Automatica 45 (2009) 2802–2814

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Distributed learning and cooperative control for multi-agent systems[☆]

Jongeun Choi[a,b,∗], Songhwai Oh[c], Roberto Horowitz[d]

[a] Department of Mechanical Engineering, Michigan State University, East Lansing, MI 48824-1226, USA
[b] Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI 48824-1226, USA
[c] School of Electrical Engineering and Computer Science, Seoul National University, Seoul, Republic of Korea
[d] Department of Mechanical Engineering, University of California, Berkeley, CA 94720-1740, USA

ARTICLE INFO

Article history:
Received 29 March 2008
Received in revised form
4 May 2009
Accepted 15 September 2009
Available online 7 October 2009

Keywords:
Cooperative control
Multi-agent systems
Recursive parameter estimation

ABSTRACT

This paper presents an algorithm and analysis of distributed learning and cooperative control for a multi-agent system so that a global goal of the overall system can be achieved by locally acting agents. We consider a resource-constrained multi-agent system, in which each agent has limited capabilities in terms of sensing, computation, and communication. The proposed algorithm is executed by each agent independently to estimate an unknown field of interest from noisy measurements and to coordinate multiple agents in a distributed manner to discover peaks of the unknown field. Each mobile agent maintains its own local estimate of the field and updates the estimate using collective measurements from itself and nearby agents. Each agent then moves towards peaks of the field using the gradient of its estimated field while avoiding collision and maintaining communication connectivity. The proposed algorithm is based on a recursive spatial estimation of an unknown field. We show that the closed-loop dynamics of the proposed multi-agent system can be transformed into a form of a stochastic approximation algorithm and prove its convergence using Ljung's ordinary differential equation (ODE) approach. We also present extensive simulation results supporting our theoretical results.

1. Introduction

Recently, there has been a growing interest in wireless sensor networks (Culler, Estrin, & Srivastava, 2004; Estrin, Culler, Pister, & Sukhatme, 2002). A wireless sensor network consists of a large number of sensor nodes. Each sensor node can perform sensing and computation, and sensor nodes form an ad hoc wireless network for communication. Applications of wireless sensor networks include, but not limited to, environment monitoring, building comfort control, traffic control, manufacturing and plant automation, and surveillance systems (Oh, Schenato, Chen, & Sastry, 2007, and the references therein). However, faced with the dynamic nature of environment, stationary sensor networks are sometimes inadequate and a mobile sensing technology shows superior performance in terms of its adaptability and high-resolution sampling capability (Singh et al., 2007).

Mobility in a sensor network can increase its sensing coverage both in space and time and robustness against dynamic changes in the environment. As in wireless sensor networks, each mobile agent is resource constrained; it operates under a short communication range, limited memory, and limited computational power. These mobile agents form an ad hoc wireless network for communication. Although each agent has limited capabilities, as a group, they can perform various tasks at a level which is compatible to a small number of high-end mobile agent. To perform various tasks such as exploration, surveillance, and environmental monitoring, distributed coordination of mobile sensing agents is required to achieve a global goal; and it has received significant attention recently (Cortes, Martinez, Karatas, & Bullo, 2004; Jadbabie, Lin, & Morse, 2003; Olfati-Saber, 2006; Ren & Beard, 2005; Tanner, Jadbabaie, & Pappas, 2003).

Among challenging problems in distributed coordination of mobile sensing agents, gradient climbing over an unknown field of interest has attracted much attention of environmental scientists and control engineers (Őgren, Fiorelli, & Leonard, 2004, DOD/ONR MURI). This is due to numerous applications of tracking toxins in a distributed environment. An interesting practical application is to trace harmful algal blooms in a lake. For certain environmental conditions, rapidly reproducing harmful algal blooms in lakes

☆ The material in this paper was partially presented at the 46th IEEE Conference on Decision and Control, December 12–14, 2007 at the Hilton New Orleans Riverside in New Orleans, Louisiana USA. This paper was recommended for publication in revised form by Associate Editor Brett Ninness under the direction of Editor Torsten Söderström.

∗ Corresponding author at: Department of Mechanical Engineering, Michigan State University, East Lansing, MI 48824-1226, USA. Tel.: +1 517 432 3164; fax: +1 517 353 1750.

E-mail addresses: jchoi@egr.msu.edu (J. Choi), songhwai@snu.ac.kr (S. Oh), horowitz@me.berkeley.edu (R. Horowitz).

0005-1098/$ – see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.automatica.2009.09.025

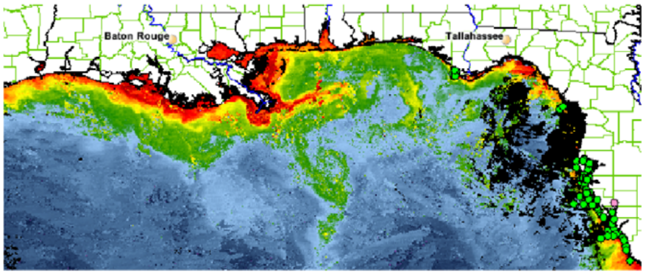

Fig. 1. The estimated field of chlorophyll generated by the harmful algal blooms observation system (Harmful Algal BloomS Observing System) by NOAA. (Photo courtesy of NOAA).

and in oceans can produce cyanotoxins (Center of Excellence for Great Lakes). Exposure to water contaminated with algal cyanotoxins causes serious acute and chronic health effects to humans and adverse effects to aquatic life (Center of Excellence for Great Lakes; Harmful Algal BloomS Observing System). The level of chlorophyll is a measure closely related to algal and cyanobacterial biomass. Hence, there have been efforts to generate the estimated fields of chlorophyll over the areas of concern (Fig. 1). Having had the aforementioned motivation, the objective of our work is to develop theoretically-sound control algorithms for multi-agent systems to trace peaks of a scalar field of interest (for example, harmful algal blooms, temperature, pH, salinity, toxins, and chemical plumes etc.). In general, theses scalar parameters provide rich information about quality of environments, such as the air, lakes, and public water systems.

The most common approach to this tracing problem has been biologically inspired *chemotaxis* (Adler, 1966; Dhariwal, Sukhatme, & Requicha, 2004), in which a mobile sensing agent is driven according to a local gradient of a field of interest. However, with this approach, the convergence rate can be slow and the mobile robot may get stuck in the local maxima of the field. The cooperative network of agents that performs adaptive gradient climbing in a distributed environment was presented in DOD/ONR MURI and Őgren et al. (2004). The centralized network can adapt its configuration in response to the sensed environment in order to optimize its gradient climb.

This problem of gradient climbing constantly occurs in biological species. Aquatic organisms search for favorable regions that contain abundant resources for their survival. For example, it is well known that fish schools climb gradients of nutrients to locate the densest source of food. To locate resources, fish schools use "taxis", a behavior in which they navigate habitats according to local gradients in uncertain environments. Grünbaum (1998) showed that schooling behavior can improve the ability of performing taxis to climb gradients, since the swarming alignment tendency can average out the stochastic sampling errors of individuals.

Olfati-Saber (2006) and Tanner et al. (2003) presented comprehensive analyses of the flocking algorithm by Reynolds (1987). This flocking algorithm was originally developed to simulate the movements of flocking birds in computer graphics where each artificial bird follows a set of rather simple distributed rules (Reynolds, 1987). A bird in a flock coordinates with the movements of its neighboring flock mates and tries to stay close to its neighbors while avoiding collisions. In general, the collective swarm behaviors of birds/fish/ants/bees are known to be the outcomes of natural optimization (Bonabeau, Dorigo, & Theraulaz, 1999; Eberhart, Shi, & Kennedy, 2001).

The state-of-the-art technology in spatial estimation by mobile sensor networks is as follows. A distributed interpolation scheme described in Martinez (in press) for field estimation by mobile sensor networks uses a distributed nonparametric inference method and made to be compatible with

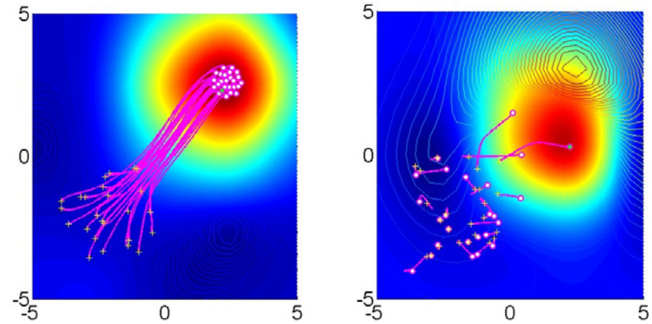

Fig. 2. Left: Trajectories of the proposed multi-agent system. Right: Trajectories of field estimating agents without communication and the swarming effort. The estimated field by agent 1 is shown as a background in colors. In the color map, red color denotes the highest scalar value while blue color represents the lowest value. Agent 1 is plotted as a green dot. Thin contour lines represent the error field between the true field and the estimated field. (+) and (o) represent, respectively, initial and final locations. Solid lines represent trajectories of agents. See more details about the simulation in Section 5.

coverage control (e.g., Cortes et al., 2004). A distributed Kriged Kalman filter for robotic sensor networks to estimate a field of interest is described in Cortés (in press). In each iteration of this Kriged Kalman filter, agents execute a consensus algorithm based on new measurements, e.g., Olfati-Saber and Shamma (2005), for computing average values of interests. Then an iterative weighted least-squares algorithm is performed to compute state estimates of the field.

In this paper, we develop novel distributed learning and cooperative control algorithms for multi-agent systems by extending the recent development in the flocking algorithm (Olfati-Saber, 2006; Tanner et al., 2003). The learning and control algorithms are performed at each agent using only local information. However, they are designed so that agents as a whole exhibit *collective intelligence*, i.e., a collection of agents achieves a global goal. In a resource-constrained multi-agent system, the communication range of each agent is limited as compared to the size of a surveillance region. Hence, agents cannot perform the coverage control as in Cortes et al. (2004), Graham and Cortés (2009) and Martinez (in press). Instead, each agent takes a measurement and also receives measurements from its neighboring agents within its communication range. Upon receiving *collective* measurements, each agent recursively updates its own estimate of an unknown static field of interest. The recursive estimation is based on a radial basis function network in order to represent a wide range of physical phenomena. To locate the maximum of the field, the sensing agent will climb the gradient of its own estimated field. In our proposed approach, each agent has its own recursive estimation of the field based on *collective* measurements from itself and its neighbors without requiring the consensus step of the Kriged Kalman filter (Cortés, in press). The proposed cooperative control mimics the individual and social behaviors of a distributed pack of animals communicating locally to search for their densest resources in an uncertain environment. The fish school's efficient performance of climbing nutrient gradients to search food resources and the exceptional geographical mapping capability of biological creatures have motivated the development of our multi-agent systems. Simulation results in Section 5 strongly support our idea and validate the effectiveness of the proposed multi-agent systems with cooperative control as compared to field estimating agents without cooperative control. As shown in Fig. 2, the proposed multi-agent system collectively locate the maximum of the unknown field rapidly while, without communication and the swarming effort, only a couple of agents near the maximum point can slowly estimate and climb the gradient of the field.

This paper also presents convergence properties of the proposed distributed learning and cooperative control algorithms by transforming the closed-loop dynamics of the multi-agent system into a form of a stochastic approximation algorithm. Our theoretical results are based on the ordinary differential equation (ODE) approach (Kushner & Yin, 1997; Ljung, 1977). We also present a set of sufficient conditions for which the convergence is guaranteed with probability one.

This paper is organized as follows. In Section 2, we briefly introduce the mobile sensing network model, notations related to a graph, and artificial potentials to form a swarming behavior. A recursive radial basis function learning algorithm for mapping the field of interest is presented in Section 3.1. In Section 3.2, cooperatively learning control is described with a stochastic approximation gain. Section 4 analyzes the convergence properties of the proposed coordination algorithm based on the ODE approach. In Section 5, the effectiveness of the proposed multi-agent system is demonstrated by simulation results with respect to different fields of interest and conditions.

2. Mobile sensing agent network

In this section, we describe the mathematical framework for mobile sensing agent networks and explain notations used in this paper.

Let $\mathbb{R}, \mathbb{R}_{\geq 0}, \mathbb{R}_{>0}, \mathbb{Z}, \mathbb{Z}_{\geq 0}, \mathbb{Z}_{>0}$ denote, respectively, the set of real, non-negative real, positive real, integer, non-negative integer, and positive integer numbers. The positive definiteness (respectively, semi-definiteness) of a matrix A is denoted by $A \succ 0$ (respectively, $A \succeq 0$). $I_n \in \mathbb{R}^{n\times n}$ and $0_n \in \mathbb{R}^{n\times n}$ denote the identity and zero matrices, respectively. $0_{m\times n} \in \mathbb{R}^{m\times n}$ denotes the m by n zero matrix. However, the subscript can be omitted for the sake of brevity when it is obvious from the context. The gradient of a differentiable real function $\phi(q) : \mathbb{R}^{2n} \to \mathbb{R}$ with respect to its vector domain q is denoted by

$$\nabla\phi(q) := \begin{bmatrix} \frac{\partial\phi(q)}{\partial q_1}^{\mathrm{T}} & \frac{\partial\phi(q)}{\partial q_2}^{\mathrm{T}} & \cdots & \frac{\partial\phi(q)}{\partial q_n}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{2n},$$

where $q = \mathrm{col}(q_1, \ldots, q_n) \in \mathbb{R}^{2n}$ and $q_i \in \mathbb{R}^2$. To be concise, the gradient of $\phi(q)$ with respect to q_i is denoted by

$$\nabla\phi(q_i) := \frac{\partial\phi(q)}{\partial q_i} \in \mathbb{R}^2.$$

The norm $\|\cdot\|$ will denote the standard Euclidean norm (or 2-norm) on vectors. The induced matrix 2-norm is defined as

$$\|A\| := \sup_{x\neq 0} \frac{\|Ax\|}{\|x\|},$$

where $A \in \mathbb{R}^{m\times n}$. Other notation will be explained in due course.

2.1. Models for mobile sensing agents

Let N_s be the number of sensing agents distributed over the surveillance region $\mathcal{M} \subset \mathbb{R}^2$, which is assumed to be a convex and compact set. The identity of each agent is indexed by $\mathcal{I} := \{1, 2, \ldots, N_s\}$. Let $q_i(t) \in \mathcal{M}$ be the location of the ith sensing agent at time $t \in \mathbb{R}_{\geq 0}$ and let $q := \mathrm{col}(q_1, q_2, \ldots, q_{N_s}) \in \mathbb{R}^{2N_s}$ be the configuration of the multi-agent system. The discrete time, high-level dynamics of agent i is modeled by

$$q_i(t + \Delta_t) = q_i(t) + \Delta_t v_i(t), \quad (1)$$

where $q_i(t) \in \mathbb{R}^2$ and $v_i(t) \in \mathbb{R}^2$ are, respectively, the position and the control input of agent i at time $t \in \mathbb{R}_{\geq 0}$. $\Delta_t \in \mathbb{R}_{>0}$ denotes the iteration step size (or sampling time). We assume that the measurement $y(q_i(t))$ of the ith sensor includes the scalar value of the field $\mu(q_i(t))$ and sensor noise $w(t)$, at its position $q_i(t)$ and a sampled time t,

$$y(q_i(t)) := \mu(q_i(t)) + w(t), \quad (2)$$

where $\mu: \mathcal{M} \to [0, \mu_{\max}]$ is an unknown field of interest.

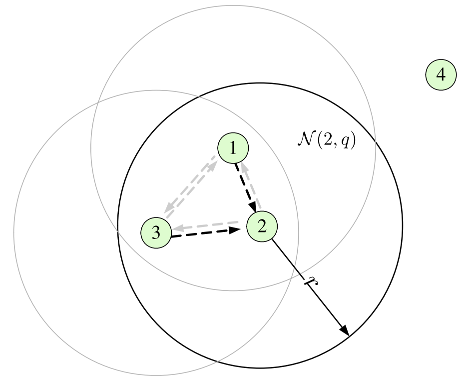

Fig. 3. The model of the mobile sensing agent network. The agent 2 gathers measurements from two neighboring sensing agents 1 and 3 in an r interactive range. Hence, the collective measurements of agent 2 will be sampled at locations denoted by agents 1, 2 and 3.

The proposed algorithm will be executed by each agent independently to estimate an unknown field of interest from noisy measurements and to coordinate multiple agents in a distributed manner to discover peaks of the unknown field. Each mobile agent will maintain its own local estimate of the field and will update the estimate using collective measurements from itself and nearby agents. Each agent will then be programmed to move towards peaks of the field using the gradient of its estimated field.

2.2. Graph-theoretic representation

The group behavior of mobile sensing agents and their complicated interactions with neighbors are best treated by a graph with edges. Let $G(q) := (\mathcal{I}, \mathcal{E}(q))$ be an undirected communication graph such that an edge $(i, j) \in \mathcal{E}(q)$ if and only if agent i can communicate with agent $j \neq i$. We assume that each agent can communicate with its neighboring agents within a limited transmission range given by a radius of r, as depicted in Fig. 3. Therefore, $(i, j) \in \mathcal{E}(q)$ if and only if $\|q_i(t) - q_j(t)\| \leq r$. For example, as shown in Fig. 3, agent 2 communicates with and collects measurements from agents 1 and 3 within its communication range. We define the neighborhood of agent i with a configuration of q by $\mathcal{N}(i, q) := \{j \in \mathcal{I} \mid (i, j) \in \mathcal{E}(q)\}$. The adjacency matrix $A := [a_{ij}]$ of an undirected graph G is a symmetric matrix such that $a_{ij} = k_3 \in \mathbb{R}_{>0}$ if vertex i and vertex j are neighbors and $a_{ij} = 0$ otherwise. Notice that an adjacency matrix A can be also defined in a smooth fashion in terms of q (Olfati-Saber, 2006). The scalar graph Laplacian $L = [l_{ij}] \in \mathbb{R}^{N_s\times N_s}$ is a matrix defined as $L := D^A - A$, where D^A is a diagonal matrix whose diagonal entries are row sums of A, i.e., $D^A := \mathrm{diag}\left(\sum_{j=1}^{N_s} a_{ij}\right)$. The two-dimensional graph Laplacian is defined as $\hat{L} := L \otimes I_2$, where $\otimes$ is the Kronecker product. For instance, the corresponding A, L and $\hat{L}$ for the example shown in Fig. 3 are:

$$A = k_3 \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \qquad L = k_3 \begin{bmatrix} 2 & -1 & -1 & 0 \\ -1 & 2 & -1 & 0 \\ -1 & -1 & 2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\hat{L} = L \otimes I_2 = k_3 \begin{bmatrix} 2I_2 & -I_2 & -I_2 & 0_2 \\ -I_2 & 2I_2 & -I_2 & 0_2 \\ -I_2 & -I_2 & 2I_2 & 0_2 \\ 0_2 & 0_2 & 0_2 & 0_2 \end{bmatrix}.$$

Let $p_i \in \mathbb{R}^2$ be the state of agent i for $i \in \mathcal{I}$ under the topology of an undirected graph G. Two agents i and j are said to agree whenever they have the same states, i.e., $p_i = p_j$. The quadratic disagreement function $\Psi_G: \mathbb{R}^{2N_s} \to \mathbb{R}_{\geq 0}$ evaluates the group disagreement in the network of agents:

$$\Psi_G(p) := \frac{1}{4} \sum_{(i,j)\in\mathcal{E}(q)} a_{ij}\|p_j - p_i\|^2, \tag{3}$$

where $p := \mathrm{col}\,(p_1, p_2, \ldots, p_{N_s}) \in \mathbb{R}^{2N_s}$. A disagreement function (Godsil & Royle, 2001; Olfati-Saber, 2006) can be obtained via the Laplacian $\hat{L}$:

$$\Psi_G(p) = \frac{1}{2} p^{\mathrm{T}} \hat{L} p, \tag{4}$$

and hence the gradient of $\Psi_G(p)$ with respect to p is given by

$$\nabla \Psi_G(p) = \hat{L} p. \tag{5}$$

The properties shown in (4) and (5) will be used in the convergence analysis in Section 4.

2.3. Swarming behavior

A group of agents are coordinated to collect (noisy) samples from a stationary field at diverse locations for the purpose of estimating the field of interest. A set of artificial potential functions creates a swarming behavior of agents and provides agents with obstacle avoidance capabilities. We use attractive and repulsive potential functions similar to ones used in Choi, Oh, and Horowitz (2007), Olfati-Saber (2006) and Tanner et al. (2003) to generate a swarming behavior. To enforce a group of agents to satisfy a set of algebraic constraints $\|q_i - q_j\| = d$ for all $j \in \mathcal{N}(i,q)$, we introduce a smooth collective potential function

$$\begin{aligned} U_1(q) &:= \sum_i \sum_{j\in\mathcal{N}(i,q), j\neq i} U_{ij}(\|q_i - q_j\|^2) \\ &= \sum_i \sum_{j\in\mathcal{N}(i,q), j\neq i} U_{ij}(r_{ij}), \end{aligned} \tag{6}$$

where $r_{ij} := \|q_i - q_j\|^2$. The pair-wise attractive/repulsive potential function $U_{ij}(\cdot)$ in (6) is defined by

$$U_{ij}(r_{ij}) := \frac{1}{2}\left(\log(\alpha + r_{ij}) + \frac{\alpha + d^2}{\alpha + r_{ij}}\right), \quad \text{if } r_{ij} < d_0^2, \tag{7}$$

otherwise (i.e., $r_{ij} \geq d_0^2$), it is defined according to the gradient of the potential, which will be described shortly. Here $\alpha, d \in \mathbb{R}_{>0}$ and $d < d_0$. The gradient of the potential with respect to q_i for agent i is given by

$$\nabla U_1(q_i) = \frac{\partial U_1(q)}{\partial q_i} = \sum_{j\neq i} \left.\frac{\partial U_{ij}(r)}{\partial r}\right|_{r=r_{ij}} 2(q_i - q_j)$$
$$= \begin{cases} \displaystyle\sum_{j\neq i} \frac{(r_{ij} - d^2)(q_i - q_j)}{(\alpha + r_{ij})^2} & \text{if } r_{ij} < d_0^2 \\ \displaystyle\sum_{j\neq i} \rho\left(\frac{\sqrt{r_{ij}} - d_0}{|d_1 - d_0|}\right) \frac{\|d_0^2 - d^2\|}{(\alpha + d_0^2)^2}(q_i - q_j) & \text{if } r_{ij} \geq d_0^2, \end{cases} \tag{8}$$

where $\rho : \mathbb{R}_{\geq 0} \to [0, 1]$ is the bump function (Olfati-Saber, 2006)

$$\rho(z) := \begin{cases} 1, & z \in [0, h); \\ \frac{1}{2}\left[1 + \cos\left(\pi \frac{(z-h)}{(1-h)}\right)\right], & z \in [h, 1]; \\ 0, & \text{otherwise.} \end{cases}$$

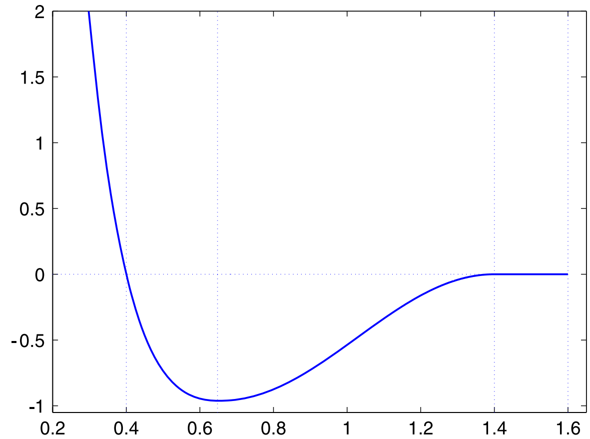

Fig. 4. The reaction force (vertical axis) between two agents is generated by the potential function in (6)–(8) with respect to $\|q_i - q_j\|$ (horizontal axis). Here parameters $d = 0.4$, $d_0 = 0.648$, $d_1 = 1.4$ and $r = 1.6$ are used.

Notice that ρ varies smoothly from 1 to 0 as the scalar input increases. (6)–(8) will be used to produce a continuously differentiable ($\mathcal{C}^1$) reaction potential force between any two agents as depicted in Fig. 4. Parameters α, d, d_0, and d_1 will shape the artificial potential function. A typical way to choose those parameters are explained as follows. In Eqs. (6)–(8), a non-zero gain factor α is introduced to prevent the reaction force from diverging at $r_{ij} = \|q_i - q_j\|^2 = 0$. As illustrated in Fig. 4, this potential yields a reaction force that is attracting when the agents are apart and repelling when a pair of two agents are too close. It has an equilibrium point at a distance of d. d_0 will be chosen at the location where the slope of the potential force first becomes zero (Fig. 4) as $\sqrt{r_{ij}}$ increases from zero. For $\sqrt{r_{ij}} > d_0$, the bump function will shape the potential force to become zero smoothly when the relative distance reaches to d_1 which is slightly shorter than the radius of the transmission range r. Hence, in general, we configure parameters such that $d < d_0 < d_1 < r$, which will force the gradient of the potential function due to agent j in (8) to be a zero vector before the communication link to agent i is disconnected from agent j. In this way, we can construct a continuously differentiable collective potential force between any two agents in spite of the limited communication range. We also introduce a potential U_2 to model the environment. U_2 enforces each agent to stay inside the closed and connected surveillance region in $\mathcal{M}$ and prevents collisions with obstacles in $\mathcal{M}$. Define the total artificial potential by

$$U(q) := k_1 U_1(q) + k_2 U_2(q), \tag{9}$$

where $k_1, k_2 \in \mathbb{R}_{>0}$ are weighting factors. A swarming behavior and an obstacle avoidance capability of each agent will be developed in Section 3.2.

3. Distributed learning and cooperative control

In this section, we describe distributed learning and cooperative control algorithms. The sensing agent will receive measurements from its neighboring agents within a limited transmission range. Upon receiving measurements, each mobile sensing agent will recursively update the estimate of an unknown static field of interest using the distributed learning algorithm. Based on the estimated field, each agent moves to the peak of the field using the cooperative control algorithm.

3.1. Distributed learning

We introduce a distributed learning algorithm for each mobile sensing agent to estimate a static field of interest $\mu : \mathcal{M} \to [0, \mu_{\max}]$. Suppose that the scalar field $\mu(\nu)$ is generated by a