore, without loss of generality, we assume that the sensor location dynamics of agent \( i \) is given by

\[ \dot{q}_i(t) = p_i(t), \quad \dot{p}_i(t) = u_i(t) \]  

(5)

where \( p_i(t) \) is the velocity of agent \( i \) and \( u_i(t) \) is the input of agent \( i \). For the case of the nonholonomic model described in Eq. (1), the proposed control \( u_i(t) \) in this paper can be implemented by \( F_i(t) \) and \( \tau_i(t) \), which are obtained by plugging \( u_i(t) \) in Eq. (4).

2.2 Limited Communication Capability. We assume that each agent can communicate with its neighboring agents within a limited transmission range, which is given by a radius of \( r \). This limited communication among agents can be modeled by an undirected graph with edges, i.e., \( G(q) = (\mathcal{N}, E(q)) \), where an edge \((i,j) \in E(q)\) if and only if \( ||q_i(t) - q_j(t)|| \leq r \). The neighborhood of agent \( i \) with a configuration of \( q \) is defined by \( \mathcal{N}(i,q) = \{ j \in \mathcal{N} | (i,j) \in E(q) \} \). We often use \( \mathcal{N}_i \) instead of using \( \mathcal{N}(i,q) \) for notational simplicity.

We define \( \mathcal{N}(i,q) \) as the union of index \( i \) and indices of its neighbors, i.e., \( \mathcal{N}_i \). We use the adjacency matrix \( A := [a_{ij}] \) of the undirected graph \( G \) as defined in Ref. [1]. Note that \( A := [a_{ij}] \) is symmetrical. The element \( a_{ij} \) of adjacency matrix is defined as \( a_{ij} := \sigma_{ai}(x - d_{ij}) \), with \( \sigma_{ai}(x) = \frac{1}{1+e^{-x}} \), where \( d_{ij} \) is a distance between neighboring agent \( j \) and agent \( i \) itself, \( \sigma_{ai} \) is the sigmoid function with constants \( w > 0 \) and \( \varepsilon > 0 \). The scalar graph Laplacian \( L := [l_{ij}] \in \mathbb{R}^{n \times n} \) is a matrix defined as \( L := D(A) - A \), where \( D(A) \) is a diagonal matrix given by, i.e., \( D(A) := \text{diag}(\sum_{i \in \mathcal{N}_i} a_{ij}) \). The Laplacian matrix \( L \) has an eigenvector of \( L_1 = [1, … , 1]^T \) associated with a zero eigenvalue, i.e., \( L_1 = 0 \). The two-dimensional graph Laplacian is defined as \( L_2 := L \otimes L_2 \), where \( \otimes \) is the Kronecker product.

3 Static Scalar Environmental Field

When the dynamics of the environmental scalar field are much slower (e.g., biomass of harmful algae blooms) than those of mobile agents, we may consider that the scalar field is static for the purpose of finding peaks. Suppose that the scalar environmental field \( \mu(\nu) \) is generated by a network of radial basis functions [5]

\[ \mu(\nu) = \sum_{j=1}^{m} \phi_j(\nu) \theta = \phi^T(\nu) \theta \]  

(6)

where \( \phi^T(\nu) \) and \( \theta \) are defined respectively by

\[ \phi^T(\nu) = [\phi_1(\nu) \phi_2(\nu) \cdots \phi_m(\nu)] \in \mathbb{R}^{1 \times m}, \]

\[ \theta = [\theta^1 \theta^2 \cdots \theta^m]^T \in \mathbb{R}^{m \times 1} \]  

(7)

Gaussian radial basis functions \( \phi_j(\nu) \) are given by

\[ \phi_j(\nu) = \frac{1}{\beta_j} \exp \left( -\frac{||\nu - \xi_j||^2}{\sigma_j^2} \right), \quad \forall j \in M \]  

(8)

where \( M := \{ 1, … , m \}, \sigma_j \) is the width of the Gaussian basis function, and \( \beta_j \) is a normalizing constant. Centers of basis functions \( \{ \xi_j \} \in \mathbb{M} \) are assumed to be uniformly distributed in the surveillance region \( Q \).

We will use the gradient ascent control based on the estimated gradient of \( \mu(q) \) to find peaks. To this end, we introduce some notations. The partial derivative of \( \phi(\nu) \in \mathbb{R}^{m \times 1} \) with respect to \( x \in \mathbb{R}^{2 \times 1} \) evaluated at \( x^* \) is denoted by \( \phi'(x^*) \) and is given as follows

\[ \phi'(x^*) := \frac{\partial \phi(x)}{\partial x} \bigg|_{x=x^*} \in \mathbb{R}^{m \times 2} \]

Using Eq. (6), the gradient of the field at \( q_i \) can be obtained in terms of \( \theta \)

\[ \nabla \mu(q_i) = \left[ \frac{\partial \mu(x)}{\partial x} \right]_{x=q_i} = \nabla \phi^T(x) \bigg|_{x=q_i} \theta = \phi^T(q_i) \theta \in \mathbb{R}^{2 \times 1} \]

The estimate of \( \nabla \mu(q_i) \) based on \( \hat{\theta} \) is denoted by \( \nabla \hat{\mu}(q_i) \).
\[
\rho(z) := \begin{cases} 
1, & z \in [0, h); \\
\frac{1}{2} + \cos\left(\frac{\pi (z-h)}{1-h}\right), & z \in [h, 1]; \\
0, & \text{otherwise}
\end{cases}
\]

A potential \( U_2 \) [5] is also used to model the environment. \( U_2 \) enforces each agent to stay inside the closed and connected surveillance region in \( Q \) and prevents collisions with obstacles in \( Q \). We construct \( U_2 \) such that it is radially unbounded in \( q \), i.e., \( U_2(q) \to \infty \) as \( ||q|| \to \infty \). This condition will be used for making a Lyapunov function candidate radially unbounded. Define the total artificial potential by

\[
U(q) := k_1U_1(q) + k_2U_2(q)
\]

where \( k_1, k_2 \in \mathbb{R}_{>0} \) are weighting factors.

The flocking rule of alignment will be implemented by adding velocity consensus that minimizes a quadratic disagreement function. The quadratic disagreement function \( \Psi_G : \mathbb{R}^{2n} \to \mathbb{R}_{>0} \) is used to evaluate the group disagreement in the network of agents \( \Psi_G(p) := \frac{1}{2} \sum_{(i,j) \in \mathcal{E}(q)} a_{ij} \| p_i - p_j \|^2 \), where \( p := \text{col}(p_1, p_2, \ldots, p_n) \in \mathbb{R}^{2n} \). The disagreement function [2,12] can be expressed via the Laplacian \( L_2 : \mathcal{E}(q) = \mathbb{R}^d(L_2) \), and hence the gradient of \( \Psi_G(p) \) with respect to \( p \) is given by \( \nabla \Psi_G(p) = L_2 p \).

### 4.2 Distributed Adaptive Control

In this subsection, we propose a distributed adaptive control algorithm. The adaptive control law for each agent will be generated using only local information from neighboring agents. Recall that the dynamics of agent \( i \) in Eq. (5) is given by \( \dot{q}_i(t) = p_i(t), \) \( \tilde{p}_i(t) = u_i(t), \) where \( u_i(t) \) is the input of agent \( i \). The control input \( u_i(t) \) is then proposed as follows

\[
u_i(t) = -\nabla U(q_i(t)) - \nabla \Psi_G(p_i(t)) - k_d p_i(t) + k_d \Phi^T(q_i(t)) \tilde{\theta}(t)
\]

where the first two terms of the right-hand side of Eq. (11) provide the swarming effort as discussed in Sec. 4.1. The third term in Eq. (11) provides damping at the gradient of the disagreement function at \( \tilde{\theta}(t) \) will be used in the adaptation laws, which is given by

\[
\nabla \Psi_G(\tilde{\theta}(t)) = \sum_{j \in \mathcal{N}_i} a_{ij} \Phi(q_j(t))\tilde{\theta}(t) - \tilde{\theta}(t)
\]

Here, the quadratic disagreement function \( \Psi_G(\tilde{\theta}(t)) : \mathbb{R}^{mn} \to \mathbb{R}_{>0} \) is defined as

\[
\Psi_G(\tilde{\theta}(t)) = \frac{1}{2} \sum_{(i,j) \in \mathcal{E}(q)} a_{ij} \tilde{\theta}(t) - \tilde{\theta}(t) \|
\]

where \( L_m \) and \( \tilde{\theta}(t) = \text{col}(\tilde{\theta}_1(t), \ldots, \tilde{\theta}_n(t)) \).

To develop recursive estimators, the error vector \( e_i(t) \) of agent \( i \) between the estimated values and measured values is defined by

\[
e_i(t) = -\begin{bmatrix}
\mu_i(q_i(t)) - \mu(q_i(t)) \\
\Phi^T(q_i(t)) \tilde{\theta}(t)
\end{bmatrix}
\]

where \( j, \ldots, k \in \mathcal{N}_i \), and the estimate of \( \mu(i) \) at \( \nu \) in Eq. (6) by agent \( i \) is denoted by \( \hat{\mu}_i(\nu) \) and is given as \( \hat{\mu}_i(\nu) := \phi^\nu(\nu) \theta_i(t) \). In addition, \( e_i(t) \) can be rewritten by

\[
e_i(t) = -\begin{bmatrix}
\phi^T(q_i(t)) \tilde{\theta}_i(t) \\
\phi^T(q_i(t)) \tilde{\theta}_i(t)
\end{bmatrix}
\]

where \( \tilde{\theta}_i(t) := \hat{\theta}_i(t) - \theta \) and

\[
\Phi_i(t) = \begin{bmatrix}
\Phi(q_i(t)) \phi(q_i(t)) \cdots \phi(q_i(t))
\end{bmatrix}^T \in \mathbb{R}^{(n) \times m}
\]

With the aforementioned notations, we propose that agent \( i \) updates \( \tilde{\theta}_i(t) \) based on the following two adaptation laws. Using the gradient-based estimator, \( \tilde{\theta}_i(t) \) is updated by the following adaptation law

\[
\tilde{\theta}_i(t) = \gamma_i \Phi_i(t) e_i(t) - k_d \phi(q_i(t)) p_i(t)
\]

where \( \gamma_i \) is the estimation gain and \( k_d \) is a consensus parameter for parameter estimates.

Using the recursive least squares (RLS) estimator, \( \hat{\theta}_i(t) \) is updated as follows

\[
\hat{\theta}_i(t) = P_i(t) \Phi_i(t) e_i(t) - P_i(t) k_d \phi(q_i(t)) p_i(t)
\]

where \( P_i(t) \) is defined by

\[
P_i(t) := \left( \int_0^t \Phi_i^T(t) \Phi_i(t) \, dt \right)^{-1} \in \mathbb{R}^{m \times m}
\]

### 4.3 Collective Dynamics of All Agents

In this subsection, we derive the collective dynamics of all agents under the proposed coordination algorithms using a collective cost function. The collective cost function \( C_d(q(t)) \) for all agents is defined by

\[
C_d(q(t)) := \sum_{i \in \mathcal{N} \setminus d} \left| \mu_i(q_i(t)) - \mu(q_i(t)) \right| = k_d \sum_{i \in \mathcal{N} \setminus d} \left| \mu_i(q_i(t)) - \phi^T(q_i(t)) \theta_d(t) \right|
\]

Hence, the collective estimate of \( C_d(q(t)) \) by all agents at time \( t \) is \( C_d(q(t)) \) and is given by

\[
C_d(q(t)) = k_d \sum_{i \in \mathcal{N} \setminus d} \left| \mu_i(q_i(t)) - \phi(q_i(t)) \tilde{\theta}_i(t) \right|
\]

The gradient of \( C_d(q(t)) \) is given by

\[
\nabla C_d(q(t)) = -k_d \left[ \begin{bmatrix}
\phi^T(q_i(t)) \\
\phi^T(q_i(t))
\end{bmatrix} \tilde{\theta}(t) \right]
\]

where \( \tilde{\theta}(t) := \text{col}(\tilde{\theta}_1(t), \ldots, \tilde{\theta}_n(t)) \), and \( A_d(t) \) is defined by

\[
A_d(t) := k_d \text{diag}(\phi^T(q_1(t)), \ldots, \phi^T(q_n(t))) \in \mathbb{R}^{2n \times mn}
\]

The collective estimate of \( \nabla C_d(q(t)) \) by all agents is denoted by \( \nabla C_d(q(t)) \) and is given by

\[
\nabla C_d(q(t)) = -k_d \left[ \begin{bmatrix}
\phi^T(q_1(t)) \\
\phi^T(q_1(t))
\end{bmatrix} \tilde{\theta}(t) \right]
\]
Recall that agents is given by

\[ \dot{q}(t) = p(t), \]
\[ \dot{p}(t) = -\nabla U(q(t)) - \nabla \Psi_G(p(t)) - K_d p(t) - \nabla \tilde{C}_d(q(t)) \]

where \( q(t) = \text{col}(q_1(t), \ldots, q_n(t)) \), and \( p(t) = \text{col}(p_1(t), \ldots, p_n(t)) \).

The collective dynamics of all agents from Eqs. (5) and (11) are obtained by

\[ \tilde{\theta}_d(t) = \Gamma_d \Phi_d(t) \tilde{\theta}_d(t) - \Gamma_d \tilde{p}_d(t) + \Gamma_d \tilde{L}_m \tilde{\theta}_d(t) \]

where \( \Gamma_d = \Gamma_0 \otimes L_m \) and \( \Gamma = \Gamma^T \succ 0 \) is the diagonal matrix given by \( \Gamma = \text{diag}(\gamma_1, \ldots, \gamma_n) \). The collective error \( e_d(t) \) is defined and expressed as follows

\[ e_d(t) := \begin{bmatrix} e_1(t) \\ \vdots \\ e_n(t) \end{bmatrix} = \begin{bmatrix} -\Phi_d(t) \tilde{e}_d(t) \\ \vdots \\ -\Phi_d(t) \tilde{e}_d(t) \end{bmatrix} = -\Phi_d(t) \tilde{e}_d(t) \]

where \( \Phi_d(t) := \text{diag}(\Phi_{d1}(t), \ldots, \Phi_{dn}(t)) \in \mathbb{R}^{(\sum_{i=1}^n \gamma_i) \times n} \). Note that \( L_m \tilde{\theta}_d(t) = \tilde{L}_m \tilde{\theta}_d(t) \) in Eq. (15), where \( \tilde{\theta}_d(t) := \tilde{\theta}_d(t) - \tilde{\theta}_d \).

On the other hand, the collective version of the adaptive law in Eq. (13) for all agents is given by

\[ \tilde{\theta}_d(t) = -p_d(t) \Phi_d(t) \tilde{\theta}_d(t) - p_d(t) \tilde{L}_m \tilde{\theta}_d(t) \]

where \( p_d(t) \) is defined by \( p_d(t) := \text{diag}(p_1(t), \ldots, p_n(t)) \in \mathbb{R}^{n_{\text{max}} \times n} \). Using Eq. (14), the time derivative of \( V_d(x) \) in Eq. (18) is obtained by

\[ \dot{V}_d = \Phi_d(t) \tilde{p}_d(t) \Phi_d(t) + \Gamma_d \tilde{\theta}_d(t) \leq 0 \]

Let \( \tilde{x} \) be a solution that belongs to \( D_{\text{ad}} \) if \( (L_d + K_d) > 0 \) and \( (\Phi_d(t) \Phi_d(t) + \tilde{L}_m) > 0 \), \forall t \in D_{\text{ad}} \), from Eq. (22), any point \( x^* \) in \( M_d \) is of the form \( x^*(t) = \text{col}(q^*(t), p^* t \equiv 0, \tilde{\theta}^* \equiv 0). \)

Moreover, \( \tilde{\theta}^* \equiv 0 \Rightarrow 0 \equiv \tilde{\theta}^* \Rightarrow \nabla \tilde{C}_d(q^*) \equiv \nabla \tilde{C}_d(q^*). \) From Eq. (14), we have \( \tilde{\theta}^* \equiv 0, p^* t \equiv 0 \Rightarrow \dot{q}^* \equiv 0 \equiv -\nabla U(q^*) - \nabla \tilde{C}_d(q^*). \) This implies that \( x^* \) is a critical point of the cost function \( V_d(x) \) and \( \tilde{\theta}_d(t) \) converges to \( \tilde{\theta}_d \), QED.

Proof of Theorem 1 for the case of the gradient-based estimator. Using Eq. (14), the time derivative of \( V_d(x) \) in Eq. (18) is obtained by

\[ \dot{V}_d = \Phi_d(t) \tilde{p}_d(t) \Phi_d(t) + \Gamma_d \tilde{\theta}_d(t) \leq 0 \]

As an example, we can define the global performance cost function \( V_d(x) \) along with the global performance cost function \( V_d(x) \) in Eq. (18) by as follows

\[ V_d(x) = \sum_{i=1}^n \tilde{x}_i \]

where \( \tilde{x}_i := \text{col}(x_1(t), \ldots, x_n(t)) \in \mathbb{R}^{n_{\text{max}}} \). Let \( D_{\text{ad}} := \{ x \in D \mid V_d(x) \leq 0 \} \) be a level-set of the collective cost function. Let \( D_{\text{ad}} \) be the set of all points in \( D_{\text{ad}} \) where \( \frac{\partial V_d(x)}{\partial x} = 0 \).

Then, every solution starting from \( D_{\text{ad}} \) approaches the largest invariant set \( M_d \) contained in \( D_{\text{ad}} \) as \( t \to \infty \).

Proof of Theorem 1 for the case of the RLS estimator. Using Eq. (14), the time derivative of \( V_d(x) \) in Eq. (19) is obtained by

\[ \dot{V}_d = \begin{bmatrix} \nabla U(q(t)) + \nabla C_d(q(t)) \end{bmatrix}^T \begin{bmatrix} p(t) \\ \dot{p}(t) \end{bmatrix} - \begin{bmatrix} \nabla U(q(t)) - \nabla \Psi_G(p(t)) - K_d p(t) - \nabla \tilde{C}_d(q(t)) \end{bmatrix} \]

Theorem 1. For any initial state \( x_0 = \text{col}(q_0, p_0, \tilde{\theta}_d) \in D_d \), where \( D_d \) is a compact set, we consider the distributed control law in Eq. (11) based on the gradient-based estimator in Eq. (12)
\[ \dot{V}_d = -p^T(t)(\dot{L}_2 + K_d)p(t) \]
\[ -\dot{U}_d(t)\left(\frac{1}{2} \Phi_d^2(t)\Phi_d(t) + \dot{L}_m\right)\Phi_d(t) \leq 0 \]

The rest of the proof follows as in the case of the gradient estimator. QED.

5 A Sampling Scheme for Helping Convergence

To improve the possibility of satisfying the sufficient conditions of convergence in Theorem 1, we provide a sampling scheme that will help on making \( \Phi_d^2(t)\Phi_d(t) \) positive definite.

In this sampling scheme, at every sampling time \( t \), a fixed number \( (m) \) of measurements sampled previously will be augmented to the fresh measurements available at time \( t \) for agent \( i \). The

![Fig. 3 The trajectories of agents in cases 1, 2, and 3 are shown in the first, second, and third columns, respectively. The first, second, third, and fourth rows correspond to initial pose 1 and field 1, initial pose 1 and field 2, initial pose 2 and field 1, and initial pose 2 and field 2, respectively. The locations of robots are marked by snapshots of poses at \( t = 0 \) s, \( t = 100 \) s, and \( t = 1000 \) s, by white, magenta, and black arrowheads, respectively, showing their positions and heading angles. Horizontal and vertical axes are x and y coordinates, respectively, and the background colors show the scalar fields.](image-url)
selection of such \( m \) additional measurements at time \( t \) for agent \( i \) is as follows
\[
\tilde{q}_j(t) = \arg \min_{q \in \Omega_i(t)} \| q - \xi_j \|, \quad \forall j \in M
\]  
(24)
where \( M := \{1, \ldots, m\} \) and \( \xi_j \) is the center location of the \( j \)th basis function as defined in Eq. (8).\( \Omega_i(t) \) is defined by
\[
\Omega_i(t) := \left( \bigcup_{k \in \mathcal{N}_i(t)} \tilde{q}_k(t) \right) \cup \left( \bigcup_{j \in M} \{ \tilde{q}_j(t) \} \right)
\]
where \( t^- \) is the sampling time taken prior to \( t \). Notice that this selection process in Eq. (24) is scalable and distributed.

Expanded \( \Phi(t) \) due to the augmented sampled data at time \( t \) is as follows
\[
\Phi(t) = \begin{bmatrix} \phi(q_1(t)) & \ldots & \phi(q_m(t)) \end{bmatrix}^T \in \mathbb{R}^{\sum_{i=1}^{m} m \times n}
\]

6 Simulation Results

We have applied the proposed distributed adaptive control to fully actuated nonholonomic differentially driven mobile robots introduced in Sec. 2.1 under different conditions. For the numerical simulation study, two scalar fields illustrated as color maps in Fig. 3 were generated by the model in Eq. (6) with nine radial basis functions. Agents were launched at a set of randomly distributed initial poses (positions and angles). Two sets of scalar fields and initial poses were selected and used for all simulations for a fair comparison. Each of initial poses of agents and scalar fields were indexed by 1 and 2. A limited transmission radius was chosen to be \( r = 0.5 \). The initial value for the parameter vector of each agent was given as a zero vector, i.e., \( \hat{\theta}_i(0) = 0_{m+1} \in \mathbb{R}^m \). Table 1 shows the parameters used for the numerical evaluation.

The trajectories of agents from the simulation are shown in Fig. 3. The error norm \( \| \tilde{\theta}_i(t) \| \), where \( \tilde{\theta}_i(t) := \hat{\theta}_i(t) - \theta \) for agent \( i \) is shown for all \( i \in I \) in Fig. 4. The simulation results under three Table 1 Parameters in the simulation study

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of agents</td>
<td>30</td>
</tr>
<tr>
<td>Number of basis functions</td>
<td>9</td>
</tr>
<tr>
<td>Surveillance region ( Q )</td>
<td>([0,3] \times [0,3])</td>
</tr>
<tr>
<td>((d_{i0},d_{i1},r_0))</td>
<td>((0.3,0.39,0.5,0.5))</td>
</tr>
<tr>
<td>((k_1,k_2,k_4,k_6))</td>
<td>((5,1,1,1))</td>
</tr>
<tr>
<td>(S_{i0})</td>
<td>(0.5S_{i0})</td>
</tr>
<tr>
<td>(\hat{\theta}_i(0))</td>
<td>(\hat{\theta}_i(0))</td>
</tr>
<tr>
<td>(x)</td>
<td>0.05</td>
</tr>
</tbody>
</table>

Fig. 4 The norm of the parameter estimation error, i.e., \( \| \tilde{\theta}_i(t) \| \) by agent \( i \) for all \( i \in I \) in cases 1, 2, and 3 are shown in the first, second, and third columns, respectively. The first, second, third, and fourth rows correspond to initial pose 1 and field 1, initial pose 1 and field 2, initial pose 2 and field 1, and initial pose 2 and field 2, respectively. Vertical and horizontal axes are \( \| \tilde{\theta}_i(t) \| \) (in a logarithmic scale) and time, respectively.
different combinations of the proposed algorithms are summarized as follows:

**Case 1:** The trajectories of agents under the distributed adaptive control using the gradient-based estimator in Eq. (12) and the sampling scheme in Eq. (24) are shown in Figs. 3(a), 3(d), 3(g), and 3(l). The corresponding parameter convergence rates are shown in Figs. 4(a), 4(d), 4(g), and 4(l).

**Case 2:** The trajectories of agents under distributed adaptive control using the RLS estimator in Eq. (13) and the sampling scheme are shown in Figs. 3(b), 3(e), 3(h), and 3(k). The corresponding parameter convergence rates are shown in Figs. 4(b), 4(e), 4(h), and 4(k).

**Case 3:** The trajectories of agents under the distributed adaptive control based on the gradient-based estimator without using the sampling scheme are shown in Figs. 3(c), 3(f), 3(i), and 3(l). The corresponding parameter convergence rates are shown in Figs. 4(c), 4(f), 4(i), and 4(l).

In particular, Fig. 3 shows the trajectories of agents under initial conditions 1 and 2 and fields 1 and 2 (2 × 2 combinatorial scenarios corresponding to four rows of Fig. 3). The locations of robots are marked by snapshots of poses at $t = 0$ s, $t = 100$ s, and $t = 1000$ s, by white, magenta, and black arrowheads, respectively, showing their positions and heading angles.

From the simulation results in Figs. 3 and 4, we see that agents with distributed adaptive control and learning algorithms successfully find major peaks of the uncertain fields. No particular behavioral difference between cases 1 and 2 was observed. Note that some of agents may converge to the local minima of the global performance cost function, which are related to the local maxima of the scalar field.

Under the limited communication range, different groups of agents are formed to share measurements and interact with each other in a distributed fashion. Therefore, the final configuration and the parameter convergence rate of each agent will depend on the initial positions of the multiagent system and the uncertain scalar field as shown in Figs. 3 and 4. For example, with the initial positions of agents as shown in Fig. 3(g), the communication graph is not connected, which results in converging to local maxima of the field. Similarly, Fig. 4(g) shows that parameter estimates by agents do not converge to the correct ones. On the other hand, with a different initial condition, better collective exploratory behaviors can be obtained as shown in Figs. 3(a) and 4(a) as compared to those in Figs. 3(g) and 4(g).

The multiagent system with the proposed sampling scheme in Eq. (24) outperforms the multiagent system without the sampling scheme, as can be seen clearly by comparing parameter error convergence rates in first (case 1) and third (case 3) columns of Fig. 4.

Recall that agent $i$ starts by an initial guess of $\hat{\theta}_i(0)$ and recursively updates $\hat{\theta}_i(t)$ as more measurements collected. Using zero, initial conditions for $\hat{\theta}_i(0)$ may be plausible in detecting accidental release of toxic chemical plumes, which is not supposed to be found in a normal situation. On the other hand, our additional simulation study suggests that using randomized initial conditions for $\hat{\theta}_i(0)$ yields exploratory behaviors by dispatching multiple agents in random directions initially to find potential peaks.

### 7 Conclusions

In this paper, we designed and analyzed a class of multiagent systems that locate peaks of static scalar fields of interest based on adaptive control. Each agent was driven by swarming and gradient ascent efforts based on its own recursively estimated field via locally collected measurements. The convergence properties of the proposed multiagent systems were analyzed. A sampling scheme to help the convergence was provided. The simulation results under different scenarios matched well with the predicted behaviors from the convergence analysis and demonstrated the usefulness of the proposed coordination and sampling algorithms. The proposed multiagent systems and coordination algorithms were developed under a set of assumptions such as fully actuated nonholonomic differentially driven mobile robots, no sensor noise, and a simple communication model (r-disk). Therefore, a future research direction is to extend the proposed approach in more realistic conditions and develop the stochastic version of this problem, taking into account the measurement sensor noise.

### Acknowledgment

This work has been supported, in part, by the National Science Foundation through CAREER Award No. CMMI-0846547. This support is gratefully acknowledged.

We appreciate valuable suggestions from the editors and the reviewers on improving the quality of the paper.

### References


