Automatica 80 (2017) 284–294

Contents lists available at ScienceDirect

Automatica

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

Multi-robots Gaussian estimation and coverage control: From client–server to peer-to-peer architectures☆

Marco Todescato [a], Andrea Carron [b], Ruggero Carli [a], Gianluigi Pillonetto [a], Luca Schenato [a]

[a] Department of Information Engineering, University of Padova, Italy
[b] Institute for Dynamic Systems and Control, ETH, Zürich, Switzerland

ARTICLE INFO

Article history:
Received 30 October 2015
Received in revised form
10 February 2017
Accepted 13 February 2017

Keywords:
Gaussian estimation
Coverage control
Robotic networks
Centralized communications
Distributed communications

ABSTRACT

In this work we study the problem of multi-robot coverage of a planar region when the sensory field used to approximate the density of event appearance is not known in advance. We address the problem by considering two different communication architectures: *client–server* and *peer-to-peer*. In the first architecture the robots are allowed to communicate with a central server/base station. In the second the robots communicate among neighboring peers by means of a *gossip* protocol in a distributed fashion. For both the architectures, we resort to nonparametric Gaussian regression approach to estimate the unknown sensory field of interest from a collection of noisy samples. We propose a probabilistic control strategy based on the posterior of the estimation error variance, which lets the robots to estimate the true sensory field with any arbitrary accuracy while simultaneously computing and exploiting the corresponding centroidal Voronoi partitions. We also present a numerically efficient approximation based on a spatial discretization to trade-off the accuracy of the estimated map against the required computational complexity. This trade-off can be tuned based on explicit estimation error bounds which depend on the spatial resolution and the Gaussian kernel parameters. Finally, we test the proposed solutions via extensive numerical simulations.

1. Introduction

The growing sensing capabilities and the development of autonomous robot vehicles able to coordinate themselves to achieve desired tasks are expected to revolutionize our capability to control the physical environment (Leonard et al., 2007). In this context, the coverage of an area of interest is one important and interesting task. In many applications the ability of a group of robots to sense and automatically cover the surrounding environment to maximize the likelihood of detecting an event of interest is appealing. On the other hand, knowledge about the spatial distribution of the event of interest is needed. As an example, consider a group of robots monitoring a forest to detect possible wildfires. Since the probability of a wildfire is likely to be proportional to the temperature, the robots should more densely cover areas with higher temperature which, if not known in advance, must be reconstructed from collected samples. At the same time, to minimize the time to reach a wildfire, the robots should station near the centroids of the partitioned area. This highlights the issue of *simultaneous estimation and coverage* associated with the problem of interest.

In this work we analyze the problem of *covering* the area of interest while *estimating* the non-uniform measurable field of event appearance from *noisy measurements* collected by the robots. There has been considerable effort in the analysis of estimation and coverage separately. Historically, classical identification techniques are based on parametric estimation paradigms, like ML and PEM (Ljung, 1999). However, these techniques often require persistent excitation to ensure convergence of the parameter (Choi, Oh, & Horowitz, 2009) and may be unsatisfactory when tested on experimental data (Pillonetto, Chiuso, & De Nicolao, 2011). To overcome these issues techniques, grouped under the *nonparametric learning* framework, have been recently developed. The main idea is to ex-

☆ This work is supported by the University of Padova, Progetto di Ateneo # CPDA147754/14 "New statistical learning approach for multi-agents adaptive estimation and coverage control". The material in this paper was partially presented at the 14th annual European Control Conference, July 15–17, 2015, Linz, Austria (Carron, Todescato, Carli, Schenato, & Pillonetto, 2015). This paper was recommended for publication in revised form by Associate Editor Michael M. Zavlanos under the direction of Editor Christos G. Cassandras.

E-mail addresses: todescat@dei.unipd.it (M. Todescato), carrona@ethz.ch (A. Carron), carlirug@dei.unipd.it (R. Carli), giapi@dei.unipd.it (G. Pillonetto), schenato@dei.unipd.it (L. Schenato).

http://dx.doi.org/10.1016/j.automatica.2017.02.045
0005-1098/

ploit black box models to estimate a function from examples collected on input locations drawn from a fixed probability sensory function (Pillonetto et al., 2011; Poggio & Girosi, 1990). The drawback is that the computational complexity grows unbounded as the cube of the number of collected samples. Thus, efficient approaches (Xu, Choi, Dass, & Maiti, 2015) based on, e.g., suitable measurements truncation (Xu, Choi, Dass, & Maiti, 2012; Xu, Choi, & Oh, 2011) or Gaussian Markov random fields (Xu, Choi, Dass, & Maiti, 2013), have been proposed.

Classical approaches to the coverage problem assume the sensory function to be perfectly known in advance. In this spirit, works (Cortés & Bullo, 2005; Cortes, Martinez, Karatas, & Bullo, 2004; Durham, Carli, Frasca, & Bullo, 2012) exploit the concept of Centroidal Voronoi partitioning and present solutions based on gradient descent strategies. In Leonard and Olshevsky (2011) a policy for the coverage of a 1-D environment is presented. In Davison, Schwemmer, and Leonard (2012) a limited number of noise-free samples are considered yet no convergence results are presented. The work (Davison, Leonard, Olshevsky, & Schwemmer, 2015) extends (Davison et al., 2012; Leonard & Olshevsky, 2011) by proving convergence in probability to the optimal configuration. A distributed solution in the presence of known time-varying functions is given in Lee, Diaz-Mercado, and Egerstedt (2015). A different line of research deals with adaptive/optimal sampling strategies to enhance estimation accuracy (Xu & Choi, 2011). In particular, in Xu et al. (2011) is proposed a distributed efficient solution, where each robot independently estimates the function of interest based on a truncated subset of its own measurements and those gathered by its neighbors. In Xu et al. (2013) instead, each robot is in charge of monitoring a fixed area of interest, thus not requiring any exchange of measurements between robots.

Some results to the coupled problem, i.e. when both coverage and estimation are considered, have appeared recently. In Choi, Lee, and Oh (2008) the authors exploit Kalman filtering to perform Gaussian estimation. The final objective is to perform estimation and maximum seeking of a function of interest by means of a coordinated group of robots. They propose a two stage algorithm in which, first, based on information on the posterior variance, the robots are spread throughout the space in order to achieve a good estimate of the sensory function; once achieved a predefined threshold, the robots are driven towards the maximum of the estimated function. However, no convergence results during the estimation phase are shown. In Choi and Horowitz (2010) the authors propose a strategy to drive a formation of robots towards the coverage of an area of interest characterized by an unknown probability density function of event appearance. The result builds on learning diffeomorphic functions with kernels. However, it applies only to one dimensional environments and, if needed, it does not provide any estimate of the function of interest. In Schwager, Rus, and Slotine (2009) the authors propose an algorithm for simultaneous distributed consensus-like parametric estimation from noise-free measurements and optimal coverage based on centroidal Voronoi partitioning. However, to prove estimation convergence an infinite amount of noise-free measurements are assumed to be collected in finite time.

In this work, of which a preliminary version can be found in Carron et al. (2015), we analyze the problem of *simultaneous* estimation and coverage. The main contribution is twofold: the first is to consider a strategy that smoothly moves from estimation to coverage at the benefit of better transient behavior as compared to traditional approaches. The second contribution is to exploit the better estimation performance of non-parametric Gaussian regression as compared to parametric approaches while being able to bound its computational complexity. More specifically, we consider two different communication architectures to address the problem both in a centralized as well as in a distributed framework, namely *client–server* and *peer-to-peer* (p2p), respectively. In the client–server architecture (even referred to as one-to-base station communication (Pater, Frasca, Durham, Carli, & Bullo, 2016)) the robots can communicate with a server/base station. In the p2p architecture robots are allowed to communicate with neighboring peers by means of a *gossip* protocol. The goal is to perform *nonparametric* estimation of an unknown sensory distribution function from noisy samples while driving the robots to optimally cover an area of interest. This is achieved via a probabilistic control strategy which allows the robots to seamless transition between estimation and coverage. Differently from the standard approach (Choi et al., 2008), our control never completely switches from the estimation to the coverage phase but always trade-offs between them in order to achieve the best solution in terms of estimation and coverage. This (i) let us prove convergence in probability of the estimated function to the true one. As so, we obtain a final coverage configuration which is arbitrarily close to a partitioning obtained with the exact prior knowledge of the sensory function. Moreover, (ii) the strategy, compared to threshold-based approaches, e.g., in the same spirit of the algorithm proposed in Choi et al. (2008), can lead to smaller average coverage time. To alleviate the computational burden needed to implement the nonparametric estimation procedure, we also propose an alternative algorithm, based on a spatial discretization, to trade-off between accuracy on the estimated map and computational requirements.

The remainder of the paper is as follows. Section 2, recalls the necessary preliminaries. Section 3 contains the problem at hand. Sections 4–6 present the server-based algorithm, its efficient version and the distributed solution with their convergence analysis, respectively. Section 7 presents compelling simulations to test the proposed solution against other possible strategies as well as in the presence of practical limitations. Section 8 concludes the paper. All the proofs can be found in Appendix.

2. Preliminaries

2.1. Voronoi partitions

Let $\mathscr{X} \subset \mathbb{R}^2$ be compact and convex. Let $\mu : \mathscr{X} \to \mathbb{R}_{>0}$ be a distribution sensory function defined over $\mathscr{X}$. Within the context of this paper, a *partition* of $\mathscr{X}$ is a collection of N convex polygons $\mathscr{P} = (\mathscr{P}_1, \dots, \mathscr{P}_N)$ with disjoint interiors whose union is $\mathscr{X}$. Given the list of N distinct points in $\mathscr{X}$, $\mathbf{x} = (x_1, \dots, x_N)$, we define the *Voronoi partition* $\mathscr{W}(\mathbf{x}) = (\mathscr{W}_1(\mathbf{x}), \dots, \mathscr{W}_N(\mathbf{x}))$ generated by $\mathbf{x}$ as

$$\mathscr{W}_i(\mathbf{x}) = \left\{ q \in \mathscr{X} \mid \|q - x_i\| \leq \|q - x_j\|, \ \forall j \neq i \right\} \tag{1}$$

$\|\cdot\|$ being the Euclidean norm, which can be shown to be convex (Du, Faber, & Gunzburger, 1999). Given a partition $\mathscr{P} = (\mathscr{P}_1, \dots, \mathscr{P}_N)$, for each region $\mathscr{P}_i$, $i \in \{1, \dots, N\}$, we define its *centroid* with respect to the sensory function μ as

$$c_i(\mathscr{P}_i) = \left(\int_{\mathscr{P}_i} \mu(q) dq \right)^{-1} \int_{\mathscr{P}_i} q \mu(q) dq.$$

We denote with $\mathbf{c}(\mathscr{P}) = (c_1(\mathscr{P}_1), \dots, c_N(\mathscr{P}_N))$ the vector of regions centroids. A partition $\mathscr{P} = (\mathscr{P}_1, \dots, \mathscr{P}_N)$ is said to be a *Centroidal Voronoi partition* of the pair $(\mathscr{X}, \mu)$ if $\mathscr{P} = \mathscr{W}(\mathbf{c}(\mathscr{P}))$, i.e., $\mathscr{P}$ coincides with the Voronoi partition generated by $\mathbf{c}(\mathscr{P})$. Given a partition $\mathscr{P} = (\mathscr{P}_1, \dots, \mathscr{P}_N)$, a sensory function μ and a set of pints $\mathbf{x} = (x_1, \dots, x_N)$, we introduce the *cost function* $H(\mathscr{P}, \mathbf{x}, \mu)$ defined as

$$H(\mathscr{P}, \mathbf{x}, \mu) = \sum_{i=1}^{N} \int_{\mathscr{P}_i} \|q - x_i\|^2 \mu(q) dq. \tag{2}$$

Observe that the cost $H(\mathscr{P}, \mathbf{c}(\mathscr{P}), \mu)$ coincides with the locational optimization function defined in Cortes et al. (2004). It can be shown that, for a fixed sensory function μ, the set of local minima of $H(\cdot, \mathbf{c}(\cdot), \mu)$ coincides with the Centroidal Voronoi partitions of the pair $(\mathscr{X}, \mu)$ (Du et al., 1999).

2.2. Coverage control algorithm

Let $\mathscr{X} \subset \mathbb{R}^2$ be a convex and closed polygon and let $\mu : \mathscr{X} \mapsto \mathbb{R}$ be a sensory function. Consider the following optimization problem

$$\min_{\mathscr{P}} H(\mathscr{P}, \mathbf{c}(\mathscr{P}), \mu).$$

The *coverage algorithm* we consider is a version of the classic Lloyd algorithm (Lloyd, 1982), which, given an initial condition $\mathscr{P}(0)$, consists of:

$$\mathscr{P}^L(k+1) = \mathscr{W}(\mathbf{c}(\mathscr{P}^L(k))), \tag{3}$$

where the upperscript L indicates the sequence generated by the Lloyd algorithm. That is, (i) it computes the centroids $\mathbf{c}(\mathscr{P})$ of the current partition and (ii) updates $\mathscr{P}$ to the partition $\mathscr{W}(\mathbf{c}(\mathscr{P}))$. Clearly, by construction $\mathscr{P}^L(k)$ are all Voronoi partitions for $k \geq 1$. It can be shown (Cortes et al., 2004) that the function $H(\mathscr{P}, \mathbf{c}(\mathscr{P}), \mu)$ is monotonically non-increasing along the solutions of (3) and that all the solutions of (3) converge asymptotically to the set of centroidal Voronoi partitions. It is well known (Cortes et al., 2004) that the set of centroidal Voronoi partitions of the pair $(\mathscr{X}, \mu)$ are the critical points of the coverage function $H(\mathscr{P}, \mathbf{c}(\mathscr{P}), \mu)$.

2.3. Nonparametric estimation

Assume $\mu : \mathscr{X} \mapsto \mathbb{R}$ is a Gaussian random field of zero mean and covariance, even referred to as kernel, $K : \mathscr{X} \times \mathscr{X} \mapsto \mathbb{R}_{\geq 0}$. According to standard notation [] we denote the Gaussian random field as $\mu \sim \mathscr{N}(0, K)$. Assume to have a set of m noisy measurements of the form $y^{(h)} = \mu(x^{(h)}) + v^{(h)}$, where $v^{(h)} \sim \mathscr{N}(0, \sigma^2)$ is zero mean Gaussian noise with variance σ^2. Then, by defining the information set I as

$$I = \left\{ \left(x^{(h)}, y^{(h)}\right) \mid h \in \{1, \ldots, m\} \right\},$$

that is the set containing all the m input locations defined as the pairs $(x^{(h)}, y^{(h)})$, as show in Cucker and Smale (2001) and Tikhonov and Arsenin (1977), the minimum variance estimate of μ given I can be computed as:

$$\widehat{\mu}(x) = \mathbb{E}[\mu(x)|I] = \sum_{h=1}^{m} c^{(h)} K(x^{(h)}, x), \quad x \in \mathscr{X}, \tag{4}$$

$\mathbb{E}$ being the expectation operator, where the coefficients $c^{(h)}$'s are given by

$$\begin{bmatrix} c^{(1)} \\ \vdots \\ c^{(m)} \end{bmatrix} = (\bar{K} + \sigma^2 \mathbb{I})^{-1} \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(m)} \end{bmatrix},$$

$$\bar{K} = \begin{bmatrix} K(x^{(1)}, x^{(1)}) & \cdots & K(x^{(1)}, x^{(m)}) \\ \vdots & & \vdots \\ K(x^{(m)}, x^{(1)}) & \cdots & K(x^{(m)}, x^{(m)}) \end{bmatrix}.$$

Moreover, the a posteriori variance of the estimate, in a generic location $x \in \mathscr{X}$, is given by Anderson and Moore (2012)

$$\begin{aligned} V(x) &= \mathrm{Var}[\mu(x)|I] = K(x,x) \\ &\quad - \begin{bmatrix} K(x^{(1)}, x) & \cdots & K(x^{(m)}, x) \end{bmatrix} \\ &\quad \times (\bar{K} + \sigma^2 \mathbb{I})^{-1} \begin{bmatrix} K(x^{(1)}, x) \\ \vdots \\ K(x^{(m)}, x) \end{bmatrix}. \end{aligned} \tag{5}$$

Next, we introduce a useful result which, under the assumption that the estimated sensory function $\widehat{\mu}$ converges to the true function μ, states that the centroidal Voronoi coverage algorithm which exploits the estimated function mimics in probability the behavior of the standard Lloyd algorithm, which uses the true μ, with arbitrary accuracy. Namely, if the algorithms are initialized from the same configuration, then their respective centroids sequences evolve arbitrarily close to each other for an arbitrary but finite time.

Proposition 1 (*Standard Lloyd Mime*). *Assume $\widehat{\mu}_k \xrightarrow{\mathbb{P}} \mu$. Pick any $0 < \delta < 1$, $\varepsilon > 0$ and integer N. There exists a sufficiently large $\bar{k}$ such that, assuming $\mathbf{c}^L_{\bar{k}} = \widehat{\mathbf{c}}_{\bar{k}}$, then*

$$\mathbb{P}\left[\left\| \widehat{\mathbf{c}}_{\bar{k}+k} - \mathbf{c}^L_{\bar{k}+k} \right\| \leq \varepsilon \right] \geq 1 - \delta, \quad k = 0 \ldots, N, \tag{6}$$

where $\xrightarrow{\mathbb{P}}$ denotes convergence in probability in the space of continuous functions (sup-norm) while $\mathbf{c}^L_{\bar{k}} = \widehat{\mathbf{c}}_{\bar{k}}$ means that it is possible to start the classical Lloyd algorithm from the configuration reached by the alternative algorithm at $\bar{k}$.

3. Problem formulation

We consider a group of N robots allowed to move in an area represented by the convex set $\mathscr{X}$. The final goal is to simultaneously estimate an unknown map $\mu : \mathscr{X} \mapsto \mathbb{R}$ and to provide a good partitioning $\mathscr{P}$ to minimize $H(\mathscr{P}, \mathbf{c}(\mathscr{P}), \mu)$. We assume to model the unknown μ as the realization of a zero-mean Gaussian random field with covariance $K : \mathscr{X} \times \mathscr{X} \mapsto \mathbb{R}_{\geq 0}$. We restrict our attention to radial Mercer kernel functions, i.e. $K(a,b) = h(\|a-b\|)$, such that if $\|a-b\| \leq \|c-d\|$ then $h(\|a-b\|) \geq h(\|c-d\|)$ and $K(x,x) = \lambda$, $\forall x \in \mathscr{X}$. Additionally, each agent $i \in \{1, \ldots, N\}$ is assumed to have some mild computation, communication and sensing capabilities. In particular, (i) denoting with x_i its current position, it can take a noisy measures of μ of the form

$$y(x_i) = \mu(x_i) + v_i, \tag{7}$$

where $v_i \backsim \mathscr{N}(0, \sigma^2)$, independent from μ and all other measurement noises v_j; (ii) it can send information either to a central server/base station or to closely located agents; (iii) it can move to a certain target-point b_i assuming a simple robots' dynamics with the following discrete update law:

$$x_{i,k+1} = x_{i,k} + u_{i,k}, \quad \forall i \in \{1, \ldots, N\}, \tag{8}$$

where $x_{i,k} = x_i(kT)$, i.e., each robot can move from location $x_{i,k}$ at time $t = kT$ to any desired location $x_{i,k+1} = b_i$ at time $t = (k+1)T$. The ultimate goal is to position the robots in the centroids of a good partition that minimizes H. Then, a good strategy would initially promote estimation and later coverage of $\mathscr{X}$. The proposed strategy, based on information about the posterior variance, probabilistically drives the robot towards estimation or coverage. Namely, when the posterior variance is high, the robots are more prone to perform estimation. When the value of the variance decreases, the robots are more inclined to perform

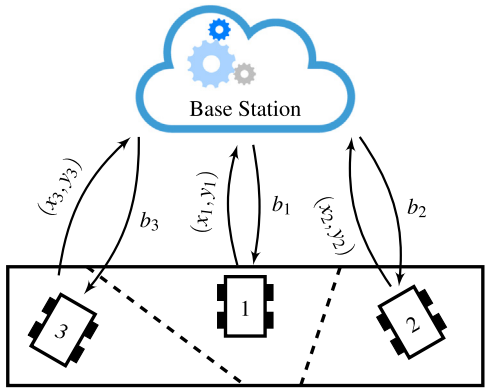


Fig. 1. Client–server communication architecture: robots send their input locations to the server/base station while the server, after performing all the computation, sends to the robots the target points.

coverage. However, conversely to the approach suggested in Choi et al. (2008) for maximum seeking, since we never completely switch from one phase to the other, convergence in probability of $\widehat{\mu}_k$ to μ holds.

4. Server-based algorithm

In this section we present the first algorithm proposed. It is based on a *client–server communication architecture*, see Fig. 1, where the robots are allowed to communicate with a central server which (i) can store all the measurements taken by all the robots; (ii) can store the last position of all the robots; (iii) can compute centroids and Voronoi regions of all the robots; (iv) can send information periodically to all robots every T seconds; (v) can send information to each robot i; (vi) can compute and store an estimate $\widehat{\mu}$ of the function μ and its posterior variance V.

To achieve the goal described in Section 3, we propose the *Server-based algorithm* described in Algorithm 1 (denoted hereafter as SB algorithm). It is convenient to divide the algorithm in two parts. The first describes the operations executed by the server; the second those executed by the robots.

We assume that each robot i, $i = 1, \ldots, N$, collects only one measurement $y_{i,k} = y(x_{i,k})$ of the form (7) within the time window $(kT, (k+1)T)$, $k \in \mathbb{N}$ (line 29). Once the measurement is taken, it is immediately transmitted to the server (line 30) which stores it in memory. Then each robot listen to the server to receive the next target location (line 32). By denoting with J_k the set of measurements received by the server at iteration k, i.e., $J_k := \{(x_{i,k}, y_{i,k}) \mid i = 1, \ldots, N\}$, the complete information set I_k available at the server at iteration k can be computed, $\forall k \geq 1$, simply as (lines 4–8)$I_k = I_{k-1} \cup J_k$, assuming $I_0 = \emptyset$. For notational convenience, it is possible to relabel the elements of I_k as

$$I_k = \left\{(x^{(h)}, y^{(h)}) \mid h \in \{1, \ldots, kN\}\right\},$$

where $(x^{(h)}, y^{(h)}) = \left(x_{i,t}, y_{i,t}\right)$ for $i \in \{1, \ldots, N\}$ and some $t \in \{1, \ldots, k\}$. Then, the server stores in memory an estimate $\widehat{\mu}_k(x)$ of $\mu(x)$ (line 10) and its corresponding posterior variance $V_k(x)$ (line 11) computed according to Eqs. (4)–(5), respectively. The server moreover computes a Voronoi partition $\mathscr{P}_k = \{\mathscr{P}_{1,k}, \ldots, \mathscr{P}_{N,k}\}$ (line 13) and the corresponding list of centroids $\widehat{\mathbf{c}}_k = \{\widehat{c}_{1,k}, \ldots, \widehat{c}_{N,k}\}$ (line 14). It is quite intuitive that in order to improve the quality of the estimate of the function μ, the measurements should be taken to reduce as much as possible the posterior variance $V_k(x)$. To do so, the SB algorithm uses a strictly monotonically increasing function $F(M_{i,k})$ of the maximum $M_{i,k}$ of the posterior (line 16) in the Voronoi region of the agent i at time k. Each robot is forced to perform estimation (line 20)

Algorithm 1 Server-based (SB)

1: **SERVER**
Require: The server stores in memory $\widehat{\mu}_k$,$V_k(x)$, I_k, and has a clock that triggers an event every T seconds.
2: **if** $k = nT$, $n \in \mathbb{N}$ **then**
3: **Listen (input locations reception)**:
4: $J_k = \emptyset$
5: **for** i=1,…,N **do**
6: $J_k = J_k \cup \{x_{i,k}, y_{i,k}\}$
7: **end for**
8: $I_k = I_{k-1} \cup J_k$

9: **Estimate update**:
10: $\widehat{\mu}_k(x) = \mathbb{E}[\mu(x) \mid I_k]$ Eq. (4)
11: $V_k(x) = \mathrm{Var}[\mu(x) \mid I_k]$ Eq. (5)

12: **Partition and centroids update**:
13: $\mathscr{P}_{i,k} = \mathscr{W}_i(\widehat{\mathbf{c}}_{k-1})$, Eq. (1)
14: $\widehat{c}_{i,k} = (\int_{\mathscr{P}_{i,k}} \widehat{\mu}_k(q)dq)^{-1} \int_{\mathscr{P}_{i,k}} q\widehat{\mu}_k(q)dq$.

15: **Target-Points computation**:
16: $M_{i,k} = \max_{x \in \mathscr{P}_{i,k}} V_k(x)$, $\forall i$
17: $p_{i,k} = F(M_{i,k})$
18: $\eta_{i,k} \sim \mathscr{B}(p_{i,k})$
19: **if** $\eta_{i,k} = 1$ **then**
20: $b_{i,k} = \mathrm{argmax}_{x \in \mathscr{P}_{i,k}} V_k(x)$
21: **else**
22: $b_{i,k} = \widehat{c}_{i,k}$
23: **end if**

24: **Target-Points Transmission**:
25: $x_{i,k+1} = b_{i,k}$ (i.e. $u_{i,k} = b_{i,k} - x_{i,k}$), $\forall i$
26: **end if**

27: **ROBOTS**
Require: A clock with sample time T or a submultiple of T.
28: **if** $k = nT$, $n \in \mathbb{N}$ **then**
29: **Measurement collection**: $y_{i,k} = \mu(x_{i,k}) + v_{i,k}$

30: **Measurement transmission**: $\left(x_{i,k}, y_{i,k}\right) \longrightarrow$ Server

31: **Listen (target points reception)**:
32: $x_{i,k+1} = b_i$ (alternatively $u_{i,k} = b_{i,k} - x_{i,k}$)

33: **Move to the new target-point**.
34: **end if**

or to move toward the centroid (line 22) by using a randomized strategy (line 18) where $\mathscr{B}$ denotes the *Bernoulli* distribution and which is based on the value $F(M_{i,k})$, i.e. the higher the value of $F(M_{i,k})$, the higher is the probability to perform an estimation step (line 17–19). If an agent i is selected to perform estimation, the server identifies the next target point for this agent by determining the point with maximum posterior variance in its current region $\mathscr{P}_{i,k}$. The target points are then sent by the server to each robot every period T (line 25).

The following proposition states that, under a mild condition on $F(\cdot)$, which determines whether an agent has to perform either an estimation or a coverage step, the map $\widehat{\mu}_k$, estimated by means the SB algorithm, converges to the true μ.

Proposition 2 (*Convergence of SB Estimate*)**.** *Let us consider the SB algorithm. Let* $F(M) : [0, 1] \to [0, 1]$ *be a continuous and strictly monotonically increasing function such that* $F(M) > 0$ *for* $M > 0$. *Then* $\widehat{\mu}_k \xrightarrow{\mathbb{P}} \mu$.

We remark that the choice for the function $F(\cdot)$ leaves a certain degree of freedom to the designer since it allows to regulate estimation vs. coverage. For example, by choosing $F(M) \approx 1, \forall M$

Algorithm 2 SB-grid

19: **if** $\eta_{i,k} = 1$ **then**
20: $\quad b_{i,k} = \text{argmax}_{x \in \mathcal{P}_{i,k} \cap \mathcal{X}_{\text{grid}}} V_k(x)$
21: **else**
22: $\quad b_{i,k} = \Pi(\widehat{c}_{i,k})$
23: **end if**

(but always strictly increasing), which corresponds to the strategy in which the robots perform almost only exploration, the proposed algorithm can also be interpreted as a cooperative strategy for optimal sampling. Finally, observe that, thanks to Proposition 2, the SB algorithm satisfies Proposition 1 as well.

5. Server-based discretized algorithm

It is well known the non-parametric estimation procedure, described in Section 2.3, suffers from the curse of dimensionality. In particular, due to matrix inversions the computational cost at iteration k is of order $\mathcal{O}(k^3N^3)$. By exploiting a suitable recursive implementation based on the Schur complement to compute the matrix inverse, it is possible to reduce the cost to $\mathcal{O}(k^2N^3)$ which, however, still grows unbounded. The standard approach to address this problem is to discard old measurements (Xu et al., 2012, 2011). However this strategy might lead to poor performance, hard to explicitly quantify. For this reason, we present here an approximated version of the SB algorithm (denoted as SB-grid) which, relying on a spatial discretization of the working area $\mathcal{X}$, is light, fast and with bounded computational burden per iteration. As drawback, since the modified algorithm is based on a spatial discretization, the result stated in Proposition 2 must be relaxed. However, we derive explicit bounds on the estimation error as function of the spacial resolution used to discretize $\mathcal{X}$. The idea is to constrain the robots to collect measurements only from a set of predetermined finite number of input locations which are obtained thanks to a spatial discretization of the continuous convex domain $\mathcal{X}$ formally defined as follows:

Definition 3 (*Sampled Space*)**.** Consider the finite set of m input locations $\mathcal{X}_{\text{grid}} := \{x_{\text{grid},1}, \dots, x_{\text{grid},m}\} \subset \mathcal{X}$ where $\mathcal{X} \subset \mathbb{R}^2$ is a convex and closed polygon. Given the scalar $\Delta > 0$, we say that set $\mathcal{X}_{\text{grid}}$ forms a *sampled space* of resolution Δ if

$$\min_{i=1,\dots,m} \|x_{\text{grid},i} - x\| \leq \Delta, \quad \forall x \in \mathcal{X}. \tag{9}$$

Moreover, it is convenient to introduce the operator Π that projects $x \in \mathcal{X}$ onto its closest point[1] in $\mathcal{X}_{\text{grid}}$ that is

$$\mathcal{X} \longmapsto \mathcal{X}_{\text{grid}} : x \longmapsto \Pi(x) = \arg \min_{a \in \mathcal{X}_{\text{grid}}} \|x - a\|.$$

To force the evolution of the robots on $\mathcal{X}_{\text{grid}}$, lines 19 ÷ 23 of Algorithm 1 are changed as shown in Algorithm 2. Note that, according to line 20, the server now computes the input location owning to $\mathcal{X}_{\text{grid}}$ which maximizes the posterior variance restricted on the grid, for each Voronoi region. This is generally different from projecting the location where the posterior variance maximum is located onto the grid, i.e.,

$$\underset{x \in \mathcal{P}_{i,k} \cap \mathcal{X}_{\text{grid}}}{\text{argmax}} \; V_k(x) \neq \Pi\left(\underset{x \in \mathcal{P}_{i,k}}{\text{argmax}} \, V_k(x)\right).$$

Conversely, line 22 simply says to project the centroids, $\widehat{c} \in \mathcal{X}$, onto the closer points owing to $\mathcal{X}_{\text{grid}}$. Observe that, since the grid is composed of a finite number of locations, new measurements can fall exactly over the same input location. Thus, to avoid storing all the collected measurements, it is convenient to associate to each location $x_i \in \mathcal{X}_{\text{grid}}$, the *virtual measurements* w_{i,ℓ_i} computed as

$$w_{i,\ell_i} = \frac{\ell_i - 1}{\ell_i} w_{i,\ell_i - 1} + \frac{1}{\ell_i} y_i,$$

where y_i is the ℓ_ith measurement taken in x_i. Accordingly, the variance associated to w_{i,ℓ_i} is given by $\sigma_i^2 = \frac{\sigma^2}{\ell_i}$ and we defined with Σ a diagonal matrix collecting these noise variances, i.e. $\Sigma = \text{diag}(\{\sigma_i^2\}_{i=1}^n)$. When two or more measurements are collected on the same input location, the size of the sampled Kernel $\bar{K}$ does not vary. It increases only when an input location on the grid is visited for the first time. When all of these locations are visited at least once, the sampled kernel does not change any more: it reaches its maximum possible size, becoming the $m \times m$ matrix K_{grid}, covariance of μ sampled on $\mathcal{X}_{\text{grid}}$, whose elements are given by $[K_{\text{grid}}]_{ki} = K(x_{\text{grid},k}, x_{\text{grid},i})$. Conversely, every time a new measurement is collected, one has to update the variance matrix Σ and the vector with the virtual measurements $w = [w_{1,\ell_1}, \dots, w_{n,\ell_n}]^T$ with $n \leq m$. The function estimate and its related posterior variance can then be computed using, as usual, Eqs. (4)–(5), evaluated using only the n virtual measurements and their associated noise variances.

The following proposition characterizes of the asymptotic behavior of the SB-grid algorithm.

Proposition 4 (*Convergence of SB-grid Estimate*)**.** *Let us consider the SB-grid algorithm. If $F(M) : [0, 1] \to [0, 1]$ is a continuous and strictly monotonically increasing function such that $F(M) > 0$ for $M > 0$, then*

$$\widehat{\mu}_k(x) \xrightarrow{\mathbb{P}} \mu(x), \quad \forall x \in \mathcal{X}_{\text{grid}}. \tag{10}$$

The following result instead characterizes the asymptotic performance of the estimator on a generic input location, possibly falling outside $\mathcal{X}_{\text{grid}}$. In particular it provides an explicit bound on the estimation error due to the spatial discretization introduced. Before stating it, it is convenient to denote with $k_{\text{grid}}(x)$ the row vector

$$k_{\text{grid}}(x) = \begin{bmatrix} K(x, x_{\text{grid},1}) & \dots & K(x, x_{\text{grid},m}) \end{bmatrix}.$$

Finally, recall that $K(a, b) = h(\|a - b\|)$ and $\lambda = K(x, x)$.

Proposition 5 (*SB-grid Estimation Error Bounds*)**.** *Let the assumptions of Proposition 4 hold and $F(0) = 0$. If $\mathcal{X}_{\text{grid}}$ is a sampled subset of the space $\mathcal{X}$ of resolution Δ, as in Definition 3, one has*

$$\lim_{k \to \infty} V_k(x) = \lambda - k_{\text{grid}}(x) K_{\text{grid}}^{-1} k_{\text{grid}}(x)^\top, \tag{11}$$

where convergence is in probability and holds also uniformly w.r.t. x. In addition, the following uniform bound holds

$$\lim_{k \to \infty} V_k(x) \leq \lambda - \frac{h^2(\Delta)}{\lambda}, \quad \forall x \in \mathcal{X}. \tag{12}$$

Finally, for sufficiently small Δ and the specific case of Gaussian kernel $K(a, b) = \lambda e^{-\frac{\|a-b\|^2}{\zeta^2}}$, we have

$$\lambda - \frac{h^2(\Delta)}{\lambda} \approx \frac{\lambda \Delta^2}{\zeta^2}. \tag{13}$$

[1] If such point is not unique, we randomly select one point in the minimizer set.

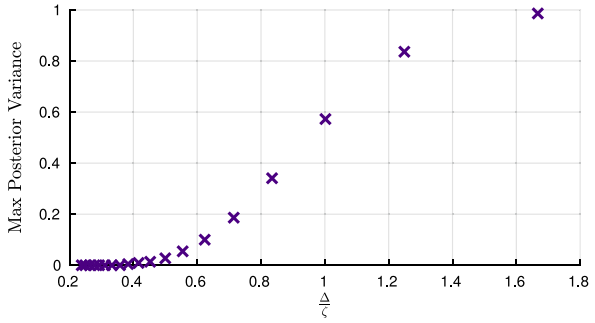


Fig. 2. Asymptotic values of the max of the posterior V as function of $\frac{\Delta}{\zeta}$ for a Gaussian kernel with standard deviation ζ (Eq. (11)).

When adopting the grid-based strategy, measurements can be collected only over the input locations contained in the finite set $\mathscr{X}_{\text{grid}}$. Using basic results on estimation of Gaussian processes, one can see that Eq. (11) is the posterior variance of μ conditional on the perfect knowledge of the function on the grid. Hence, the above result shows that our updating mechanism ensures convergence to the minimum possible error compatible with $\mathscr{X}_{\text{grid}}$. Eq. (12) then shows how the posterior variance can be made uniformly and arbitrarily small by choosing a Δ sufficiently small. In particular, from (13) one sees that the error converges to zero at least quadratically in Δ when a Gaussian kernel is adopted. Fig. 2 shows the behavior of the bound of the asymptotic posterior variance of Eq. (11).

6. Distributed gossip algorithm

Here, we present a distributed algorithm to solve the problem of estimation and coverage. Differently from the client–server communication architecture used in both the SB and SB-grid algorithms, here the robots exploit a *peer-to-peer* communication architecture. According to this architecture, each robot (peer) is allowed to communicate with one neighboring peer at a time in order to exchange local information. The idea is to combine a local estimation performed by each robot, based on all its collected measurements, with the *gossip coverage algorithm* presented in Bullo, Carli, and Frasca (2012). To be more precise, each robot is assumed to be able to:

- store its collected measurements in the local set I_i;
- store its region of competence $\mathscr{P}_i$;
- compute a local estimate $\widehat{\mu}_i|_{\mathscr{P}_i}$ of μ and the corresponding posterior variance $V_i|_{\mathscr{P}_i}$ over $\mathscr{P}_i$, based on I_i;
- compute its local centroid $\widehat{c}_i$, according to $\widehat{\mu}_i$;
- communicate with only one robot at a time (bidirectional gossip or peer-to-peer communication), to update $\mathscr{P}_i$.

The algorithm we propose, which we refer to as *Distributed Gossip algorithm* (denoted hereafter as DG) is formally described in Algorithms 3 and 4.

Observe that the algorithm consists of two main parts, namely, (i) *local estimation and coverage* (Algorithm 3); and (ii) *gossip communication and partitions update* (Algorithm 4). The two phases are completely asynchronous and uncorrelated one from the other. During phase (i) each robot acts independently according to a local clock. For ease of notation we used the subscript k to identify the local time instants t_k^i. Phase (ii) of the algorithm concerns with the communication between agents and the update of the partition. Observe that this point is crucial for the distributed implementation of the algorithm. Indeed, the partition update cannot be implemented according to the classical approach since to iteratively update the partition $\mathscr{P}_k$ following Lloyd's equation (3), full information about the current, at the next centroids of all robots are required. Thus, in a distributed framework, where only local exchange of partial information is allowed, this cannot be achieved. To workaround this issue, following the approach proposed in Bullo et al. (2012), we implement a gossip coverage in which only a pair of robots, say (i, j), establishes a bidirectional gossip-like communication: firstly the robots check if they are neighbors, namely, if

$$\mathscr{P}_i \cap \mathscr{P}_j \neq \emptyset; \tag{14}$$

secondly, if the above condition is satisfied, the robots exchange their centroids, and update the partition as

$$\begin{aligned} \mathscr{P}_{i,k} &= \left\{ q \in \bigcup_{\ell \in \{i,j\}} \mathscr{P}_{\ell,k-1} \,\middle|\, \|q - \widehat{c}_{i,k}\| \leq \|q - \widehat{c}_{j,k}\| \right\}, \\ \mathscr{P}_{j,k} &= \left\{ q \in \bigcup_{\ell \in \{i,j\}} \mathscr{P}_{\ell,k-1} \,\middle|\, \|q - \widehat{c}_{j,k}\| \leq \|q - \widehat{c}_{i,k}\| \right\}. \end{aligned} \tag{15}$$

If they are no neighbors, then, no update is performed, i.e., $\mathscr{P}_{i,k} = \mathscr{P}_{i,k-1}$, $\mathscr{P}_{j,k} = \mathscr{P}_{j,k-1}$. Note that, only agents i, j are involved in the updating process at iteration k which means that all the other partitions remain unchanged, namely, $\mathscr{P}_{\ell,k} = \mathscr{P}_{\ell,k-1}$, $\forall\, \ell \neq i, j$. In general, the T_k iterations, describing phase (ii) of the DG algorithm, are in general different from all the t_k^i. Moreover, during the interval $[T_{k-1}, T_k]$ the partition $\mathscr{P}$ does not change and the robots perform phase (i). Similarly to Proposition 2, we are able to prove asymptotic convergence of the estimated function $\widehat{\mu}$ to μ.

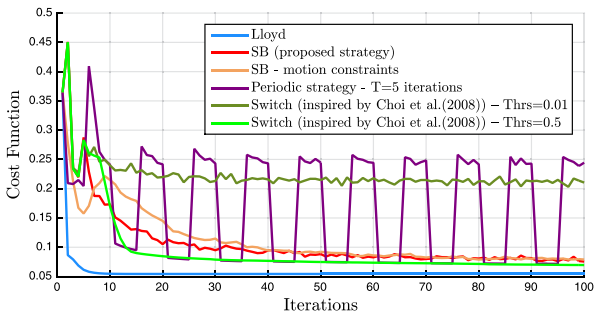
Algorithm 3 DG – Local Estimation and Coverage

Require: Each robot stores in its local memory $\widehat{\mu}_{i,k}$, $V_{i,k}(x)$, $I_{i,k}$.
for $t_1^i, t_2^i, \ldots, t_k^i, \ldots$ **do**
  Measurement collection:
  $I_{i,k} = I_{i,k-1} \cup \{x_{i,k}, y_{i,k}\}$
  Estimate update:
  $\widehat{\mu}_{i,k}|_{\mathscr{P}_{i,k}} = \mathbb{E}\left[\mu(x) \mid I_{i,k}\right]$ constrained to $\mathscr{P}_{i,k}$
  $V_{i,k}|_{\mathscr{P}_{i,k}} = \text{Var}\left[\mu(x) \mid I_{i,k}\right]$ constrained to $\mathscr{P}_{i,k}$
  Centroid update:
  $\widehat{c}_{i,k} = (\int_{\mathscr{P}_{i,k}} \widehat{\mu}_{i,k}(q)dq)^{-1} \int_{\mathscr{P}_{i,k}} q\widehat{\mu}_{i,k}(q)dq.$
  Target-Point computation and Movement:
  $M_{i,k} = \max_{x \in \mathscr{P}_{i,k}} V_{i,k}(x)$
  $p_{i,k} = F(M_{i,k})$
  $\eta_{i,k} \sim \mathcal{B}(p_{i,k})$
  if $\eta_{i,k} = 1$ **then**
    $x_{i,k+1} = \text{argmax}_{x \in \mathscr{P}_{i,k}} V_{i,k}(x)$
  else
    $x_{i,k+1} = \widehat{c}_{i,k}$
  end if
end for

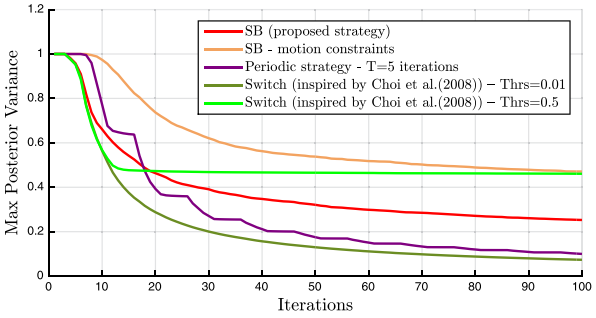
Algorithm 4 DG – Gossip Comm. and Partitions Update

for $T_1, T_2, \ldots, T_k, \ldots$ **do**
  Two robots, say i,j, communicate with each other
  if i and j are neighbors (i.e., Eq. (14) is verified) **then**
    Centroids transmission:
    The robots exchange their centroids $\widehat{c}_{i,k}, \widehat{c}_{j,k}$
    Partitions update:
    They update their partitions according to Eq. (15)
  end if
end for

Proposition 6 (*Convergence of DG Estimate*). *Assume* $F(\cdot) : [0, 1] \mapsto [0, 1]$ *is continuous and strictly monotonically increasing*



(a) Evolution of the cost function H of Eq. (2) computed with respect to the robots' positions.



(b) Evolution of the max M of the posterior variance V.

Fig. 3. Comparison of the proposed SB algorithm against standard Lloyd, the SB under motion constraints, the Switch algorithm inspired by Choi et al. (2008), for two different values of switching threshold, and the Periodic Strategy with period $T = 5$ iterations. Average over 100 Monte Carlo simulations.

and satisfying $\sum_k F(\sigma^2/k) = \infty$. *Then* $\widehat{\mu}_k \xrightarrow{\mathbb{P}} \mu$, *where the estimate is point-wise defined as*

$$\widehat{\mu}_k(x) := \widehat{\mu}_{i,k}(x) \quad \text{s.t. } x \in \mathscr{P}_{i,k},\ i = \{1, \dots, N\}.$$

Note that, thanks to Proposition 6, a result similar to Proposition 1 holds for the DG algorithm as well. In this case, instead of the classical Lloyd, it is possible to mimic in probability the evolution of the Gossip Coverage (Bullo et al., 2012) using μ.

Some remarks about the implementation of the DG algorithm are due. First note that $\{\mathscr{P}_i\}_{i=1}^N$ no more coincide with standard Voronoi partitions. They can indeed consist of non-convex regions. Yet, as proved in Bullo et al. (2012), the partitions converge to a set of proper centroidal Voronoi partitions. Nonetheless, in general, a non-convex set can be expressed as the union of a finite number of convex polygons with disjoint interiors and that each polygon can be suitably described as an ordered list of vertices. Thus, in practice, condition (14) is verified if two robots share a piece of their boundaries which can be checked by the robots just exchanging lists of point. Once verified (14), two robots need to update their partitions. It is common engineering practice to discretize $\mathscr{X}$ into a finite number of cells. In this case, the partitions consist of subsets of cells and the robot, to perform (15), simply need to exchange subsets of cells between each other (Durham et al., 2012). Finally, even under robots' motion constraints (see Section 7), a careful analysis of the proof shows that probabilistic convergence still holds just requiring that, during the estimation phase, a neighborhood of radius $\varepsilon > 0$ of the location where the maximum of the posterior is attained is reached with probability not less than a constant $M(\varepsilon) > 0$.

7. Simulations

In this section we provide some simulations showing the performances of the proposed algorithms. All the simulations are run in MATLAB on an Intel Core i7-4790 desktop machine with 8 GB of RAM. We consider a team of $N = 8$ robots in the normalized squared domain $\mathscr{X} = [0, 1] \times [0, 1]$. As kernel, we choose the Gaussian

$$K(x, x') = \lambda e^{-\frac{\|x-x'\|^2}{2\zeta^2}}, \quad \lambda = 1,\ \zeta = 0.2.$$

The unknown sensory function μ is chosen to be a combination of two bi-dimensional Gaussians[2]:

$$\mu(x) = 5 \sum_{i=1}^{2} e^{-\frac{\|x-\mu_i\|^2}{0.01}}, \qquad \mu_1 = \begin{bmatrix} 0.8 \\ 0.2 \end{bmatrix}, \qquad \mu_2 = \begin{bmatrix} 0.5 \\ 0.7 \end{bmatrix}.$$

[2] Given the universal representing properties of the Gaussian kernel, we do not report tests with different sensory functions.

We analyze the behavior of the algorithms for $F_\alpha(M) = M^\alpha$, $\alpha = 2$, being the max of posterior variance M normalized between $[0, 1]$. This choice forces the robot to be slightly more prone to perform estimation rather than coverage. It is recalled that by changing α it is possible to force the robots to be more prone either to one phase or the other.

7.1. Comparison among centralized strategies

Here, the SB is compared against the following algorithms:

- The standard Lloyd algorithm (3), which assumes perfect knowledge of μ.
- A "Switch" strategy, inspired by Choi et al. (2008), in which the robots, after an *estimation-only* phase, when the maximum of the posterior variance reaches a predefined threshold, switch to a *coverage-only* phase permanently. We compare the performance for two values of the threshold.
- A "Periodic Strategy" in which robots perform estimation and coverage periodically for T iteration each.
- Also, we simulate our SB algorithm in the presence of motion constraints. Namely, the robots are constrained within a disk of radius $r = 0.1$.

We compare the algorithms in terms of (i) cost function $H(\mathscr{P}, \mathbf{x}, \mu)$ (Eq. (2)) computed with respect to the robots positions; and (ii) evolution of the maximum of the posterior variance defined as $M_k = \max_{x \in \mathscr{X}} V_k(x)$. The choice of plotting H w.r.t. $\mathbf{x}$ rather than $\mathbf{c}(\mathscr{P})$ is dictated by the simple fact that the robots, conversely to the standard Lloyd algorithm, do not always lie on the corresponding centroids. Thus, we believe that $H(\mathscr{P}, \mathbf{x}, \mu)$ better captures the actual coverage cost. Observe that this choice is consistent with the standard Lloyd. Indeed, since $\mathbf{x} \equiv \mathbf{c}^L$, in this case we retrieve the standard *locational optimization cost*.

Fig. 3(a) shows the evolution of H as a function of number of iterations k. As expected, the classical Lloyd is characterized by a non-increasing behavior. Differently, the other algorithms are characterized by a non-monotonic behavior. This is due to the fact that they do not always move on the centroids of the partition. Indeed, in the SB the robots randomly move either to the estimated centroids or to the locations where the maximum of the posterior variance is attained. Differently, the behavior of the Switch algorithm inspired by Choi et al. (2008) largely depends on the values of the threshold. Indeed, for small values the robots are forced to perform estimation for an unreasonable number of iterations. For large values the robots are not able to perform a good function estimation as suggested by the evolution of the posterior variance shown in Fig. 3(b). Moreover, far any value of the threshold, the estimated function never converges to the true one. Regarding the Periodic strategy, even if it should be able

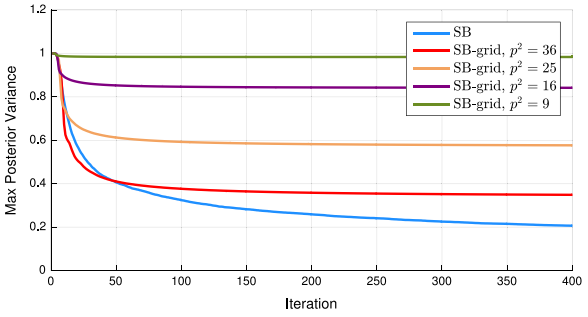


Fig. 4. Comparison between the SB and the SB-grid algorithms for different total number of points p^2: evolution of the max of the posterior variance averaged over 100 Monte Carlo runs.

Table 1
SB vs. SB-grid algorithms for different total number of points p^2.

p^2	9	16	25	36	SB
Exe. time (s)	2.3	2.4	3.9	4.7	865.4

to asymptotically reconstruct the sensory function, it does not well behave in terms of coverage cost. Finally, the proposed SB under practical motion constraints is characterized by a slower convergence rate but still behaves well in terms of coverage. Thus, the proposed probabilistic strategy, by letting the robots to automatically trade-off between estimation and coverage is able to simultaneously estimate the true sensory function while providing a good coverage performance.

7.2. Comparison between the SB and SB-grid algorithms

In this section we compare the SB and the SB-grid algorithms. Since $\mathscr{X}$ has been chosen to be equal to the square set $[0, 1]^2$, it is convenient to let $\mathscr{X}_{\text{grid}}$ be a grid of p equally spaced points per side. Thus, $|X_{\text{grid}}| = p^2$. Fig. 4 shows the performance, in terms of max of the posterior variance, of the algorithms for different levels of space discretization, i.e., different total number of points p^2. It can be seen that the grid based approximation is slightly faster during the first iterations but it reaches its asymptotic estimation error variance in more or less 150 iterations. Conversely, in the SB the max of the posterior decreases, asymptotically converging to zero. However, in terms of execution time the SB-grid algorithm is much lighter than the SB algorithm, see Table 1. This is the major advantage of using this implementation since, given a desired final value for the max of the posterior variance according to Eq. (11) of Proposition 5 (see Fig. 2 as well which shows the limit values of the max of the posterior as function of Δ/ζ), the execution time can be reduced of several orders of magnitude.

7.3. Comparison among distributed strategies

In this section we compare the proposed DG algorithm with an algorithm, which we refer to as DG Switch, in the same flavor the Switch strategy presented in the previous sections. In the DG Switch, as in the DG algorithm, the robots perform estimation based on their own set I_i and communicate only with neighboring peers to update their partitions. However, differently from the our DG approach, the controller, instead of driving the robots based on a stochastic strategy, forces the robots to perform estimation until the maximum of the posterior variance, computed within each robot's partition, reaches a certain threshold. After that, the robots permanently switch to coverage. Fig. 5(a)–(b) plot the evolution of the cost function H and of the maximum among the maxima of the posterior variances of all the robots, i.e.,

$$M_k := \max_{i\in\{1,\dots,N\}} \left(\max_{x\in\mathscr{P}_{i,k}} V_{i,k}(x) \right).$$

Similarly to the corresponding centralized strategies, the proposed DG algorithm falls in between the DG Switch, which has been simulated for two different values of the predefined threshold. Once again for small values of the threshold the DG Switch keeps doing estimation not being able to perform a good coverage. While for large values it is not able to perform estimation. As concluding remark, it is worth stressing that, even if from Fig. 5(b) the DG Switch seems to outperform the proposed strategy, the DG algorithm converges to the true sensory function and thus $M_k \to 0$ asymptotically.

8. Conclusions

In this work we considered the problem of multi-robot simultaneous estimation and coverage via non-parametric estimation. We explored two different communication architectures, namely client–server (centralized) and peer-to-peer (distributed), for which we were able to prove convergence in probability of the estimated map to the true map while also providing asymptotic behavior of the robots similar to the one provided by the Loyd's algorithm for optimal coverage. The transition from exploration to coverage is smooth and can be easily tuned via the transition function $F(\cdot)$. We also provided an efficient numerical approximated estimation algorithm for which the asymptotic estimation error can be computed a-priori. One interesting future direction regards the extension to dynamic scenarios.

Appendix

A.1. Proof of Proposition 1

Let $\mathscr{U}$ be the set of all the continuously differentiable sensory functions defined over $\mathscr{X}$. Let $\mathscr{U}$ be equipped with a norm $\|\cdot\|$, for instance, $\|\mu\| = \max_{x\in\mathscr{X}} \mu(x)$.

Let us define $G(c;\mu) : \mathscr{X}^N \times \mathscr{U} \mapsto \mathscr{X}^N$ as $G(c;\mu) := \mathbf{c}(\mathscr{W}(c))$, where the operator $\mathbf{c}$ computes the centroids according to the sensory function μ. It is known that the map G, above defined, is continuous on both arguments (see Bullo et al., 2012). Observe that $c^L_{k+1} = G\left(c^L_k;\mu\right)$ and $\hat{c}_{k+1} = G\left(\hat{c}_k;\hat{\mu}_k\right)$.

We can write

$$\begin{aligned}\|\hat{c}_{k+1} - c^L_{k+1}\| &= \left\|G\left(\hat{c}_k;\hat{\mu}_k\right) - G\left(c^L_k;\mu\right)\right\| \\ &\le \left\|G\left(\hat{c}_k;\hat{\mu}_k\right) - G\left(\hat{c}_k;\mu\right)\right\| + \left\|G\left(\hat{c}_k;\mu\right) - G\left(c^L_k;\mu\right)\right\|.\end{aligned}$$

For continuity of the operator G on both $\mathscr{X}^N$, which is compact, and on $\mathscr{U}$, it follows that there exist $L_\mu > 0$ and $\bar{L}_\mu = \max_{c\in\mathscr{X}^N} L_{\mu;c}$ such that

$$\begin{aligned}\left\|G\left(\hat{c}_k;\mu\right) - G\left(c^L_k;\mu\right)\right\| &\le L_\mu \|\hat{c}_k - c^L_k\|, \\ \left\|G\left(\hat{c}_k;\hat{\mu}_k\right) - G\left(\hat{c}_k;\mu\right)\right\| &\le \bar{L}_\mu \|\hat{\mu}_k - \mu\|.\end{aligned}$$

Hence $\|\hat{c}_{k+1} - c^L_{k+1}\| \le L_\mu \|\hat{c}_k - c^L_k\| + \bar{L}_\mu \|\hat{\mu}_k - \mu\|$. Assuming there exists a positive integer $\bar{k}$ such that $\|\hat{\mu}_k - \mu\| \le \xi$ for all $k \ge \bar{k}$, and such that $\hat{c}_{\bar{k}} = c^L_{\bar{k}}$, then it follows, for $k > \bar{k}$,

$$\|\hat{c}_k - c^L_k\| \le \sum_{j=\bar{k}}^{k} L_\mu^{(j-\bar{k}-1)} \bar{L}_\mu \xi.$$

To conclude the proof, note that from the assumption that $\widehat{\mu}_k \xrightarrow{\mathbb{P}} \mu$ and the definition of convergence in probability, it follows that for

any positive integer N, positive real number ε and real number δ such that $0 < \delta < 1$, there exist a positive real number $\xi > 0$ and a positive integer number $\bar{k}$ such that the following two facts are verified

- $\mathbb{P}\left[\left\|\hat{\mu}_k - \mu\right\| \leq \xi \text{ for all } k \in \left\{\bar{k}, \ldots, \bar{k}+N\right\}\right] \geq 1-\delta$,
- $\sum_{j=\bar{k}}^{N} L_{\mu}^{(j-\bar{k})} \bar{L}_{\mu} \xi \leq \varepsilon$.

A.2. Proof of Proposition 2

Without loss of generality, a system with only one agent is considered. For every $\varepsilon > 0$, define the process $\bar{x}_k$ as follows:

$$\bar{x}_k = \begin{cases} x_k & \text{if } M_k \geq \varepsilon \\ C_k & \text{with probability } (1-p_\varepsilon) \text{ if } M_k < \varepsilon \\ E_k & \text{with probability } p_\varepsilon \text{ if } M_k < \varepsilon \end{cases} \tag{A.1}$$

where $p_\varepsilon = \min_{x\in[\varepsilon,1]} F(x)$, C_k is the location of the centroid and E_k is the location where the posterior variance attains its maximum. We also define $\bar{I}_k$ as the set of measurements collected by the process $\bar{x}_k$ up to instant k, $\bar{V}_k(x) = \mathrm{Var}\left[\mu(x)|\bar{I}_k\right]$ the posterior variance at the input location x associated to the process $\bar{x}_k$ and $\bar{M}_k$ the maximum of $\bar{V}_k$ w.r.t. x. First, we prove that $\forall \varepsilon \geq 0$, $\forall \delta \in (0, 1]$ there exists k_0 such that, $\forall k \geq k_0$, one has $\mathbb{P}[M_k \leq \varepsilon] \geq 1-\delta$. Note that, in view of the definition of x_k and $\bar{x}_k$, the two processes coincide at k and/or it holds that $M_k \leq \varepsilon, \bar{M}_k \leq \varepsilon$. Then, one always has

$$\mathbb{P}\left[\bar{M}_k \leq \varepsilon\right] = \mathbb{P}\left[M_k \leq \varepsilon\right]. \tag{A.2}$$

In view of (A.2), our strategy is to prove that $\bar{x}_k$ satisfies the desired condition which then will immediately extend to x_k. As also clear in the sequel, the key advantage of using $\bar{x}_k$, in place of x_k, is that it avoids the introduction of conditional probability measures difficult both to define and to handle.

Now, consider the subsequence $\bar{x}_{k_j}$ extracted by $\bar{x}_k$ such that $k_i < k_{i+1}$ for every i and for every k_j the agent is moving to E_{k_j}. The length of this subsequence is infinite with probability one since $\bar{x}_k$ can move to the maximum posterior variance location with probability at least p_ε at every k. It is an elementary algebraic fact that, for every $\varepsilon > 0$, there exists a pair $\bar{\alpha}$ and $\bar{m}$ such that:

$$\lambda - \frac{(\lambda-\bar{\alpha})^2}{\lambda + \frac{\sigma^2}{\bar{m}}} \leq \varepsilon.$$

By continuity of the kernel, there exists a finite partition, function of $\varepsilon, \bar{\alpha}, \bar{m}$, given by subsets $\mathscr{D}_j \subseteq \mathscr{X}$ such that $K(x, x^*) \geq \lambda - \bar{\alpha}$, $\forall x, x^* \in \mathscr{D}_j$ (recall that $K(x,x) = \lambda$). Since there is a finite number of subsets $\mathscr{D}_j$, at least one of them is visited infinite times by the subsequence $\bar{x}_{k_j}$ with probability one. This implies that, with probability not smaller than $1-\delta$, there always exists a time instant k_a such that $\mathscr{D}_j$ has been visited at least $\bar{m}$ times and another instant $k_b > k_a$ where $\mathscr{D}_j$ is visited again. Now it is not restrictive to consider only $\bar{m}$ measurements falling in $\mathscr{D}_j$, denoted by $z_1^j, \ldots, z_{\bar{m}}^j$ and collected on the input locations $\bar{x}_1^j, \ldots, \bar{x}_{\bar{m}}^j$. Let $\bar{K}_j$ be the $\bar{m}\times\bar{m}$ kernel matrix with (k,i) entry $[\bar{K}_j]_{ki} = K(\bar{x}_k^j, \bar{x}_i^j)$, i.e. obtained sampling the kernel K on the input locations falling in $\mathscr{D}_j$. We have $Tr(\bar{K}_j) = \sum \Lambda(\bar{K}_j) = \bar{m}\lambda$, where $\Lambda(\bar{K}_j)$ is the set of real and non negative eigenvalues of $\bar{K}_j$. Then, one has $\bar{K}_j \preceq \bar{m}\lambda\mathbb{I}$ so that

$$(\bar{K}_j + \sigma^2\mathbb{I}) \preceq (\bar{m}\lambda + \sigma^2)\mathbb{I} \Rightarrow (\bar{K}_j + \sigma^2\mathbb{I})^{-1} \succeq (\bar{m}\lambda + \sigma^2)^{-1}\mathbb{I}.$$

It comes that, with probability at least $1-\delta$, for every input location $x \in \mathscr{D}_j$ one has

$$\begin{aligned} \mathrm{Var}\left[\mu(x)|\bar{I}_k\right] &\leq \mathrm{Var}\left[\mu(x)|z_1^j, \ldots, z_{\bar{m}}^j\right] = K(x,x) \\ &\quad - \begin{bmatrix} K(\bar{x}_1^j, x) & \cdots & K(\bar{x}_{\bar{m}}^j, x) \end{bmatrix} \\ &\quad \times (\bar{K}_j + \sigma^2\mathbb{I})^{-1} \begin{bmatrix} K(\bar{x}_1^j, x) \\ \vdots \\ K(\bar{x}_{\bar{m}}^j, x) \end{bmatrix} \\ &\leq \lambda - \frac{\sum_{h=1}^{\bar{m}} K(\bar{x}_h^j, x)^2}{\bar{m}\lambda + \sigma^2} \leq \lambda - \frac{\bar{m}(\lambda-\bar{\alpha})^2}{\bar{m}\lambda + \sigma^2} \\ &= \lambda - \frac{(\lambda-\bar{\alpha})^2}{\lambda + \frac{\sigma^2}{\bar{m}}} \leq \varepsilon. \end{aligned}$$

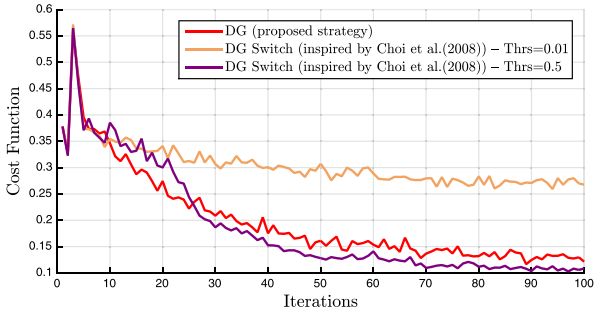
The above equations show that, with probability not smaller than $1-\delta$, $\max_{x\in\mathscr{D}_j} \bar{V}_{k_a}(x) \leq \varepsilon$ and $\bar{M}_{k_b} \leq \varepsilon$. In fact, since at instant k_b the subset $\mathscr{D}_j$ is visited again, the input location where $\bar{V}_{k_b}$ is maximized falls again in $\mathscr{D}_j$ so that the maximum of the posterior variance over the entire domain cannot be large than ε. In view of (A.2), this implies that $\mathrm{Var}[\mu(x)|I_k]$ is converging uniformly to zero in probability. Now, define $\widetilde{\mu}_k = \mu - \widehat{\mu}_k$. From the Chebyshev inequality, we have that $\forall x, s$

$$\mathbb{P}\left[|\widetilde{\mu}_k(x)| \leq sd \mid \mathrm{Var}\left[\mu(x)|I_k\right] = d^2\right] \geq 1 - 1/s^2. \tag{A.3}$$

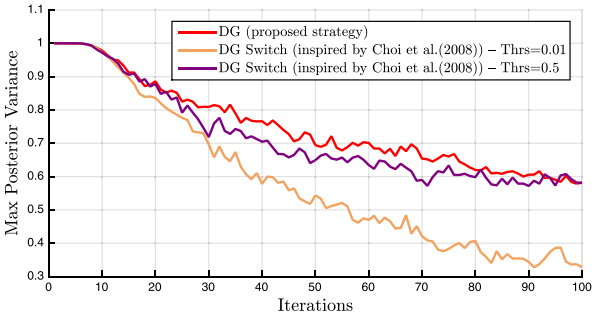
Note that the conditioning event is the posterior variance of the reconstruction error which has been just proved to go uniformly to zero in probability. Now, fix another arbitrary $\varepsilon > 0$, $0 < \delta < 1$ and define $\bar{\delta} = 1 - \sqrt{1-\delta}$. We can find k_0 such that $\forall k \geq k_0$, with probability at least $1-\bar{\delta}$, one has

$$\max_x \mathrm{Var}\left[\mu(x)|I_k\right] \leq \bar{\delta}\varepsilon^2.$$

Setting $s^2 = 1/\bar{\delta}$ in the Chebyshev inequality above, since the conditioning event holds with probability at least $1-\bar{\delta}$, we obtain



(a) Evolution of the cost function H of Eq. (2) computed with respect to the robots' positions.



(b) Evolution of the maximum $M_k = \max_i(\max_{x\in\mathscr{D}_{i,k}} V_{i,k}(x))$.

Fig. 5. Comparison of the proposed DG algorithm against the DG Switch algorithm inspired by Choi et al. (2008), for two different values of switching threshold. Average over 100 Monte Carlo simulations.

that $\forall k \geq k_0$ the event

$$|\widetilde{\mu}_k(x)| \leq sd = \varepsilon\sqrt{\bar{\delta}}/\sqrt{\bar{\delta}} = \varepsilon$$

has probability not smaller than $(1-s^{-2})(1-\bar{\delta}) = (1-\bar{\delta})^2 = 1-\delta$. This shows that, in probability, $\widetilde{\mu}_k$ is going to zero in the sup-norm topology and concludes the proof.

A.3. Proof of Propositions 4 and 5

As already noticed in Section 5, when measurements can be collected only over the input locations contained in the finite set $\mathscr{X}_{\text{grid}}$, the estimation process at a generic instant k is equivalent to reconstructing the function μ from measurements $w_{i,\ell_i} = \mu(x_{\text{grid},i}) + v_i$, where, conditional on the process history up to instant k, $v_i \sim \mathcal{N}(0, \sigma_i)$, $\sigma_i^2 = \frac{\sigma^2}{\ell_i}$ being ℓ_i the number of visits at $x_{\text{grid},i}$. The proof of Proposition 2 can be now followed just replacing the function domain $\mathscr{X}$ with $\mathscr{X}_{\text{grid}}$, with the Voronoi regions covering the entire $\mathscr{X}$ defined at every instant k by a map having as arguments only the estimates of the m random variables $\mu(x_{\text{grid},i})$. One then obtains that $\ell_i \to \infty$ for $i = 1, \ldots, m$, i.e. the posterior variances of all the $\mu(x_{\text{grid},i})$ go to zero. Hence, Eqs. (10) and (11) immediately follow. To obtain Eq. (12), note the following two facts. First, given any $x \in \mathscr{X}$ there exists $x_{\text{grid},i} \in \mathscr{X}_{\text{grid}}$ such that $\|x - x_{\text{grid},i}\| \leq \Delta$. Second, the r.h.s. in (12) is exactly the posterior variance of $\mu(x)$ conditional on the perfect knowledge of $\mu(x_{\text{grid},i})$ with $\|x - x_{\text{grid},i}\| = \Delta$. Eq. (12) is then obtained recalling that, if $\|a - b\| \leq \|c - d\|$ then $h(\|a - b\|) \geq h(\|c - d\|)$ and $K(x, x) = \lambda$, $\forall x \in \mathscr{X}$. Finally, Eq. (13) is just the expansion of the r.h.s. of (12) around $\Delta = 0$ for the Gaussian kernel case.

A.4. Proof of Proposition 6

The proof consists of two steps. First, for each agent we prove that it performs infinitely many times "estimation", that is, it moves infinitely many times on the max of the posterior variance computed inside its region. Second, we prove that, given any arbitrarily small value ε, for every agent, the posterior variance goes below ε over any point of its region. More specifically, by denoting with $\mathscr{P}(k)$ the sequence of regions associated to one robots, we show that $\forall\varepsilon\ \exists k_0$ s.t. $\forall k \geq k_0$, $\max_{x\in\mathscr{P}(k)} V_k(x) \leq \varepsilon$. Now, by letting $v_k := \frac{\lambda\sigma^2}{k\lambda+\sigma^2}$ and assuming to collect one measurement per iteration, if a is an integer satisfying $a \geq \sigma^2/\lambda$, one easily has $M_k := \max_x V_k(x) \geq v_k \geq \frac{\sigma^2}{k+a}$. Monotonicity of $F(\cdot)$ then ensures that

$$\sum_{k=1}^{\infty} F(M_k) \geq \sum_{k=1}^{\infty} F(v_k) \geq \sum_{k=a+1}^{\infty} F(\sigma^2/k) = \infty. \tag{A.4}$$

Letting $\chi(\cdot)$ be the indicator function of an event, for any agent it holds that

$$\begin{aligned} \#\text{ of estimation events} &= \sum_k \chi(u_k \leq F(M_k)) \\ &\leq \sum_k \chi(u_k \leq F(v_k)), \end{aligned} \tag{A.5}$$

where u_k are independent uniform random variables on the unit interval. Combining Eqs. (A.4), (A.5) and the second Borel–Cantelli lemma,[3] we have

$$\mathbb{P}\Big[\text{any robot performs infinitely many times estimation}\Big] = 1. \tag{A.6}$$

[3] To apply the Borel–Cantelli lemma, (A.5) is necessary. Indeed, since the M_k are correlated r.v., $\chi(u_k \leq F(M_k))$ are correlated too, while the $\chi(u_k \leq F(v_k))$ are all mutually independent.

Hereby, ω denotes an element in the probability space contained in the event in the lhs of (A.6). Then, consider the trajectory of one agent corresponding to ω. Similarly to Appendix A.2, by continuity of the kernel, there exists a finite partition, function of ε, given by subsets $\mathscr{D}_j \subseteq \mathscr{X}$ such that the value of the max of the posterior in $\mathscr{D}_j$ goes below ε if we take at least $m(\varepsilon)$ measurements inside $\mathscr{D}_j$.

Recall the regions associated to the robot are denoted by $\mathscr{P}(k)$. To each $\mathscr{P}(k)$ we associate I_k which is the union of smallest group of $\mathscr{D}_j$ containing $\mathscr{P}(k)$, i.e.,

$$I_k := \left\{ \bigcup_j \mathscr{D}_j \mid \mathscr{D}_j \cap \mathscr{P}(k) \neq \emptyset,\ \mathscr{P}(k) \subseteq \bigcup_j \mathscr{D}_j \right\}.$$

One $\mathscr{D}_j$ contributes to the computation of the max of the posterior if for infinitely many times has non null intersection with the $\mathscr{P}(k)$. Indeed, if $\mathscr{D}_j$ is not intersected infinitely many times by the $\mathscr{P}(k)$, there exists $\bar{k}$ such that $\mathscr{D}_j$ does not influence the computation of the max for $k \geq \bar{k}$.

Fix one $\mathscr{D}_\ell$ which contributes to the computation of the max. Let us define the subsequence S_k from I_k such that (i) $\mathscr{D}_\ell \subseteq S_k$, $\forall k$ and, (ii) if the pair (k, j) satisfies $\mathscr{D}_j \subseteq S_k$, then the event $\mathscr{D}_j \subseteq S_k$ occurs for infinitely many k.

The length of S_k is infinite because, by construction, $\mathscr{D}_\ell$ intersects I_k for an infinite number of k and I_k can be chosen among a finite number of subsets independent of k.

Now, partition the $\mathscr{D}_j$ contained in all the S_k into two groups (of arbitrary cardinality) denoted by A and B. For the sake of contradiction, assume that the max of the variance in any $\mathscr{D}_j \in A$ goes below ε and that this does not happen in B. There exist two scenarios: (i) S_k contains infinitely many times only $\mathscr{D}_j \in A$ or only $\mathscr{D}_j \in B$. This is absurd for construction. (ii) S_k contains infinitely many times a non null intersection of A and B. Thus, at least one $\mathscr{D}_j \in B$ is visited infinitely many times (for estimation purposes) since the value of the posterior is larger than that over A. Then, the value of the posterior in at least one $\mathscr{D}_j \in B$ must go uniformly below ε which is absurd by definition of B. So, in all the $\mathscr{D}_j$ forming S_k, and so also in $\mathscr{D}_\ell$, the posterior variance goes uniformly below ε. The proof is concluded by repeating the argument for any robot, any $\mathscr{D}_\ell$ and ω.

References

Anderson, B. D. O., & Moore, J. B. (2012). *Optimal filtering*. Courier Corporation.

Bullo, F., Carli, R., & Frasca, P. (2012). Gossip coverage control for robotic networks: Dynamical systems on the space of partitions. *SIAM Journal on Control and Optimization*, *50*(1), 419–447.

Carron, A., Todescato, M., Carli, R., Schenato, L., & Pillonetto, G. (2015). Multi-agents adaptive estimation and coverage control using Gaussian regression. In *European control conference (ECC'15)*.

Choi, J., & Horowitz, R. (2010). Learning coverage control of mobile sensing agents in one-dimensional stochastic environments. *IEEE Transactions on Automatic Control*, *55*(3), 804–809.

Choi, J., Lee, J., & Oh, S. (2008). Swarm intelligence for achieving the global maximum using spatio-temporal Gaussian processes. In *American control conference, 2008* (pp. 135–140). IEEE.

Choi, J., Oh, S., & Horowitz, R. (2009). Di stributed learning and cooperative control for multi-agent systems. *Automatica*, *45*(12), 2802–2814.

Cortés, J., & Bullo, F. (2005). Coordination and geometric optimization via distributed dynamical systems. *SIAM Journal on Control and Optimization*, *44*(5), 1543–1574.

Cortes, J., Martinez, S., Karatas, T., & Bullo, F. (2004). Co verage control for mobile sensing networks. *IEEE Transactions on Robotics and Automation*, *20*(2), 243–255.

Cucker, F., & Smale, S. (2001). On the mathematical foundations of learning. *Bulletin of the American Mathematical Society*, *39*, 1–49.

Davison, P., Leonard, N. E., Olshevsky, A., & Schwemmer, M. (2015). Nonuniform line coverage from noisy scalar measurements. *IEEE Transactions on Automatic Control*, *60*(7), 1975–1980.

Davison, P., Schwemmer, M., & Leonard, N. E. (2012). Distributed nonuniform coverage with limited scalar measurements. In *2012 50th annual Allerton conference on communication, control, and computing (Allerton)* (pp. 1455–1460). IEEE.

Du, Q., Faber, V., & Gunzburger, M. (1999). Cent roidal Voronoi tessellations: applications and algorithms. *SIAM Review*, *41*(4), 637–676.

Durham, J. W., Carli, R., Frasca, P., & Bullo, F. (2012). Discrete partitioning and coverage control for Gossiping robots. *IEEE Transactions on Robotics*, *28*, 364–378.
Lee, S. G., Diaz-Mercado, Y., & Egerstedt, M. (2015). Mu ltirobot control using time-varying density functions. *IEEE Transactions on Robotics*, *31*(2), 489–493.
Leonard, N. E., & Olshevsky, A. (2011). Nonuniform coverage control on the line. In *2011 50th IEEE conference on decision and control and European control conference*, (CDC-ECC). (pp. 753–758). IEEE.
Leonard, N. E., Paley, D. A., Lekien, F., Sepulchre, R., Fratantoni, D. M., & Davis, R. E. (2007). Colle ctive motion, sensor networks, and ocean sampling. *Proceedings of the IEEE*, *95*(1), 48–74.
Ljung, L. (Ed.) (1999). *System identification (2nd ed.): theory for the user*. Upper Saddle River, NJ, USA: Prentice Hall PTR.
Lloyd, S. (1982). Le ast squares quantization in PCM. *IEEE Transactions on Information Theory*, *28*(2), 129–137.
Pater, R., Frasca, P., Durham, J. W., Carli, R., & Bullo, F. (2016). Dynamic partitioning and coverage control with asynchronous one-to-base-station communication. *IEEE Transactions on Control of Network Systems*, *3*(1), 24–33.
Pillonetto, G., Chiuso, A., & De Nicolao, G. (2011). Pr ediction error identification of linear systems: A nonparametric Gaussian regression approach. *Automatica*, *47*(2), 291–305.
Poggio, T., & Girosi, F. (1990). Ne tworks for approximation and learning. In *Proceedings of the IEEE*, Vol. 78 (pp. 1481–1497).
Schwager, M., Rus, D., & Slotine, J.-J. (2009). Decentra lized, adaptive coverage control for networked robots. *International Journal of Robotics Research*, *28*(3), 357–375.
Tikhonov, A. N., & Arsenin, V. Y. (1977). *Solutions of ill-posed problems*. Washington, DC: Winston/Wiley.
Xu, Y., & Choi, J. (2011). Adaptive sampling for learning Gaussian processes using mobile sensor networks. *Sensors*, *11*(3), 3051–3066.
Xu, Y., Choi, J., Dass, S., & Maiti, T. (2012). Sequential Bayesian prediction and adaptive sampling algorithms for mobile sensor networks. *IEEE Transactions on Automatic Control*, *57*(8), 2078–2084.
Xu, Y., Choi, J., Dass, S., & Maiti, T. (2013). Ef ficient Bayesian spatial prediction with mobile sensor networks using Gaussian Markov random fields. *Automatica*, *49*(12), 3520–3530.
Xu, Y., Choi, J., Dass, S., & Maiti, T. (2015). *Bayesian prediction and adaptive sampling algorithms for mobile sensor networks: Online environmental field reconstruction in space and time*. Springer.
Xu, Y., Choi, J., & Oh, S. (2011). Mobile sensor network navigation using Gaussian processes with truncated observations. *IEEE Transactions on Robotics*, *27*(6), 1118–1131.

Marco Todescato received his Dr. Eng. Master's degree in Control Engineering in 2012 and his Ph.D. degree in Information Engineering in 2016 from the University of Padova, Italy. During his doctoral studies he was Visiting Researcher at the Max Planck Institute, Tübingen, Germany, at the Laboratory of Information and Decision Systems (LIDS), MIT, at the Department Electrical Engineering, University of California Los Angeles, at the Department of Mechanical Engineering, University of California Santa Barbara, at the Automatic Control Laboratory, ETH Zürich. He is currently a Postdoctoral Fellow with the Control group of the Department of Information Engineering, University of Padova. His research interests include smart grid, distributed optimization, and nonparametric estimation.

Andrea Carron was born in Padova, Italy, in 1988. He received the Dr. Eng. Bachelor's and Master's degrees in control engineering from the University of Padova, Padova, Italy, in 2010 and 2012, respectively. He received his Ph.D. degree in 2016 from the University of Padova. He is currently a Postdoctoral Fellow with the Department of Mechanical and Process Engineering at ETH Zürich. His interests include multi-agent robotics, distributed optimization, and nonparametric estimation.

Ruggero Carli received the Laurea Degree in Computer Engineering and the Ph.D. degree in Information Engineering from the University of Padova, Padova, Italy, in 2004 and 2007, respectively. From 2008 through 2010, he was a Post-doctoral Fellow with the Department of Mechanical Engineering, University of California at Santa Barbara. He is currently an Associate Professor with the Department of Information Engineering, University of Padova. His research interests include control theory and, in particular, control under communication constraints, cooperative control, and distributed estimation.

Gianluigi Pillonetto (M03) was born in Montebelluna (TV), Italy on January 21, 1975. He received the Doctoral degree in computer science engineering (with highest honours) from the University of Padova, Padova, Italy, in 1998 and the Ph.D. degree in bioengineering from the Polytechnic of Milan, Milan, Italy, in 2002. In 2000 and 2002, he was Visiting Scholar and Visiting Scientist, respectively, at the Applied Physics Laboratory, University of Washington, Seattle. In 2005, he became Assistant Professor of Control and Dynamic Systems at the Department of Information Engineering, University of Padova where he currently serves as an Associate Professor. He is an Associate Editor of Automatica, Systems and Control Letters and IEEE Transactions on Automatic Control. His research interests are in the field of system identification and machine learning.

Luca Schenato (M07) received the Laurea degree from the University of Padova, Padova, Italy, in 1999 and the Ph.D. degree in electrical engineering from the University of California at Berkeley in 2003. Currently, he is an Associate Professor with the University of Padova. He was Associate Editor for the *IEEE Transactions On Automatic Control*. His research interests include distributed control, estimation and optimization for multi-agent systems, control subject to communication constraints in networked control systems, and the control of biomimetic locomotion. Prof. Schenato was the recipient of the Italian Professorship "Returning Brains" in 2004, the Eli Jury Award from the EECS Department of UC Berkeley in 2006, and of the EUCA European Control Award in 2014. He was one of the organizers of the Conference on Robotics, Communication and Coordination (Robocomm) and the Workshop on Estimation and Control of Networked Systems (Necsys). He is a member of the IFAC Technical Committee on "Networked Systems".