Sensor Deployment for Network-like Environments

Luca Greco, *Member, IEEE*, Matteo Gaeta, Benedetto Piccoli

Abstract

This paper considers the problem of optimally deploying omnidirectional sensors, with potentially limited sensing radius, in a network-like environment. This model provides a compact and effective description of complex environments as well as a proper representation of road or river networks. We present a two-step procedure based on a discrete-time gradient ascent algorithm to find a local optimum for this problem. The first step performs a coarse optimization where sensors are allowed to move in the plane, to vary their sensing radius and to make use of a reduced model of the environment called collapsed network. It is made up of a finite discrete set of points, barycenters, produced by collapsing network edges. Sensors can be also clustered to reduce the complexity of this phase. The sensors' positions found in the first step are then projected on the network and used in the second finer optimization, where sensors are constrained to move only on the network. The second step can be performed on-line, in a distributed fashion, by sensors moving in the real environment, and can make use of the full network as well as of the collapsed one. The adoption of a less constrained initial optimization has the merit of reducing the negative impact of the presence of a large number of local optima.

The effectiveness of the presented procedure is illustrated by a simulated deployment problem in an airport environment.

I. Introduction

Imagine a scenario where a toxic gas is spreading in an area or a building and safe paths have to be found to evacuate people. Or think of an airport environment where people moving through rooms and corridors has to be surveilled in order to detect and avoid terroristic actions. Or consider the need of measuring environmental quantities, such as temperature or

L. Greco and M. Gaeta are with the DIIMA, University of Salerno, Via Ponte don Melillo 1, 84084 Fisciano (SA), Italy; e-mail: lgreco@ieee.org, gaeta@crmpa.unisa.it

B. Piccoli is with the Istituto per le Applicazioni del Calcolo "Mauro Picone", Via dei Taurini 19, 00185 Roma, Italy; e-mail: b.piccoli@iac.cnr.it

humidity, on wide areas to the aim of improving theoretical models or making more accurate weather forecast.

There is a great number of situations that would greatly enjoy the use of network of sensors. Indeed, many of the previous tasks are difficult, or impossible, to be accomplished by a single sensor. The employment of a large number of sensors increases the robustness to sensor failure and communication disruption and make spatially-distributed observations possible. If sensors are able to move, the number of tasks they can perform is still greater.

Static and dynamic sensors' networks need to be deployed in the environment, and the way this problem is solved can significantly affect the quality of service they have to provide.

A. Static Deployment and Locational Optimization

Sensors' deployment problems are strictly related to facilities location-allocation problems, which are the subject of the locational optimization discipline ([1]).

In locational optimization objective functions are used to describe the interactions between users and facilities and among them. Users may find facilities desirable, hence they would like to exert an attractive force to facilities, or undesirable and they would repel them. The attractive model can describe allocation problems of useful services or facilities such as mailboxes, hospitals, fire stations, malls, etc. (see [1]). The repulsive one, instead, can be used to model problems where polluting or dangerous facilities (i.e. nuclear reactors, garbage dumps, etc.) are to be located far enough from urban conglomerations. An excellent survey on undesirable facility locations problems is given by [2] (see also [3]). These operational research problems can be converted in sensors' deployment problems by considering sensors as facilities and points or areas, where events can happen or some quantities has to be measured, as users.

Two well known problems, involving one facility only, are the classical Weber and the obnoxious facility location problems (see [4], [5] for a recent heuristic solution). Three problems involving p facilities are the p-center, p-median and p-dispersion problems. Some recent results on the p-center problem are in [6] and [7]. The latter paper addresses also the p-dispersion problem.

A classical p-median problem, close to the one considered in this paper, can be simply described as the one of finding the optimal location of p facilities by minimizing the average distance of the demand points to the nearest facility (see [8] for a recent survey on heuristics methods to solve it). Close to the p-median problem is the multisource Weber problem, for

which many heuristics exist ([9]). A more general formulation of these problems can be found in [10] and [11]. In [10] a dynamical (gradient descent) version of the Lloyd's algorithm [12] has been presented to find a local optimum for a generalized p-median problem. A deterministic annealing optimization algorithm to solve the classical version is reported in [13]. The aforementioned solutions to the p-median problem, as well as many solutions to p-facilities problems, are based on the construction of a Voronoi Tesselation ([11], [14]).

B. Dynamic Deployment and Distributed Solutions

The use of moving, instead of static, sensor networks provide a great flexibility in solving sensing tasks, mainly when the environment is partially or completely unknown or is not directly accessible for safety reasons. In these cases, sensors are usually initially deployed randomly and hence need to move in order to acquire knowledge of the environment and to optimally re-deploy for their task. Furthermore, environments are usually not static and the network may experience sensor failure or loss. In these situations the properties of adaptivity and reconfigurability owned by a network of moving sensors turn out very useful.

A general tendency in robotic networks is to have sensors (agents) endowed with the same computational and sensing capabilities. This choice increases the overall robustness of the network, but usually calls for distributed coordination algorithms. Having equal sensors, indeed, naturally leads to define optimization and coordination algorithms based on local observations and local decisions ([15], [16], [17]). Many of the algorithms proposed in the previous section involve the solution of a global optimization problem requiring a complete knowledge of the environment and of sensors' distribution. The solutions to p-center, p-dispersion and p-median problems proposed in [7], [10], [18], instead, are all spatially distributed, with the meaning that each sensor requires only the knowledge of positions of its neighbors (or even less if it has a limited sensing radius). This fact allows a distributed implementation where each sensor computes its next movement without centralized coordination.

Other solutions to the area-coverage problem look at sensors like particles subject to virtual forces or potential fields. The compositions of suitably defined attractive and repulsive forces is then used to make the network behaving in the desired fashion (spread sensors, avoid obstacles, keep connectivity, etc.). Representative for this kind of approach are the algorithms presented in [19], [20], or in [21], where also secure connectivity issues are considered. In [22], instead, it is raised the relevant problem of power consumption in wireless networks and three energy-efficient algorithms are presented for sensors' deployment.

C. Network-like Environments and Paper Contributions

In this paper, we focus on network-like environments as there are surveillance or monitoring problems where such a kind of model can provide a more suitable description.

Network models represent a natural choice whenever environments have an intrinsic network structure. It is the case, for instance, when sensors have to be optimally deployed over a network of roads to monitor vehicular traffic, or in a river network to measure temperature or pollutants concentration. Even some location-allocation problems can involve networks. Consider, for instance, the case where useful facilities (i.e. schools, hospitals) have to be located in the interior of a network of roads, which is the source of a nuisance (i.e. noise, pollutants), with the goal of minimizing its harmful impact on them (see [5] for the case with one facility); or the dual problem of locating obnoxious facilities (i.e. dumps, industrial plants, mobile phone repeater antennas) reducing the hazard on the network.

Most notably, we think that a network-like model can provide an effective and compact description for complex environments, focusing only on major features and abstracting from those geometrical details which are less important for the deployment problem. The coverage of nonconvex environments with holes or obstacles, for instance, is a challenging task ([23]), which can enjoy significantly the use of a reduced network-like model. Environments with a complex structure, accounting for a large number of variously sized and shaped rooms, passages, forbidden areas and obstacles, can be reduced to a set of connected paths where the sensing task is more requested or where sensors are forced to pass. An airport is a very representative example of this kind of environments. In this case people moving throughout the airport can be aptly compared with a network flow and focus can be on paths more than on corridors, halls and lounges.

Many of the problems introduced in previous sections have been formulated even for a network-like environment ([24], [25]), but they usually consider a finite discrete set of demand points located on the network's nodes and try to optimize the locations of facilities w.r.t. some objective function accounting for the distance from them. An important thread of works for the deployment of sensors both on plane and on a network is represented by the papers [26], [27], [28]. In such works, the authors propose powerful greedy algorithms that provide a constant approximation of the optimal solution. Their method, however, aims at solving a global static deployment problem, considering a finite set of demand points and allowing sensors to jump among positions.

In this paper, instead, we consider a deployment problem where sensors use local information to dynamically solve the optimization problem while they are moving in the environment. The constraints induced by the environment to the motion of sensors are explicitly considered. More precisely, we address a generalized p-median problem involving omnidirectional sensors with potentially limited sensing radius and we extend the formulation presented in [18], [10], [16] to network-like environments. The task is to find sensors' positions that optimize an objective function defined on the network and accounting for the sensor's features and preferential areas. This is a mixed problem, since the network is considered embedded in the plane (it is a continuous set of demand points) and the planar euclidean norm is used to measure the distance between sensors and network.

The core of the cited formulation and of our solution is a discrete-time gradient ascent algorithm based on Voronoi partitions and aiming at maximizing the objective function. It is a well known fact, however, that such a kind of algorithms can get stuck early in local optima, especially when sensors are forced to move in an over-constrained environment like a network. Moreover, the local maximum found by the algorithm is often greatly related to the initial sensors' position.

For these reasons, we present a novel two-step procedure performing an initial coarse optimization, whose purpose is to provide a good starting point for a second finer optimization. The first step can be carried out off-line, either by a central unit, or by each sensor individually (without doing real movements). The impact of local optima is reduced by allowing sensors, in the initial optimization, to virtually move in $\mathbb{R}^2$ and to vary their sensing radius arbitrarily. In order to reduce the complexity of this phase, sensors can be initially clustered and the optimization problem solved for the clusters' centers. After that, a desired number of sensors is spread close to clusters' centers and the sensors' positions thus found are projected on the closest edges of the network. The projected positions are then used in the second optimization, where sensors are constrained to move only on the network. The second step can be performed on-line, in a distributed fashion, by sensors moving in the real environment.

The use of a two-step procedure is motivated also by the very nature of some surveillance tasks, such as for instance airport surveillance, where a large number of individuals (sometimes referred to as mass objects [29]) are monitored. In these cases, sensors can solve the first step optimization using imprecise or estimated information and keeping still; then they can move to reach the final projected positions using planned routes compatible with

the network. After this initial deployment, sensors can change their positions by dynamically solving a distributed optimization problem (second step) based on real measures taken from the (potentially varying) environment.

It is worth noting that the present procedure can be used to solve both static and dynamic deployment problems. Moreover, the first step deserves attention by its own, since it provides a solution to those problems involving facilities located in the interior of the network mentioned at the beginning of this section.

Another contribution of this paper is the introduction of a simplified model of the network (similar to the discretization in [30]) called *collapsed network* and consisting of finite many points. It is obtained by decomposing each segment of the original network in one or more sub-segments and collapsing each sub-segment in its barycenter. This model allows a coarser but faster optimization, since computations with barycenters are remarkably less than those needed by the full network. Collapsed network, hence, is intended mainly for fastening the first optimization, but can be used profitably also for the second step. Indeed, it turns out particularly useful in practical implementation involving hardware with limited computational capabilities.

As mentioned above, our work is related to that of [18], [10], [16]. In particular, the first step of the optimization, allowing sensors to move in $\mathbb{R}^2$, could be regarded as a specialization of the problem described in [10]. However, the different topology induced by the network introduces issues related to the explicit computation of the gradient and to the convergence of the maximization algorithm, which deserve special solutions. A relevant difference is that the gradient of the objective function presents discontinuity points caused by barycenters on the boundary edge of two neighboring Voronoi cells. Such barycenters can change allocation during sensors' motion, inducing abrupt variations to the value of the objective function associated to each cell. This fact prevents a classical convergence proof for gradient algorithms, hence we consider our proof as a minor contribution of the paper. Some results about convergence may alternatively be derived by using the method of Kushner and Borkar ([31], [32]) of stochastic approximation to deal with our differential inclusion.

The outline of the paper is as follows. In section II the mathematical definitions of sensors, network and Voronoi covering are introduced along with the objective function to be maximized to solve the deployment problem. Section III is devoted to the introduction of the collapsed network and to the formulation, and proof of convergence, of a gradient ascent algorithm to solve the first step (subsection III-A) and the second step (subsection

III-B) of the optimization procedure. Section IV addresses the solution of the second step optimization involving the full network. Finally, a network describing an airport environment is used in section V to illustrate by simulations the effectiveness of the proposed optimization procedure. Conclusions and future research directions are reported in section VI.

II. Preliminaries and Problem Formulation

In this section we introduce the mathematical framework to describe the sensors, the network and its Voronoi covering.

Definition 1: Given two points $p_1, p_2 \in \mathbb{R}^2$, with $p_1 \neq p_2$, $s_{12} = [p_1, p_2] \subset \mathbb{R}^2$ is the segment joining p_1 and p_2 and $s_{12}^o = (p_1, p_2)$ is the open segment between them. We define length of a segment s_{12} as $\ell(s_{12}) = \|p_2 - p_1\|$, where $\|\cdot\|$ is the Euclidean norm; barycenter of a segment s_{12} the point $b(s_{12}) = \frac{1}{2}(p_2 + p_1) \in s_{12}$; partition of a segment $s = [p_1, p_2]$ in k sub-segments, the set of segments $\{s_i\}_{i=1,\ldots,k}$ given by

$$s_i = \left[p_1 + (i-1)\frac{(p_2 - p_1)}{k}, p_1 + i\frac{(p_2 - p_1)}{k}\right].$$

Definition 2: A network $\mathcal{N} = (\mathcal{V}, \mathcal{S})$ is a subset of $\mathbb{R}^2$ consisting of a set of points $\mathcal{V} = \{v_1, \ldots, v_n \in \mathbb{R}^2,\ v_i \neq v_j\ \forall i \neq j\}$ and a set of segments $\mathcal{S} \subseteq \{s_{ij} = [v_i, v_j] \subset \mathbb{R}^2,\ i, j \in \{1, \ldots, n\}\ i \neq j\}$, such that:

i) $\forall v_i \in \mathcal{V}$, $\exists v_j \in \mathcal{V}$, $v_i \neq v_j$ such that $s_{ij} \in \mathcal{S}$ (no isolated vertex);

ii) $\forall i, j, h, k \in \{1, \ldots, n\}$, $(i, j) \neq (h, k)$, $s_{ij}^o \cap s_{hk}^o = \emptyset$ (no segment intersection).

Definition 3: Given a network $\mathcal{N}$ and a set of points $\mathcal{P} = \{p_1, \ldots, p_m\} \subset \mathcal{N}$, the Voronoi covering of $\mathcal{N}$ generated by $\mathcal{P}$ with respect to the Euclidean norm is the collection of sets $\left\{V_i^{\mathcal{N}}(\mathcal{P})\right\}_{i \in \{1,\ldots,m\}}$ defined by

$$V_i^{\mathcal{N}}(\mathcal{P}) = \{q \in \mathcal{N} \mid \|q - p_i\| \leq \|q - p_j\|,\ \forall p_j \in \mathcal{P}\}.$$

Remark 4: It is straightforward to recognize that $V_i^{\mathcal{N}}(\mathcal{P})$ can be equivalently defined as $V_i^{\mathcal{N}}(\mathcal{P}) = V_i(\mathcal{P}) \cap \mathcal{N}$, where $V_i(\mathcal{P})$ is the i-th cell of the usual Voronoi partition of $\mathbb{R}^2$ generated by $\mathcal{P}$. The previous definition is about a covering and not a partition since neighboring cells can have a nontrivial intersection: a portion of a segment can belong to the shared edge of two cells $V_i(\mathcal{P})$ and $V_j(\mathcal{P})$.

We adapt the framework provided in [18] to describe the sensors' and network features. Each sensor is modeled by the (same) *performance function* $f : \mathbb{R}_+ \to \mathbb{R}$, that is a non-increasing and piecewise differentiable map having a finite number of bounded discontinuities

at $R_1, \ldots, R_N \in \mathbb{R}_+$, with $R_1 < \ldots < R_N$. We can set $R_0 = 0$, $R_{N+1} = +\infty$ and write

$$f(x) = \sum_{\alpha=1}^{N+1} f_\alpha(x) 1_{[R_{\alpha-1}, R_\alpha)}(x), \tag{1}$$

with $f_\alpha : [R_{\alpha-1}, R_\alpha] \to \mathbb{R}$, $\alpha \in \{1, \ldots, N+1\}$ non-increasing continuously differentiable functions such that $f_\alpha(R_\alpha) > f_{\alpha+1}(R_\alpha)$ for $\alpha \in \{1, \ldots, N\}$. In order to model regions of the network with different importance, we can use a *density function* $\phi : \mathcal{N} \to \mathbb{R}_+$, which is bounded and measurable on $\mathcal{N}$. Given $g : \mathbb{R}^2 \to \mathbb{R}$ we indicate by $\int_{\mathcal{N}} g(q)dq$ (respectively $\int_{V_i^{\mathcal{N}}(\mathcal{P})} g(q)dq$) the sum of the linear integrals of g over the segments of $\mathcal{N}$ (respectively $V_i^{\mathcal{N}}(\mathcal{P})$) using an arc-length parameterization. With these functions we can define the multi-center function $\mathcal{H} : \mathcal{N}^m \to \mathbb{R}$ for m sensors located in $\mathcal{P} = \{p_1, \ldots, p_m\} \subset \mathcal{N}$

$$\mathcal{H}(\mathcal{P}) = \int_{\mathcal{N}} \max_{i \in \{1,\ldots,m\}} f(\|q - p_i\|) \phi(q) \, dq. \tag{2}$$

We can also provide an alternative expression for (2) based on the Voronoi covering induced by $\mathcal{P}$ as follows

$$\mathcal{H}(\mathcal{P}) = \sum_{i=1}^{m} \int_{V_i^{\mathcal{N}}(\mathcal{P})} f(\|q - p_i\|) \phi(q) \, dq - \sum_{\Delta_{hk}^{\mathcal{N}} \in \Delta^{\mathcal{N}}} \int_{\Delta_{hk}^{\mathcal{N}}} f(\|q - p_h\|) \phi(q) \, dq, \tag{3}$$

where $\Delta_{hk}^{\mathcal{N}} \triangleq V_h^{\mathcal{N}}(\mathcal{P}) \cap V_k^{\mathcal{N}}(\mathcal{P})$ and $\Delta^{\mathcal{N}} \triangleq \left\{\Delta_{hk}^{\mathcal{N}} \mid h < k, \ \forall h, k \in \{1, \ldots, m\}\right\}$. The second term in (3) is not null if and only if there exists a non trivial segment $s \subseteq s_{ij} \in \mathcal{S}$ such that $s \subset \Delta_{hk}^{\mathcal{N}}$ for some $i, j \in \{1, \ldots, n\}$ and $h, k \in \{1, \ldots, m\}$.

III. Deployment over a Collapsed Network

In a collapsed network each segment of the original network is decomposed in one or more sub-segments and each sub-segment is collapsed in its barycenter. Chosen a value for r guaranteeing a good approximation, we can build the r-collapsed network $\mathcal{C}_r^{\mathcal{N}}$ as follows:

Definition 5 (r-Collapsed Network): Given a network $\mathcal{N} = (\mathcal{V}, \mathcal{S})$ and $r > 0$, $\forall s \in \mathcal{S}$ consider its partition in $k_s = \left\lceil \frac{\ell(s)}{r} \right\rceil$ sub-segments s_i (having at most length r) and the associated set of barycenters $\{b(s_i)\}_{i=1,\ldots,k_s}$. We define the r-collapsed network associated to $\mathcal{N}$ the set of points $\mathcal{C}_r^{\mathcal{N}} = \bigcup_{s \in \mathcal{S}} \{b(s_i)\}_{i=1,\ldots,k_s}$.

The multi-center function must be re-defined since the integration domain is now a discrete set represented by the barycenters. Hence we set

$$\mathcal{H}(\mathcal{P}) = \sum_{b_e \in \mathcal{C}_r^{\mathcal{N}}} \max_{i \in \{1,\ldots,m\}} f(\|b_e - p_i\|) \phi_{b_e}, \tag{4}$$

where ϕ_{b_e} are suitable (density) weights assigned to barycenters.

A. Sensors Moving in $\mathbb{R}^2$

In this section we solve the deployment problem for a collapsed network and sensors moving in $\mathbb{R}^2$, which constitutes the first step of our optimization procedure.

Also for the multi-center function (4) we can provide an alternative expression using the Voronoi covering. We need the following definition[1]:

Definition 6: Given an r-collapsed network $\mathcal{C}_r^{\mathcal{N}}$ for some $r \in \mathbb{R}_+$ and a set of points $\mathcal{P} = \{p_1, \ldots, p_m\} \subset \mathbb{R}^2$, the Voronoi covering of $\mathcal{C}_r^{\mathcal{N}}$ generated by $\mathcal{P}$ with respect to the Euclidean norm is the collection of sets $\left\{V_i^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P})\right\}_{i\in\{1,\ldots,m\}}$ defined by

$$V_i^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P}) = \left\{b \in \mathcal{C}_r^{\mathcal{N}} \mid \|b - p_i\| \leq \|b - p_j\|, \ \forall p_j \in \mathcal{P}\right\}.$$

We define also the boundary of a Voronoi cell as

$$\partial V_i^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P}) = \left\{b \in \mathcal{C}_r^{\mathcal{N}} \mid \|b - p_i\| = \|b - p_j\|, \ \forall p_j \in \mathcal{P}\right\},$$

and, in order to simplify the problem, we make the following assumption

Assumption 7: $\partial V_i^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P}) = \emptyset \ \forall i \in \{1, \ldots, m\}$.

With this assumption the multi-center function (4) can be written also as

$$\mathcal{H}(\mathcal{P}) = \sum_{i=1}^{m} \sum_{b_e \in V_i^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P})} f\left(\|b_e - p_i\|\right) \phi_{b_e}, \tag{5}$$

or, setting $\mathrm{dist}(q, \mathcal{P}) = \min_{i\in\{1,\ldots,m\}} \|q - p_i\|$, as

$$\mathcal{H}(\mathcal{P}) = \sum_{b_e \in \mathcal{C}_r^{\mathcal{N}}} f\left(\mathrm{dist}(b_e, \mathcal{P})\right) \phi_{b_e}. \tag{6}$$

Remark 8: $\mathcal{H}(\mathcal{P})$ is not globally Lipschitz as $f(\cdot)$ is not a continuous function. However, if $f(x)$ is continuous and piecewise differentiable with bounded derivative, then $\mathcal{H}(\mathcal{P})$ is globally Lipschitz. In order to prove the global Lipschitz continuity, let us consider two sets of points $\mathcal{P} = \{p_1, \ldots, p_m\} \subset \mathbb{R}^2$ and $\mathcal{P}' = \{p'_1, \ldots, p'_m\} \subset \mathbb{R}^2$ and compute

$$\begin{aligned}
\mathcal{H}(\mathcal{P}) - \mathcal{H}(\mathcal{P}') &= \sum_{b_e \in \mathcal{C}_r^{\mathcal{N}}} \left[f\left(\mathrm{dist}(b_e, \mathcal{P})\right) - f\left(\mathrm{dist}(b_e, \mathcal{P}')\right)\right] \phi_{b_e} \\
&\leq \sum_{b_e \in \mathcal{C}_r^{\mathcal{N}}} \left\|\frac{\partial f}{\partial x}\right\|_\infty \left|\mathrm{dist}(b_e, \mathcal{P}) - \mathrm{dist}(b_e, \mathcal{P}')\right| \phi_{b_e} \\
&\leq \left(\sum_{b_e \in \mathcal{C}_r^{\mathcal{N}}} \phi_{b_e}\right) \left\|\frac{\partial f}{\partial x}\right\|_\infty \|\mathcal{P} - \mathcal{P}'\|,
\end{aligned}$$

[1]For this definition similar remarks as Remark 4 also apply.

where $\left\|\frac{\partial f}{\partial x}\right\|_\infty$ is the L_∞-norm of $\frac{\partial f}{\partial x}$.

Theorem 9: The multi-center function $\mathcal{H}$ is continuously differentiable on $(\mathbb{R}^2)^m \backslash \left(\mathcal{D}_{\mathcal{C}_r^{\mathcal{N}}}\right)^m$, where

$$\mathcal{D}_{\mathcal{C}_r^{\mathcal{N}}} \triangleq \bigcup_{b_e \in \mathcal{C}_r^{\mathcal{N}}} \left\{q \in \mathbb{R}^2 \mid \|b_e - q\| = R_i,\ \forall i = 1, \ldots, N\right\}$$

is the discontinuity set of $f(\cdot)$ in $\mathbb{R}^2$. Moreover, for each $h \in \{1, \ldots, m\}$

$$\frac{\partial \mathcal{H}(\mathcal{P})}{\partial p_h} = \sum_{b_e \in V_h^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P})} \frac{\partial}{\partial p_h} f\left(\|b_e - p_h\|\right) \phi_{b_e}. \tag{7}$$

Proof: The continuous differentiability of $\mathcal{H}$ on $(\mathbb{R}^2)^m \setminus \left(\mathcal{D}_{\mathcal{C}_r^{\mathcal{N}}}\right)^m$ is a straight consequence of the same property of $f(\cdot)$ on $\mathbb{R}^2 \setminus \mathcal{D}_{\mathcal{C}_r^{\mathcal{N}}}$. As concerns the gradient, using (6) we have

$$\begin{aligned}\frac{\partial \mathcal{H}(\mathcal{P})}{\partial p_h} &= \frac{\partial}{\partial p_h} \sum_{b_e \in \mathcal{C}_r^{\mathcal{N}}} f\left(\mathrm{dist}(b_e, \mathcal{P})\right) \phi_{b_e} \\ &= \sum_{b_e \in \mathcal{C}_r^{\mathcal{N}}} \frac{\partial}{\partial p_h} f\left(\mathrm{dist}(b_e, \mathcal{P})\right) \phi_{b_e}.\end{aligned}$$

In the hypothesis of Assumption 7, we have that the index set $I_{be} = \mathrm{argmin}_{i \in \{1,\ldots,m\}} \left(\|b_e - p_i\|\right)$ has cardinality 1. Hence, $\forall b_e$ such that $I_{be} \equiv i \neq h$, $\frac{\partial}{\partial p_h} f\left(\mathrm{dist}(b_e, \mathcal{P})\right) = 0$ and

$$\frac{\partial \mathcal{H}(\mathcal{P})}{\partial p_h} = \sum_{\{b_e \in \mathcal{C}_r^{\mathcal{N}} \mid I_{be} \equiv h\}} \frac{\partial}{\partial p_h} f\left(\|b_e - p_h\|\right) \phi_{b_e},$$

whereby the thesis follows using the definition of $V_h^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P})$. ■

The sensors' location-allocation problem can be addressed by means of a gradient-like algorithm. If a continuous time implementation is looked for, the following fictitious dynamics would be associated to the sensors' positions

$$\dot{\mathcal{P}} = \nabla \mathcal{H}(\mathcal{P}). \tag{8}$$

Unfortunately, this dynamics conveys some problems. It is well defined as long as Assumption 7 and Theorem 9 are fulfilled, but these hypotheses are, in fact, too stringent for the algorithm to work properly. Indeed, they would require the evolution of the sensors to avoid any position in the discontinuity set and the barycenters not to enter or exit the Voronoi cells where they are at the initial time instant.

First of all we can reduce the analysis to continuously differentiable functions to avoid issues related to the existence of the gradient. Still the relaxation of Assumption 7 induces some problems on the definition of the gradient. Barycenters on a boundary edge of a

Voronoi cell belong to all the cells sharing that edge. All sensors' positions producing these configurations are discontinuity points for $\nabla\mathcal{H}(\mathcal{P})$. Roughly speaking, the gradient takes different values depending on which cell the shared barycenters are assumed to belong to. This fact makes the equation (8) a set of differential equations with discontinuous right-hand side.

The problem of shared barycenters can be solved by adding a lexicographic criterion to the definitions based on the euclidean distance. Indeed, with this criterion barycenters on boundary edges are allocated univocally to the sensor having the lower index (w.r.t. the Lexicographic Order (L.O.)) among sharing sensors. This fact allows us to define a genuine Voronoi partition, no longer a covering, whose generic cell is given by (compare with Definition 6):

$$V_i^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P}) = \{b \in \mathcal{C}_r^{\mathcal{N}} \mid \|b - p_i\| \leq \|b - p_j\| \ \ \forall p_j \in \mathcal{P} \ \wedge \\ \|b - p_i\| < \|b - p_j\| \ \text{ if } j < i \text{ w.r.t. the L.O.}\}. \quad (9)$$

We must now define a generalized (lexicographic) gradient of $\mathcal{H}$, $\nabla_l\mathcal{H}(\mathcal{P})$, according to this new definition. $\forall b_e \in V_i^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P}) \setminus \partial V_i^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P})$ we use the classical formula given by (7). $\forall b_e \in \partial V_i^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P})$ notice that the partial derivative of f, $\frac{\partial}{\partial p_h} f\left(\|b_e - p_h\|\right)$, exists and is well defined. In the light of this remark we can write the h-th component of the generalized (lexicographic) gradient of $\mathcal{H}$ as

$$\frac{\partial_l \mathcal{H}(\mathcal{P})}{\partial p_h} \triangleq \sum_{b_e \in V_h^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P})} \frac{\partial}{\partial p_h} f\left(\|b_e - p_h\|\right) \phi_{b_e}, \quad (10)$$

which is formally equal to the formula (7) provided by Theorem 9.

The differential equation using this new definition for the gradient, however, does not imply the existence and uniqueness of the solution, and this proof may turn out to be complex due to special sensors' and barycenters' configurations. Moreover, the formula (10) accounts only for the infinitesimal perturbations of sensors' position not inducing any barycenters to enter or exit Voronoi cells, hence changing their allocation.

In order to simplify the convergence proof and to provide an algorithm which is more suitable for a realistic implementation, we consider here a discrete-time version of the gradient algorithm. In our case, the discrete-time implementation can overcome the problems with discontinuity gradients due to the properties of the function $\mathcal{H}$ and its discontinuity points, as it is shown by the following theorem.

Theorem 10: Consider the following discrete-time evolution for the sensors' positions

$$\mathcal{P}^{(k+1)} = \mathcal{P}^{(k)} + \delta_k \nabla_l \mathcal{H}(\mathcal{P}^{(k)}), \tag{11}$$

where the h-th component of $\nabla_l \mathcal{H}$ is given by (10) and $\mathcal{H} : \mathbb{R}^{2m} \to \mathbb{R}$ as in (5). If $f(\cdot)$ has locally bounded second derivatives, then, for suitable δ_k, $\mathcal{P}^{(k)}$ lies in a bounded set and

i) $\mathcal{H}(\mathcal{P}^{(k)})$ is monotonically nondecreasing;

ii) $\mathcal{P}^{(k)}$ converges to the set of critical points of $\mathcal{H}$.

Proof: It is easy to see that there exists a ball $B \supseteq \mathcal{N}$ such that, if $p_i \in \partial B$, then $\frac{\partial_l \mathcal{H}(\mathcal{P})}{\partial p_i}$ points inside B. Thus the fact that $\mathcal{P}^{(k)}$ is bounded for δ_k sufficiently small, easily follows. According to the definition in equation (9) of a Voronoi cell using the lexicographic rule, we can define

$$\mathcal{H}_{p_i}(\mathcal{P}) = \sum_{b_e \in V_i^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P})} f\left(\|b_e - p_i\|\right) \phi_{b_e},$$

hence we can write $\mathcal{H}(\mathcal{P}) = \sum_{i=1}^{m} \mathcal{H}_{p_i}(\mathcal{P})$. We must prove that $\mathcal{H}_{p_i}(\mathcal{P}^{(k+1)}) \geq \mathcal{H}_{p_i}(\mathcal{P}^{(k)})$ for $\forall p_i \in \mathcal{P}^{(k)}$ obeying the discrete-time evolution (11) and that $\exists p_i \in \mathcal{P}^{(k)}$ such that $\mathcal{H}_{p_i}(\mathcal{P}^{(k+1)}) > \mathcal{H}_{p_i}(\mathcal{P}^{(k)})$ if $\mathcal{P}^{(k)}$ does not belong to the set of critical points of $\mathcal{H}$.

Define $\mathcal{H}^u_{p_i}(\mathcal{P}^{(k+1)}) = \sum_{b_e \in V_i^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P}^{(k)})} f\left(\left\|b_e - p_i^{(k+1)}\right\|\right) \phi_{b_e}$, that is the cost $\mathcal{H}_{p_i}(\cdot)$ computed by using the new sensors' positions $\mathcal{P}^{(k+1)}$ but the old allocation of barycenters to sensors, i.e. the Voronoi partition generated by $\mathcal{P}^{(k)}$. Therefore, we write

$$\begin{aligned}
\mathcal{H}^u_{p_i}(\mathcal{P}^{(k+1)}) &= \mathcal{H}_{p_i}(\mathcal{P}^{(k)} + \delta_k^i \nabla_l \mathcal{H}(\mathcal{P}^{(k)})) \\
&= \mathcal{H}_{p_i}(\mathcal{P}^{(k)}) + \delta_k^i \left(\frac{\partial \mathcal{H}_{p_i}(\mathcal{P}^{(k)})}{\partial p_i} \cdot \left(\nabla_l \mathcal{H}(\mathcal{P}^{(k)})\right)_i \right) + o\left((\delta_k^i)^2\right) \\
&= \mathcal{H}_{p_i}(\mathcal{P}^{(k)}) + \delta_k^i \left\| \frac{\partial \mathcal{H}_{p_i}(\mathcal{P}^{(k)})}{\partial p_i} \right\|^2 + o\left((\delta_k^i)^2\right),
\end{aligned} \tag{12}$$

where the (generalized) gradient is computed by means of the lexicographic assignment based on the Voronoi partition generated by $\mathcal{P}^{(k)}$.

If the Voronoi partitions generated by $\mathcal{P}^{(k+1)}$ and $\mathcal{P}^{(k)}$ coincide (same barycenters' allocation), then $\mathcal{H}_{p_i}(\mathcal{P}^{(k+1)}) = \mathcal{H}^u_{p_i}(\mathcal{P}^{(k+1)})$ and we can assert directly that $\mathcal{H}_{p_i}(\mathcal{P}^{(k+1)}) \geq \mathcal{H}_{p_i}(\mathcal{P}^{(k)})$ or $\mathcal{H}_{p_i}(\mathcal{P}^{(k+1)}) > \mathcal{H}_{p_i}(\mathcal{P}^{(k)})$ if $\frac{\partial \mathcal{H}_{p_i}(\mathcal{P}^{(k)})}{\partial p_i} \neq 0$, that is if $\mathcal{P}^{(k)}$ does not belong to the set of critical points of $\mathcal{H}$. It is worth noting that similar remarks about the strict inequality apply also in what follows and they will not be repeated. If the barycenters' allocation is different, $\mathcal{H}_{p_i}(\mathcal{P}^{(k+1)})$ may be smaller than $\mathcal{H}^u_{p_i}(\mathcal{P}^{(k+1)})$ and we cannot say anything about its

relation with $\mathcal{H}_{p_i}(\mathcal{P}^{(k)})$. This fact is due to the presence of barycenters changing allocation during the sensors' evolution, hence the cells involved in the change cannot be considered independently.

Let us consider for simplicity only one barycenter $b_e \in V_j^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P}^{(k)})$ and suppose that at step $k+1$ $b_e \in V_i^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P}^{(k+1)})$. With this assumption no barycenters can enter or exit the union of the two cells but b_e. We have

$$\begin{aligned}\mathcal{H}_{p_i}(\mathcal{P}^{(k+1)}) &= \mathcal{H}^u_{p_i}(\mathcal{P}^{(k+1)}) + f\left(\left\|b_e - p_i^{(k+1)}\right\|\right)\phi_{b_e}\\ \mathcal{H}_{p_j}(\mathcal{P}^{(k+1)}) &= \mathcal{H}^u_{p_j}(\mathcal{P}^{(k+1)}) - f\left(\left\|b_e - p_j^{(k+1)}\right\|\right)\phi_{b_e}.\end{aligned} \tag{13}$$

In other words, $\mathcal{H}_{p_i}(\mathcal{P}^{(k+1)})$ is grown w.r.t. the ideal value $\mathcal{H}^u_{p_i}(\mathcal{P}^{(k+1)})$ due to the allocation of b_e to $p_i^{(k+1)}$, whereas, $\mathcal{H}_{p_j}(\mathcal{P}^{(k+1)})$ is decreased w.r.t. $\mathcal{H}^u_{p_i}(\mathcal{P}^{(k+1)})$ by the contribution that b_e would have given if it were allocated to $p_j^{(k+1)}$ even at the step $k+1$. It is worth noting that if b_e changes allocation from $p_j^{(k+1)}$ to $p_i^{(k+1)}$, then $\left\|b_e - p_i^{(k+1)}\right\| \leq \left\|b_e - p_j^{(k+1)}\right\|$ (the equality holds only if $b_e \in \partial V_i^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P}^{(k+1)}) \cap \partial V_j^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P}^{(k+1)})$ and, being $i < j$, the allocation is induced by the lexicographic rule). Therefore, due to the monotonicity of f, $f\left(\left\|b_e - p_i^{(k+1)}\right\|\right) \geq f\left(\left\|b_e - p_j^{(k+1)}\right\|\right)$. Summing up, we have

$$\begin{aligned}\mathcal{H}_{p_i}(\mathcal{P}^{(k+1)}) + \mathcal{H}_{p_j}(\mathcal{P}^{(k+1)}) &\geq \mathcal{H}^u_{p_i}(\mathcal{P}^{(k+1)}) + \mathcal{H}^u_{p_j}(\mathcal{P}^{(k+1)})\\ &= \mathcal{H}_{p_i}(\mathcal{P}^{(k)}) + \delta_k^i\left\|\frac{\partial\mathcal{H}_{p_i}(\mathcal{P}^{(k)})}{\partial p_i}\right\|^2 + o\left((\delta_k^i)^2\right)\\ &+ \mathcal{H}_{p_j}(\mathcal{P}^{(k)}) + \delta_k^j\left\|\frac{\partial\mathcal{H}_{p_j}(\mathcal{P}^{(k)})}{\partial p_j}\right\|^2 + o\left((\delta_k^j)^2\right).\end{aligned}$$

There exist some constants $c_k^{ij}, \delta_k^{ij} > 0$ such that

$$\mathcal{H}_{p_i}(\mathcal{P}^{(k+1)}) + \mathcal{H}_{p_j}(\mathcal{P}^{(k+1)}) \geq \mathcal{H}_{p_i}(\mathcal{P}^{(k)}) + \mathcal{H}_{p_j}(\mathcal{P}^{(k)}) + \delta_k^{ij}c_k^{ij} + o\left((\delta_k^{ij})^2\right).$$

Extending the same reasoning to more complex configurations involving more than one common boundary edge and more than two neighboring cells, we can say that $\exists c_k, \delta_k > 0$ such that

$$\mathcal{H}(\mathcal{P}^{(k+1)}) = \sum_{i=1}^{m}\mathcal{H}_{p_i}(\mathcal{P}^{(k+1)}) \geq \mathcal{H}(\mathcal{P}^{(k)}) + \delta_k c_k + o(\delta_k^2).$$

This proves assertion i).

Assertion ii) can be proved by exploiting the results of Section 3.2 of [18]. More precisely, using the fact that $\mathcal{P}^{(k)}$ is bounded and f has locally bounded second derivatives, then there

exists $\bar{\delta} > 0$ such that we can choose $\delta_k \geq \bar{\delta}$. Then we conclude, using Proposition 3.4 of [18], that assertion ii) is true. ■

Remark 11 (Distributed Implementation): The use of a gradient ascent algorithm based on a Voronoi partition, allows us to solve not only a static deployment problem, but also a dynamic one. As shown in [18], this kind of algorithms is spatially distributed, with the meaning that the i-th sensor needs only to know the position of its neighbors in order to determine the boundary of its cell and, hence, to compute $\frac{\partial \mathcal{H}_{p_i}}{\partial p_i}$. For the same reason the i-th sensor can choose the value of the step-size δ_k^i independently of the other sensors simply performing locally a classical line search algorithm. This property makes the algorithm suitable for a spatially distributed implementation.

The independence in the choice of the step-sizes δ_k^i is obviously preserved in each period k, as long as a synchronous implementation is considered. In this case sensors have access to a global clock, or perform a synchronization algorithm. At the beginning of the the k-th period (instant t_k), all sensors are idle, build their Voronoi cells and compute their gradients and step-sizes, then they move until, at most, the end of the period (instant t_{k+1}). If, instead, an asynchronous implementation is considered, further hypotheses are necessary to ensure that independence is preserved. Unfortunately, the discontinuity of the gradient prevent us from using the results of [33]. But, if a sensor has the capability to detect when its neighbors start and stop moving and when a new sensor joins the neighborhood, the asynchronous algorithm presented in [10] (Table IV), can be applied, thus automatically recovering the independence.

Remark 12: In the previous theorem, for sake of simplicity, we did not consider degenerate configurations where different sensors have the same position ($p_i = p_j$ for $i \neq j$). But it can be proved that if the initial positions of sensors are not degenerate, sensors can always choose a suitable δ_k^i to avoid the occurrence of these configurations.

B. Sensors Moving on the Network

This section is devoted to the case of sensors constrained to move on the network and sensing a collapsed network. Therefore, these results are suitable for an implementation of the second step of our procedure on hardware with limited computational capabilities.

We still assume $f(\cdot)$ to be a continuously differentiable function and we make use of the lexicographic criterion for the barycenter allocation. As concerns sensors' motion, however,

we cannot use directly the gradient since the sensors have to remain on the network. We must consider now the directional derivative of $\mathcal{H}$ along the edges of the network.

Following the guidelines of the previous section, the following theorem can be proved.

Theorem 13: Given a network $\mathcal{N} = (\mathcal{V}, \mathcal{S})$ and the related r-collapsed network $\mathcal{C}_r^{\mathcal{N}}$, the multi-center function $\mathcal{H}$ is continuously differentiable almost everywhere on $\mathcal{N}^m$. In particular, on each open segment s_{ij}^o such that $s_{ij} \in \mathcal{S}$, given the unit vector w_{ij} such that $s_{ij} \cdot w_{ij} = \|s_{ij}\|$, the directional derivative in $p_h \in s_{ij}^o$ along w_{ij} is

$$D_{w_{ij}}\mathcal{H}(\mathcal{P})[p_h] = \left(\frac{\partial_l \mathcal{H}}{\partial p_h}(\mathcal{P}) \cdot w_{ij}\right) w_{ij} \tag{14}$$

$$\frac{\partial_l \mathcal{H}}{\partial p_h}(\mathcal{P}) = \sum_{b_e \in V_h^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P})} \frac{\partial}{\partial p_h} f\left(\|b_e - p_h\|\right) \phi_{b_e}.$$

The directional derivative is a multivalued function on the vertices of the network as more than one edge can share the same vertex, but we need a univocal definition. Hence, we fix a choice rule such that the directional derivative in a vertex is given by the maximum among all the derivatives defined for each possible direction that does not lead the sensor out of the network. If all the directional derivatives in a vertex point outward the network, then the derivative is set equal to zero.

Definition 14: Given the set

$$S_{v_i} = \big\{ s \in \mathcal{S} \mid \exists v_j \in \mathcal{V}, \exists \bar{\delta} > 0 \text{ s.t. } s = [v_i, v_j] \vee s = [v_j, v_i], \\ \forall \delta \in [0, \bar{\delta}] \; v_i + \delta D_{w_{ij}}\mathcal{H}(\mathcal{P})[v_i] \in \mathcal{N} \big\}, \tag{15}$$

we define the directional derivative of $\mathcal{H}(\mathcal{P})$ in any point $p_h \in \mathcal{N}$ as follows

$$\widetilde{D}_h \mathcal{H}(\mathcal{P}) = \begin{cases} D_{w_{ij}}\mathcal{H}(\mathcal{P})[p_h] \text{ given by (14)} & \forall p_h \in \mathcal{N} \setminus \mathcal{V} \\ \max\limits_{s_{ij} \in S_{v_i}} D_{w_{ij}}\mathcal{H}(\mathcal{P})[p_h] & \begin{array}{c} \forall p_h \equiv v_i \in \mathcal{V} \\ S_{v_i} \neq \emptyset \end{array} \\ 0 & \begin{array}{c} \forall p_h \equiv v_i \in \mathcal{V} \\ S_{v_i} = \emptyset \end{array} \end{cases} \tag{16}$$

We can now define the discrete-time gradient-like algorithm.

Theorem 15: Consider the following discrete-time evolution for the sensors' positions

$$\mathcal{P}^{(k+1)} = \mathcal{P}^{(k)} + \delta_k \widetilde{D}\mathcal{H}(\mathcal{P}^{(k)}), \tag{17}$$

where the h-th component of $\widetilde{D}\mathcal{H}$ is given by (16) and $\mathcal{H} : \mathcal{N}^m \to \mathbb{R}$ as in (5). If $f(\cdot)$ has locally bounded second derivatives, then, for suitable δ_k, $\mathcal{P}^{(k)}$ lies in a bounded set and

i) $\mathcal{H}(\mathcal{P}^{(k)})$ is monotonically nondecreasing;

ii) $\mathcal{P}^{(k)}$ converges to the set of critical points of $\mathcal{H}$.

Proof: The proof is essentially the same of the Theorem 10, except for the use of the derivative (16) in place of the gradient (10). In particular equality (12) becomes

$$\begin{aligned}\mathcal{H}^u_{p_i}(\mathcal{P}^{(k+1)}) &= \mathcal{H}_{p_i}(\mathcal{P}^{(k)} + \delta^i_k \widetilde{D}\mathcal{H}(\mathcal{P}^{(k)})) \\ &= \mathcal{H}_{p_i}(\mathcal{P}^{(k)}) + \delta^i_k \left(\frac{\partial \mathcal{H}_{p_i}(\mathcal{P}^{(k)})}{\partial p_i} \cdot \left(\widetilde{D}\mathcal{H}(\mathcal{P}^{(k)}) \right)_i \right) + o\left((\delta^i_k)^2\right) \\ &= \mathcal{H}_{p_i}(\mathcal{P}^{(k)}) + \delta^i_k \left(\frac{\partial \mathcal{H}_{p_i}(\mathcal{P}^{(k)})}{\partial p_i} \cdot w_{hl} \right)^2 + o\left((\delta^i_k)^2\right),\end{aligned}$$

for p_i moving along w_{hl}. ■

IV. Deployment over a Full Network

In this section we consider a more accurate version of the second step of the optimization procedure, namely sensors constrained on the network and sensing the full network. To start with, let us define the boundary of a Voronoi cell as $\partial V_i^{\mathcal{N}}(\mathcal{P}) = \{q \in \mathcal{N} \mid \|q - p_i\| = \|q - p_j\|, \exists p_j \in \mathcal{P}\}$, and the instantaneous discontinuity set of $f(\cdot)$ as

$$\mathcal{D}_{\mathcal{N}}(\mathcal{P}) \triangleq \bigcup_{p_j \in \mathcal{P}} \{q \in \mathcal{N} \mid \|q - p_j\| = R_i, \ \forall i = 1, \ldots, N\}.$$

Assumption 16: We make the following assumptions:

i) orthogonality assumption: $\forall h, k \in \{1, \ldots, n\}$, $\forall i \in \{1, \ldots, m\}$ and for any segment $s = [a, b] \subseteq s_{hk} \in \mathcal{S}$ with $a \neq b$, $s \not\subset \partial V_i^{\mathcal{N}}(\mathcal{P})$;

ii) $\partial V_i^{\mathcal{N}}(\mathcal{P}) \cap \mathcal{D}_{\mathcal{N}}(\mathcal{P}) = \emptyset$, $\forall i \in \{1, \ldots, m\}$;

iii) $\mathcal{V} \cap \mathcal{D}_{\mathcal{N}}(\mathcal{P}) = \emptyset$;

iv) $\forall h, k \in \{1, \ldots, n\}$, $\forall i \in \{1, \ldots, m\}$, $\forall q \in s_{hk} \cap V_i^{\mathcal{N}}(\mathcal{P})$, if $(q - p_i) \cdot (v_h - v_k) = 0 \Rightarrow \|q - p_i\| \notin \{R_1, \ldots, R_N\}$;

With the orthogonality assumption the expression (3) simplifies to

$$\mathcal{H}(\mathcal{P}) = \sum_{i=1}^{m} \int_{V_i^{\mathcal{N}}(\mathcal{P})} f\left(\|q - p_i\|\right) \phi\left(q\right) dq. \tag{18}$$

Theorem 17: Given a network $\mathcal{N} = (\mathcal{V}, \mathcal{S})$ if Assumption 16 holds, the multi-center function $\mathcal{H}$ is continuously differentiable almost everywhere on $\mathcal{N}^m$. In particular, on each

open segment s_{ij}^o such that $s_{ij} \in \mathcal{S}$, given the unit vector w_{ij} such that $s_{ij} \cdot w_{ij} = \|s_{ij}\|$, the directional derivative in $p_h \in s_{ij}^o$ along w_{ij} is

$$D_{w_{ij}} \mathcal{H}(\mathcal{P})[p_h] = \left(\frac{\partial \mathcal{H}}{\partial p_h}(\mathcal{P}) \cdot w_{ij} \right) w_{ij} \tag{19}$$

$$\frac{\partial \mathcal{H}}{\partial p_h}(\mathcal{P}) = \sum_{k=1}^{M_h(\mathcal{P})} \mathcal{I}_k$$

$$\mathcal{I}_k = \|b_k - a_k\| \int_{[0,1] \setminus \left\{ t_{1,1}^k, t_{1,2}^k, \ldots, t_{N,1}^k, t_{N,2}^k \right\}} \frac{\partial}{\partial \nu} f(\nu) \frac{p_h - \gamma_k(t)}{\|p_h - \gamma_k(t)\|} \phi(\gamma_k(t))\, dt$$
$$+ \|b_k - a_k\| \sum_{\alpha=1}^{N} \left(f_{\alpha+1}(R_\alpha) - f_\alpha(R_\alpha) \right) \sum_{j=1}^{2} \frac{p_h - \gamma_k(t_{\alpha,j}^k)}{\left\| p_h - \gamma_k(t_{\alpha,j}^k) \right\|} \phi\left(\gamma_k(t_{\alpha,j}^k) \right),$$

where $\gamma_k(t) = a_k + (b_k - a_k)\, t$, $t \in [0,1]$ is a parameterization for the k-th segment $[a_k, b_k] \in V_h^{\mathcal{N}}(\mathcal{P})$, $M_h(\mathcal{P})$ is the number of segments in $V_h^{\mathcal{N}}(\mathcal{P})$ and $t_{\alpha,j}^k \in [0,1],\ j \in \{1,2\}$ are the zeros of $\|\gamma_k(t) - p_h\| - R_\alpha = 0$ (if any).

Proof: Consider the gradient of $\mathcal{H}(\mathcal{P})$ in the form (18)

$$\begin{aligned} \frac{\partial \mathcal{H}}{\partial p_h}(\mathcal{P}) &= \frac{\partial}{\partial p_h} \sum_{i=1}^{m} \int_{V_i^{\mathcal{N}}(\mathcal{P})} f(\|q - p_i\|)\, \phi(q)\, dq \\ &= \frac{\partial}{\partial p_h} \int_{V_h^{\mathcal{N}}(\mathcal{P})} f(\|q - p_h\|)\, \phi(q)\, dq \\ &+ \frac{\partial}{\partial p_h} \sum_{\substack{i=1 \\ i \neq h}}^{m} \int_{V_i^{\mathcal{N}}(\mathcal{P})} f(\|q - p_i\|)\, \phi(q)\, dq. \end{aligned} \tag{20}$$

Let us consider the second term of (20) for each $i \neq h$

$$\begin{aligned} &\frac{\partial}{\partial p_h} \int_{V_i^{\mathcal{N}}(\mathcal{P})} f(\|q - p_i\|)\, \phi(q)\, dq \\ &= \lim_{\varepsilon \to 0} \frac{1}{\|\varepsilon\|} \left(\int_{V_i^{\mathcal{N}}(\{p_1, \ldots, p_h + \varepsilon, \ldots, p_m\})} f(\|q - p_i\|)\, \phi(q)\, dq - \int_{V_i^{\mathcal{N}}(\mathcal{P})} f(\|q - p_i\|)\, \phi(q)\, dq \right) \\ &= \lim_{\varepsilon \to 0} \frac{1}{\|\varepsilon\|} \left(\int_{\Delta V_{ih}^{\mathcal{N}}(\mathcal{P})^+} f(\|q - p_i\|)\, \phi(q)\, dq - \int_{\Delta V_{ih}^{\mathcal{N}}(\mathcal{P})^-} f(\|q - p_i\|)\, \phi(q)\, dq \right), \end{aligned} \tag{21}$$

where $\Delta V_{ih}^{\mathcal{N}}(\mathcal{P})^+ = V_i^{\mathcal{N}}(\{p_1, \ldots, p_h + \varepsilon, \ldots, p_m\}) \setminus V_i^{\mathcal{N}}(\mathcal{P})$ and $\Delta V_{ih}^{\mathcal{N}}(\mathcal{P})^- = V_i^{\mathcal{N}}(\mathcal{P}) \setminus V_i^{\mathcal{N}}(\{p_1, \ldots, p_h + \varepsilon, \ldots, p_m\})$. It is worth noting that $V_i^{\mathcal{N}}(\{p_1, \ldots, p_h + \varepsilon, \ldots, p_m\})$ can be different from $V_i^{\mathcal{N}}(\mathcal{P})$ if and only if $p_i \in \mathcal{N}_{\mathcal{G}_D}(p_h, \mathcal{P})$.[2]

[2]With $\mathcal{N}_{\mathcal{G}_D}(p_h, \mathcal{P})$ we represent the set of neighbors of p_h in $\mathcal{P}$. The neighboring property is given w.r.t. the proximity graph $\mathcal{G}_D(\mathcal{P})$, that is the Delaunay graph associated to the Voronoi partition induced by $\mathcal{P}$.

Let us consider now the first term of (20)

$$\frac{\partial}{\partial p_h}\int_{V_h^{\mathcal{N}}(\mathcal{P})} f\left(\|q-p_h\|\right)\phi\left(q\right)dq$$
$$=\lim_{\varepsilon\to 0}\frac{1}{\|\varepsilon\|}\left(\int_{V_h^{\mathcal{N}}(\{p_1,\ldots,p_h+\varepsilon,\ldots,p_m\})} f\left(\|q-(p_h+\varepsilon)\|\right)\phi\left(q\right)dq-\int_{V_h^{\mathcal{N}}(\mathcal{P})} f\left(\|q-p_h\|\right)\phi\left(q\right)dq\right)$$
$$=\lim_{\varepsilon\to 0}\frac{1}{\|\varepsilon\|}\int_{V_h^{\mathcal{N}}(\{p_1,\ldots,p_h+\varepsilon,\ldots,p_m\})}\left(f\left(\|q-(p_h+\varepsilon)\|\right)-f\left(\|q-p_h\|\right)\right)\phi\left(q\right)dq$$
$$+\lim_{\varepsilon\to 0}\frac{1}{\|\varepsilon\|}\left(\int_{V_h^{\mathcal{N}}(\{p_1,\ldots,p_h+\varepsilon,\ldots,p_m\})} f\left(\|q-p_h\|\right)\phi\left(q\right)dq-\int_{V_h^{\mathcal{N}}(\mathcal{P})} f\left(\|q-p_h\|\right)\phi\left(q\right)dq\right)$$

$$=\lim_{\varepsilon\to 0}\frac{1}{\|\varepsilon\|}\int_{V_h^{\mathcal{N}}(\mathcal{P})}\left(f\left(\|q-(p_h+\varepsilon)\|\right)-f\left(\|q-p_h\|\right)\right)\phi\left(q\right)dq$$
$$+\lim_{\varepsilon\to 0}\frac{1}{\|\varepsilon\|}\int_{\Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^+}\left(f\left(\|q-(p_h+\varepsilon)\|\right)-f\left(\|q-p_h\|\right)\right)\phi\left(q\right)dq$$

$$-\lim_{\varepsilon\to 0}\frac{1}{\|\varepsilon\|}\int_{\Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^-}\left(f\left(\|q-(p_h+\varepsilon)\|\right)-f\left(\|q-p_h\|\right)\right)\phi\left(q\right)dq$$
$$+\lim_{\varepsilon\to 0}\frac{1}{\|\varepsilon\|}\left(\int_{\Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^+} f\left(\|q-p_h\|\right)\phi\left(q\right)dq-\int_{\Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^-} f\left(\|q-p_h\|\right)\phi\left(q\right)dq\right), \qquad (22)$$

where $\Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^+ = V_h^{\mathcal{N}}(\{p_1,\ldots,p_h+\varepsilon,\ldots,p_m\}) \setminus V_h^{\mathcal{N}}(\mathcal{P})$ and $\Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^- = V_h^{\mathcal{N}}(\mathcal{P}) \setminus V_h^{\mathcal{N}}(\{p_1,\ldots,p_h+\varepsilon,\ldots,p_m\})$.

Now we want to prove that the sum of the second term of (20) and the last term of (22) is null. First of all, recall that the sum in the second term of (20) can be limited to the cells in the neighborhood of the h-th cell, namely $\forall i \in I_h$ with $I_h = \{j \in \{1,\ldots,m\} \mid p_j \in \mathcal{N}_{\mathcal{G}_D}(p_h,\mathcal{P})\}$. This fact implies that $\bigcup_{i\in I_h}\left(\Delta V_{ih}^{\mathcal{N}}(\mathcal{P})^+ \cup \Delta V_{ih}^{\mathcal{N}}(\mathcal{P})^-\right) = \Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^+ \cup \Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^-$. Moreover it can be easily seen that $\Delta V_{ih}^{\mathcal{N}}(\mathcal{P})^+ \subset \Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^-$ and $\Delta V_{ih}^{\mathcal{N}}(\mathcal{P})^- \subset \Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^+$ $\forall i \in I_h$. Indeed, any segment $s \in \Delta V_{ih}^{\mathcal{N}}(\mathcal{P})^+$ is such that $s \in V_i^{\mathcal{N}}(\{p_1,\ldots,p_h+\varepsilon,\ldots,p_m\})$ and $s \notin V_i^{\mathcal{N}}(\mathcal{P})$, and, for any infinitesimal perturbation of p_h, it is possible only if $s \in V_h^{\mathcal{N}}(\mathcal{P})$

and $s \notin V_h^{\mathcal{N}}(\{p_1, \ldots, p_h + \varepsilon, \ldots, p_m\})$, hence if $s \in \Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^-$. Therefore we have

$$\begin{aligned}
&\frac{\partial}{\partial p_h} \sum_{i \in I_h} \int_{V_i^{\mathcal{N}}(\mathcal{P})} f\left(\|q - p_i\|\right) \phi\left(q\right) dq \\
&= \lim_{\varepsilon \to 0} \frac{1}{\|\varepsilon\|} \sum_{i \in I_h} \left(\int_{\Delta V_{ih}^{\mathcal{N}}(\mathcal{P})^+} f\left(\|q - p_i\|\right) \phi\left(q\right) dq - \int_{\Delta V_{ih}^{\mathcal{N}}(\mathcal{P})^-} f\left(\|q - p_i\|\right) \phi\left(q\right) dq \right) \\
&= \lim_{\varepsilon \to 0} \frac{1}{\|\varepsilon\|} \left(\int_{\bigcup_{i \in I_h} \Delta V_{ih}^{\mathcal{N}}(\mathcal{P})^+} f\left(\|q - p_i\|\right) \phi\left(q\right) dq - \int_{\bigcup_{i \in I_h} \Delta V_{ih}^{\mathcal{N}}(\mathcal{P})^-} f\left(\|q - p_i\|\right) \phi\left(q\right) dq \right) \\
&= \lim_{\varepsilon \to 0} \frac{1}{\|\varepsilon\|} \left(\int_{\Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^-} f\left(\|q - p_i\|\right) \phi\left(q\right) dq - \int_{\Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^+} f\left(\|q - p_i\|\right) \phi\left(q\right) dq \right).
\end{aligned}$$

The conclusion follows from the fact that $\lim_{\varepsilon \to 0} \left(\Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^+ \cup \Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^-\right) = \partial V_h^{\mathcal{N}}(\mathcal{P})$ and $\forall q \in \partial V_h^{\mathcal{N}}(\mathcal{P})$ $\|q - p_i\| = \|q - p_h\|$. As concerns the second-last term of (22), recalling again that $\lim_{\varepsilon \to 0} \left(\Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^+ \cup \Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^-\right) = \partial V_h^{\mathcal{N}}(\mathcal{P})$ and the Assumptions 16, we can write

$$\begin{aligned}
&\lim_{\varepsilon \to 0} \frac{1}{\|\varepsilon\|} \int_{\Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^+} \left(f\left(\|q - (p_h + \varepsilon)\|\right) - f\left(\|q - p_h\|\right)\right) \phi\left(q\right) dq \\
&\leq \lim_{\varepsilon \to 0} \frac{1}{\|\varepsilon\|} \int_{\Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^+} \left\| \frac{\partial f}{\partial x} \right\|_{[0, \mathrm{diam}(\mathcal{N})]} \|\varepsilon\| \, \|\phi\|_{[0, \mathrm{diam}(\mathcal{N})]} \, dq \\
&= \lim_{\varepsilon \to 0} \left\| \frac{\partial f}{\partial x} \right\|_{[0, \mathrm{diam}(\mathcal{N})]} \|\phi\|_{[0, \mathrm{diam}(\mathcal{N})]} \, \mu\left(\Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^+\right) = 0,
\end{aligned}$$

where $\mathrm{diam}(\mathcal{N}) \triangleq \max_{p,q \in \mathcal{N}} \|q - p\|$. The same argument holds for the term with $\Delta V_{hh}^{\mathcal{N}}(\mathcal{P})^-$, hence we have

$$\begin{aligned}
\frac{\partial \mathcal{H}}{\partial p_h}(\mathcal{P}) &= \lim_{\varepsilon \to 0} \frac{1}{\|\varepsilon\|} \int_{V_h^{\mathcal{N}}(\mathcal{P})} \left(f\left(\|q - (p_h + \varepsilon)\|\right) - f\left(\|q - p_h\|\right)\right) \phi\left(q\right) dq \\
&= \int_{V_h^{\mathcal{N}}(\mathcal{P})} \frac{\partial}{\partial p_h} f\left(\|q - p_h\|\right) \phi\left(q\right) dq \\
&= \sum_{k=1}^{M_h(\mathcal{P})} \mathcal{I}_k,
\end{aligned}$$

with $M_h(\mathcal{P})$ the number of segments in $V_h^{\mathcal{N}}(\mathcal{P})$ and $\mathcal{I}_k = \int_{s_k} \frac{\partial}{\partial p_h} f\left(\|q - p_h\|\right) \phi\left(q\right) dq$ and $s_k = [a_k, b_k] \in V_h^{\mathcal{N}}(\mathcal{P})$. If we choose the parameterization $\gamma_k(t) = a_k + (b_k - a_k)\, t$, $t \in [0, 1]$ for s_k we can apply the Theorem 19 in appendix. Recall that in this case $\nu\left(x, q\right) = \|q - x\|$, hence the equation $\|\gamma_k(t) - p_h\| - R_\alpha = 0$ may have at most two zeros at $t_{\alpha,1}^k$ and $t_{\alpha,2}^k$ $\forall \alpha \in \{1, \ldots, N\}$. It is worth noting that assumptions 16 iii) and iv) play here the same role of assumptions i) and ii) in Theorem 19. Therefore, from the definition of $f(\cdot)$, equation (1), we have the thesis. ■

In order to define a gradient-like algorithm, also in this case, we must relax Assumptions 16. First of all, focus on the orthogonality assumption. It has been introduced to avoid the presence of entire segments in the boundary of a cell, because these configurations induce problems in the definition of the gradient (they represent points on which the gradient may assume different values). Even in this case we opt to use the lexicographic rule in order to univocally assign a segment on the boundary to only one cell, and, again, we consider a discrete-time dynamics for the gradient-like algorithm.

Using the lexicographic rule, we re-define the Voronoi cell as follows

$$V_i^{\mathcal{N}}(\mathcal{P}) = \{q \in \mathcal{N} \mid \|q - p_i\| \leq \|q - p_j\| \ \ \forall p_j \in \mathcal{P} \ \wedge \\ \|q - p_i\| < \|q - p_j\| \ \text{ if } j < i \text{ w.r.t. the L.O.}\}$$

and verify that the expression (18) for $\mathcal{H}(\mathcal{P})$ is still formally correct. We remove the orthogonality hypothesis by adding to (19) an $\mathcal{I}_k$ term for each segment entirely included in the boundary of a Voronoi cell. This fact does not change the expression (19), since, with the new definition $V_h^{\mathcal{N}}(\mathcal{P})$, $M_h(\mathcal{P})$ accounts now also for segments on the boundary.

The relaxation of the other assumptions would imply some discontinuities in the integration domain induced by the discontinuities of the function f. These discontinuities, without additional assumptions, would prevent us from guaranteeing $\mathcal{H}(\mathcal{P})$ to be monotonically nondecreasing along the evolution of $\mathcal{P}$ given by the gradient dynamics. Hence, we assume now f to be continuous and piecewise differentiable. Being f continuous, the second term in $\mathcal{I}_k$ in (19) is null.

As made in the previous section, the directional derivative must be univocally defined on the vertices. To this aim, we use the expression (16) given in Definition 14, but with reference to the formula (19) for the directional derivative in a point in the interior of a segment. Using these definitions we can state the following theorem.

Theorem 18: Consider the following discrete-time evolution for the sensors' positions

$$\mathcal{P}^{(k+1)} = \mathcal{P}^{(k)} + \delta_k \widetilde{D}\mathcal{H}(\mathcal{P}^{(k)}), \tag{23}$$

where the h-th component of $\widetilde{D}\mathcal{H}$ is given by (16) and $D_{w_{ij}}\mathcal{H}(\mathcal{P})[p_h]$ by (19) and $\mathcal{H} : \mathcal{N}^m \to \mathbb{R}$ as in (18). If $f(\cdot)$ has locally bounded second derivatives, then, for suitable δ_k, $\mathcal{P}^{(k)}$ lies in a bounded set and

i) $\mathcal{H}(\mathcal{P}^{(k)})$ is monotonically nondecreasing;

ii) $\mathcal{P}^{(k)}$ converges to the set of critical points of $\mathcal{H}$.

a	20	20	20	10	4	20	20	20	10	10	4
c_x	4.3	5	6	3.5	9	12.5	13.5	15	14.5	17	20
c_y	2.3	4	5.5	5	8.5	8.5	7.2	6.2	10.5	9	7
σ_x	1.5	1.5	1.5	2	4	1.5	1.5	1.5	2	2	4
σ_y	1.5	1.5	1.5	2	4	1.5	1.5	1.5	2	2	2

TABLE I

PARAMETER VALUES FOR THE 11 GAUSSIAN FUNCTIONS $G(a, c_x, c_y, \sigma_x, \sigma_y) = a \exp\left(-\left(\frac{x-c_x}{\sigma_x}\right)^2 - \left(\frac{y-c_y}{\sigma_y}\right)^2\right)$ MAKING UP THE DENSITY FUNCTION ϕ.

Proof: As long as sensors' configurations not violating orthogonality assumption are considered, the gradient is smooth and the proof is canonical. In the case of discontinuity points, segments belonging to the boundary of a cell can change allocation during sensors' motion. Hence, we can proceed as in the proof of Theorem 10 and 15 replacing barycenters with segments. In particular, being s_h a segment changing allocation, equations (13) are now replaced by

$$\mathcal{H}_{p_i}(\mathcal{P}^{(k+1)}) = \mathcal{H}^u_{p_i}(\mathcal{P}^{(k+1)}) + \mathcal{I}^i_h$$
$$\mathcal{H}_{p_j}(\mathcal{P}^{(k+1)}) = \mathcal{H}^u_{p_j}(\mathcal{P}^{(k+1)}) - \mathcal{I}^j_h,$$

with $\mathcal{I}^l_k = \int_{s_h} \frac{\partial}{\partial p_l} f\left(\|q - p_l\|\right) \phi\left(q\right) dq$. Again, due to the fact that any point $q \in s_h$ is such that $\left\|q - p_i^{(k+1)}\right\| \leq \left\|q - p_j^{(k+1)}\right\|$ and the monotonicity of f, we have $\mathcal{I}^i_h \geq \mathcal{I}^j_h$, whereby the thesis follows as in the proof of Theorem 10. ■

V. A CASE STUDY

In this section we apply the proposed two-step optimization procedure to a network representing a wing of the Amsterdam Schiphol airport. The network is made up of 63 vertices and 87 segments and the density function ϕ is the sum of 11 Gaussian functions of the form $G(a, c_x, c_y, \sigma_x, \sigma_y) = a \exp\left(-\left(\frac{x-c_x}{\sigma_x}\right)^2 - \left(\frac{y-c_y}{\sigma_y}\right)^2\right)$ with parameters assuming the values given in table I.

The network and a contour plot of the density function are shown in fig. 1 (darker colors denote preferential areas). For sake of providing a clear graphical representation, the density function shown here is defined on $\mathbb{R}^2$, but the one used in all simulations is restricted to the network.

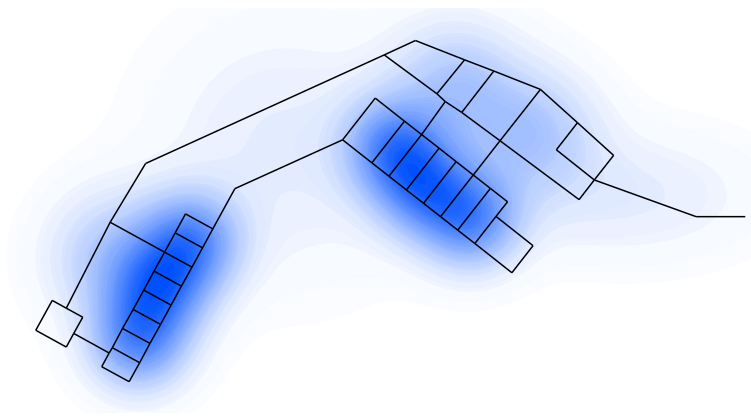


Fig. 1. Network representing a wing of the Amsterdam Schiphol airport and contour plot of the density function ϕ used in simulations (darker colors denote preferential areas).

The 50 sensors to be deployed have performance function $f(x) = \frac{1}{2}\left(1 - \tanh\left(\frac{x-\frac{R}{2}}{\frac{R}{6}}\right)\right)$, where R is a parameter considered as variable in the first step and as fixed in the second step of the optimization. Even if the previous function describes sensors with an infinite sensing radius, they will be represented as shaded circles of radius $\frac{7}{8}R$ to emphasize that the performance function assumes values lesser than 0.01 for larger distances.

A. *First step*

The first step of the optimization is performed on a collapsed network with collapsing factor $r = 0.3$ (see the small dots along the grey network in fig. 2-a,-c)). Sensors are grouped in 10 clusters of 5 elements each, and each cluster is represented as a single sensor. Clusters set initially $R = 10$ and decrease linearly its value up to 1 during the simulation (compare fig. 2-a) with fig. 2-c)). As apparent by the flows in fig. 2-b), clusters are allowed to move in $\mathbb{R}^2$.

It is important to recall that, both the variation of the sensing radius and the unconstrained motion of sensors are allowed in the first step as it is performed off-line. This step makes use of the algorithm described in section III-A and is thought to provide a good starting point for the second step. However, if sensors are initially located on the network as in fig. 2-a), they can execute the first step independently, using partial or rough information of the environment, without moving, and then plan a route on the network to reach the previously

computed final positions. Since final positions can be not on the network (see fig. 2-c)), they must be projected on it to be reachable. Anyway, this projection has to be performed before the second step to provide a valid starting point.

B. Second step

The second step considers a full network with sensors having fixed radius $R = 1$ and initially deployed in the positions shown in fig. 3-a). Such positions are obtained by spreading randomly 5 sensors close to each cluster center and projecting them on the closest segments of the network. Sensors now can take real measures from the environment and perform the optimization on-line, while moving, according to the algorithm described in section IV. They are constrained to move on the network as shown in fig. 3-b). Final positions (see fig. 3-c)) show how sensors, originally clustered, diffused to better cover preferential areas (see fig 1).

VI. Conclusions and Future Works

This paper focused on the problem of optimally deploying sensors in an environment modeled as a network. An optimization problem for the allocation of omnidirectional sensors with potentially limited sensing radius has been formulated. A novel two-step optimization procedure based on a discrete-time gradient ascent algorithm has been presented. In order the algorithm to not get stuck early in one of the many local minima, in the first (off-line) step, sensors are allowed to move in the plane. Moreover, a reduced model of the environment, called collapsed network, and sensors' clustering, are used to speed up the first optimization. The positions found in the first step are then projected on the network and used in the second (on-line) finer optimization, where sensors are constrained to move only on the network.

The proposed procedure can be used to solve both static and dynamic deployment problems and the first step alone can provide solutions to location-allocation problems involving facilities located in the interior of the network.

A main future research direction will consider the integration of classical Operative Research methods with the present gradient algorithm. In particular the first optimization could be addressed by adapting methods and heuristics developed for the solution of the multi-source Weber problem ([9]). The aim is to build an overall global optimization technique to solve location-allocation problems of large dimensions with many facilities. Moreover, future

research, more related to deployment problems, will consider other sensor's models such as those with limited sensing cone.

APPENDIX

Theorem 19: Let $\varphi : \mathbb{R}^2 \times \mathbb{R}_+ \to \mathbb{R}$ be a smooth function w.r.t. its second argument and a non-increasing, piecewise differentiable map with a finite number of bounded discontinuities at $R_1, \ldots, R_N \in \mathbb{R}_+$, $R_1 < \ldots < R_N$, w.r.t. its first argument. Let $\nu : \mathbb{R}^2 \times \mathbb{R}^2 \to \overline{\mathbb{R}}_+$ be a continuously differentiable map w.r.t. both its arguments. Let $s = [a, b] \subset \mathbb{R}^2$ be a segment and assume that

i) $\nu(\bar{x}, a), \nu(\bar{x}, b) \notin \{R_1, \ldots, R_N\}$;

ii) $\forall q \in [a, b]$, if $\nabla\nu(\bar{x}, q) \cdot (b - a) = 0 \Rightarrow \nu(\bar{x}, q) \notin \{R_1, \ldots, R_N\}$,

then

$$\begin{aligned}
&\frac{d}{dx} \int_s \varphi\left(\nu\left(x, q\right), q\right) dq \Big|_{x=\bar{x}} \\
&= \int_{[0,1]\setminus\left\{t_{1,1},\ldots,t_{1,k_1},\ldots,t_{N,1},\ldots,t_{N,k_N}\right\}} \frac{\partial}{\partial\nu}\varphi(\nu(\bar{x}, \gamma(t)), \gamma(t)) \left.\frac{\partial\nu(x, \gamma(t))}{\partial x}\right|_{x=\bar{x}} \|b - a\| \, dt \\
&+ \sum_{i=1}^{N}\sum_{j=1}^{k_i} \left(\varphi(R_i^+, \gamma(t_{i,j})) - \varphi(R_i^-, \gamma(t_{i,j}))\right) \left.\frac{\partial\nu(x, \gamma(t))}{\partial x}\right|_{\substack{x=\bar{x}\\ t=t_{i,j}}} \|b - a\|,
\end{aligned}$$

where $\gamma(t) = a + (b - a)\, t,\ t \in [0, 1]$ is a parameterization for s and $t_{i,j} \in [0, 1],\ j \in \{1, \ldots, k_i\}$ are the zeros of $\nu(\bar{x}, \gamma(t)) - R_i = 0$.

Proof: By using the Dirac's delta formalism we have

$$\begin{aligned}
&\frac{d}{dx} \int_s \varphi\left(\nu\left(x, q\right), q\right) dq \Big|_{x=\bar{x}} \\
&= \int_s \frac{\partial}{\partial x}\varphi(\nu(x, q), q) dq \Big|_{x=\bar{x}} \\
&= \int_s \frac{\partial}{\partial\nu}\varphi(\nu(x, q), q)\Big|_{\nu\notin\{R_1,\ldots,R_N\}} \left.\frac{\partial\nu(x, q)}{\partial x}\right|_{x=\bar{x}} dq \\
&+ \int_s \sum_{i=1}^{N} \left(\varphi(R_i^+, q) - \varphi(R_i^-, q)\right) \delta\left(\nu - R_i\right) \left.\frac{\partial\nu(x, q)}{\partial x}\right|_{\nu=R_i} dq.
\end{aligned}$$

With the chosen parameterization of s, the equation $\nu(\bar{x}, \gamma(t)) - R_i = 0$ may have k_i zeros $t_{i,j} \in [0, 1],\ j \in \{1, \ldots, k_i\}$. Thanks to i) and ii) the set of zeros $t_{i,j}$ does not change cardinality for x near $\bar{x}$, thus, $t_{i,j}$ depends smoothly on x. Recalling that for every arc $\gamma : [l, u] \to \Gamma$ we have

$$\int_\Gamma f(x) dx = \int_l^u f(\gamma(t)) \|\dot{\gamma}(t)\| \, dt,$$

and that for the special case of $s\ \|\dot{\gamma}(t)\| = \|b-a\|$, the derivative becomes

$$\begin{aligned}
&\frac{d}{dx}\int_s \varphi\left(\nu\left(x,q\right),q\right)dq\Big|_{x=\bar{x}} \\
&= \int_{[0,1]\setminus\left\{t_{1,1},\ldots,t_{1,k_1},\ldots,t_{N,1},\ldots,t_{N,k_N}\right\}} \frac{\partial}{\partial\nu}\varphi(\nu(\bar{x},\gamma(t)),\gamma(t))\ \frac{\partial\nu(x,\gamma(t))}{\partial x}\Big|_{x=\bar{x}} \|b-a\|\,dt \\
&+ \int_{[0,1]}\sum_{i=1}^{N}\left(\varphi(R_i^+,\gamma(t)) - \varphi(R_i^-,\gamma(t))\right)\delta\left(\nu-R_i\right)\frac{\partial\nu(x,\gamma(t))}{\partial x}\Big|_{\nu=R_i}\|b-a\|\,dt,
\end{aligned}$$

from which, using the property $\int_l^u f(x)\delta\left(x-\bar{x}\right)dx = f(\bar{x})$ for $\bar{x}\in[l,u]$, we have the thesis.

■

REFERENCES

[1] Z. Drezner, *Facility Location: A Survey of Applications and Methods*, ser. Series in Operations Research, Z. Drezner, Ed. Springer Velag, New York, 1995.

[2] E. Erkut and S. Neuman, "Analytical models for locating undesirable facilities," *Eur. J. Oper. Res.*, vol. 40, no. 3, pp. 275–291, 1989.

[3] P. Cappanera, "A survey on obnoxious facility location problems," Dept. of Informatics - Univ. of Pisa, Tech. Rep. TR-99-11, April 1999.

[4] Z. Drezner and A. Suzuki, "The big triangle small triangle method for the solution of nonconvex facility location problems," *Operations Research*, vol. 52, no. 1, pp. 128–135, 2004.

[5] T. Drezner and Z. Drezner, "Location of a facility in a planar network," in *38th SWDSI Annual Conf.*, San Diego, CA, USA, 13–17 March 2007.

[6] A. Suzuki and Z. Drezner, "The p-center location problem in an area," *Location Science*, vol. 4, no. 1-2, pp. 69–82, 1996.

[7] J. Cortés and F. Bullo, "Coordination and geometric optimization via distributed dynamical systems," *SIAM J. Contr. Optim.*, vol. 44, no. 5, pp. 1543–1574, 2005.

[8] N. Mladenović, J. Brimberg, P. Hansen, and J. A. Moreno-Pérez, "The p-median problem: A survey of metaheuristic approaches," *Eur. J. Oper. Res.*, vol. 179, no. 3, pp. 927–939, 2007.

[9] J. Brimberg, P. Hansen, N. Mladenović, and E. C. Taillard, "Improvements and comparison of heuristics for solving the uncapacitated multisource Weber problem," *Operations Research*, vol. 48, no. 3, pp. 444–460, 2000.

[10] J. Cortés, S. Martínez, T. Karataş, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Trans. Robot. and Automat.*, vol. 20, no. 2, pp. 243–255, 2004.

[11] Q. Du, V. Faber, and M. Gunzburger, "Centroidal Voronoi tessellations: Applications and algorithms," *SIAM Rev.*, vol. 41, no. 4, pp. 637–676, 1999.

[12] S. P. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 129–137, 1982.

[13] P. Sharma, S. Salapaka, and C. Beck, "A scalable deterministic annealing algorithm for resource allocation problems," in *Proc. American Control Conf.*, Minneapolis, MN, USA, 14–16 June 2006, pp. 3092–3097.

[14] A. Okabe and A. Suzuki, "Locational optimization problems solved through Voronoi diagrams," *Eur. J. Oper. Res.*, vol. 98, no. 3, pp. 445–456, 1997.

[15] A. Ganguli, S. Susca, S. Martínez, F. Bullo, and J. Cortés, "On collective motion in sensor networks: Sample problems and distributed algorithms," in *Proc. IEEE Int. Conf. on Decision and Control and Eur. Control Conference*, Seville, Spain, Dec. 2005, pp. 4239–4244.

[16] S. Martínez, J. Cortés, and F. Bullo, "Motion coordination with distributed information," *IEEE Contr. Syst. Mag.*, vol. 27, no. 4, pp. 75–88, 2007.

[17] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Trans. Automat. Contr.*, vol. 48, no. 6, pp. 988–1001, 2003.

[18] J. Cortés, S. Martínez, and F. Bullo, "Spatially-distributed coverage optimization and control with limited-range interactions," *ESAIM Contr. Optim. & Calc. of Variations*, vol. 11, pp. 691–719, 2005.

[19] A. Howard, M. J. Matarić, and G. S. Sukhatme, "Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem," in *Proc. 6th Int. Symp. on Distrib. Auton. Rob. Systems*, H. Asama, T. Arai, T. Fukuda, and T. Hasegawa, Eds. Springer, 2002, pp. 299–308.

[20] Y. Zou and K. Chakrabarty, "Sensor deployment and target localization in distributed sensor networks," *ACM Trans. Embedded Computing Sys.*, vol. 3, no. 1, pp. 61–91, 2004.

[21] Y. Mao and M. Wu, "Coordinated sensor deployment for improving secure communications and sensing coverage," in *3rd ACM Wor. On Security of Ad Hoc and Sensor Networks*. Alexandria, VA, USA: ACM Press New York, NY, USA, Nov. 2005, pp. 117–128.

[22] N. Heo and P. K. Varshney, "Energy-efficient deployment of intelligent mobile sensor networks," *IEEE Trans. Syst. Man and Cyber. – Part A*, vol. 35, no. 1, pp. 78–92, Jan. 2005.

[23] C. H. Caicedo-Núñez and M. Žefran, "Performing coverage on nonconvex domains," in *17th IEEE Int. Conf. on Control Applications*, Sept. 2008, pp. 1019–1024.

[24] B. C. Tansel, R. L. Francis, and T. J. Lowe, "Location on networks: A survey. Part I: The p-center and p-median problems," *Manag. Science*, vol. 29, no. 4, pp. 482–497, 1983.

[25] ——, "Location on networks: A survey. Part II: Exploiting tree network structure," *Manag. Science*, vol. 29, no. 4, pp. 498–511, 1983.

[26] A. Krause, C. Guestrin, A. Gupta, and J. Kleinberg, "Near-optimal sensor placements: Maximizing information while minimizing communication cost," in *Proc. of Inf. Processing in Sensor Networks*, 2006.

[27] A. Krause, J. Leskovec, C. Guestrin, J. VanBriesen, and C. Faloutsos, "Efficient sensor placement optimization for securing large water distribution networks," *J. of Water Resources Plan. and Manag.*, vol. 134, no. 6, pp. 516–526, 2008.

[28] A. Krause, R. Rajagopal, A. Gupta, and C. Guestrin, "Simultaneous placement and scheduling of sensors," in *Proc. ACM/IEEE Int. Conf. on Inf. Processing in Sensor Networks*, 2009.

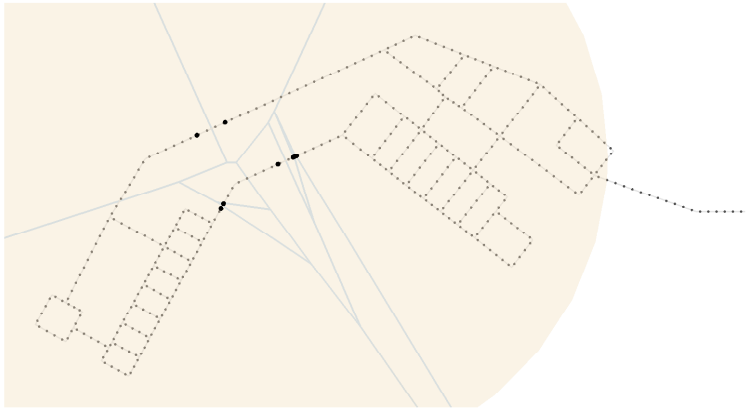
[29] Z. Guo, M. Zhou, and G. Jiang, "Adaptive sensor placement and boundary estimation for monitoring mass objects," *IEEE Trans. Syst. Man and Cyber. – Part B: Cybernetics*, vol. 38, no. 1, pp. 222–232, Feb. 2008.

[30] C. Gao, J. Cortés, and F. Bullo, "Notes on averaging over acyclic digraphs and discrete coverage control," *Automatica*, vol. 44, no. 8, pp. 2120–2127, 2008.

[31] H. J. Kushner and G. Yin, *Stochastic approximation and recursive algorithms and applications*. Springer, 2003.

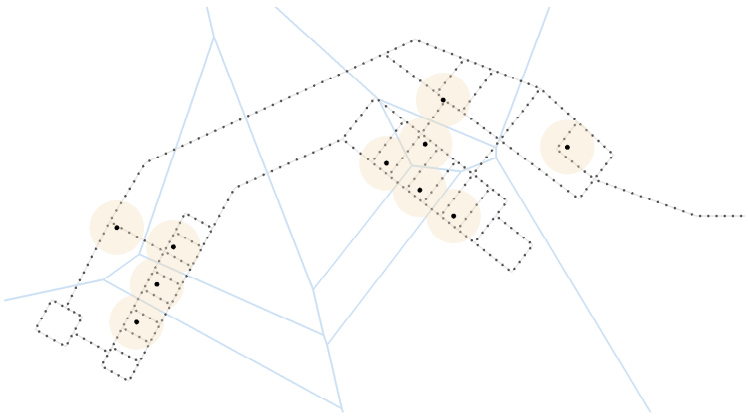
[32] V. S. Borkar, *Stochastic approximation: A dynamical systems viewpoint*. Cambridge University Press, 2008.

[33] J. N. Tsitsiklis, D. P. Bertsekas, and M. Athans, "Distributed asynchronous deterministic and stochastic gradient optimization algorithms," *IEEE Trans. Automat. Contr.*, vol. 31, no. 9, pp. 803–812, Sep. 1986.



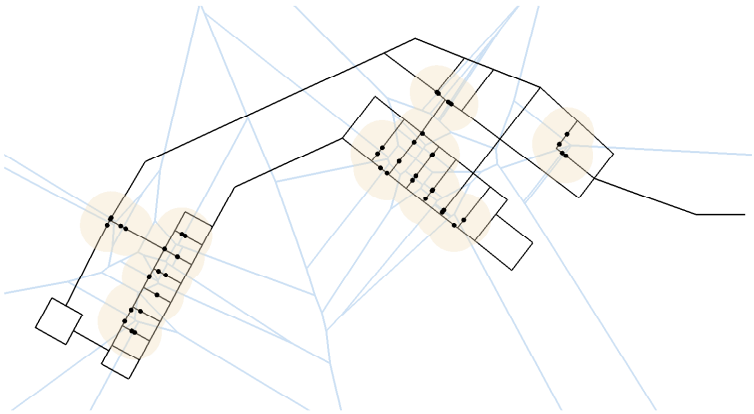
a)

b)



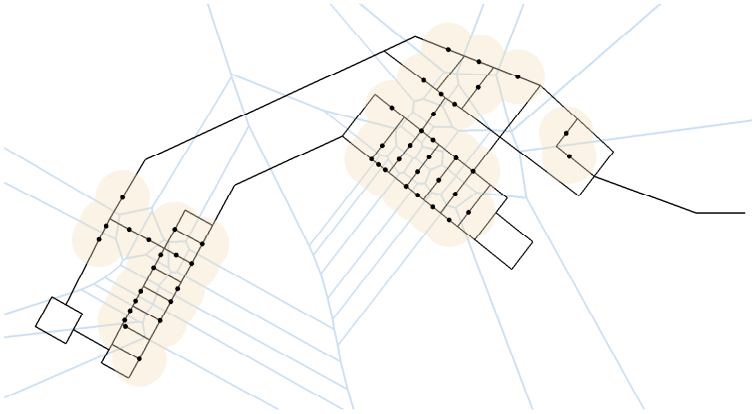
c)

Fig. 2. First step of the optimization procedure for the deployment over a collapsed network of 50 sensors grouped in 10 clusters: a) initial positions of clusters (sensing radius $R = 10$) and related Voronoi partition; b) gradient ascent flows (clusters allowed to move in $\mathbb{R}^2$); c) final positions, not necessarily belonging to the network, (sensing radius $R = 1$) and related Voronoi partition.



a)

b)



c)

Fig. 3. Second step of the optimization procedure for the deployment of 50 sensors with $R = 1$ over a full network: a) initial positions obtained by spreading randomly 5 sensors close to each cluster position found in the first step (see fig. 2-c)) and then projecting them on the network, related Voronoi partition; b) gradient ascent flows (sensors constrained to move on the network); c) final positions and related Voronoi partition.