

Adaptive-based, Scalable Design for Autonomous Multi-Robot Surveillance

Alessandro Renzaglia, Lefteris Doitsidis, Agostino Martinelli, Elias Kosmatopoulos

▶ **To cite this version:**

Alessandro Renzaglia, Lefteris Doitsidis, Agostino Martinelli, Elias Kosmatopoulos. Adaptive-based, Scalable Design for Autonomous Multi-Robot Surveillance. 49th IEEE Conference on Decision and Control (CDC), Dec 2010, Atlanta, United States. inria-00579310

HAL Id: inria-00579310
https://inria.hal.science/inria-00579310

Submitted on 23 Mar 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Adaptive-based, Scalable Design for Autonomous Multi-Robot Surveillance

Alessandro Renzaglia, Lefteris Doitsidis, Agostino Martinelli and Elias B. Kosmatopoulos

***Abstract*— In this paper the problem of positioning a team of mobile robots for a surveillance task in a non-convex environment with obstacles is considered. The robots are equipped with global positioning capabilities (for instance they are equipped with GPS) and visual sensors able to monitor the surrounding environment. The goal is to maximize the area monitored by the team, by identifying the best configuration of the team members. Due to the non-convex nature of the problem, an analytical solution cannot be obtained. The proposed method is based on a new cognitive-based, adaptive optimization algorithm (CAO). This method allows getting coordinated and scalable controls to accomplish the task, even when the obstacles are unknown. Extensive simulations are presented to show the efficiency of the proposed approach.**

I. INTRODUCTION

The use of multi-robot teams has gained a lot of attention in recent years. This is due to the extended capabilities that the teams have to offer comparing to the use of a single robot for the same task. Robot teams can be used in a variety of mission including: surveillance in hostile environments (i.e. areas contaminated with biological, chemical or even nuclear wastes), environmental monitoring (i.e. air quality monitoring, forest monitoring) and law enforcement missions (i.e. border patrol), etc. In all the aforementioned tasks the deployment of limited resources (robots) to maximize the area monitored is the key issue. This can be achieved by optimizing the way that the robots are deployed so that the area monitored by each team member is maximized and at the same time the overlap of these areas is minimized.

As far as it concerns the optimal coverage using a team of robots, two problems have been identified and formally approached up to now, both by introducing a suitable optimization function. The first problem deals with the optimal arrangement of the team members, so that for every point in the area to be covered, the closest robot is as close as possible to that point. This corresponds to the exigency of having the possibility to intervene as fast as possible, in all the points of the area with at least one robot. In this case, the corresponding cost function which will be minimized depends on the distance of the robots from the points of the area to be covered.

A. Renzaglia and A. Martinelli are with INRIA Rhône-Alpes, Grenoble, France email: firstname.lastname@inrialpes.fr

L. Doitsidis is with the Department of Electronics, Technological Educational Institute of Crete, Chania, Greece ldoitsidis@chania.teicrete.gr

E.B. Kosmatopoulos is with the Department of ECE, Democritus University of Thrace, Xanthi, Greece & Informatics & Telematics Institute, CERTH, Thessaloniki, Greece kosmatop@dssl.tuc.gr

The second problem deals with the maximization of an area monitored by a team of robots using vision sensors. In this case, a point is considered monitored regardless of its distance from a robot, as long as it is in the field of view of at least one vision sensor. Furthermore, a typical assumption is that the robots are equipped with omnidirectional vision sensors.

In this paper we consider a case similar to the second problem, for a non-convex environment with obstacles, considering a threshold on both the maximum distance of a point to be monitored and a limited angle of visibility. The presence of these constraints dramatically changes the nature of the cost function and makes the problem analytically unsolvable even for a convex environment.

A. Previous contributions

Several approaches have been proposed in the literature considering the first problem described in the previous section. In [3], the authors present a solution for the optimal coverage with a team of mobile robots in a convex environment, i.e. without obstacles, based on the Voronoi partition. A similar approach, for a convex environment, is proposed in [11], where additionally the robots estimate a function indicating the relative importance of different areas in the environment, using information from the sensors. A case for a non-convex region without obstacles (i.e. in a simply connected space), is analyzed in [10]. In this work the Voronoi partition is obtained by using the geodesic distance instead of the Euclidean one taking into account the particular topology of the problem. All the aforementioned approached have limited value in real scenarios since they are based on strong assumptions and they can not deal with scenarios which consider one or more obstacles.

As far as it concerns the second problem described in the previous section, different solutions have been proposed in the literature. In [4] the authors propose a gradient-based algorithm for the case of a single robot case and they prove that the visible area is almost everywhere a locally Lipschitz function of the observer location. In [5], an approach for the multi-robot problem is presented based on the assumption that the environment is simply connected. The visibility problem is also related with the Art Gallery Problem where the goal is to find the optimum number of guards in a non-convex environment so that each point of the environment is visible by at least one guard [1], [12]. All the aforementioned solutions are based on the hypothesis that a given point can be monitored regardless of its distance from the vision sensor and of the robots' orientation.

B. Paper contribution

In this paper we propose a cooperative algorithm to maximize the monitored areas in a 2D non-convex environment, even if it is unknown[1], by using a team of mobile robots. In particular, we consider the second problem previously mentioned by also accounting a maximum distance in the region visible from a vision sensor and a limited visibility angle. To the best of our knowledge this problem has never been approached by following a coordinated control strategy. On the other hand, an incremental deployment algorithm for the omnidirectional case can be found in [6]. Our goal is to approach this problem by introducing a learning strategy able to provide a coordinated control algorithm for all the team members. In particular, the proposed approach is based on the Cognitive-based Adaptive Optimization (CAO) methodology. The CAO methodology, which was recently introduced in [8], [9], possesses the capability of being able to efficiently handle optimization problems for which an analytical form of the function to be optimized is unknown, but the function is available for measurement at each iteration of the algorithm employed to optimize it. As a result, it perfectly suits for multi-robot optimal coverage in non-convex environments, where the analytical form of the function to be optimized is unknown but the function is available for measurement (through the robots' sensors) for each multi-robot configuration. The CAO approach extends the popular Simultaneous Perturbation Stochastic Approximation (SPSA) algorithm [13]. The difference between the SPSA and the CAO approach is that SPSA employs an approximation of the gradient of an appropriate cost function using only the most recent experiments, while the CAO approach employs linear-in-the-parameters approximators that incorporate information of a user specified time window of the past experiments together with the concept of candidate perturbations for efficiently optimizing the unknown function. It has to be emphasized that for complicated optimization problems like the one treated in this paper, the SPSA algorithm may fail to produce efficient solutions contrary to the CAO approach which always achieves to efficiently and rapidly move the robots to locations that optimize the particular coverage criterion. It is mentioned that the CAO or the SPSA do not create an approximation or estimation of the obstacles location and geometry; instead, they on-line produce a **local approximation** of the – unknown – cost function the robots are called to optimize. For this reason, they require simple – and thus scalable – approximation schemes to be employed.

This paper is organized as follows. In the next section we describe the stochastic optimization approach used in this work, the CAO algorithm, and how it is applicable to a generic coverage-like problem. Then, in section III, we show explicitly the proposed solution based on the CAO algorithm applied to the particular problem we want to solve. Finally, in section IV, we present the results of the numerical simulations in order to evaluate the performance of the proposed method.

[1]In this case each robot has to be equipped also with range sensors in order to get metric information from the environment (as it has been done in the past when the environment is unknown [5], [6]).

II. The Cognitive-based Adaptive Optimization Approach

The Cognitive-based Adaptive Optimization (CAO) approach [7]-[9] was originally developed and analyzed for the optimization of functions for which an explicit form is unknown but their measurements are available as well as for the adaptive fine-tuning of large-scale nonlinear control systems. In this section, we will describe how the CAO approach can be appropriately adapted and extended so that it is applicable to the problem of multi-robot coverage. More explicitly, let us consider the problem where M robots are involved in a coverage task, attempting to maximize a given *coverage criterion*. Apparently, the coverage criterion is a function of the robots' positions or poses (positions and orientations), i.e.

$$J_k = \mathcal{J}\left(x_k^{(1)}, \ldots, x_k^{(M)}\right) \tag{1}$$

where $k = 0, 1, 2, \ldots$ denotes the time-index, J_k denotes the value of the coverage criterion at the k-th time-step, $x_k^{(1)}, \ldots, x_k^{(M)}$ denote the position/pose vectors of robots $1, \ldots, M$, respectively, and $\mathcal{J}$ is a nonlinear function which depends – apart from the robots' positions/poses – on the particular environment where the robots live; for instance, in the 2D case the function $\mathcal{J}$ depends on the location of the various obstacles that are present, while in the 3D case with flying robots monitoring a terrain, the function $\mathcal{J}$ depends on the particular terrain morphology.

Due to the dependence of the function $\mathcal{J}$ on the particular environment characteristics, the *explicit form of the function $\mathcal{J}$ is not known* in most practical situations; as a result, standard optimization algorithms (e.g. steepest descent) are not applicable to the problem in hand. However, in most practical cases – like the one treated in this paper – the current value of the coverage criterion can be estimated from the robots' sensor measurements. In other words, at each time-step k, an estimate of J_k is available through robots' sensor measurements,

$$J_k^n = \mathcal{J}\left(x_k^{(1)}, \ldots, x_k^{(M)}\right) + \xi_k \tag{2}$$

where J_k^n denotes the estimate of J_k and ξ_k denotes the noise introduced in the estimation of J_k due to the presence of noise in the robots' sensors. Please note that, although it is natural to assume that the noise sequence ξ_k is a stochastic *zero-mean* signal, it is not realistic to assume that it satisfies the typical Additive White Noise Gaussian (AWNG) property even if the robots' sensor noise is AWNG: as $\mathcal{J}$ is a nonlinear function of the robots' positions/poses (and thus of the robots' sensor measurements), the AWNG property is typically lost.

Apart from the problem of dealing with a criterion for which an explicit form is not known but only its noisy measurements are available at each time, efficient robot coverage algorithms have additionally to deal with the problem of

restricting the robots' positions so that obstacle avoidance as well as robot formation constraints are met. In other words, at each time-instant k, the vectors $x_k^{(i)}, i = 1, \ldots, M$ should satisfy a set of constraints which, in general, can be represented as follows:

$$\mathcal{C}\left(x_k^{(1)}, \ldots, x_k^{(M)}\right) \leq 0 \tag{3}$$

where $\mathcal{C}$ is a set of nonlinear functions of the robots' positions/poses. As in the case of $\mathcal{J}$, the function $\mathcal{C}$ depends on the particular environment characteristics (e.g. location of obstacles, terrain morphology) and an explicit form of this function may be not known in many practical situations; however, it is natural to assume that the coverage algorithm is provided with information whether a particular selection of robots' positions/poses satisfies or violates the set of constraints (3).

Given the mathematical description presented above, the multi-robot coverage problem can be mathematically described as the problem of moving $x_k^{(1)}, \ldots, x_k^{(M)}$ to a set of positions/poses that solves the following constrained optimization problem:

$$\begin{array}{ll} \text{maximize} & (1) \\ \text{subject to} & (3)\,. \end{array} \tag{4}$$

As already noticed, the difficulty in solving – in real-time and in real-life situations – the constrained optimization problem (4) lies in the fact that explicit forms for the functions $\mathcal{J}$ and $\mathcal{C}$ are not available. To circumvent this difficulty, the CAO approach is adopted – appropriately modified to be applicable to the problem in hand – which is capable of efficiently dealing with optimization problems for which the explicit forms of the objective function and constraints are not known, but noisy measurements/estimates of these functions are available at each time-step. Next we describe the CAO approach as applied to the multi-robot coverage problem described above. The proof of this theorem – not presented here for brevity purposes – is among the same lines as the main results of [8], [9]; the main difference between the proof of the theorem presented below and that of [8], [9] is that while in the case of [8], [9] it is established that the CAO algorithm used there is approximately a *gradient-descent* algorithm, the CAO algorithm used in this paper is proven to be approximately a *projected gradient-descent* algorithm.

As a first step, the CAO approach makes use of function approximators for the estimation of the unknown objective function $\mathcal{J}$ at each time-instant k according to

$$\hat{J}_k\left(x_k^{(1)}, \ldots, x_k^{(M)}\right) = \vartheta_k^\tau \phi\left(x_k^{(1)}, \ldots, x_k^{(M)}\right)\,. \tag{5}$$

Here $\hat{J}_k\left(x_k^{(1)}, \ldots, x_k^{(M)}\right)$ denotes the approximation/estimation of $\mathcal{J}$ generated at the k-th time-step, ϕ denotes the nonlinear vector of L *regressor terms*, ϑ_k denotes the vector of *parameter estimates* calculated at the k-th time-instant and L is a positive user-defined integer denoting the *size* of the function approximator (5). The parameter estimation vector ϑ_k is calculated according to

$$\vartheta_k = \underset{\vartheta}{\operatorname{argmin}} \frac{1}{2} \sum_{\ell=\ell_k}^{k-1} \left(J_\ell^n - \vartheta^\tau \phi\left(x_\ell^{(1)}, \ldots, x_\ell^{(M)}\right)\right)^2 \tag{6}$$

where $\ell_k = \max\{0, k-L-T_h\}$ with T_h being a user-defined nonnegative integer. Standard least-squares optimization algorithms can be used for the solution of (6).

Remark 1: In order for the proposed methodology to guarantee with efficient performance, special attention has to be paid in the selection of the regressor vector ϕ. Polynomial or polynomial-like regressor vectors as well as sigmoidal regressor vectors can be employed for the construction of ϕ. The particular choice adopted for the application treated in this paper is described in section III. See [7]-[9] for more details on the design considerations for the regressor vector. $\diamond$

As soon as the estimator $\hat{J}_k$ is constructed according to (5), (6), the set of new robots' positions/poses is selected as follows: firstly, a set of N *candidate robots' positions/poses* is constructed according to[2]

$$x_k^{i,j} = x_k^{(i)} + \alpha_k \zeta_k^{i,j}, i \in \{1, \ldots, M\}, j \in \{1, \ldots, N\}\,, \tag{7}$$

where $\zeta_k^{i,j}$ is a zero-mean, unity-variance random vector with dimension equal to the dimension of $x_k^{(i)}$ and α_k is a positive real sequence which satisfies the conditions:

$$\lim_{k\to\infty} \alpha_k = 0, \quad \sum_{k=1}^{\infty} \alpha_k = \infty, \quad \sum_{k=1}^{\infty} \alpha_k^2 < \infty\,. \tag{8}$$

Among all N candidate new positions $x_k^{1,j}, \ldots, x_k^{M,j}$, the ones that correspond to *non-feasible positions/poses* – i.e. the ones that violate the constraints (3) – *are neglected* and then the new robots' positions/poses are calculated as follows:

$$\left[x_{k+1}^{(1)}, \ldots, x_{k+1}^{(M)}\right] = \underset{\substack{j \in \{1, \ldots, N\} \\ x_k^{i,j} \text{not neglected}}}{\operatorname{argmax}} \hat{J}_k\left(x_k^{1,j}, \ldots, x_k^{M,j}\right)$$

The idea behind the above logic is simple: at each time-instant a set of many candidate new robots' positions/poses is generated and the candidate – among the ones that provide with a feasible solution – that provides the "best" estimated value $\hat{J}_k$ of the coverage criterion is selected as the new set of robots' positions/poses. The random choice for the candidates is essential and crucial for the efficiency of the algorithm, as such a choice guarantees that $\hat{J}_k$ is a reliable and accurate estimate for the unknown function $\mathcal{J}$; see [8], [9] for more details. On the other hand, the choice of a slowly decaying sequence α_k – a typical choice of adaptive gains in stochastic optimization algorithms, see e.g. [2] – is essential for "filtering out" the effects of the noise term ξ_k [cf. (2)]. The next theorem summarizes the properties of the CAO algorithm described above; it has to be emphasized that the proof of this theorem is among the same lines as the main results of [8], [9]:

[2]Here, N is a sufficiently large user-defined positive integer; typically it suffices to choose $N \in \{20, \ldots, 30\}$.

Theorem 1: Let $x^{(1^*)}, \ldots, x^{(M^*)}$ denote any – local – minimum of the constrained optimization problem (4). Assume also that the functions $\mathcal{J}, \mathcal{C}$ are either continuous or discontinuous with a finite number of discontinuities. Then, the CAO-based multi-robot coverage algorithm as described above guarantees that the robots' positions/poses $x_k^{(1)}, \ldots, x_k^{(M)}$ will converge to one of the local minima $x^{(1^*)}, \ldots, x^{(M^*)}$ with probability 1, provided that the size L of the regressor vector ϕ is larger than a lower bound $\bar{L}$.

Remark 2: Strictly speaking, Theorem 1 is valid as long as the zero-mean, unity variance vectors $\zeta_k^{i,j}$ satisfy some extra technical conditions (which are satisfied if e.g. $\zeta_k^{i,j}$ are Bernoulli random vectors). However, extensive simulation investigations have shown that – in practice – Theorem 1 is still valid even if the random vectors $\zeta_k^{i,j}$ are Gaussian random vectors, despite the fact that such a choice does not satisfy the aforementioned technical conditions. $\diamond$

Remark 3: As already noticed in section I, the CAO algorithm requires only a local approximation of the unknown function $\mathcal{J}$ and as a result the lower bound $\bar{L}$ has not to be large (as opposed to methods that construct a global approximation of the unknown function $\mathcal{J}$). Although, there exist no theoretical results for providing the lower bound $\bar{L}$ for the size of the regressor vector ϕ, practical investigations on many different problems (even in cases where the dimension of the variables to be optimized is as high as 500; see [7]-[9] for more details) indicate that for the choice of the regressor vectors according to Remark 1 such a bound is around 20. $\diamond$

Remark 4: As an alternative to the CAO approach, the SPSA approach [13] may be employed in multi-robot coverage applications. According to the SPSA approach, the robot positions/poses are updated according to

$$\begin{cases} x_{k+1}^{(i)} = x_k^{(i)} + \beta_k \zeta_k^i, & \text{if } k \text{ is even} \\ x_{k+1}^{(i)} = x_k^{(i)} + \gamma_k \frac{J_k^n - J_{k-1}^n}{\zeta_{k-1}^{(i)}}, & \text{if } k \text{ is odd} \end{cases} \tag{9}$$

where $\zeta_k^{(i)}$ are zero-mean, unity-variance random vectors and β_k, γ_k are slowly decaying sequences (similar as the sequence α_k). The SPSA algorithm is computationally simpler than the CAO one, but it does not perform as efficient as the CAO approach as have been demonstrated in a variety of approaches, see [7]-[9]. However, extensive simulation experiments have demonstrated that a hybird scheme which uses SPSA at the first 10-20 time-steps and then switches to the CAO algorithm can have significant improvements over schemes that employ only the CAO algorithm. This is due to the fact that CAO, at its initial steps, may preserve a poor performance because it takes some iterations for the CAO estimator (5) in order to come up with a reliable estimate $\hat{J}_k$ of the unknown coverage function $\mathcal{J}$. $\diamond$

Remark 5: We close this section by mentioning that similarly to the proposed approach global optimization methods such as simulated annealing and genetic algorithms do not require that the explicit form of the function $\mathcal{J}$ is known; Moreover, these methods can guarantee global convergence as opposed to the proposed approach which guarantees only a local one. However, simulated annealing, genetic algorithms and other similar global optimization methods require that a large amount of different combinations of robots' positions is being evaluated all over the robots' application area. Such a requirement renders these methods practically infeasible as a huge amount of time would have to be spent in order for the robots to visit many different locations all over their application area. $\diamond$

III. The proposed CAO-based solution

The objective of this work is to maximize the monitored area in a given region by using a team of mobile robots, without any assumption on the topology of the environment. We can mathematically define the problem in the following way. Let us consider a planar non-convex environment and let Ω be the region accessible by the robots. Let $\mathcal{P} = \{x_k^{(i)}\}_{i=1}^M$ denote the position/pose of the M robots at the time step k and $C = \{r_i, \phi_i\}_{i=1}^M$ the relative maximum distances of monitoring and the camera's visibility angle (when $\phi_i = 2\pi$ the robot i^{th} is equipped with an omnidirectional camera). In our approach, we consider the monitoring of a point $q \in \Omega$ a binary function

$$f(q, \mathcal{P}; C) = \begin{cases} 1 & \text{if } q \text{ is monitored} \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

Let us assume that a robot can monitor the points which satisfy all the following conditions:

- are connected by a line-of-sight with it;
- are at a distance smaller than a given threshold value;
- are within the visibility angle.

Thus, we can define the cost function $\mathcal{J}$ as follows:

$$\mathcal{J}(\mathcal{P}; C) = \frac{1}{V} \int_\Omega f(q, \mathcal{P}; C)\, dq \tag{11}$$

where $V = \int_\Omega dq$. Obviously, this is only an implicit expression of the cost function and it is impossible to get an explicit form because of the dependency on the particular environment. However, as explained in the previous section, we just need the numerical value of the cost function for each time step and not its explicit expression. This is the key advantage of CAO which does not require an a priori knowledge of the environment.

Once defined the optimization problem, a fundamental point for a good behavior of the algorithm is an appropriate choice of the form of the regressor vector ϕ, introduced in equation (5). As mentioned in remark 1, several different choices for its explicit expression are admissible. However, for the particular application treated in this paper, it was found that it suffices to choose the regressor vector as follows:

1) choose the size of the function approximator L to be an odd number;
2) select the first term of the regressor vector ϕ to be the constant term;
3) select randomly the next $(L-1)/2$ terms of ϕ to be any 2nd-order terms of the form $x_a^{(i)} \cdot x_b^{(j)}$

[with $a, b \in \{1, \ldots, \dim(x^{(i)})\}, i, j \in \{1, \ldots, M\}$ randomly-selected positive integers];

4) select the last $(L-1)/2$ terms of ϕ to be any 3rd-order terms of the form $x_a^{(i)} \cdot x_b^{(k)} \cdot x_c^{(j)}$ [with $a, b, c \in \{1, \ldots, \dim(x^{(i)})\}, i, k, j \in \{1, \ldots, M\}$ randomly-selected positive integers].

Once the regressor vector ϕ has been set and once the values of the cost function (11) are available for measurement at each time step, it is possible to find at each time step the vector of parameter estimates θ_k and thus the approximation of the cost function $\hat{J}_k$. The other important choice in order to assure the convergence of the algorithm is the expression of the sequence α_k, defined in equation (7). A typical choice for such a sequence is given by

$$\alpha_k = \frac{c}{(k+1)^\eta}, \tag{12}$$

where c is a positive user-defined constant and $\eta \in (0, 0.5)$. In our implementations we set $c = 0.2$ and $\eta = 0.15$.

Remark 6: Please note that the CAO algorithm's computational requirements are dominated by the requirement for solving the least-squares problem (6). As the number of free parameters in this optimization problem is L, most popular algorithms for solving least-squares problems have – in the worst case – $\mathcal{O}(L^3)$ complexity (polynomial complexity with respect to L). Please note that for a realistic situation where 3-5 robots are employed for optimal coverage, our simulation investigations indicate that a "good" value for L is around 20. ◇

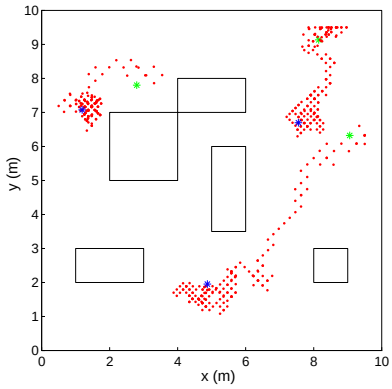
IV. Simulations

To evaluate the efficiency of the proposed algorithm, several simulations with varying number of robots and different monitoring constraints, have been performed in a variety of environments. The number of robots in the considered teams for the simulations here presented is varying between three and five. The teams are considered to be homogeneous since the maximum distance of monitoring for each robot is the same, although it is not the same in all simulated scenarios. This assumption has been made for simplification purposes and easier comprehension of the results.

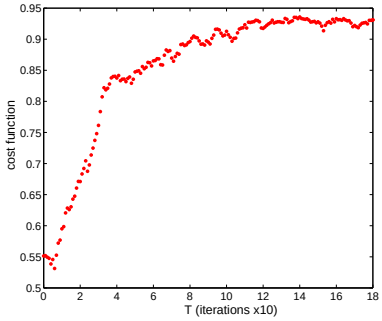
A. Omnidirectional monitoring

We start by considering the case where all the robots are equipped with omnidirectional cameras, i.e. the camera's visibility angle is 2π for all the robots. In this case the symmetry of the problem allows neglecting the robots' orientation, reducing the dimension of the problem.

In the first simulation presented in fig. 1, the team is composed by three robots with a maximum monitoring distance $r = 5m$. The cost function, in fig. 1(b), indicates that the algorithm is able to provide a very good solution. The efficiency of the proposed solution can also be evaluated by observing the robot trajectories in fig. 1(a). The robots move in order to eliminate all the shadow regions generated by the obstacles and to minimize the overlapping zones monitored by more than one sensor.



(a)



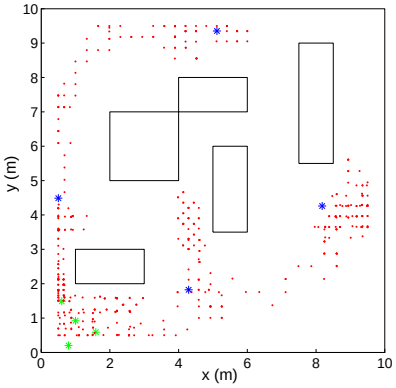
(b)

Fig. 1. Three robots with a maximum monitoring distance $r = 5m$ and omnidirectional vision sensors. In fig. (a) it is shown the robots' motion: the green points show the initial positions of the robots, the final ones are in blue, in red the trajectories. In fig. (b) the cost function $\mathcal{J}(\mathcal{P})$.

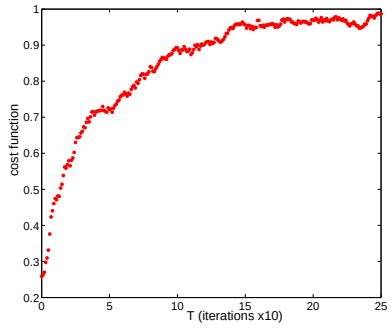
In the other simulations here presented (fig. 2, 3), it is possible to see that such spreading out can be obtained also when the robots are initially very close one to each other, which could be a more realistic case for the starting configuration of a multi-robot task. These simulations include a larger number of robots because the position and the number of the obstacles make the monitoring of the environment more complicated. As for the case illustrated in fig. 1, the proposed algorithm works very well. This is proved by the behaviour of the cost function and also by looking the robot trajectories (fig. 2(a) and 3(a)). We have performed several simulations by changing all the parameters characterizing the environment and the team. The results which were obtained are similar to the ones here presented.

B. Limited monitoring angle

Here we present the results for the more complicated case in which each robot has a limited visibility angle. In order to have a simple comparison with the omnidirectional vision case we have used the same environments of the previous simulations. Since now it is important not only the robots'

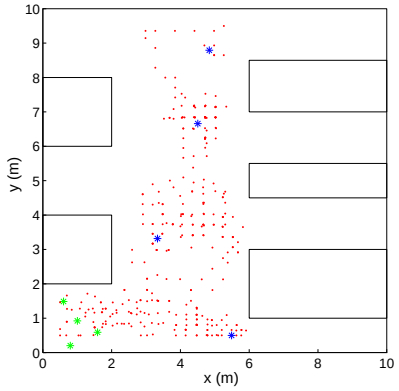


(a)

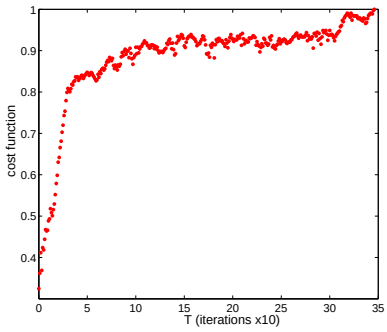


(b)

Fig. 2. Four robots with a maximum monitoring distance $r = 5m$.



(a)



(b)

Fig. 3. Four robots in with a maximum monitoring distance $r = 6m$.

positions but also their orientation, for clarity's sake we have shown explicitly in the figures the monitored area for each robots (fig. 4(a), 5(a)).

By observing the cost function (fig. 4(b), 5(b)) it is evident that in this case it is less smooth. Because of this the proposed method needs of more iterations to converge.

V. Conclusion and Future Work

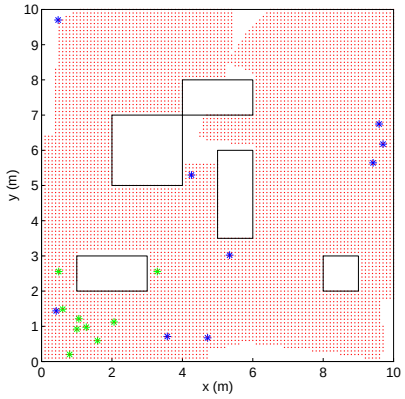
A new method for obtaining cooperative and scalable multi-robot controls for an optimal surveillance problem, in a 2D non-convex environment with unknown obstacles has been proposed. The goal is to maximize the area monitored by all the robots through their visual sensors. The optimization problem is solved by using a new stochastic method, the cognitive-based adaptive optimization algorithm.

The proposed approach has the following key advantages with respect to previous works:

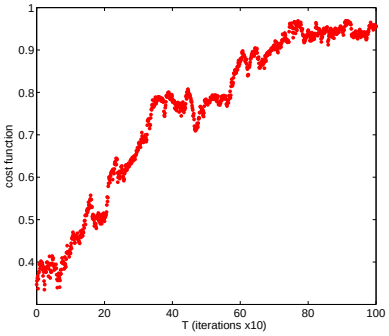
- it does not require any a priori knowledge on the environment;
- it works in any given environment, without the necessity to make any kind of assumption about its topology;
- it can incorporate any kind of constraints, for instance regarding a possible existing threshold on the maximum distance on the monitored region, or a limited visibility angle;
- it does not require a knowledge about these constraints since they are learnt during the task execution;
- its complexity is low allowing real time implementations.

The advantages of the proposed methodology make it suitable for real implementations and the results obtained through numerical simulations give us the motivation to adopt the CAO also in other frameworks. We are interested in extending the problem to the 3D case. Our aim is to develop a strategy for the surveillance of an unknown urban-like environment with a MAV swarm. For this case we will consider a cone of visibility for each robot, instead of an omnidirectional monitoring. Thus, apart from the position the orientation of each robot has to be considered.

Furthermore, we expect that many important tasks in mobile robotics can be approached by CAO-based algorithms: for example coordinated exploration, optimal target tracking, multi-robot localization, and so on. This is basically due to the fact that the CAO approach does not require an a priori knowledge of the environment and it has low complexity.

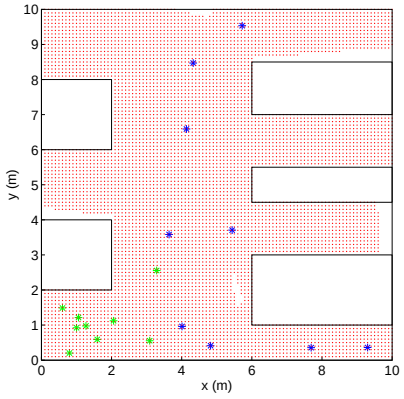


(a)

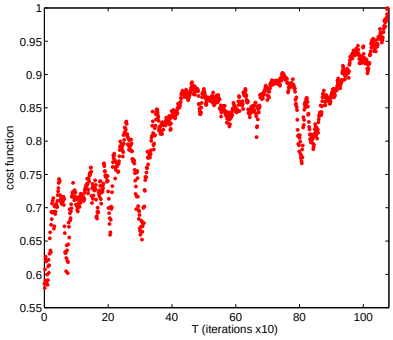


(b)

Fig. 4. Nine robots with a maximum monitoring distance $r = 5m$ and 120 degree of monitoring angle.



(a)



(b)

Fig. 5. Nine robots in with a maximum monitoring distance $r = 6m$ and 120 degree of monitoring angle.

Both these issues are fundamental in mobile robotics.

VI. ACKNOWLEDGMENTS

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n. 231855 (sFly).

REFERENCES

[1] P.K. Agarwal and M. Sharir, "Efficient algorithms for geometric optimization," *ACM Computing Surveys,* vol. 30, no.4, pp. 412-458, 1998.

[2] D. P. Bertsekas and J. N. Tsitsiklis, "Gradient convergence in gradient methods with errors," *SIAM Journal in Optimization,* vol. 10, no. 3, pp. 627-642, 2000.

[3] J. Cortés, S. Martínez, T. Karataş and F. Bullo, "Coverage Control for Mobile Sensing Networks," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, pp. 243-255, 2004.

[4] A. Ganguli, J. Cortés and F. Bullo, "Maximizing visibility in nonconvex polygons: nonsmooth analysis and gradient algorithm design," in *American Control Conference*, vol. 2, pp. 792-797, 2005.

[5] A. Ganguli, J. Cortés and F. Bullo, "Visibility-based multi-agent deployment in orthogonal environments," in *American Control Conference,* New York, USA, pp. 3426-3431, 2007.

[6] A. Howard, M.J. Matarić and G.S. Sukhatme, "An incremental deployment algorithm for mobile robot teams," in *IEEE International Conference on Robotics and Intelligent System (IROS),* Lausanne, Switzerland, pp. 2849-2854, 2002.

[7] E.B. Kosmatopoulos, Papageorgiou, M., Vakouli, A. Kouvelas, A., "Adaptive fine-tuning of nonlinear control systems with application to the urban traffic control strategy TUC," *IEEE Transactions on Control Systems Technology,* Vol. 15, no. 6, pp. 991-1002, 2007.

[8] E.B. Kosmatopoulos, "An adaptive optimization scheme with satisfactory transient performance," *Automatica,* Vol. 45, No. 3, pp. 716-723, 2009.

[9] E.B. Kosmatopoulos and A. Kouvelas, "Large-Scale Nonlinear Control System Fine-Tuning through Learning," *IEEE Transactions Neural Networks,* Vol. 20, No. 6, pp. 1009-1023, 2009.

[10] L. Pimenta, V. Kumar, R.C Mesquita and G. Pereira, "Sensing and coverage for a network of heterogeneous robots," in *47th IEEE Conference on Decision and Control,* Cancun, Mexico, pp. 3947-3952, 2008.

[11] M. Schwager, J. McLurkin and D. Rus, "Distributed Coverage Control with Sensory Feedback for Networked Robots," in *Proceedings of Robotics: Science and Systems,* Philadelphia, USA, 2006.

[12] T.C. Shermer, "Recent results in art galleries," in *IEEE Proceedings,* vol. 80, no.9, pp. 1384-1399, 1992.

[13] J.C. Spall, "Multivariate stochastic approximation using a simultaneous perturbation gradient approximation," *IEEE Trans. Autom. Control,* vol. 37, no. 3, pp. 332-341, Mar. 1992.