Proceedings of the
47th IEEE Conference on Decision and Control
Cancun, Mexico, Dec. 9-11, 2008

Deployment of sensors in a network-like environment

Luca Greco, Matteo Gaeta, Benedetto Piccoli

Abstract— **This paper considers a deployment problem involving omnidirectional sensors with potentially limited sensing radius. The environment is modeled as a network and two optimization problems are formulated and solved. The first one introduces some simplifications, allowing sensors to be located everywhere in the polytope enclosing the network, and considering a reduced model for the environment called collapsed network. It is made up of a finite discrete set of points, barycenters, produced by collapsing the network edges. The second problem considers a classical graph model and forces sensors to stay on the network. We propose a discrete-time gradient ascent algorithm to find a local optimum for these problems. The present algorithm can also be implemented in a distributed fashion.**

I. Introduction

A great number of situations ranging from surveillance to habitat and environment monitoring, from wild fire detection to search and rescue operations, from exploration to intruder detection, would greatly enjoy the use of network of sensors. 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 sensor 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 resource or facilities allocation problems which are the subject of the locational optimization discipline [1].

In locational optimization some 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 attraction 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.

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

B. Piccoli is with the Istituto per le Applicazioni del Calcolo "Mauro Picone", Viale Policlinico 137, 00161 Roma, Italy; e-mail: b.piccoli@iac.cnr.it

A basic distinction among location optimization problems can be performed according to the number of facilities involved (one or $p > 1$ facilities).

Two well known 1-facility problems are the classical Weber and the obnoxious facility location problems (see [4], [5] for a recent heuristic solution).

Three classical 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 can be simply described as the one of finding the optimal allocation of n facilities by minimizing the average distance of the demand points to the nearest facility. A more general formulation can be found in [8] and [9]. In [8] a dynamical (gradient descent) version of the Lloyd's algorithm [10] has been presented to find a local optimum for a generalized p-median problem. A different algorithm to solve the classical version is reported in [11], [12]. The aforementioned solutions to the p-median problem, as long as many solutions to p-facilities problems, are based on the construction of a Voronoi Tesselation ([9], [13]).

B. Dynamic Deployment and Distributed Solutions

The use of moving, instead of static, sensor network provides 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 losses. 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 ([14], [15], [16]). Many of the algorithms proposed in the previous sections 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], [8], [17], 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.

Many other solutions to the area-coverage problem look at sensors like particles subject to virtual forces or potential

978-1-4244-3124-3/08/$25.00 ©2008 IEEE

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 [18], [19]. Another example of this kind of approach is [20], where also secure connectivity issues are considered.

Power consumption is a relevant problem in wireless networks as remarked in [21], where three energy-efficient algorithms are presented for sensors' deployment.

C. Network-like Environments

The area where locating facilities or sensors is sometimes better described by a network-like environment. It happens, for instance, whenever facilities have to be located on road networks, or river networks, or networks of shipping lanes. Or when the use of reduced model abstracting from many of the geometrical features of the environment is desirable. Consider, for instance, a surveillance task in an airport environment. In this case people moving throughout the airport is compared with a network flow and the focus is on paths more than on corridors, halls and lounges.

Many of the previous problems have been formulated even for network-like environment [22], [23]. Usually these problems 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. A different problem involving a network is presented in [5], where one facility can be located in any point of the convex hull of the network and the network is considered as the source of (or subject to) a nuisance.

In this paper we consider a generalized p-median problem involving a network environment and omnidirectional sensors with potentially limited sensing radius. The task is to find sensors' position to cover the network optimizing an objective function defined on the network and accounting for the sensor features and preferential areas. This is a mixed problem, since the network is considered as a subset of a plane (there is not a discrete set of demand points) and the planar euclidean distance is used. We present a discrete-time gradient ascent Lloyd's algorithm to find a local optimum to this problem. It is worth noting that this solution can be used to solve a static deployment problem as much as a dynamic one, since it is suitable for a distributed implementation.

The overall formulation is indebted to the works of Bullo and his coworkers [17], [8], [15], but our network-based approach needs quite different solutions. Due to the different topology induced by the network model, many issues related to the explicit computation of the gradient must be considered along with convergence problems of the maximization algorithm. In order to aptly tackle the deployment problem, we solve first a subproblem involving a simplified network that we call *collapsed network* and consisting of finite many points (barycenters). In this problem sensors are also allowed to move in the plane ($\mathbb{R}^2$). The solution found for this simplified problem is then used to solve the general problem involving the full network and sensors moving on it.

II. Preliminaries

In this section we provide some useful definitions for the network describing the environment and its Voronoi partition.

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 $s^o_{12} = (p_1, p_2)$ 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) $\forall i, j, h, k \in \{1, \ldots, n\}$, $(i, j) \neq (h, k)$, $s^o_{ij} \cap s^o_{hk} = \emptyset$ (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 partition of $\mathcal{N}$ generated by $\mathcal{P}$ with respect to the Euclidean norm is the collection of sets $\{V_i^{\mathcal{N}}(\mathcal{P})\}_{i\in\{1,\ldots,m\}}$ defined by

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

III. Sensor deployment over a network

We adapt the framework provided in [17] 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 partition induced by $\mathcal{P}$ as follows

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

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 $h, k \in \{1, \ldots, m\}$.

A. Collapsed network and sensors moving in $\mathbb{R}^2$

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 a 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 4 (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 barycentric points. Hence we have

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

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

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

Definition 5: 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 partition 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

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

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

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

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

Theorem 1: The multi-center function $\mathcal{H}$ is continuously differentiable on $\left(\mathbb{R}^2\right)^m \setminus \left(\mathcal{D}_{\mathcal{C}_r^{\mathcal{N}}}\right)^m$, 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 $h \in \{1, \ldots, m\}$

$$\frac{\partial \mathcal{H}(\mathcal{P})}{\partial p_h} = \sum_{b_e \in V_h^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P})} \frac{\partial}{\partial p_h} f\left(\|b_e - p_h\|\right) \phi_{b_e}. \tag{6}$$

Proof: The proof of this theorem has been removed due to space limitations. The same holds for the proofs of the other theorems throughout the paper. ■

The sensors' location problem can be solved 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

$$\dot{\mathcal{P}} = \nabla \mathcal{H}(\mathcal{P}). \tag{7}$$

Unfortunately, this dynamics conveys many problems. It is well defined as long as the hypotheses of Assumption 1 and Theorem 1 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 1 induces some problems on the definition of the gradient. Barycenters on a boundary segment of a Voronoi cell belong to all the cells sharing that segment. 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 (7) a set of differential equations with discontinuous right-hand side. Being $\mathcal{H}(\cdot)$ at least locally Lipschitz ($f(r)$ is assumed to be continuously differentiable w.r.t. r), it would be natural to look for Filippov solutions of the aforementioned differential equation ([24]). This choice ensures the existence and uniqueness of solutions for the equation (7), at least in a generalized sense, but the proof of convergence of the gradient-like algorithm to the local maxima of $\mathcal{H}(\cdot)$ becomes harder. Indeed, there can be many shared barycenters on boundary segments, thus preventing the decomposition of $\mathcal{H}(\cdot)$ in a sum of functions each accounting for the contribution of a single cell.

In order to avoid the problems introduced by Filippov solutions, a possible choice is to add a lexicographic criterion to the definitions based only on the euclidean distance. This criterion allows barycenters to belong univocally to the sensor having the lower index (w.r.t. the Lexicographic Order (L.O.)) among the sharing sensors. Let us define again the Voronoi cell for a collapsed cell as follows (compare with Definition 4):

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

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 (6). $\forall b_e \in \partial V_i^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P})$ 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

$$\frac{\partial_l \mathcal{H}(\mathcal{P})}{\partial p_h} \triangleq \sum_{b_e \in V_h^{\mathcal{C}_r^{\mathcal{N}}}(\mathcal{P})} \frac{\partial}{\partial p_h} f\left(\|b_e - p_h\|\right) \phi_{b_e}, \tag{8}$$

which is formally equal to (6).

Unfortunately, the differential equation using this new definition for the gradient does not imply, as in the Filippov case, 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 (8) based on the lexicographic criterion accounts only for infinitesimal perturbations of sensors' position not inducing any change in the allocation of barycenters to cells. In other terms, each barycenter is not allowed to enter or exit the cells. In order to avoid all these problems, we look for a discrete-time implementation of the gradient algorithm. It is worth noting that the discrete-time implementation is not the solution for any problem with discontinuity gradients. It works in our case due to the properties of the function $\mathcal{H}$ and its discontinuity points, as it is shown by the following theorem.

Theorem 2: Consider the following discrete-time evolution for the sensors' positions

$$\mathcal{P}^{(k+1)} = \mathcal{P}^{(k)} + \delta_k \nabla_l \mathcal{H}(\mathcal{P}^{(k)}), \tag{9}$$

where the h-th component of $\nabla_l \mathcal{H}$ is given by (8) 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 $\mathcal{H}$.

Remark 1: 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, we can always choose a suitable δ_k to avoid the occurrence of these configurations.

Remark 2: 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 [17], this kind of algorithms is spatially distributed, with the meaning that each sensor needs only to know the position of its neighbors in order to determine the boundary of its cell and, hence, to compute its next movement. This property makes the algorithm suitable for a realistic asynchronous distributed implementation, provided that some further conditions are imposed (see [8]). In the same paper algorithms for the computation and maintenance of the Voronoi cell by each sensor, are also presented.

Remark 3: The dynamics in (7) should be considered as an abstraction of the real sensors' dynamics as well as the network model is an abstraction of the real environment. A realistic implementation of our discrete-time algorithm requires each sensor to be endowed with a local controller charged with motion planning tasks.

B. Full network and sensors moving on it

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

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

Assumption 2: We make the following assumptions:

i) orthogonality assumption: $\forall h, k \in \{1, \dots, n\}, \forall i \in \{1, \dots, m\}$ 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) $\partial V_i^{\mathcal{N}}(\mathcal{P}) \cap \mathcal{D}_{\mathcal{N}}(\mathcal{P}) = \emptyset$, $\forall i \in \{1, \dots, m\}$;
iii) $\mathcal{V} \cap \mathcal{D}_{\mathcal{N}}(\mathcal{P}) = \emptyset$;
iv) $\forall h, k \in \{1, \dots, n\}$, $\forall i \in \{1, \dots, m\}$, $\forall q \in s_{hk} \cap V_i^{\mathcal{N}}(\mathcal{P})$, if $(q - p_i) \cdot (v_h - v_k) = 0 \Rightarrow \|q - p_i\| \notin \{R_1, \dots, R_N\}$;

With the orthogonality assumption the expression (3) simplifies to

$$\mathcal{H}(\mathcal{P}) = \sum_{i=1}^{m} \int_{V_i^{\mathcal{N}}(\mathcal{P})} f\left(\|q - p_i\|\right) \phi\left(q\right) dq. \tag{10}$$

Since the sensors have to remain on the network, we cannot use directly the gradient. We must consider now the directional derivative of $\mathcal{H}$ along the edges of the network. Unfortunately, this fact implies that on the vertices of the network the directional derivative is a multivalued function as more than one edge can share the same vertex.

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

Theorem 3: Given a network $\mathcal{N} = (\mathcal{V}, \mathcal{S})$ if Assumption 2 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

$$D_{w_{ij}} \mathcal{H}(\mathcal{P})[p_h] = \frac{\partial \mathcal{H}}{\partial p_h}(\mathcal{P}) \cdot w_{ij} \tag{11}$$
$$\frac{\partial \mathcal{H}}{\partial p_h}(\mathcal{P}) = \sum_{k=1}^{M_h(\mathcal{P})} \mathcal{I}_k,$$

where $\mathcal{I}_k$ is reported at the top of the next page, $\gamma_k(t) = a_k + (b_k - a_k)\, t, \ t \in [0, 1]$ 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).

In order to define a gradient-like algorithm, also in this case, we must relax Assumptions 2. 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. As in the case with barycenters, this choice conveys some problems as long as a continuous-time implementation is looked for the deployment algorithm. In particular, with this definition, the gradient can be univocally defined if all the infinitesimal perturbations inducing a change in the assignment of entire segments are ignored. Therefore, we consider a discrete-time dynamics for the gradient-like algorithm.

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

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

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

and verify that the expression (10) for $\mathcal{H}(\mathcal{P})$ is still formally correct. We remove the orthogonality hypothesis by adding to (11) an $\mathcal{I}_k$ term for each segment entirely included in the boundary of a Voronoi cell. This fact does not change the expression (11), 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 (11) is null.

The directional derivative must be univocally defined on the vertices. To this aim, 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 6: Given the set

$$S_{v_i} = \left\{ s \in \mathcal{S} \mid \exists v_j \in \mathcal{V}, \exists \bar{\delta} > 0 \text{ s.t. } s = [v_i, v_j] \vee s = [v_j, v_i], \; \forall \delta \in [0, \bar{\delta}] \; v_i + \delta D_{w_{ij}} \mathcal{H}(\mathcal{P})[v_i] \in \mathcal{N} \right\}, \quad (12)$$

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

$$\widetilde{D}_h \mathcal{H}(\mathcal{P}) = \begin{cases} D_{w_{ij}} \mathcal{H}(\mathcal{P})[p_h] \text{ given by (11)} & \forall p_h \in \mathcal{N} \setminus \mathcal{V} \\ \max\limits_{s_{ij} \in S_{v_i}} D_{w_{ij}} \mathcal{H}(\mathcal{P})[p_h] & \begin{array}{c}\forall p_h \equiv v_i \in \mathcal{V} \\ S_{v_i} \neq \emptyset\end{array} \\ 0 & \begin{array}{c}\forall p_h \equiv v_i \in \mathcal{V} \\ S_{v_i} = \emptyset\end{array} \end{cases} \quad (13)$$

Using these definitions we can state the following theorem.

Theorem 4: Consider the following discrete-time evolution for the sensors' positions

$$\mathcal{P}^{(k+1)} = \mathcal{P}^{(k)} + \delta_k \widetilde{D} \mathcal{H}(\mathcal{P}^{(k)}), \quad (14)$$

where the h-th component of $\widetilde{D}\mathcal{H}$ is given by (13) and $D_{w_{ij}} \mathcal{H}(\mathcal{P})[p_h]$ by (11) and $\mathcal{H} : \mathcal{N}^m \to \mathbb{R}$ as in (10). 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 $\mathcal{H}$.

IV. SIMULATIONS

In this Section we show some simulations illustrating the effectiveness of the presented algorithm in the two cases previously analyzed. In the examples we used the same randomly generated network with 50 vertices and 122 segments. The network can be enclosed in a box whose side has length 5. We considered the presence of 30 sensors with the following continuous performance function $f(x) = \frac{1}{2}\left(1 - \tanh\left(\frac{x - \frac{1}{2}R}{\frac{R}{6}}\right)\right)$, where R is a parameter fixed to the value $R = 0.8$. Even if, with this function, sensors have an infinite visibility radius, we represent them with a circle whose radius is $\frac{7}{8}R = 0.7$ to emphasize that the performance function assumes values lesser than 0.01 for larger distances.

In the first example (see Fig. 1) sensors can move in $\mathbb{R}^2$, the performance function is given by $\phi = 20\exp\left(-(x-1)^2 - (y-4)^2\right) + 20\exp\left(-(x-4)^2 - (y-1)^2\right)$ and the network has been collapsed with collapsing factor $r = 0.3$. As apparent from the figures, sensors gather near the red barycenters having a higher value of the density function (preferential areas).

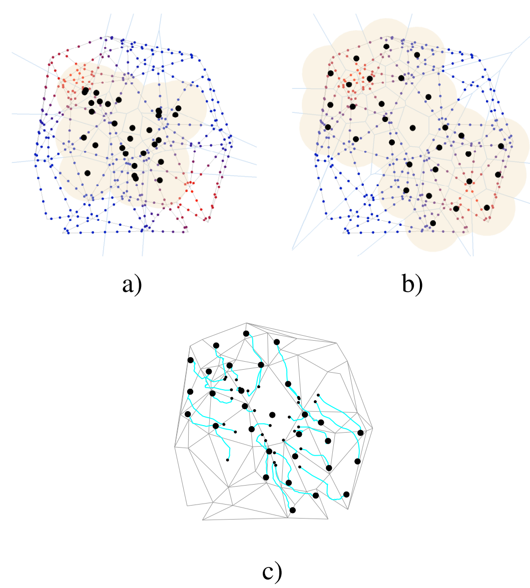
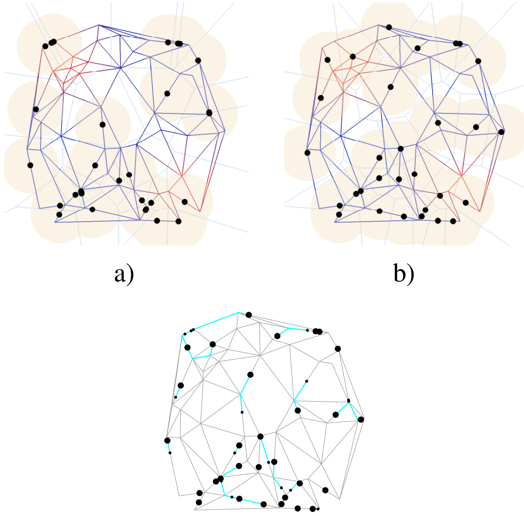


Fig. 1. Deployment problem of 30 sensors in an environment described by a collapsed network. Collapsing factor $r = 0.3$, density function $20\left(\exp\left(-(x-1)^2 - (y-4)^2\right) + \exp\left(-(x-4)^2 - (y-1)^2\right)\right)$. Figure a) illustrates the initial positions and the Voronoi partition, whereas figure b) illustrates the final positions. Figure c) shows the gradient ascent flow. Notice that sensors can move in $\mathbb{R}^2$.

The second example (see Fig. 2), shows a network not collapsed but with the same density function of the first example. Sensors are now forced to move on the network (see the gradient ascent flows in Fig. 2-c). The last example shows different solutions with respect to the first one, mainly



a) b)

c)

Fig. 2. Deployment problem of 30 sensors in an environment described by a (full) network. Density function $20\left(\exp\left(-(x-1)^2-(y-4)^2\right)+\exp\left(-(x-4)^2-(y-1)^2\right)\right)$. Figure a) illustrates the initial positions and the Voronoi partition, whereas figure b) illustrates the final positions. Figure c) shows the gradient ascent flow. Notice that sensors are now forced to move on the network.

due to the stringent constraint on the motion of sensors. The use of collapsed vs uncollapsed network also affects the results depending on the collapsing factor.

These examples also point out that sensors may get stuck early, due to the presence of many local maxima. Moreover, the local maximum found by the gradient ascent algorithm, is usually greatly related to the initial sensors' deployment. Therefore, as long as a global optimization problem is concerned, as it is the case for a static allocation, the initial deployment may be driven by the density function ϕ so as to put more sensors in preferential areas. This is a way to provide a good starting point for the optimization algorithm.

V. Conclusions and Future Works

This paper focused on the problem of optimally deploying sensors in an environment modeled as a network. The allocation of omnidirectional sensors, that can have limited sensing radius, has been considered. Two optimization problems have been formulated and solved. The first one involves some relaxations in the model of the network and in the location of sensors. We proposed a discrete-time gradient ascent algorithm capable of solving both the static and the dynamic deployment problem.

A main future research direction will consider the integration of classical Operative Research methods with the present gradient algorithm. The aim is to build an overall global optimization technique to solve 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.

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

[8] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Trans. Robot. and Automat.*, vol. 20, no. 2, pp. 243–255, 2004.

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

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

[11] P. Sharma, S. Salapaka, and C. Beck, "A deterministic annealing approach to combinatorial library design for drug discovery," in *Proc. American Control Conf.*, Portland, OR, USA, 8–10 June 2005, pp. 979–984.

[12] ——, "A scalable deterministic annealing algorithm for resource allocation problems," in *Proc. American Control Conf.*, Minneapolis, MN, USA, 14–16 June 2006, pp. 3092–3097.

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

[14] A. Ganguli, S. Susca, S. Martinez, F. Bullo, and J. Cortes, "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.

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

[16] 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.

[17] J. Cortes, S. Martinez, 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.

[18] 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.

[19] 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.

[20] 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.

[21] 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.

[22] 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.

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

[24] A. F. Filippov, *Differential Equations with Discontinuous Righthand Sides*, ser. Mathematics and its Applications. Dordrecht, The Nederlands: Kluwer Academics Publishers, 1988, vol. 18.