A Semi-Lagrangian Approach for the Minimal Exposure Path Problem in Wireless Sensor Networks^{*}

Armando Alves Neto^{a,*}, Víctor C. da Silva Campos^a, Douglas G. Macharet^b

^aDept. of Electronics Engineering, Univ. Federal de Minas Gerais, Belo Horizonte, Brazil. ^bComputer Vision and Robotics Laboratory (VeRLab), Dep. of Computer Science, Univ. Federal de Minas Gerais, Belo Horizonte, Brazil

Abstract

A critical metric of the coverage quality in Wireless Sensor Networks (WSNs) is the Minimal Exposure Path (MEP), a path through the environment that least exposes an intruder to the sensor detecting nodes. Many approaches have been proposed in the last decades to solve this optimization problem, ranging from classic (grid-based and Voronoi-based) planners to genetic meta-heuristics. However, most of them are limited to specific sensing models and obstacle-free spaces. Still, none of them guarantee an optimal solution, and the state-of-the-art is expensive in terms of run-time. Therefore, in this paper, we propose a novel method that models the MEP as an *Optimal Control* problem and solves it by using a *Semi-Lagrangian* approach. This framework is shown to converge to the optimal MEP while also incorporates different homogeneous and heterogeneous sensor models and geometric constraints (obstacles). Experiments show that our method dominates the state-of-the-art, improving the results by approximately 10 % with a relatively lower execution time.

Keywords: Wireless Sensor Network (WSN), Minimal Exposure Path (MEP), Policy Iteration, Dynamic Programming

^{*}This work has been financed by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001, Conselho Nacional de Desenvolvimento Científico e Tecnológico - Brasil (CNPq) - grant number 432241/2018-3, and the Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG) - grant number APQ-01515-18.

^{*}Corresponding author

Email addresses: aaneto@cpdee.ufmg.br (Armando Alves Neto), victor@cpdee.ufmg.br (Víctor C. da Silva Campos), doug@dcc.ufmg.br (Douglas G. Macharet)

1. Introduction

30

Wireless Sensor Networks (WSNs) are commonly used in a vast range of civilian and military applications and have been the focus of many studies. Such networks are constituted of multiple stationary wireless nodes with processor, memory, radio transceiver, power source, and a set of sensors used to collect data from the region where they have been deployed. In this context, a fundamental aspect is the coverage of the WSN, i.e., the monitoring quality of the network considering the dispersion of sensors and their properties. Most of the studies consider a full coverage of the area of interest, where every portion of the environment is within the sensing range of at least one sensor [1]. However, different sensing models can be found in the literature [2], and each one represents a target detection differently.

Detection of mobile targets using WSNs is a typical usage example of such systems, and it has been formulated as different problems in the literature, such as trap coverage [3], barrier coverage [4], and minimum exposure path [5, 6]. In the Minimal Exposure Path (MEP) problem, the goal is to determine a path through the sensing field that connects two arbitrary positions and minimizes the likelihood of a target being detected by the network during its movement. This is critical since the exposure can also be used as a quality metric of the

WSN, and the MEP represents the worst-case coverage performance. Moreover, this information allows to enhance the network during the design phase or to optimize and maintain it after deployment. The MEP problem has been the focus of many works in the literature and is generally tackled with grid-based approaches [5], methods based on the use of Voronoi diagrams [6], and heuristic solutions such as evolutionary algorithms [2].

In this paper, we propose a formulation based on optimal control theory for determining optimal control inputs for an intruder in the MEP problem. This approach uses the Dynamic Programming Principle to approximate the solution with a Semi-Lagrangian numerical scheme and policy iteration to accelerate the convergence. Fig. 1 illustrates a scenario filled with sensor nodes, over which we



Figure 1: MEP problem: find a path between a source (p_{init}) and destination (p_{goal}) that minimizes the exposure of a moving target to a set of sensor nodes. Our approach computes a value function representing an *exposure field* over the network, whose gradient approximates the optimal solution.

compute a value function capable of leading an agent from p_{init} to p_{goal} while minimizing its exposure to the network topology.

The methodology was evaluated considering recently available benchmarks, and it overcame the state-of-the-art in all instances. Specifically, we highlight the following contributions:

- we proposed a novel approach to solve the MEP problem using an optimal control framework that ensures convergence to the optimal solution – results are on average 10% better than the state-of-the-art.
- our formulation allows to incorporate the target's dynamics (which pro-
- 40
- duces smoother solutions), as well as many different sensing models and intensity functions for homogeneous and heterogeneous network topologies;
- it also allows to model geometric constraints, such as obstacles, holes, and other non-navigable areas;
- the method is faster than the best one in the literature, even when employed for a large number of nodes it is even more efficient when computing other paths in the same scenario (for the same p_{goal}).

The remainder of this paper is structured as follows: Sec. 2 reviews existing literature; Sec. 3 introduces preliminary concepts and formulates the problem;
Sec. 4 presents and details the proposed approach; Sec. 5 shows numerical results; and finally, in Sec. 6, we conclude and discuss avenues for future investigation.

2. Related work

75

Coverage is one of the most fundamental factors regarding WSNs, and it is related to problems such as deployment planning and network optimization. The coverage quality can be assessed, for example, by the exposure of an arbitrary path going through the sensing field, where a higher minimum exposure represents better coverage.

- A typical scenario for the employment of WSNs is related to the detection and tracking of mobile targets. Such a task is important, for example, in wildlife monitoring or, in the case of security applications, to detect possible intruders. Problems such as trap coverage [3, 7] and barrier coverage [4] are usually concerned about the detection of the target that reaches the sensing range of a node, and the path it takes inside the monitored area is generally not considered. On the other hand, the Minimal Exposure Path (MEP) problem allows for
- a broader understanding and representation of the strengths and weaknesses of the WSN under consideration, as it serves to better characterize the behavior of a target accordingly to its movement inside the field [5, 6]. It is usually formulated as a trajectory optimization problem, which considers the path exposure as the cost function to be minimized.

Analytical solutions to the MEP problem only exist in the trivial case where a single sensor node is considered [8]. There is no exact solution to the multiple sensor nodes case, and it has not yet been proven that it is solvable under such circumstance [9]. In this sense, different approximation solutions have been proposed to tackle this scenario [10].

Approaches tackling the MEP problem are generally separated into two cat-

egories, depending on the node sensing function used: i) *idealistic* models; and ii) *realistic probability* models. Idealistic models are more straightforward representations of the signal attenuation in WSNs, and consequently, the MEP can

- ⁸⁰ be solved by simpler algorithms. The first studies in this context used gridbased approaches and Voronoi diagrams [6, 11, 9]. The main problem with such methods is that, although easy to implement, they cannot be applied to scenarios where multiple sensors are used to detect the intruder (called the all-sensor intensity model). Also, when destination and source points do not ⁸⁵ lie in the Voronoi diagram, such approaches are unable to provide the optimal
 - solution [12].

105

Later, the results were improved by using hybrid evolutionary algorithms, and other biological-inspired solutions for fixed and mobile nodes [2, 13]. For example, in [14], a Physarum Optimization Algorithm (POA), that mimics the behavior of such organism, is used. The application of meta-heuristics provides accurate results since they usually do not rely on approximate cell/grid representations. To improve performance and efficiency, the authors of [15] also proposed a Hybrid Genetic Algorithm (HGA) based on particle swarm optimization.

⁹⁵ More realistic sensor models have been investigated in recent articles, such as directional nodes [16] and sensing probability models [17]. To tackle the drawbacks of grid-based and Voronoi-based approaches, [2] presents an HGA based on a Numerical Functional Extreme (NFE) model, transforming the MEP into a global optimization problem. Results overcame the state-of-the-art at that ¹⁰⁰ moment, however, similarly to other existing works, the method was based on a $m \ge m$ grid, whose resolution influences the output precision.

Currently, the state-of-the-art in the MEP problem is [12], which presents two approaches to solve it: (i) the GB-MEP, a grid-based method with some modifications; and (ii) the GA-MEP, based on a genetic algorithm with a featured individual representation and an effective combination of genetic opera-

tors. Since it is based on a meta-heuristic algorithm, the GA-MEP provides better precision performance than the GB-MEP between 77% to 88% of the cases,

depending on the nodes distribution (Gaussian distribution is only 40%). On the other hand, the GB-MEP is much faster, depending on the resolution used to compute $E(\cdot)$. Also, when compared with the genetic algorithm proposed in

[2], the GA-MEP dominates the *HGA-NFE*, winning in all tested instances.

110

Other aspects of the environment and the sensor field are also important to be treated in the MEP context. For example, in [18], an artificial-potential approach that considers the presence of static obstacles and evaluates the paths for multiple vehicles is proposed. Besides also considering cluttered environments, in [19], an adaptive cell decomposition approach capable of handling heterogeneous WSNs is proposed. Obstacles and heterogeneous topologies are typically more realistic and difficult to deal with than other scenarios.

More recently, the MEP formulation was combined with the Dubins Orienteering Problem (DOP), and a multi-objective evolutionary algorithm was proposed [20]. In this context, besides minimizing the exposure, the path is subjected to a bounded curvature and limited budget. Moreover, the goal is to visit a set of points that maximizes the total reward collected.

In this work, the MEP problem is considered in an optimal control setting, and a Dynamic Programming Approach is employed to solve it. Considering a continuous-time agent, it amounts to solving a partial differential equation, known as the Hamilton-Jacobi-Bellman equation, which can be solved numerically, among other possibilities, by Semi-Lagrangian approximation schemes [21]. To handle possibly infinite values for the exposure, representing unreach-

- able or forbidden regions on the search space (due to the presence of obstacles), we propose the use of a Kruzkov transformation [22, 21, 23], which maps values in $[0, \infty)$ to the [0, 1] range. A suitable Dynamic Programming Principle is presented for this transformed case, which is employed to propose a Semi-Lagrangian Approximation scheme to solve the optimal control problem
- ¹³⁵ by discretizing it in time and employing an unstructured grid in space. Our approach is shown to converge to the optimal solution of the problem utilizing the Barles-Souganidis Theorem [24, Theorem 2.1].

3. Problem formulation

In the MEP problem, the exposure function $E(\cdot)$ presents two main aspects. First, it depends on the model of the sensors composing the network, here referred to as the sensing function $S(\cdot)$. Second, it depends on how the combined energy of all sensors is computed at each location, defined as the *intensity func tion* $I(\cdot)$. Next, we discuss these aspects and formalize the Minimal Exposure Path as a problem of nonlinear optimization.

¹⁴⁵ 3.1. Node sensing functions

According to the literature, there are different sensing models used in WSNs, but all of them depend on the Euclidean distance between the sensor location s and the position p for which we must compute the detection. The most commonly used sensor functions are described below.

• Boolean disk coverage model [2],

$$S(\boldsymbol{s}, \boldsymbol{p}) = \begin{cases} 1, & \text{if } \|\boldsymbol{s} - \boldsymbol{p}\| \le r \\ 0, & \text{otherwise} \end{cases},$$
(1)

where r is the critical sensing range. As the name suggests, it emulates an *on-off* detection behavior, which might be used in very simple scenarios.

• Attenuated disk coverage model [2, 12],

$$S(\boldsymbol{s},\boldsymbol{p}) = \frac{\lambda}{\|\boldsymbol{s} - \boldsymbol{p}\|^{\mu}},\tag{2}$$

where λ and μ are positive parameters related to the propagation and attenuation of the node signal.

• Sensing probability model [2],

155

$$S(\boldsymbol{s},\boldsymbol{p}) = e^{-\alpha \|\boldsymbol{s}-\boldsymbol{p}\|^{\beta}},\tag{3}$$

where α and β also represent positive parameters of signal attenuation.

• Probability coverage model with noise [25, 12],

$$S(\boldsymbol{s},\boldsymbol{p}) = \frac{\lambda}{\|\boldsymbol{s}-\boldsymbol{p}\|^{\mu}} + \eta, \qquad (4)$$

where η represents an additive noise energy factor, generally modeled as a Gaussian distribution.

Generally speaking, functions (1) and (2) are known as *idealistic models*, while (3) and (4) are considered to be *realistic probability* representations of the real-world. In *homogeneous* networks, the same function and parameters are used to model all sensors, while in *heterogeneous* topologies, they are distinct.

165 3.2. Field intensity functions

A second aspect of the sensor field is the impact of all n sensors over the exposure function. Below we describe two main models found in the current literature.

• All-sensor field intensity function [12],

$$I(\boldsymbol{p}) = \sum_{i=1}^{n} S(\boldsymbol{s}_i, \boldsymbol{p}), \qquad (5)$$

170

175

160

where the influence of all sensors in the field is incorporated in the computation of the exposure, no matter how far they are from p.

• Maximum-sensor intensity function [2],

$$I(\boldsymbol{p}) = \max S(\boldsymbol{s}_i, \boldsymbol{p}), \qquad (6)$$

where only the influence of the node with the highest sensing function is used to compute the exposure. When the field is composed of homogeneous nodes, this sensor is the closest to p.

3.3. Minimal Exposure Path

Now we can compute the exposure $E(\cdot)$ of a path p(t), which is directly related to the WSN ability to detect mobile targets traversing its sensing field.

The exposure is defined as the time integral of the cumulative energy perceived

 $_{180}$ by the sensors in the network [5, 6], according to a given intensity function.

185

More formally:

$$E(\boldsymbol{p}(t)) = \int_0^{t_f} I(\boldsymbol{p}(t)) \left| \dot{\boldsymbol{p}}(t) \right| dt, \tag{7}$$

where t_f is the time the goal is reached. Here, $\boldsymbol{p}_{\text{init}} = \boldsymbol{p}(0)$ represents the source position, while $\boldsymbol{p}_{\text{goal}} = \boldsymbol{p}(t_f)$ is the destination position. In other words, a target moving from $\boldsymbol{p}_{\text{init}}$ to $\boldsymbol{p}_{\text{goal}}$ will be exposed to the node set according to (7). Finally, the MEP problem can be defined as finding a path throughout the environment that less exposes the target. Formally:

Problem 1 (Minimal Exposure Path). Let $\mathcal{N} = \{\mathbf{n}_i\}_{i=1}^n$ be a sensor field formed by a set of n nodes distributed in a \mathbb{R}^2 (cluttered) environment, each node represented by a sensing model $S(\cdot) : \mathbb{R}^2 \to \mathbb{R}$. In this context, the main goal is to compute a penetration path \mathbf{p}^* through \mathcal{N} , leading from the source position \mathbf{p}_{init} to the destination \mathbf{p}_{goal} , that minimizes the exposure (7) of the target, i.e.,

$$\boldsymbol{p}^*(t) = \mathop{\arg\min}_{\boldsymbol{p}(t)} E(\boldsymbol{p}(t)) \,.$$

4. Methodology

4.1. Numerical approximation scheme

In this paper, we tackle the MEP problem in an optimal control setting. First, let us consider a path generated by the dynamic system:

$$\dot{\boldsymbol{p}}(t) = \boldsymbol{u}(t), \text{ with } \boldsymbol{p}(0) = \boldsymbol{p}_{\text{init}},$$
(8)

where p(t) is the target position and u(t) the velocity vector over time. We consider that the set of admissible control inputs (velocities) is given by:

$$\mathcal{U} = \{ \boldsymbol{u} \in \mathbb{R}^m \mid 0 < \boldsymbol{u}_{\min} \leq \| \boldsymbol{u}(t) \| \leq \boldsymbol{u}_{\max} \ \forall t \}$$

With these definitions, exposure (7) can be rewritten as:

$$E(\boldsymbol{p}(t), \boldsymbol{u}(t)) = \int_0^{t_f} I(\boldsymbol{p}(t)) \| \boldsymbol{u}(t) \| dt.$$

Now, we can also define the value function $V(\cdot) : \mathbb{R}^2 \to \mathbb{R}$ such that, for every point p(t) in the space, determines the minimum exposure from p(t) to p_{goal} :

$$V(\boldsymbol{p}(t)) = \inf_{\boldsymbol{u} \in \mathcal{U}} E(\boldsymbol{p}(t), \boldsymbol{u}(t)).$$
(9)

This value function admits the Dynamic Programming Principle (Optimality Principle) since any point along the path should be optimal. Then, it can be written as:

$$V(\boldsymbol{p}(t)) = \inf_{\boldsymbol{u} \in \mathcal{U}} \left(V(\boldsymbol{p}_u(\Delta t)) + \int_0^{\Delta t} I(\boldsymbol{p}(t)) \|\boldsymbol{u}(t)\| dt \right),$$
(10)

with $p_u(\Delta t)$ representing the point at time Δt along the path, taken when considering velocities defined by u(t). In the presented form, the value function can be approximated by employing a Semi-Lagrangian approach [21].

190

To deal with constraints along the path, such as obstacles and restricted zones, we consider that the value function must be infinite in those locations. In addition, assuming that the exposure is always non-negative, the value of the goal location must always be null. When considered together, these constraints lead to the boundary conditions:

$$V(\boldsymbol{p}(t)) = \begin{cases} 0, & \text{for } \boldsymbol{p}(t) = \boldsymbol{p}_{\text{goal}} \\ \infty, & \text{for } \boldsymbol{p}(t) \in \partial \mathcal{O} \end{cases},$$
(11)

with $\partial \mathcal{O}$ representing the boundaries of geometric forbidden regions. To better tackle these infinite values, we employ a rescaling of the value function (known as Kruzkov transformation [21]), such that:

$$\overline{V}(\boldsymbol{p}(t)) = 1 - e^{-V(\boldsymbol{p}(t))}.$$
(12)

By making use of the original Dynamic Programming Principle of the value function, and concerning:

$$E_{\Delta t, \boldsymbol{u}} = \int_0^{\Delta t} I(\boldsymbol{p}(t)) \, \|\boldsymbol{u}(t)\| dt$$

in this transformed setting, we have:

$$\overline{V}(\boldsymbol{p}(t)) = 1 - e^{-\inf_{\boldsymbol{u} \in \mathcal{U}} (V(\boldsymbol{p}_u(\Delta t)) + E_{\Delta t, \boldsymbol{u}})},$$

$$= 1 - \sup_{\boldsymbol{u} \in \mathcal{U}} \left(e^{-V(\boldsymbol{p}_u(\Delta t))} e^{-E_{\Delta t, \boldsymbol{u}}} \right),$$

$$= 1 - \sup_{\boldsymbol{u} \in \mathcal{U}} \left(\left(-1 + e^{-V(\boldsymbol{p}_u(\Delta t))} \right) e^{-E_{\Delta t, \boldsymbol{u}}} + e^{-E_{\Delta t, \boldsymbol{u}}} \right),$$

$$\overline{V}(\boldsymbol{p}(t)) = 1 + \inf_{\boldsymbol{u} \in \mathcal{U}} \left(\overline{V}(\boldsymbol{p}_u(\Delta t)) - 1 \right) e^{-\int_{0}^{\Delta t} I(\boldsymbol{p}(t)) \|\boldsymbol{u}(t)\| dt}.$$
(13)

Eq. (13) can be seen as the Dynamic Programming Principle for this transformed $\overline{V}(\cdot)$, and we can also employ a Semi-Lagrangian numerical scheme to approximate the solution, by employing a time discretization, followed by a space discretization.

We consider the time discretization of (13) by applying a trapezoidal approximation for the integral term, and a trapezoidal method to solve the system of equations composed of (8). By considering a time step of Δt , this leads to:

$$\overline{V}_{k}(\boldsymbol{p}_{k}) = 1 + \inf_{\boldsymbol{u}_{k} \in \mathcal{U}} \left(\overline{V}_{k+1}(\boldsymbol{p}_{k+1}) - 1 \right) e^{-g_{k}(\boldsymbol{p}_{k}, \boldsymbol{u}_{k})},$$
(14)

with

$$g_k(\boldsymbol{p}_k, \boldsymbol{u}_k) = \frac{1}{2} \Big(I(\boldsymbol{p}_k) + I(\boldsymbol{p}_{k+1}) \Big) \| \boldsymbol{u}_k \| \Delta t,$$
(15)

and

$$\boldsymbol{p}_{k+1} = \boldsymbol{p}_k + \boldsymbol{u}_k \Delta t. \tag{16}$$

Afterwards, we perform the space discretization of $\overline{V}(\cdot)$, by considering an ²⁰⁵ unstructured grid of points covering the space. Since these points will be used to represent $\overline{V}(\cdot)$, they are the only points over the space for which the value is updated. Since $\overline{V}_{k+1}(\boldsymbol{p}_{k+1})$ might not be a part of the grid, it is replaced by a finite element linear interpolation over the grid. In this work, we have employed the Delaunay triangulation on the unstructured grid points to find a triangulation of the space (see Fig.2). We consider these triangles as our finite elements, and represent the interpolation of $\overline{V}_{k+1}(\boldsymbol{p}_{k+1})$ as $\mathcal{I}_{\overline{V}_{k+1}}[\boldsymbol{p}_{k+1}]$. Taken together, both discretizations (time and space) lead to a Semi-Lagrangian approximation scheme of (13), in the form:

$$\overline{V}_{k}(\boldsymbol{p}_{k}) = 1 + \inf_{\boldsymbol{u}_{k} \in \mathcal{U}} \left(\mathcal{I}_{\overline{V}_{k+1}}[\boldsymbol{p}_{k+1}] - 1 \right) e^{-g_{k}(\boldsymbol{p}_{k}, \boldsymbol{u}_{k})},$$
(17)

with boundary conditions

$$\overline{V}_{k}(\boldsymbol{p}_{k}) = \begin{cases} 0, & \text{if } \boldsymbol{p}_{k} = \boldsymbol{p}_{\text{goal}} \\ 1, & \text{if } \boldsymbol{p}_{k} \in \partial \mathcal{O} \end{cases}$$
(18)

Eq. (17) can be directly employed in a backwards in time-marching scheme, known as *value iteration*, to find an approximate solution to (13). Since, in the way the problem has been presented, we are interested in stationary/infinite horizon solutions, we have employed an acceleration technique known as *policy iteration* [21, Section 8.4.7]. In this technique, we alternate between finding an optimal policy u_k and an optimal value $\overline{V}(\cdot)$. At every grid point, the optimal policy is:

$$\boldsymbol{u}_{k} = \operatorname*{arg\,min}_{\boldsymbol{u}_{k} \in \mathcal{U}} \left(\mathcal{I}_{\overline{V}}[\boldsymbol{p}_{k+1}] - 1 \right) e^{-g_{k}(\boldsymbol{p}_{k}, \boldsymbol{u}_{k})}, \tag{19}$$

and fixed for this iteration. Afterwards, the value function is updated according to (18) and

$$\overline{V}(\boldsymbol{p}_k) - \mathcal{I}_{\overline{V}}[\boldsymbol{p}_{k+1}]e^{-g_k(\boldsymbol{p}_k, \boldsymbol{u}_k)} = 1 - e^{-g_k(\boldsymbol{p}_k, \boldsymbol{u}_k)}.$$

These steps are repeated until the algorithm converges to the minimum of the value function. The Minimal Exposure Path is then found by integrating the system trajectory (16), starting at $p_0 = p_{\text{init}}$ and employing the optimal policy given by (19), until the path reaches p_{goal} . The original value function in (9) can be recovered by:

$$V(\boldsymbol{p}) = -\ln\left(1 - \overline{V}(\boldsymbol{p})
ight).$$

215 4.2. Convergence analysis

The Dynamic Programming Principle of the transformed value function in (13) can be recast as a partial differential equation, known as the HJB equation, of the form:

$$\sup_{\boldsymbol{u}\in\mathcal{U}}\left(\overline{V}(\boldsymbol{p})\,\ell(\boldsymbol{p},\boldsymbol{u})-\nabla\overline{V}(\boldsymbol{p})\cdot f(\boldsymbol{p},\boldsymbol{u})-\ell(\boldsymbol{p},\boldsymbol{u})\right)=0\tag{20}$$

with $\ell(\boldsymbol{p}, \boldsymbol{u})$ being the cost, and $f(\boldsymbol{p}, \boldsymbol{u})$ defining the system dynamics, given by:

$$\ell(\boldsymbol{p}, \boldsymbol{u}) = I(\boldsymbol{p}) \|\boldsymbol{u}\|, \tag{21}$$
$$f(\boldsymbol{p}, \boldsymbol{u}) = \boldsymbol{u}.$$

Having defined this partial differential representation, we are now ready to state the main result concerning the convergence of our proposed methodology.

Theorem 1. Let our optimal control problem be represented by the HJB equation (20). As long as $\ell(\mathbf{p}, \mathbf{u})$ and $f(\mathbf{p}, \mathbf{u})$ are Lipschitz continuous in \mathbf{p} , and $\ell(\mathbf{p}, \mathbf{u}) > 0 \quad \forall \mathbf{p}, \mathbf{u}$, there is a unique viscosity solution to (20), corresponding to the optimal solution to the MEP problem. In addition, the proposed numerical

solution scheme converges to this unique viscosity solution as the time step, Δt , and the maximum distance between points on the grid, Δp , tend to zero, so long as Δp tends faster than Δt .

Proof. Note that $\ell(\boldsymbol{p}, \boldsymbol{u})$ and $f(\boldsymbol{p}, \boldsymbol{u})$ being Lipschitz continuous and $\ell(\boldsymbol{p}, \boldsymbol{u}) > 0$ $\forall \boldsymbol{p}, \boldsymbol{u}$, are sufficient conditions to ensure that there is a unique viscosity

solution to (20), and that the problem admits a comparison principle [22]. In this case, from the Barles-Souganidis Theorem [24, Theorem 2.1], as long as our numerical approximation scheme is *monotone*, a *contraction mapping*, and *consistent*, it converges to the unique viscosity solution of the HJB equation [21]. As such, the remainder of this proof shows these three properties of the approximation scheme in (17).

4.2.1. Monotonicity

Consider two functions \overline{W} and \overline{V} , with $\overline{W} \leq \overline{V}$ for every point on the grid. Suppose that the inf operator in (17) is attained by $\overline{\mathbf{w}}$ for \overline{W} , and \overline{u} for \overline{V} . It follows that:

$$\begin{split} \overline{W}_k(\boldsymbol{p}_k) &\leq 1 + \left(\mathcal{I}_{\overline{W}}[\boldsymbol{p}_{k+1}] - 1\right) e^{-g_k(\boldsymbol{p}_k, \overline{\boldsymbol{u}})},\\ \overline{V}_k(\boldsymbol{p}_k) - \overline{W}_k(\boldsymbol{p}_k) &\geq e^{-g_k(\boldsymbol{p}_k, \overline{\boldsymbol{u}})} \left(\mathcal{I}_{\overline{V}}[\boldsymbol{p}_{k+1}] - \mathcal{I}_{\overline{W}}[\boldsymbol{p}_{k+1}]\right), \end{split}$$

which implies that $\overline{V}_k(\boldsymbol{p}_k) - \overline{W}_k(\boldsymbol{p}_k) \ge 0$ since we employed a linear interpolation.

4.2.2. Contractiveness

Considering two functions \overline{W} and \overline{V} , with $\overline{\mathbf{w}}$ minimizing \overline{W} . It follows that:

$$\begin{split} \overline{V}_{k}(\boldsymbol{p}_{k}) &- \overline{W}_{k}(\boldsymbol{p}_{k}) \leq e^{-g_{k}(\boldsymbol{p}_{k},\overline{\mathbf{w}})} \left(\mathcal{I}_{\overline{V}}[\boldsymbol{p}_{k+1}] - \mathcal{I}_{\overline{W}}[\boldsymbol{p}_{k+1}] \right), \\ \overline{V}_{k}(\boldsymbol{p}_{k}) &- \overline{W}_{k}(\boldsymbol{p}_{k}) \leq e^{-g_{k}(\boldsymbol{p}_{k},\overline{\mathbf{w}})} \|\overline{V} - \overline{W}\|_{\infty}^{(k+1)}, \end{split}$$

with $\|\overline{V} - \overline{W}\|_{\infty}^{(k+1)}$ being the maximum of the error between the two functions in the next time step. Since we are assuming that $\ell(\boldsymbol{p}, \boldsymbol{u}) > 0$, then $g_k(\boldsymbol{p}_k, \boldsymbol{u}_k) > \overline{g}\Delta t > 0 \quad \forall \boldsymbol{p}_k, \boldsymbol{u}_k$. Since a similar bound can be found for $\overline{W}_k(\boldsymbol{p}_k) - \overline{V}_k(\boldsymbol{p}_k)$, then:

$$\|\overline{V} - \overline{W}\|_{\infty}^{(k)} \le e^{-\overline{g}\Delta t} \|\overline{V} - \overline{W}\|_{\infty}^{(k+1)}.$$

As we solve the problem back in time, this shows that our approximation scheme is a contraction mapping, with contraction rate $e^{-\overline{g}\Delta t}$. From the Banach fixedpoint Theorem, it guarantees that our approximation scheme converges to a unique solution.

4.2.3. Consistency

We start our consistency analysis by considering the error of time discretization, comparing solutions from (13) and (14). If we consider that the inf operator is attained by \overline{u} in (14), it follows that:

$$\overline{V}(\boldsymbol{p}(t)) - \overline{V}_k(\boldsymbol{p}_k) \le \left(\overline{V}(\boldsymbol{p}_{\overline{\boldsymbol{u}}}) - 1\right) e^{-\int_0^{\Delta t} I(\boldsymbol{p}(t)) \|\overline{\boldsymbol{u}}(t)\| dt} - \left(\overline{V}_{k+1}(\boldsymbol{p}_{k+1}) - 1\right) e^{-g_k(\boldsymbol{p}_k, \overline{\boldsymbol{u}})}$$

Some algebraic manipulations lead to:

$$\overline{V}(\boldsymbol{p}(t)) - \overline{V}_{k}(\boldsymbol{p}_{k}) \leq \left(\overline{V}(\boldsymbol{p}_{\overline{\boldsymbol{u}}}) - 1\right) \left(e^{-\int_{0}^{\Delta t} I(\boldsymbol{p}(t)) \|\boldsymbol{u}(t)\| dt} - e^{-g_{k}(\boldsymbol{p}_{k},\overline{\boldsymbol{u}})}\right) \\ + e^{-g_{k}(\boldsymbol{p}_{k},\overline{\boldsymbol{u}})} \left(\overline{V}(\boldsymbol{p}_{k+1}) - \overline{V}_{k+1}(\boldsymbol{p}_{k+1})\right) \\ + e^{-g_{k}(\boldsymbol{p}_{k},\overline{\boldsymbol{u}})} \left(\overline{V}(\boldsymbol{p}_{\overline{\boldsymbol{u}}}) - \overline{V}(\boldsymbol{p}_{k+1})\right).$$

From the dynamics considered in (8), $p_{\overline{u}} = p_{k+1}$, and by making use of the Mean Value Theorem, $g_k(p_k, u_k) > \overline{g}\Delta t$, and the fact that [26, Corollary 1.4], for a Lipschitz function $\ell(t)$ we have that the error of a trapezoidal integration

is bounded by $\frac{\Delta t^2}{8} \left(\sup \dot{\ell} - \inf \dot{\ell} \right)$. It follows that:

$$\overline{V}(\boldsymbol{p}(t)) - \overline{V}_k(\boldsymbol{p}_k) \le C_1 \Delta t^2 + e^{-\overline{g}\Delta t} \|\overline{V} - \overline{V}_k\|_{\infty},$$
(22)

for some constant C_1 . If we consider that the inf operator is attained by u^* in (13), and that:

$$\hat{\boldsymbol{u}}_k = \frac{1}{\Delta t} \int_0^{\Delta t} \boldsymbol{u}^*(\tau) d\tau$$

is the control obtained by the mean of the optimal control over a time step, it follows that

$$\begin{split} \overline{V}_k(\boldsymbol{p}_k) - \overline{V}(\boldsymbol{p}(t)) &\leq \left(\overline{V}_{k+1}(\boldsymbol{p}_{k+1}) - 1\right) e^{-g_k(\boldsymbol{p}_k, \hat{\boldsymbol{u}}_k)} \\ &- \left(\overline{V}(\boldsymbol{p}_{\boldsymbol{u}^*}) - 1\right) e^{-\int_0^{\Delta t} I(\boldsymbol{p}(t)) \|\boldsymbol{u}^*(t)\| dt} \end{split}$$

Since, for the particular case of the dynamics considered in (8), $p_{u^*} = p(t) + \hat{u}_k \Delta t$, following similar arguments to the ones used in obtaining (22), we have:

$$\overline{V}_k(\boldsymbol{p}_k) - \overline{V}(\boldsymbol{p}(t)) \le C_2 \Delta t^2 + e^{-\overline{g}\Delta t} \|\overline{V} - \overline{V}_k\|_{\infty},$$

with C_2 some constant term. This implies that:

$$(1 - e^{-\overline{g}\Delta t}) \|\overline{V} - \overline{V}_k\|_{\infty} \le C\Delta t^2,$$
$$\|\overline{V} - \overline{V}_k\|_{\infty} \le C\Delta t,$$
(23)

²⁴⁵ with $C = \max(C_1, C_2)$.

For the space discretization error, we analyze the errors of the value function on the grid points, by comparing (14) and (17). To differentiate them, we will denote the value on the grid points of (17) by \overline{v}_k . If we consider that the inf is attained by \overline{u} in (14), it follows that:

$$\overline{v}_k(\boldsymbol{p}_k) - \overline{V}_k(\boldsymbol{p}_k) \le \left(\mathcal{I}_{\overline{v}_{k+1}}[\boldsymbol{p}_{k+1}] - \overline{V}_{k+1}(\boldsymbol{p}_{k+1})\right) e^{-g_k(\boldsymbol{p}_k, \overline{\boldsymbol{u}})},$$

which, by considering that $g_k(\boldsymbol{p}_k, \overline{\boldsymbol{u}}) > \overline{g}\Delta t$, leads to:

$$\overline{v}_{k}(\boldsymbol{p}_{k}) - \overline{V}_{k}(\boldsymbol{p}_{k}) \leq e^{-\overline{g}\Delta t} \left| \mathcal{I}_{\overline{v}_{k+1}}[\boldsymbol{p}_{k+1}] - \mathcal{I}_{\overline{V}_{k+1}}[\boldsymbol{p}_{k+1}] \right| + e^{-\overline{g}\Delta t} \left| \mathcal{I}_{\overline{V}_{k+1}}[\boldsymbol{p}_{k+1}] - \overline{V}_{k+1}(\boldsymbol{p}_{k+1}) \right|.$$
(24)

Since \overline{V} is Lipschitz, then:

255

$$\left|\mathcal{I}_{\overline{V}_{k+1}}[\boldsymbol{p}_{k+1}] - \overline{V}_{k+1}(\boldsymbol{p}_{k+1})\right| \leq C_3 \Delta \boldsymbol{p},$$

with C_3 being a constant and Δp the largest distance between any point and a grid point. Combining this inequality with (24), we have that:

$$\overline{v}_k(\boldsymbol{p}_k) - \overline{V}_k(\boldsymbol{p}_k) \leq e^{-\overline{g}\Delta t} \|\overline{v} - \overline{V}_k\|_{\infty} + e^{-\overline{g}\Delta t} C_3 \Delta \boldsymbol{p}.$$

Since a similar bound can be found for $\overline{V}_k(\boldsymbol{p}_k) - \overline{v}_k(\boldsymbol{p}_k)$, then:

$$\begin{aligned} \|\overline{v} - \overline{V}_k\|_{\infty} &\leq e^{-\overline{g}\Delta t} \|\overline{v} - \overline{V}_k\|_{\infty} + C_4 \Delta \boldsymbol{p}, \\ \|\overline{v} - \overline{V}_k\|_{\infty} &\leq C_4 \frac{\Delta \boldsymbol{p}}{\Delta t} \end{aligned}$$
(25)

with C_4 a constant. Combining (23) and (25), we have that:

$$\|\overline{V} - \overline{v}\|_{\infty} \le C_5 \Delta t + C_6 \frac{\Delta p}{\Delta t},\tag{26}$$

with C_5 and C_6 being constants. As suggested in [21], the best coupling between Δt and Δp , in this case, is given by $\Delta p = \Delta t^2$, indicating that our grid resolution should be finer than our time discretization resolution.

Since we have shown that our scheme is monotone, a contraction mapping, and consistent, from the Barles-Souganidis Theorem, we have proven its convergence.

Remark 1. Although our consistency analysis considered the specific case of the system dynamics given by (8), similar bounds can be found for other dynamics (as long as $f(\mathbf{p}, \mathbf{u})$ is Lipschitz), by considering the numerical integration error of solving the ordinary differential equation by the method being used).

Remark 2. Note that our convergence results demand that $\ell(\mathbf{p}, \mathbf{u})$ in (21) is Lipschitz continuous, which demands that the Boolean disk coverage model (1) be approximated by a Lipschitz continuous function and that a maximum value be allowed for the Attenuated disk coverage model (2) and for the Probability

coverage model with noise in (4). In addition to this, the fact $\ell(\mathbf{p}, \mathbf{u}) > 0$, $\forall \mathbf{p}, \mathbf{u}$ requires that the intensity of the sensor field does not vanish at any point, which is essential to ensure that the problem admits a unique optimal solution.



Figure 2: Spatial grid discretization: points are more concentrated around nodes (black dots) and in the obstacles boundaries (black regions).

4.3. Implementation details

As previously discussed, the approach is composed of a time and a space ²⁶⁵ discretization, and our results converge to the optimal solution as these discretization errors decrease. Concerning the time discretization, if the time-step Δt is too small, it could slow down the convergence of the iterations when *value iteration* is used. This problem is somewhat mitigated when employing the *policy iteration*, but a suitable time-step must still be chosen. For the experiments ²⁷⁰ in the next section, we have set $\Delta t = 0.1$.

Although the proposed approach could be employed with a regular structured grid on the space, using rectangles as finite elements for the interpolation, unstructured grids allow for a better representation of the nonlinear profile of the node sensing functions and the transformed value function in (12). It also

allows representing obstacles with arbitrary geometries. In that regard, we have used a simple heuristic to sample grid points, concentrating them near the nodes and obstacle boundaries, as shown in Fig. 2. Basically, the higher the exposure value at p, the higher is the chance of it receiving a grid point. For each sensor node, we have used about 100 grid points.

To solve the minimization problem in (19), we perform an exhaustive search within a discrete set of allowable velocities. Although this is not the most efficient approach, it avoids local minima.

If values of (9) are too high, usually caused by having some sensing node too

close to the goal, there may be numerical problems working with the Kruzkov transformed value function in (12), as the values might get too close to 1. To mitigate this problem, we might consider the rescaling of the intensity functions (5) and (6), by dividing it by ω , leading to:

$$\overline{I}(\boldsymbol{p}) = \frac{I(\boldsymbol{p})}{\omega}, \qquad \qquad \omega \ge 1, \qquad (27)$$

employed in (15) instead of the original intensity function. For the experiments in the next section, we have set $\omega = 100$.

285

290

Finally, even though our conditions converge to the optimal solution, in practice, we have a sub-optimal path, and a local optimization procedure is performed over the path found by the policy in (19). This procedure checks along the path, described by a series of points in discrete-time, if each point could be substituted by a better point close to it, and replaces it if that is the case.

5. Experiments

In this section, we first show an illustrative example to discuss different aspects of our approach. Next, we compare our results with the state-of-the-art literature considering recently available benchmarks. Finally, we provide results with cluttered environments and heterogeneous sensor nodes.

5.1. Illustrative example

We begin by employing our method to the scenario presented in [2, Section VII.A]. It consists of a network topology with 32 nodes, modeled as attenuated disk coverage functions (2) with $\lambda = 4$ and $\mu = 2$. They have also adopted the maximum-sensor intensity function (6). Considering a 10 m x 10 m space, destination point was set to $p_{\text{goal}} = [10, 6.5]$, while source position was initially set to $p_{\text{init}} = [0, 4]$. Subsequently, we have also determined paths for $p_{\text{init}} = [0, 8]$ and $p_{\text{init}} = [5, 0]$. Fig. 3 presents the value function (gray-map background) and the MEPs computed by our approach when applied to the aforementioned scenario,



Figure 3: Example scenario (presented in [2]). The black dots are the nodes locations, while the color-map represents $V(\mathbf{p})$. We computed three different MEPs to $\mathbf{p}_{\text{goal}} = [10, 6.5]$, the first one starting from $\mathbf{p}_{\text{init}} = [0, 4]$ (red line), and other two beginning from $\mathbf{p}_{\text{init}} = [0, 8]$ and [5, 0].

³⁰⁵ concerning all three source positions. The exposure values and execution times are shown at Tab. 1.

Table 1: Exposure value and execution time for the paths shown in illustrative example of Fig.3.

$oldsymbol{p}_{ ext{init}}$	$E(\cdot)$	$\mathbf{Time}(\mathbf{s})$
[0, 4]	43.533	1349
[0,8]	43.126	138
[5, 0]	46.759	130

It is possible to notice that, the execution time for the first trial was 1349 s, while for the other two sources it was approximately ten times faster. The first path is slower because, in the first run, our method has to compute the value function (9), which is more computationally costly. However, for all subsequent computations of a MEP (for the same p_{goal}), we can use the previous V(p) illustrated at Fig.3. This is an essential advantage of our technique when compared to the literature since other approaches usually require a complete algorithm reset (e.g., evolutionary algorithms) for every new p_{init} .

315

320

Tab. 2 presents a comparison between our method and the other three discussed in [2]. The parameter m represents the grid resolution used in the algorithm. Small grid cell numbers have less exposure, especially due to the low accuracy of the results. On the other hand, high accuracy grids (with larger m) achieves values closer to the true exposure, however, by exponentially increasing time and storage costs [2]. Our approach is capable of providing the most

accurate value of $E(\cdot)$ (optimal), except for some small numerical inaccuracy.

		HGA-NFE [2]		Grid-based [2]		Voronoi-based [2]		Semi-Lagrangian	
_	m	$E(\cdot)$	$\mathbf{Time}(\mathbf{s})$	$E(\cdot)$	$\mathbf{Time}(\mathbf{s})$	$E(\cdot)$	Time(s)	$E(\cdot)$	Time(s)
	40	42.59	36.34	57.04	42.76				
	80	43.01	83.76	55.05	178.30	47.44	4.60	43.53	1349
	100	43.38	132.91	54.66	274.28				

Table 2: Comparison between our method and results in Ye et al. [2].

5.2. Comparative analysis with the state-of-the-art

In order to qualitatively evaluate the performance of our Semi-Lagrangian approach, first, we define the sensing function and the field intensity function according to [12] (state-of-the-art). The authors have used the probability coverage model with noise for the sensor node, given by Eq. (4). Assuming the noise energy as a normal distribution with zero mean, $\eta \sim \mathcal{N}(0, \sigma^2)$, they have simplified the sensing function, such as:

$$S(\boldsymbol{s}, \boldsymbol{p}) \approx -\ln\left(1 - Q\left(\frac{A - \frac{\lambda}{\|\boldsymbol{s}-\boldsymbol{p}\|^{\mu}}}{\sigma}\right)\right),$$

with

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{\frac{-t^2}{2}} dt$$

where A = 6.0 is a threshold detection, $\lambda = 100.0$, $\mu = 1.0$ and $\sigma = 1.0$. Still, for the field intensity function, results in [12] are based on the all-sensor intensity model (5). Next, we apply our method to the exact scenarios described in the aforementioned work, whose sensors were deployed into a $500 \text{ m} \times 500 \text{ m}$ environment. The authors have used three different random distributions to place the nodes:

(i) Exponential distribution method, (ii) Uniform distribution method, and (iii) Gaussian distribution method. They have tested a total of 240 configurations, with 80 different topologies for each of those three distribution methods. The set of topologies also present networks with a number of sensors n ranging from 30 to 100¹. Here, however, for the sake of space, we have simulated experiments
³³⁵ for all topologies with 30 and 100 nodes to verify the behavior of our approach subjected to a low and a high n.

All trials receive as input the source position $p_{init} = [0, 150]$ and the destination position $p_{goal} = [500, 350]$, and they were executed using Python 3.8.5 on an Intel CoreTM i7-7500U CPU 2.70 GHz x 4 and 16 GB of RAM under Ubuntu ³⁴⁰ 20.04.

Tables 3, 4 and 5 compile the results for the Exponential, Uniform and Gaussian distribution methods, respectively. When comparing the minimum exposure value of the GA-MEP and the GB-MEP with our approach, we can see that the Semi-Lagrangian method always provides results that are smaller (or at most equivalent) to those in [12].

345

Here it is important to say that the exposure values for the GA-MEP given in the tables are the best ones obtained after H runs of their main algorithm. Since it is based on meta-heuristics, there are no guarantees that the optimal values are always reached. Also, the execution times shown for the GA-MEP are the average values of all trials, such that the real-time spent to reached the best $E(\cdot)$ is H times larger than those presented. Consequently, we can claim that, based on the experiments, for $H \geq 3$ the Semi-Lagrangian approach is faster than the GA-MEP in most of the scenarios.

¹Data available at: Nguyen Thi My, Binh (2020), "Data for: Efficient Approximation Approaches to Minimal Exposure Path Problem in Probabilistic Coverage Model for Wireless Sensor Networks", Mendeley Data, V1, doi: 10.17632/5zh6cc2xww.1

Instance		GB-MEP [12]		GA-N	MEP [12]	Semi-Lagrangian	
\boldsymbol{n}	Ord	$E(\cdot)$	Time (s)	Best $E(\cdot)$	Avg. time(s)	$E(\cdot)$	Time (s)
	1	38.3990	0.631	43.3873	331	26.2863	557
	2	0.0402	0.695	0.0401	394	0.0281	504
	3	29.8977	0.974	29.5743	551	25.3489	517
	4	0.4342	0.429	0.4342	247	0.3288	431
20	5	1.8524	0.784	1.6056	411	1.6056	477
30	6	0.0481	0.929	0.0421	320	0.0406	724
	7	1.0559	1.519	1.0604	313	0.6674	437
	8	0.0027	0.427	0.0023	224	0.0021	426
	9	0.0114	2.487	0.0109	256	0.0102	443
	10	0.1734	0.849	0.1280	385	0.1239	491
	1	5.2302	1.6840	4.0037	934	4.0037	3085
	2	1035.9792	7.304	1034.8402	1050	903.3848	2619
	3	2.4163	1.324	2.4069	691	2.3007	1959
	4	1.2142	1.066	1.1727	807	1.0458	2210
100	5	6.6685	1.306	6.2369	888	6.1064	4068
100	6	7.8457	1.189	6.4425	1030	5.5539	2882
	7	3.8835	3.012	4.3673	1121	2.8832	1979
	8	61.0651	1.093	60.3105	917	51.5410	2060
	9	138.5203	1.099	120.4673	1348	105.7741	2413
	10	1.8611	1.794	1.4176	667	1.3986	2030

Table 3: Comparison with nodes placed using the Exponential distribution method [12].

Table 6 compiles the percentages of improvement provided by our method ³⁵⁵ over the minimum values given by both algorithms of [12]. The highest improvements have been reached under Exponential (30) and Gaussian (100). Also, the total average of the Exponential distribution is higher than for other distributions, indicating that the GA-MEP is more susceptible to highly concentrated nodes. The average improvement for all scenarios was approximately 10%.

360

Fig. 4 present some comparative tests compiled in the previous tables. Here, for each distribution (Exponential, Uniform, and Gaussian) and each $n = \{30, 100\}$, we have chosen the instance with the best percentage improvement reached by our method. Blue and green lines represent the MEPs obtained by

Instance		GB-MEP [12]		GA-N	MEP [12]	Semi-Lagrangian	
\boldsymbol{n}	Ord	$E(\cdot)$	Time (s)	Best $E(\cdot)$	Avg. time(s)	$E(\cdot)$	Time (s)
	1	0.0053	3.094	0.0047	515	0.0046	700
	2	0.0117	2.444	0.0109	397	0.0108	712
	3	0.0082	1.513	0.0079	371	0.0077	701
	4	0.1768	5.414	0.1749	444	0.1651	1353
20	5	0.0019	0.786	0.0017	290	0.0017	600
30	6	0.0146	4.013	0.0142	393	0.0135	528
	7	0.0026	0.477	0.0021	317	0.0021	506
	8	0.0033	1.535	0.0073	326	0.0028	937
	9	0.0128	2.504	0.0118	378	0.0103	603
	10	0.0050	0.733	0.0042	493	0.0040	784
	1	4.1743	9.036	5.3521	1160	3.3556	3215
	2	5.3815	2.811	4.2433	1054	4.2017	2095
	3	20.5244	9.3	20.4461	1110	19.7358	1786
	4	0.9734	5.185	0.8304	1040	0.7895	2198
100	5	1.2987	3.98	1.2134	1066	1.1025	2811
100	6	1.9280	4.682	4.8181	1268	1.5576	2794
	7	0.6936	3.797	1.8955	1191	0.6198	3191
	8	3.4418	5.836	3.1789	1263	2.9332	2792
	9	106.9770	9.965	106.5400	1272	101.0590	2621
	10	5.1870	5.716	8.0124	1248	4.7532	3047

Table 4: Comparison with nodes placed using the Uniform distribution method [12].

the GA-MEP and the GB-MEP, respectively, while the red line represents our
Semi-Lagrangian approach. Figures 4(a)-4(c) illustrate cases with few nodes, and figures 4(d)-4(f) are cases with a large number of sensors. Here, it is possible to see that most of the cases with greater improvement are those where the GA-MEP fails to approximate the optimal solution. And even when it doesn't happen (Fig 4(a)), there is a significant difference between our result and those provide by [12].

It is also possible to notice that the highest average value was obtained in the Exponential configuration with 100 nodes. It can be explained by the concentration of sensors near the source position.

Instance		GB-MEP [12]		GA-N	MEP [12]	Semi-Lagrangian	
\boldsymbol{n}	Ord	$E(\cdot)$	Time (s)	Best $E(\cdot)$	Avg. time(s)	$E(\cdot)$	Time (s)
	1	0.0008	0.697	0.0009	247	0.0009	704
	2	0.0035	0.6	0.0043	322	0.0029	969
	3	0.0001	0.414	0.0001	230	0.0001	640
	4	0.0111	0.767	0.0302	371	0.0090	844
20	5	0.0014	0.924	0.0012	307	0.0012	850
30	6	0.0035	1.379	0.0037	233	0.0034	751
	7	0.0031	0.687	0.0033	302	0.0031	1092
	8	0.0040	0.691	0.0031	318	0.0031	845
	9	0.0012	1.075	0.0013	240	0.0012	1393
	10	0.0002	0.47	0.0002	255	0.0002	615
	1	0.3134	2.195	0.6915	960	0.2452	3865
	2	1.1851	5.021	1.4138	811	1.0962	2965
	3	0.0009	1.621	0.0010	434	0.0009	5343
	4	0.7224	1.274	0.5442	643	0.5029	3549
100	5	0.1625	1.577	0.4453	986	0.1334	3725
100	6	0.0585	1.176	0.0579	772	0.0579	2591
	7	2.0298	3.22	2.3785	869	1.5953	3954
	8	0.1247	1.772	0.7191	798	0.0909	4074
	9	0.1766	1.404	0.1515	553	0.1511	2325
	10	0.1049	1.3	0.2841	727	0.0871	4044

Table 5: Comparison with nodes placed using the Gaussian distribution method [12].

5.3. Cluttered environments

375

As previously said, the value function $V(\cdot)$ allows the modeling of geometric constraints, given by the boundary conditions at (11). Therefore, it is possible to incorporate obstacles to the searching space, which makes the MEP problem more attainable to real-world scenarios.

In Fig. 5, we progressively added obstacles to the base scenario (Fig. 3). Fig. 5(a) shows the original MEP starting from $p_{init} = [0, 4]$, whose exposure was previously computed as 43.53 (Tab. 2). When two obstacles are added (Fig. 5(b)), a new path is computed and $E(\cdot)$ is increased to 50.78. After the incorporation of five other obstacles (Fig. 5(c)), the path exposure reaches 54.74.

	Exponential		Unif	orm	Gaussian	
Ord	30 100		30	100	30	100
1	31.5	0.0	2.6	19.6	0.0	21.8
2	29.9	12.7	1.3	1.0	18.0	7.5
3	14.3	4.4	3.2	3.5	0.0	0.0
4	24.3	10.8	5.6	4.9	19.2	7.6
5	0.0	2.1	0.0	9.1	0.0	17.9
6	3.6	13.8	5.1	19.2	2.9	0.1
7	36.8	25.8	1.9	10.6	0.0	21.4
8	10.0	14.5	16.1	7.7	1.3	27.1
9	6.4	12.2	12.8	5.1	2.5	0.3
10	3.2	1.3	4.8	8.4	5.0	17.0
Avg.	16.0	9.8	5.3	8.9	4.9	12.1
Std.	13.5	7.9	5.2	6.2	7.4	10.2

Table 6: Percentages of improvement of our technique over [12].

One can see that as more geometric constraints (i.e., non-navigable regions) are incorporated into the environment, the agent's exposure to the sensor network might remain the same or more commonly increase.

5.4. Heterogeneous networks

Finally, we applied our method to a sensor field composed of heterogeneous nodes. We considered the same scenario described in Sec. 5.1, however, here we have randomly chosen the parameter λ , as 1 or 3, for the sensing model (2).

Fig. 6 presents the MEP computed for the heterogeneous network. We highlight that the dashed circles do not represent limits to the detection range of the sensors, but they only serve to illustrate the heterogeneity of the nodes.

6. Conclusion and future work

395

The use of Wireless Sensor Networks (WSNs) is continuously increasing in several civilian and military applications. In this context, the Minimal Exposure Path (MEP) between two arbitrary positions in the sensing field represents the path that minimizes the likelihood of a moving target being detected. Finding the MEP is critical since the exposure allows estimating the coverage quality of a known deployed WSN.

400

In this paper, we proposed an algorithm based on policy iteration for determining an optimal control solution. This novel approach solves the MEP problem ensuring convergence to the optimal solution, given results that overcame the state-of-the-art in all instances, with an average improvement of about

 $10\,\%$ over available benchmarks. Our formulation allows us to incorporate the 405 target's dynamics and tackle geometric constraints (such as obstacles), as well as many different sensing models and intensity functions for homogeneous and heterogeneous network topologies. The method is faster than the best one in the literature, even when employed for a very large number of nodes.

As future work, we intend to extend our method to larger search spaces, 410 where it is possible to incorporate nonholonomic constraints to the target, threedimensional environments, and directional sensing models (such as cameras), among others. To do this, it is imperative to improve the efficiency of the algorithm, possibly resorting to meta-heuristics and numerical approximations.

We also expect to deal with dynamic scenarios, where node sensing functions 415 vary not only with the sensors' location but also along time.

References

420

- [1] S. M. Mohamed, H. S. Hamza, I. A. Saroit, Coverage in mobile wireless sensor networks (M-WSN): A survey, Computer Communications 110 (2017) 133 - 150.
- [2] M. Ye, Y. Wang, C. Dai, X. Wang, A hybrid genetic algorithm for the minimum exposure path problem of wireless sensor networks based on a numerical functional extreme model, IEEE Trans. on Vehicular Technology 65 (10) (2016) 8644-8657.

- [3] P. Balister, Z. Zheng, S. Kumar, P. Sinha, Trap coverage: Allowing coverage holes of bounded diameter in wireless sensor networks, in: IEEE INFOCOM, 2009, pp. 136–144. doi:10.1109/INFCOM.2009.5061915.
 - [4] J. Chen, J. Li, T. H. Lai, Energy-Efficient Intrusion Detection with a Barrier of Probabilistic Sensors: Global and Local, IEEE Trans. Wireless Comm. 12 (9) (2013) 4742–4755. doi:10.1109/TW.2013.072313.122083.
 - [5] S. Meguerdichian, F. Koushanfar, G. Qu, M. Potkonjak, Exposure in Wireless Ad-Hoc Sensor Networks, in: Int. Conf. on Mobile Computing and Networking, Association for Computing Machinery, New York, NY, USA, 2001, p. 139–150. doi:10.1145/381677.381691.
- [6] G. Veltri, Q. Huang, G. Qu, M. Potkonjak, Minimal and Maximal Exposure Path Algorithms for Wireless Embedded Sensor Networks, in: Int. Conf. on Embedded Networked Sensor Systems, 2003, p. 40–50. doi:10.1145/ 958491.958497.
 - [7] J. Chen, J. Li, T. H. Lai, Trapping mobile targets in wireless sensor networks: An energy-efficient perspective, IEEE Trans. on Vehicular Technology 62 (7) (2013) 3287–3300. doi:10.1109/TVT.2013.2254732.
 - [8] H. Feng, L. Luo, Y. Wang, M. Ye, R. Dong, A novel minimal exposure path problem in wireless sensor networks and its solution algorithm, Int. J. of Distributed Sensor Networks 12 (8) (2016). doi:10.1177/ 1550147716664245.
- 445

440

430

- [9] H. N. Djidjev, Efficient computation of minimum exposure paths in a sensor network field, in: J. Aspnes, C. Scheideler, A. Arora, S. Madden (Eds.), Distributed Computing in Sensor Systems, Springer Berlin Heidelberg, Berlin, Heidelberg, 2007, pp. 295–308.
- [10] H. N. Djidjev, Approximation algorithms for computing minimum exposure paths in a sensor field, ACM Trans. Sen. Netw. 7 (3) (2010). doi:10.1145/ 1807048.1807052.

- [11] V. Phipatanasuphorn, Parameswaran Ramanathan, Vulnerability of sensor networks to unauthorized traversal and monitoring, IEEE Trans. on Computers 53 (3) (2004) 364–369. doi:10.1109/TC.2004.1261841.
- [12] H. T. T. Binh, N. T. M. Binh, N. H. Ngoc, D. T. H. Ly, N. D. Nghia, Efficient approximation approaches to minimal exposure path problem in probabilistic coverage model for wireless sensor networks, Applied Soft Computing 76 (2019) 726–743. doi:https://doi.org/10.1016/j.asoc.2018.12. 022.
- [13] T. M. B. Nguyen, C. M. Thang, D. N. Nguyen, T. T. B. Huynh, Genetic algorithm for solving minimal exposure path in mobile sensor networks, in: IEEE Symp. Series on Comp. Intelligence (SSCI), 2017, pp. 1–8. doi: 10.1109/SSCI.2017.8285402.
- ⁴⁶⁵ [14] Y. Song, L. Liu, H. Ma, A. V. Vasilakos, A Biology-Based Algorithm to Minimal Exposure Problem of Wireless Sensor Networks, IEEE Trans. on Network and Service Management 11 (3) (2014) 417–430. doi:10.1109/ TNSM.2014.2346080.
- [15] S. Aravinth, J. Senthilkumar, V. Mohanraj, Y. Suresh, A hybrid swarm in telligence based optimization approach for solving minimum exposure prob lem in wireless sensor networks, Concurrency and Computation: Practice
 and Experience 33 (3) (2021). doi:https://doi.org/10.1002/cpe.5370.
 - [16] L. Liu, X. Zhang, H. Ma, Minimal exposure path algorithms for directional sensor networks, Wireless Communications and Mobile Computing 14 (10) (2014) 979–994. doi:10.1002/wcm.2250.
- 475

455

460

[17] Y. Miao, Y. Wang, W. Xuan, Hybrid particle swarm algorithm for minimum exposure path problem in heterogeneous wireless sensor network, Int. J. of Wireless and Mobile Computing 8 (2015) 74. doi:10.1504/IJWMC. 2015.066760.

- [18] S. Ferrari, G. Foderaro, A potential field approach to finding minimumexposure paths in wireless sensor networks, in: IEEE Int. Conf. on Robotics and Automation, ICRA, 2010, pp. 335–341. doi:10.1109/ROBOT.2010. 5509193.
- [19] L. Liu, G. Han, H. Wang, J. Wan, Obstacle-avoidance minimal expo sure path for heterogeneous wireless sensor networks, Ad Hoc Networks
 55 (2017) 50-61.
 - [20] D. G. Macharet, A. Alves Neto, D. Shishika, Minimal exposure dubins orienteering problem, IEEE Robotics and Automation Letters 6 (2) (2021) 2280-2287. doi:10.1109/LRA.2021.3061004.
- ⁴⁹⁰ [21] M. Falcone, R. Ferretti, Semi-Lagrangian approximation schemes for linear and Hamilton—Jacobi equations, SIAM, 2013.
 - [22] M. Bardi, I. Capuzzo-Dolcetta, Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations, Springer Science & Business Media, 1997.
- ⁴⁹⁵ [23] C. M. Chilan, B. A. Conway, Optimal nonlinear control using hamilton-jacobi-bellman viscosity solutions on unstructured grids, J. of Guidance, Control, and Dynamics 43 (1) (2020) 30–38. doi:10.2514/1. G004362.
 - [24] G. Barles, P. E. Souganidis, Convergence of approximation schemes for
- ⁵⁰⁰ fully nonlinear second order equations, Asymptotic analysis 4 (3) (1991) 271–283.
 - [25] K. Deng, Z. Liu, Minimum path exposure and detection interval setting for target detection using wireless sensor network, Int. J. of Wireless Information Networks 14 (2007) 289–294.
- 505 [26] D. Cruz-Uribe, C. Neugebauer, Sharp error bounds for the trapezoidal rule and simpson's rule., J. of Inequalities in Pure & Applied Mathematics 3 (4) (2002).

Armando Alves Neto received the B.S.E. degree in Automation and Control Engineering from the Universidade Federal de Minas Gerais in 2006, and S.M. and Ph.D. degrees in Computer Science from UFMG in 2008 and 2012, respectively. He is an Assistant Professor

at the Department of Electronic Engineering at UFMG. Research interests include real-time motion planning, multi-agent control, robust control, and collision avoidance strategies.

510

Victor Costa da Silva Campos is a Professor at the Department of Electronics Engineering at Universidade Federal de Minas Gerais (UFMG) since 2018. He received the B.Eng. degree

in Control and Automation Engineering from UFMG in 2009, and the M.Sc. and PhD degrees in Electrical Engineering also from UFMG in 2011 and 2015, respectively. His research interests include robust control, adaptive control, Takagi-Sugeno fuzzy systems, Computational Intelligence applied to control systems and motion planning.

Douglas Guimarães Macharet is an Assistant Professor at the Department of Computer
Science (DCC) of the Universidade Federal de Minas Gerais (UFMG). He received an M.Sc. and D.Sc. degrees in Computer Science from the same university in 2009 and 2013, respectively. He is with the Computer Vision and Robotics Laboratory (VeRLab), and his main research interests are in mobile robotics, focusing on motion planning and navigation, multirobot systems, and swarm robotics and human-robot interaction.





(a) 30(7) nodes, Exponential: 36.8%.











(e) 100(1) nodes, Uniform: 19.6%. (f) 100

(f) 100(8) nodes, Gaussian: 27.1%.

Figure 4: Best results of our approach in comparison with [12] for each set of topologies.



(a) Empty environment: $E(\cdot) = (b)$ Two added obstacles: $E(\cdot) = 43.533.$ 50.785.



(c) Five more obstacles added: $E(\cdot) = 54.737$.

Figure 5: MEP computed by our method in cluttered environments: a) empty scenario discussed in Sec. 5.1; b) two obstacles have been added, forcing the method to modify the MEP; c) five more obstacles have been incorporated to the space, resulting in an even higher final exposure.



Figure 6: MEP computed for a scenario with a heterogeneous sensor network: $E(\cdot) = 32.032$ and execution time of 1263 s.