

# Distributed area coverage control with imprecise robot localization: Simulation and experimental studies

Sotiris Papatheodorou<sup>1</sup> , Anthony Tzes<sup>1</sup>,  
Konstantinos Giannousakis<sup>2</sup> and Yiannis Stergiopoulos<sup>2</sup>

## Abstract

This article examines the static area coverage problem by a network of mobile, sensor-equipped agents with imprecise localization. Each agent has uniform radial sensing ability and is governed by first-order kinodynamics. To partition the region of interest, a novel partitioning scheme, the Additively Weighted Guaranteed Voronoi diagram is introduced which takes into account both the agents' positioning uncertainty and their heterogeneous sensing performance. Each agent's region of responsibility corresponds to its Additively Weighted Guaranteed Voronoi cell, bounded by hyperbolic arcs. An appropriate gradient ascent-based control scheme is derived so that it guarantees monotonic increase of a coverage objective and is extended with collision avoidance properties. Additionally, a computationally efficient simplified control scheme is offered that is able to achieve comparable performance. Several simulation studies are offered to evaluate the performance of the two control schemes. Finally, two experiments using small differential drive-like robots and an ultra-wideband positioning system were conducted, highlighting the performance of the proposed control scheme in a real world scenario.

## Keywords

Multi-agent systems, multi-robot systems, mobile robots, cooperative systems, autonomous agents

Date received: 26 October 2017; accepted: 1 August 2018

Topic: Mobile Robots and Multi-Robot Systems

Topic Editor: Nak-Young Chong

Associate Editor: M Bernadine Dias

## Introduction

Area coverage of a planar region by a swarm of robotic agents is an active field of research with great interest and potential. The task involves the dispersal inside a region of interest of autonomous, sensor-equipped robotic agents to achieve sensor coverage of that region. Some of the possible applications for multi-agent coverage are environmental and crop monitoring, search and rescue or surveillance.<sup>1</sup> Distributed control algorithms are a popular choice for this task because of their computational efficiency compared to centralized controllers, their ability to adapt in real time to changes in the robot swarm and their reliance only on local information. Thus they can be implemented on robots with low computational power and low power radio transceivers

since each agent is required to communicate only with its neighbors. The downside of distributed control schemes is that they lead to local optima of their objective function because of the lack of information on the state of all agents.

<sup>1</sup> Engineering Division, New York University Abu Dhabi, Abu Dhabi, United Arab Emirates

<sup>2</sup> Electrical and Computer Engineering Department, University of Patras, Rio, Achaia, Greece

### Corresponding author:

Sotiris Papatheodorou, Engineering Division, New York University Abu Dhabi, PO Box 129188, Abu Dhabi, United Arab Emirates.

Email: sotiris.papatheodorou@nyu.edu



Creative Commons CC BY: This article is distributed under the terms of the Creative Commons Attribution 4.0 License

(<http://www.creativecommons.org/licenses/by/4.0/>) which permits any use, reproduction and distribution of the work without further permission provided the original work is attributed as specified on the SAGE and Open Access pages (<https://us.sagepub.com/en-us/nam/open-access-at-sage>).

Area coverage can be categorized either as static<sup>2,3</sup> in which case the goal is the convergence of the agents to some static final configuration or sweep<sup>4,5</sup> in which the agents continuously move inside the region of interest in order to satisfy some changing coverage objective.

Other factors that affect the choice of control scheme are the dynamics of the agents used,<sup>6,7</sup> the geometrical characteristics of the region of interest,<sup>8,9</sup> and the sensing performance of the agents' sensors.<sup>10,11</sup>

Several varying approaches to the coverage problem have been proposed, including annealing algorithms,<sup>12</sup> game theory,<sup>13</sup> distributed optimization,<sup>14</sup> model predictive control,<sup>15,16</sup> optimal control,<sup>17</sup> and event-triggered control.<sup>18</sup>

Due to the inherent uncertainty in the measurements of all localization systems, there is great interest in incorporating this uncertain information in the control scheme. There have been approaches such as the use of probabilistic methods,<sup>19</sup> data fusion<sup>20</sup> or safe trajectory planning.<sup>21</sup> In this article, we use a Voronoi-like partitioning scheme that takes into account the agents' positioning uncertainty and design a suitable gradient-based control law to solve the area coverage problem. Other partitioning schemes based on Voronoi diagrams have been proposed for collision avoidance with static obstacles<sup>22</sup> and between mobile agents.<sup>23</sup>

The problem of static area coverage of a convex planar region of interest by a network of heterogeneous agents is examined in this article. The agents are equipped with isotropic sensors whose performance is allowed to differ among agents, resulting in a heterogeneous team with respect to each agent's sensing performance. The goal of the agent team is achieving coverage over as large part of the region of interest as possible in a cooperative manner using their on-board sensors. The localization uncertainty is taken into account by assuming that each agent resides somewhere within a circular positioning uncertainty region. These uncertainty regions along with the agents' sensed regions are used to create an Additively Weighted Guaranteed Voronoi (AWGV) partitioning of the region of interest and subsequently, a gradient-based distributed control law is derived. The control law guarantees the monotonic increase of an area coverage objective. This control law is then extended with a collision avoidance scheme to ensure the safe operation of agents. A computationally efficient simplified control law is also presented. Both control schemes are evaluated through simulation studies. Moreover, experimental results are presented to provide a complete overview of the proposed control scheme. This article is an extension of the work by Papatheodorou et al.,<sup>24,25</sup> which did not account for heterogeneous agents, implemented only the simplified control law, and did not provide collision avoidance guarantees. The main contribution of this article is the incorporation of the agents' localization uncertainty in the coverage control scheme by introducing the AWGV diagram and presenting an experimental implementation of the devised control scheme.

## Mathematical preliminaries

Assume a compact convex region  $\mathcal{Q} \in \mathbb{R}^2$  under surveillance and a network of  $n$  dimensionless mobile agents. Each agent's dynamics is governed by

$$\dot{q}_i = u_i, \quad q_i \in \mathcal{Q}, \quad u_i \in \mathbb{R}^2, \quad i \in I_n \quad (1)$$

where  $u_i$  is the control input for each agent and  $I_n = \{1, 2, \dots, n\}$ .

The agents' positions are not known precisely and thus in the general case each agent has a positioning uncertainty of different measure. We assume that an upper bound  $r_i$  for the positioning uncertainty of each agent is known and thus its center resides within a disk

$$C_i^u(q_i, r_i) = \{q \in \mathcal{Q}: \|q - q_i\| \leq r_i\}, \quad i \in I_n \quad (2)$$

where  $\|\cdot\|$  is the Euclidean metric and  $q_i$  the agent's position as reported by its localization equipment.

All agents are equipped with omnidirectional sensors with circular sensing footprints and as such the sensed region of each agent can be defined as

$$C_i^s(q_i, R_i) = \{q \in \mathcal{Q}: \|q - q_i\| \leq R_i\}, \quad i \in I_n$$

where  $R_i$  is the sensing range of each sensor.

We also define for each agent a "Guaranteed Sensed Region" (GSR) as the region that is sensed by the agent for all of its possible positions inside  $C_i^u$ . The GSRs are defined as

$$C_i^{gs}(C_i^u, C_i^s) = \left\{ \bigcap_{q_i \in C_i^u} C_i^s(q_i, R_i), \forall q_i \in C_i^u \right\}, \quad i \in I_n$$

and since  $C_i^u$  and  $C_i^s$  are disks, the previous expression can be further simplified into

$$C_i^{gs}(q_i, R_i^g) = \{q \in \mathcal{Q}: \|q - q_i\| \leq R_i^g\}, \quad i \in I_n \quad (3)$$

where  $R_i^g = R_i - r_i$ , provided that  $R_i \geq r_i$ .

When an agent's position is known precisely, that is,  $r_i = 0$ , its GSR is equivalent to its sensed region, whereas when an agent's positioning uncertainty is too large compared to its sensing capabilities, that is,  $r_i > R_i$ , then  $C_i^{gs} = \emptyset$ .

## Space partitioning

Most distributed area coverage schemes assign an area of responsibility to each agent using a Voronoi or Voronoi-inspired partitioning scheme. In this article, due of the localization uncertainty of the agents as well as their heterogeneous sensing patterns, the AWGV diagram described in section "AWGV diagram" section is developed for the assignment of areas of responsibility.

### Guaranteed Voronoi diagram

The GV diagram<sup>26</sup> is defined for a set of uncertain regions  $D = \{D_1, D_2, \dots, D_n\}, D_i \subset \mathcal{Q}$ . Each uncertain region  $D_i$

contains all possible positions of a point  $q_i \in \mathbb{R}^2$  whose localization cannot be measured precisely. The GV diagram generates cells  $W_i$ , defined as

$$W_i = \{q \in \mathcal{Q} : \max \|q - q_i\| \leq \min \|q - q_j\| \mid \forall j \in I_n, j \neq i, q_i \in D_i, q_j \in D_j\}, \quad i \in I_n \quad (4)$$

Each cell  $W_i$  contains the points on the plane that are closer to the corresponding point  $q_i$  of region  $D_i$  for all possible configurations of the uncertain points. It is important to note that when two or more uncertain regions overlap, none of them are assigned GV cells. In addition, if the uncertain regions degenerate into points, the GV diagram converges to the classic Voronoi diagram.

**Remark 1.** In contrast to the classic Voronoi diagram, the GV diagram does not constitute a tessellation of  $\mathcal{Q}$ , since  $\bigcup_{i \in I_n} W_i \subset \mathcal{Q}$ .

### AWGV diagram

The AWGV diagram is a generalization of the GV diagram in which each uncertain region is assigned an importance weight, similarly to the Additively Weighted Voronoi diagram.<sup>27–30</sup> Uncertain regions with greater weights are assigned larger portions of the region  $\mathcal{Q}$ . Using the agents' guaranteed sensing radii  $R_i^g$  as weights, it is possible to assign them regions of responsibility based on their sensing performance as well as their positioning uncertainty. The AWGV diagram for a set of uncertain regions  $D = \{D_1, D_2, \dots, D_n\}, D_i \subset \mathcal{Q}$  with weights  $R = \{R_1^g, R_2^g, \dots, R_n^g\}, R_i^g \in \mathbb{R}$  is defined as

$$W_i^g = \{q \in \mathcal{Q} : \max \|q - q_i\| - R_i^g \leq \min \|q - q_j\| - R_j^g \mid \forall j \in I_n, j \neq i, q_i \in D_i, q_j \in D_j\}, \quad i \in I_n \quad (5)$$

As such, each cell  $W_i^g$  contains the points on the plane that are closer to the corresponding weighted point  $q_i$  of region  $D_i$  for all possible configurations of the uncertain points. In contrast to the GV diagram, when two or more uncertain regions overlap, it is possible that the one with the largest weight is assigned an AWGV cell. It can be shown that if the weights are equal, the AWGV converges to the GV diagram. Moreover, if the uncertain regions degenerate into points, the AWGV diagram converges to the additively weighted Voronoi diagram and if in addition to that the weights are equal, then it converges to the classic Voronoi diagram.

**Remark 2.** Similar to the GV diagram, the AWGV diagram does not constitute a tessellation of  $\mathcal{Q}$ , since  $\bigcup_{i \in I_n} W_i^g \subset \mathcal{Q}$ . We define a neutral region  $\mathcal{O}$  so that  $\mathcal{O} \cup \bigcup_{i \in I_n} W_i^g = \mathcal{Q}$ .

It is useful to define the  $R$ -limited AWGV cells as

$$W_i^R = W_i^g \cap C_i^{g_s}, \quad i \in I_n \quad (6)$$

which are the parts of the AWGV cells that are guaranteed to be sensed by their respective agents. It can be shown that since AWGV cells are disjoint, these  $R$ -limited cells are also disjoint regions.

### AWGV diagram of disks

Given the fact that the uncertain regions have been assumed to be disks  $C^u = \{C_1^u, C_2^u, \dots, C_n^u\}, C_i^u \subset \mathcal{Q}$ , the definition of the AWGV diagram can be simplified into

$$W_i^g = \{q \in \mathcal{Q} : \|q - q_i\| + r_i - R_i^g \leq \|q - q_j\| - r_j - R_j^g, \quad \forall j \in I_n, j \neq i\}, \quad i \in I_n \quad (7)$$

An equivalent definition is

$$W_i^g = \mathcal{Q} \cap \bigcap_{j \in I_n, j \neq i} H_{ij}$$

where

$$H_{ij} = \{q \in \mathcal{Q} : \|q - q_i\| + r_i - R_i^g \leq \|q - q_j\| - r_j - R_j^g\} \\ = \{q \in \mathcal{Q} : \|q - q_j\| - \|q - q_i\| \geq r_i + r_j - R_i^g + R_j^g\}$$

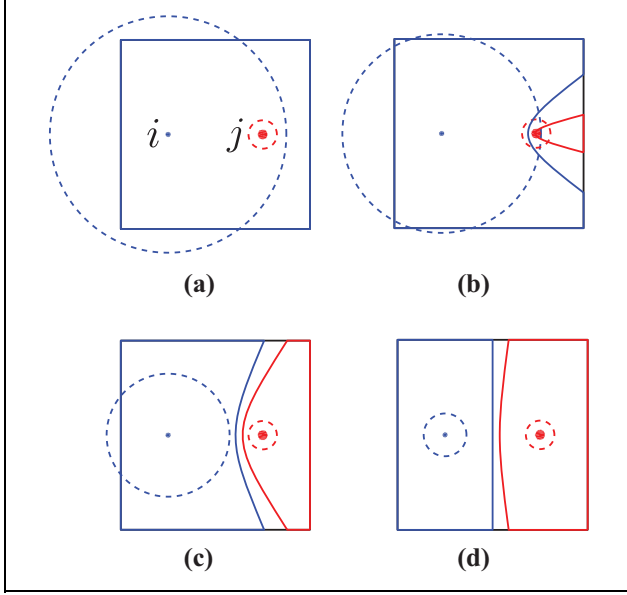
Thus, the boundary  $\partial H_{ij}$  is given by

$$\partial H_{ij} = \{q \in \mathcal{Q} : \|q - q_j\| - \|q - q_i\| = r_i + r_j - R_i^g + R_j^g\}$$

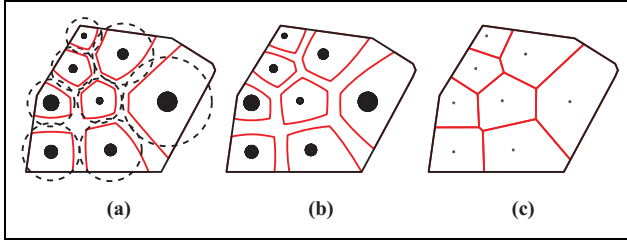
which is the equation of one branch of a hyperbola with foci at  $q_i$  and  $q_j$  and semi-major axis  $a_{ij} = \frac{r_i + r_j - R_i^g + R_j^g}{2}$ . Since  $a_{ij}$  can be either positive or negative,  $\partial H_{ij}$  will correspond to the “East” or “West” branch of a hyperbola respectively, thus the region  $H_{ij}$  can be either convex or non-convex, respectively.

Figure 1 shows the change in the AWGV cells of two agents as the sensing radius of the  $i$ th agent decreases. The uncertainty regions are shown as filled disks, the sensing regions as dashed circles, and the boundaries of the AWGV cells as well as the region  $\mathcal{Q}$  are shown in solid lines. The cell, uncertainty, and sensing region of agent  $i$  are shown in blue, while those of agent  $j$  are shown in red and the region  $\mathcal{Q}$  in black. In Figure 1(a), agent  $i$  has been assigned the whole region, while the cell of agent  $j$  is empty. In Figure 1(b) and (c), the cell of agent  $i$  is non-convex, while that of agent  $j$  is convex. As the sensing radius of agent  $i$  decreases, the area of its cell decreases, while the area of the cell of agent  $j$  increases. In Figure 1(d), the cell of agent  $i$  is bounded by a straight line because  $a_{ij} = 0$ . Figure 2 shows a comparison between the classic Voronoi, the GV, and the AWGV diagrams. The agent positions and uncertainty regions are represented by black disks, their respective cells in red and their sensing regions in dashed black lines.

**Remark 3.** We define the AWGV neighbors  $N_i$  of an agent  $i$  with respect to the AWGV partitioning as those agents that are required to construct the AWGV cell of agent  $i$ . These are defined as



**Figure 1.** AWGV cells as the sensing radius of the  $i$ th agent decreases. AWGV: Additively Weighted Guaranteed Voronoi.



**Figure 2.** Comparison of partitioning schemes: (a) classic Voronoi, (b) GV, and (c) AWGV. AWGV: Additively Weighted Guaranteed Voronoi.

$$N_i = \{j \in I_n, j \neq i : \partial H_{ij} \cap \partial W_i^g \neq \emptyset\} \quad (8)$$

Thus, the cell of agent  $i$  can be written as

$$W_i^g = \Omega \cap \bigcap_{j \in N_i} H_{ij}$$

### Problem statement – distributed control law

By taking into account the AWGV partitioning of the space, the following area coverage criterion is constructed

$$\begin{aligned} \mathcal{H} &= \int \bigcup_{i \in I_n} W_i^R \phi(q) dq \\ &= \sum_{i \in I_n} \int_{W_i^R} \phi(q) dq = \sum_{i \in I_n} \mathcal{H}_i \end{aligned} \quad (9)$$

where the function  $\phi : \mathbb{R}^2 \rightarrow \mathbb{R}_+$  is related to the a priori knowledge of importance of a point  $q \in \Omega$ , indicating the probability of an event to take place at  $q$  in a coverage

scenario. This criterion expresses the area that is guaranteed to be closest to and covered by the agents.

A distributed control law is designed to maximize the coverage criterion (9), while taking into account the agents' guaranteed sensing pattern in equation (3) and the AWGV partitioning equation (7). It is shown that this control law leads to a monotonic increase of the coverage criterion.

**Theorem 1.** For a network of mobile agents with uncertain localization as in equation (2), uniform range-limited radial sensing performance as in equation (3), governed by the individual agent's kinodynamics described in equation (1), and the AWGV partitioning of  $\Omega$  defined in equation (7), the coordination scheme

$$\begin{aligned} u_i &= K_i \int_{\partial W_i^R \cap \partial C_i^{gs}} n_i \phi dq \\ &+ K_i \sum_{j \in N_i} \left[ \int_{\partial W_i^R \cap \partial H_{ij}} v_i^j n_i \phi dq \right. \\ &\left. + \int_{\partial W_j^R \cap \partial H_{ji}} v_j^j n_j \phi dq \right] \end{aligned} \quad (10)$$

where  $n_i$  is the outward unit normal on  $\partial W_i^R$  and  $K_i$  a positive constant, maximizes the performance criterion (9) along the agents' trajectories in a monotonic manner, leading to a locally area-optimal configuration of the network.

**Proof.** In order to guarantee monotonic increase of the coverage criterion (9), we evaluate its time derivative

$$\frac{d\mathcal{H}}{dt} = \sum_{i \in I_n} \frac{\partial \mathcal{H}_i}{\partial q_i} \dot{q}_i = \sum_{i \in I_n} \frac{\partial \mathcal{H}_i}{\partial q_i} u_i$$

Using the gradient-based control law

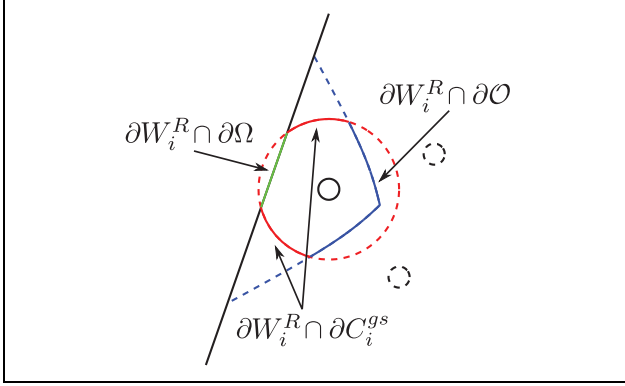
$$u_i = K_i \frac{\partial \mathcal{H}_i}{\partial q_i}^T, i \in I_n$$

where  $K_i > 0$ , monotonic increase of  $\mathcal{H}$  is achieved.

We now evaluate the partial derivative of  $\mathcal{H}$  with respect to  $q_i$  as

$$\frac{\partial \mathcal{H}}{\partial q_i} = \frac{\partial}{\partial q_i} \int_{W_i^R} \phi dq + \sum_{j \in N_i} \frac{\partial}{\partial q_i} \int_{W_j^R} \phi dq$$

Considering the second summation term, infinitesimal motion of  $q_i$  may only affect  $\partial W_j^R$  at  $\partial W_j^R \cap \partial \mathcal{O}$ , where  $\mathcal{O}$  is the neutral region defined in remark 2, since both hyperbola branches are affected by alteration of one of the foci. In addition, only the AWGV neighbors (equation (8)) of the  $i$ th agent are considered in the summation, as a major property of AWGV-partitioning. Therefore, the former expression can be written via the generalized Leibniz integral rule<sup>31</sup> (by converting surface integrals to line ones) as



**Figure 3.** Decomposition of  $\partial W_i^R$  into disjoint sets.

$$\frac{\partial \mathcal{H}^T}{\partial q_i} = \int_{\partial W_i^R} v_i^j n_i \phi \, dq + \sum_{j \in N_i} \int_{\partial W_j^R \cap \partial O} v_j^j n_j \phi \, dq$$

where  $v_i^j, v_j^j$  stand for the transpose Jacobian matrices with respect to  $q_i$  of the points  $q \in \partial W_i^R, q \in \partial W_j^R$ , respectively, that is,

$$v_j^i(q) \triangleq \frac{\partial q^T}{\partial q_i}, \quad q \in \partial W_j^R, \quad i, j \in I_n \quad (11)$$

The boundary  $\partial W_i^R$  can be decomposed in disjoint sets as

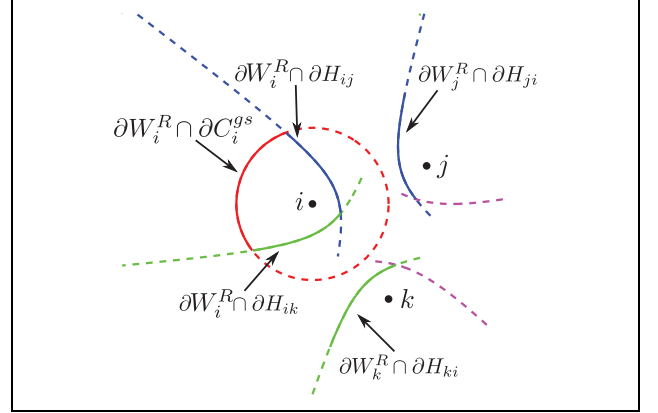
$$\begin{aligned} \partial W_i^R = & \{ \partial W_i^R \cap \partial \Omega \} \cup \{ \partial W_i^R \cap \partial C_i^{gs} \} \\ & \cup \{ \partial W_i^R \cap \partial O \} \end{aligned} \quad (12)$$

These sets represent the parts of  $\partial W_i^R$  that lie on the boundary of  $\Omega$ , the boundary of the agent's GSR, and the boundary of the unassigned neutral region of  $\Omega$ , respectively. This decomposition is shown in Figure 3 with  $\partial W_i^R \cap \partial \Omega$  in green,  $\partial W_i^R \cap \partial C_i^{gs}$  in solid red, and  $\partial W_i^R \cap \partial O$  in solid blue.

Hence  $\frac{\partial \mathcal{H}^T}{\partial q_i}$  can be written as

$$\begin{aligned} \frac{\partial \mathcal{H}^T}{\partial q_i} = & \int_{\partial W_i^R \cap \partial \Omega} v_i^j n_i \phi \, dq + \int_{\partial W_i^R \cap \partial C_i^{gs}} v_i^j n_i \phi \, dq \\ & + \int_{\partial W_i^R \cap \partial O} v_i^j n_i \phi \, dq + \sum_{j \in N_i} \int_{\partial W_j^R \cap \partial O} v_j^j n_j \phi \, dq \end{aligned}$$

It is apparent that  $v_i^j = 0$  at  $q \in \partial W_i^R \cap \Omega$  since all  $q \in \partial \Omega$  remain unaltered by infinitesimal motions of  $q_i$ , assuming no alteration of the environment. Considering the second integral, for any point  $q \in \partial W_i^R \cap \partial C_i^{gs}$  it holds that  $v_i^j(q) = \mathbb{I}_2$ , where  $\mathbb{I}$  stands for the identity matrix, since they translate along the direction of motion of  $q_i$  at the same rate. As far as the sets  $\partial W_i^R \cap \partial O$  and  $\partial W_j^R \cap \partial O, j \in N_i$  are concerned, they can be merged in pairs via utilization of the left and right hyperbolic branches, as introduced in AWGV-partitioning (equation (7)), as follows



**Figure 4.** Illustration of the domains of integration of the control law (10).

$$\begin{aligned} \frac{\partial \mathcal{H}^T}{\partial q_i} = & \int_{\partial W_i^R \cap \partial C_i^{gs}} n_i \phi \, dq \\ & + \sum_{j \in N_i} \left[ \int_{\partial W_i^R \cap \partial H_{ij}} v_i^j n_i \phi \, dq \right. \\ & \left. + \int_{\partial W_j^R \cap \partial H_{ji}} v_j^j n_j \phi \, dq \right] \end{aligned} \quad (13)$$

The unit normal vectors  $n_i, n_j$  and Jacobian matrices  $v_i^j, v_j^j$  in the second part of equation (13) can be evaluated via utilization of the parametric representations of the sets over which the integration takes place. These sets are parts of the branches of a hyperbola ( $\partial H_{ij}$  and  $\partial H_{ji}$ ) that assigns the space among two arbitrary agents  $i, j$ . The evaluation of the normal vectors and Jacobian matrices is examined in more detail in the Appendix 1.

The decomposition of the set  $\partial W_i^R \cap \partial O$  into mutually disjoint hyperbolic arcs  $\partial W_i^R \cap \partial H_{ij}$  and  $\partial W_i^R \cap \partial H_{ik}$  is shown in Figure 4. Figure 4 also shows the other domains of integration used in equation (13)  $\partial W_i^R \cap \partial C_i^{gs}$ ,  $\partial W_j^R \cap \partial H_{ji}$ , and  $\partial W_k^R \cap \partial H_{ki}$ .

The proposed law equation (10) leads to a gradient flow of  $\mathcal{H}$  along the agents' trajectories, while  $\mathcal{H}$  increases monotonically, since

$$\frac{d\mathcal{H}}{dt} = \sum_{i \in I_n} \frac{\partial \mathcal{H}}{\partial q_i} \dot{q}_i = \sum_{i \in I_n} \|u_i\|^2 \geq 0$$

**Remark 4.** The integrals over  $\partial W_i^R \cap \partial C_i^{gs}$  and  $\partial W_i^R \cap \partial H_{ij}$  require knowledge of the AWGV neighbors  $N_i$  in order to compute  $W_i^R$  and  $H_{ij}$ . Similarly, the integral over  $\partial W_j^R \cap \partial H_{ji}$  requires knowledge of the whole cell  $W_j^R$  of agent  $j$  and as such requires information from the AWGV neighbors  $N_j$  of agent  $j$ . Thus, agent  $i$  requires information from all agents in its control neighbors set

$$\tilde{N}_i = \bigcup_{j \in N_i} N_j$$

which correspond to the AWGV neighbors of all AWGV neighbors of agent  $i$ . Given the fact that  $\tilde{N}_i \subseteq I_n$ , the resulting control law is distributed, since each agent computes its own control input based on local information.

### Constraining agents inside the region

When using control law (10), there is no guarantee that the positioning uncertainty regions of all agents remain completely inside  $\Omega$ , that is, it is possible that  $C_i^u \cap \Omega \neq C_i^u$  for some agent  $i$ . Consequently, the control law may lead some agent  $i$  outside the region  $\Omega$ , given that it may reside anywhere within its positioning uncertainty region  $C_i^u$ .

In order to prevent this from happening, instead of using  $\Omega$ , a subset  $\Omega_i^s \subseteq \Omega$  is used instead, which in the general case differs among agents because their uncertainty radii are allowed to be different. This subset of  $\Omega$  is computed as the Minkowski difference of  $\Omega$  with the disk  $C(r_i) = \{q \in \Omega : \|q\| \leq r_i\}$

$$\Omega_i^s = \{q \in \Omega | q + C(r_i) \subseteq \Omega\}, \quad i \in I_n$$

Using the subset  $\Omega_i^s$ , instead of constraining the whole positioning uncertainty regions inside  $\Omega$ , we need only constrain their centers  $q_i$  inside the respective regions  $\Omega_i^s$ . This can be achieved by projecting the velocity control input  $u_i$  of each agent on  $\partial\Omega_i^s$ , resulting in movement along the region's boundary. This control extension is only required if  $q_i \in \partial\Omega_i^s$  and the control input  $u_i$  leads agent  $i$  towards the outside of  $\Omega_i^s$ . As a result, the control law extension that guarantees agents remain inside the region is

$$\tilde{u}_i = \begin{cases} \text{proj}_{\partial\Omega_i^s} u_i & \text{if } q_i \in \partial\Omega_i^s \wedge q_i + \epsilon u_i \notin \Omega_i^s \\ u_i & \text{otherwise} \end{cases} \quad (14)$$

where  $\epsilon$  is an infinitesimally small positive constant.

This control enhancement does not affect the monotonic increase of  $\mathcal{H}$ . This is due to the fact that the inner product of a vector with its projection is always non-negative, thus  $u_i^T \tilde{u}_i \geq 0 \Rightarrow K_i \frac{\partial \mathcal{H}}{\partial q_i}^T \tilde{u}_i \geq 0$ . Consequently  $\frac{\partial \mathcal{H}}{\partial q_i} \tilde{u}_i \geq 0$  which results in monotonic increase of  $\mathcal{H}$  as shown in the proof of theorem 1.

### Inter-agent collision avoidance

Although the derived control scheme guarantees monotonic increase of the coverage objective function  $\mathcal{H}$ , it cannot guarantee collision avoidance between the agents, since it can result in the overlapping of two or more positioning uncertainty regions. In such a case, it is possible that the agents collide, since their actual positions may reside anywhere within their respective positioning uncertainty regions. In order to avoid overlapping of positioning uncertainty regions, an extension to the control law is

devised. Agent  $i$  is stopped when the reported distance  $\|q_i - q_j\|$  between itself and some other agent  $j$  becomes less than or equal to a critical distance  $d_{ij}^c = r_i + r_j + \epsilon$ , and if the control input leads agent  $i$  towards agent  $j$ , that is, if  $u_i^T(q_j - q_i) \geq 0$ .  $\epsilon$  is an infinitesimally small positive constant used so that the control law extension comes into effect before a possible collision. Consequently, each agent  $i$  implements

$$\tilde{u}_i = \begin{cases} 0, & \text{if } \exists j \in N_i : \|q_i - q_j\| \leq d_{ij}^c \wedge u_i^T(q_j - q_i) \geq 0 \\ u_i, & \text{otherwise} \end{cases}$$

This procedure is also presented in Algorithm 1.

It should be noted that this control enhancement does not affect the monotonic increase of  $\mathcal{H}$  since it either has the same value as the initial control law  $u_i$  or zero which results in  $\mathcal{H}$  being unaffected by this particular agent.

Algorithm 1: Collision avoidance – control input  $\tilde{u}_i$  of agent  $i$

---

**Input:**  $q_i, r_i, u_i, q_j, r_j \forall j \in N_i$   
**Output:**  $\tilde{u}_i$

```

1  $\tilde{u}_i \leftarrow u_i$ 
2  $\epsilon \leftarrow$  an infinitesimally small value
3 foreach  $j \in N_i$  do
4    $d_{ij}^c \leftarrow r_i + r_j + \epsilon$ 
5   if  $\|q_i - q_j\| \leq d_{ij}^c$  and  $u_i^T(q_j - q_i) \geq 0$  then
6      $\tilde{u}_i \leftarrow 0$ 
7   end
8 end

```

---

### Simplified control law

Due to the complex integrals over  $H_{ij}$  and  $H_{ji}$  of the complete control law (10), the large amount of information required in order to implement it, as described in remark 4, a simplified control law, using only the first of the three terms of equation (10) is proposed.

**Conjecture 1.** The control law

$$u_i = K_i \int_{\partial W_i^R \cap \partial C_i^s} n_i \phi \, dq \quad (15)$$

is computationally efficient and leads to performance comparable with that of equation (10). However it does not guaranteed the fast convergence rate of the complete control law (10).

**Remark 5.** The two control law enhancements described in sections “Constraining agents inside the region” and “Inter-agent collision avoidance” are needed both for the complete (equation (10)) and simplified (equation (15)) control laws.



### Computational complexity

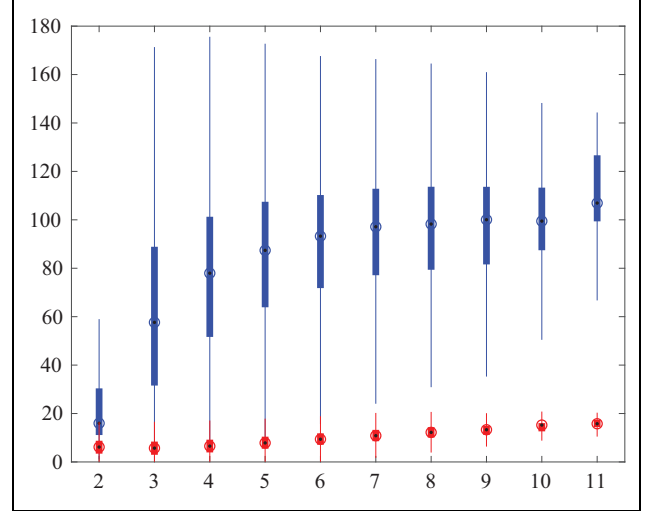
We will first examine the computational complexity of constructing a single AWGV cell. From equation (3) we get that  $W_i^g$  is the result of  $N_i + 1$  intersections, where  $N_i$  are the AWGV neighbors of agent  $i$ . Thus the computational complexity of a single AWGV cell is linear with the number of AWGV neighbors or more formally  $O(N_i)$ .

In order to compute the complete control law (10) each agent  $i$  needs to compute its own AWGV cell as well as the cells of its control neighbors  $\tilde{N}_i$ . Thus the total number of intersections required is  $(N_i + 1) + \sum_{j \in \tilde{N}_i} (N_j + 1)$  which

results in the worst case complexity of this process being  $O(N_i^2)$ . The complete control law also requires the computation of integrals over the agent's own cell as well as the cells of all of its control neighbors  $\tilde{N}_i$ . This results in linear complexity  $O(\tilde{N}_i)$ , thus the total complexity of the optimal control law is dominated by the quadratic complexity of the creation of the AWGV cells and is  $O(N_i^2)$ . It should be noted that this is only an upper bound and in practice the computational complexity will be lower than  $O(N_i^2)$ .

The simplified control law (15) improves on this quadratic complexity by only requiring information from the agent's AWGV neighbors  $N_i$ . Thus, both the cell and integral computations require  $O(N_i)$  time, a significant improvement over the complete control law.

A Monte–Carlo numerical study of the computational cost of both control laws was conducted on an *Intel i7-6500U* CPU operating at 2.5 GHz. A set of random agent configurations were generated with the number of agents ranging from 3 to 40 and both control laws were computed for all of the agents of each random configuration, resulting in 100,633 measurements. A box plot of the results can be seen in Figure 5, depicting the time needed for the computation of the control input for a single agent with respect to the number of its AWGV neighbors  $N_i$ , in blue and red for the complete and simplified control laws, respectively. The circles indicate the median, the bottom and top edges of the filled rectangles the 25th and 75th percentiles, respectively, and the lower and upper whiskers represent measurements that are within 1.5 times the interquartile range (the distance between the 25th and 75th percentiles) from the 25th and 75th percentiles, respectively. It is observed that the simplified control law does indeed approach linear complexity, while the complete control law's complexity is lower than quadratic in practice, as was expected. The significantly greater variance of the measurements of the complete control law compared to the simplified one is due to the fact that the length of the hyperbolic arcs  $\partial W_i^R \cap \partial H_{ij}$  and  $\partial W_j^R \cap \partial H_{ji}$  varies more than the length of the circular arcs  $\partial W_i^R \cap \partial C_i^{gs}$ . It is also observed that the computational time required for the complete control law is significantly longer than that required for the simplified control law.



**Figure 5.** Control law computation time in ms versus the number of AWGV neighbors  $N_i$  for the complete (blue) and simplified (red) control laws. AWGV: Additively Weighted Guaranteed Voronoi.

### Simulation studies

Simulations have been conducted in order to evaluate the efficiency of the complete control law (10) and compare it with the simplified control law (15). The a priori importance of points inside the region was  $\phi(q) = 1, \forall q \in \Omega$ . In this case, the maximum possible value for  $\mathcal{H}$  is

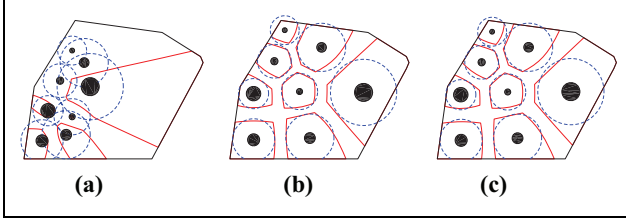
$$\mathcal{H}^{\max} = \sum_{i \in I_n} \mathcal{A}(C_i^{gs})$$

where  $\mathcal{A}(\cdot)$  is the area function, given that  $n$  guaranteed sensing disks can be packed inside  $\Omega$ . In all simulations, the control law extensions as described in sections “Constraining agents inside the region” and “Inter-agent collision avoidance” are used.

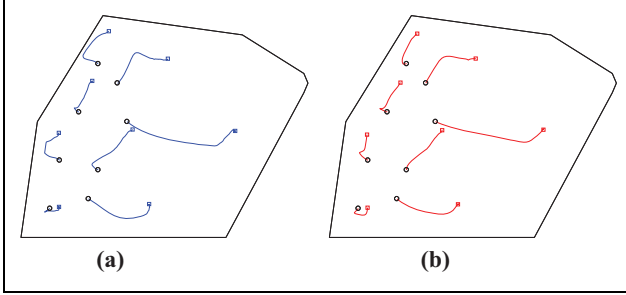
Figures 6, 9, and 12 show the initial and final states of the network for both control laws, with the uncertainty disks shown as filled black circles, the guaranteed sensing disks in dashed blue and the AWGV cells in solid red. The trajectories of the agents are shown in Figures 7, 10, and 13 for both the complete and simplified control laws, with the initial (final) positions represented by circles (squares). Figures 8, 11, and 14 show the evolution of the coverage objective  $\mathcal{H}$  with time as a percentage of its maximum value, again in blue for the complete and red for the simplified control law.

### Case study I

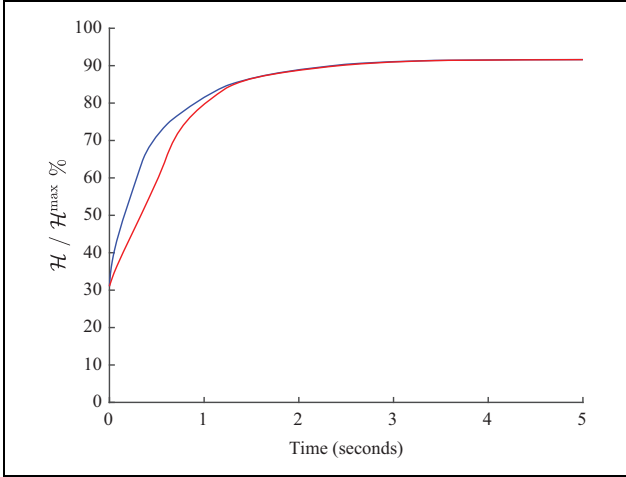
In this case study, an indicative network of eight agents is simulated. Both the uncertainty radii  $r_i$  and sensing radii  $R_i$  differ between agents. Figure 6 shows the initial and final states of the network for both control laws, Figure 7 shows the agent trajectories, and Figure 8 shows the evolution of the coverage objective  $\mathcal{H}$  with time as a percentage of its



**Figure 6.** Case study I: initial (a) and final configuration for the complete (b) and simplified (c) control laws.



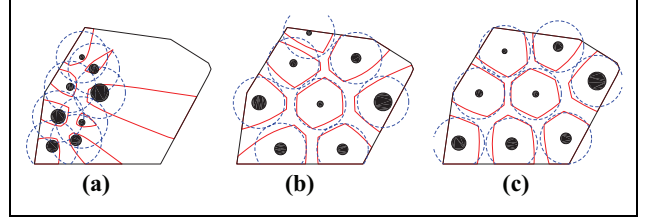
**Figure 7.** Case study I: agents' trajectories for the complete (a) and simplified (b) control laws.



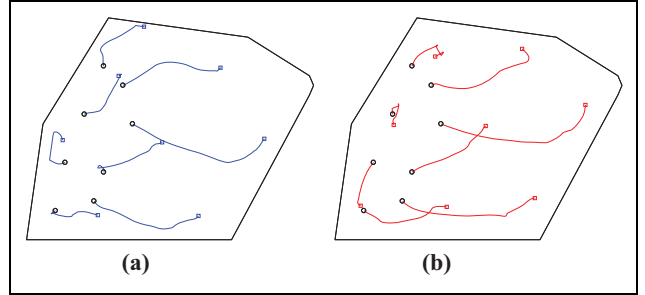
**Figure 8.** Case study I: evolution of the coverage objective  $\mathcal{H}/\mathcal{H}^{\max}$  through time for the complete (blue) and simplified (red) control laws.

maximum value. It is observed that for both control laws the agents converge to almost identical final configurations, with a mean distance between final agent positions of just 0.40% of the region's diameter. The coverage objective  $\mathcal{H}$  increased monotonically in both cases, although this is not guaranteed to happen when using the simplified control law.  $\mathcal{H}/\mathcal{H}^{\max}$  has reached a final value of 91.60% for both control laws.

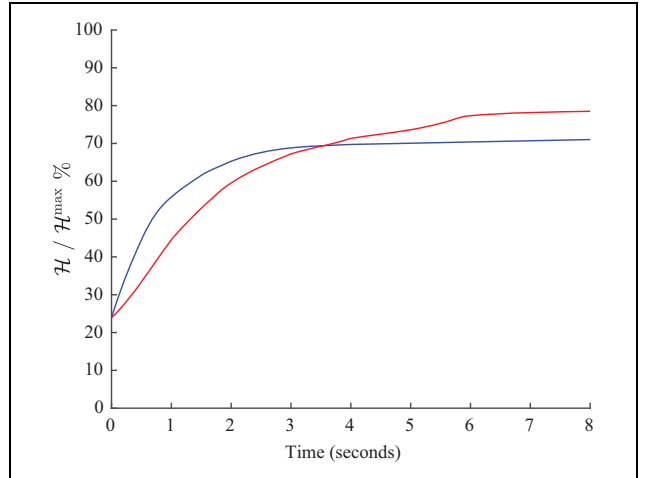
In order to better show the effect of positioning uncertainty in the performance of the agent team, this case study was repeated several times with the same initial conditions for various values of the agents' positioning



**Figure 9.** Case study II: initial (a) and final configuration for the complete (b) and simplified (c) control laws when using the virtual sensing radii, resulting in a common guaranteed sensing radius.



**Figure 10.** Case study II: agents' trajectories for the complete (a) and simplified (b) control laws.



**Figure 11.** Case study II: evolution of the coverage objective  $\mathcal{H}/\mathcal{H}^{\max}$  through time for the complete (blue) and simplified (red) control laws.

**Table 1.** Dependence of the agents' performance on their positioning uncertainty.

Uncertainty (%)	0%	25%	50%	75%	100%
Complete	4.5165	4.1024	3.6843	3.3185	2.9742
Simplified	4.5168	4.1104	3.6885	3.3188	2.9742

uncertainty. Table 1 shows the final value of the coverage objective  $\mathcal{H}$  for both the complete and simplified control laws where 100% corresponds to the positioning uncertainty in the simulation shown in Figures 6 to 8. It is



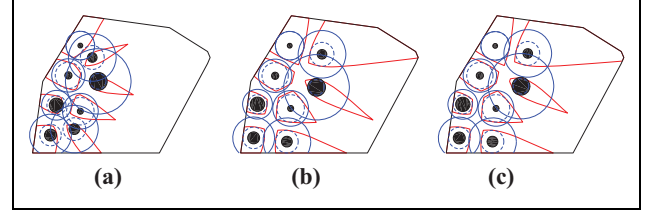
observed that as positioning uncertainty increases, the final value of the coverage objective decreases, since the area of both the GSRs  $C_i^{gs}$  and the AWGV cells  $W_i^g$  decreases. It is also observed that the simplified control law achieves the same better performance than the complete one, although not more than 0.2% in these particular simulations. This behavior is further discussed in section “Case study II”.

### Case study II

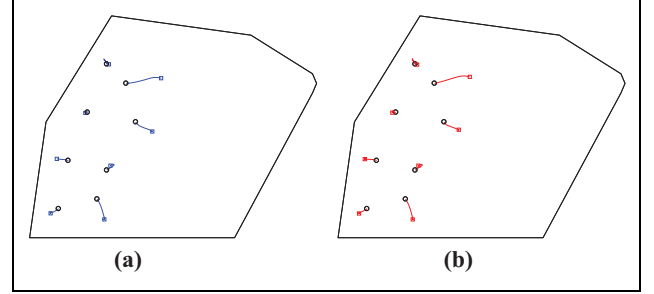
This simulation study is used to present a case where the simplified control law provides better coverage performance than the complete control law. The agents have the same initial configuration as in case study I and the same heterogeneous positioning uncertainty. Their sensing radii are selected so that they have a common guaranteed sensing radius  $R^g$ , thus the sensing radius of each agent is  $\tilde{R}_i = R^g + r_i, \forall i \in I_n$ . In this case the AWGV partitioning converges to the GV partitioning and the assigned cells are based solely on the agents’ positioning uncertainty regions. The common sensing radius  $R^g$  was selected as the mean of the sensing radii used in case study I.

Figure 9 shows the initial and final states of the network for both control laws, Figure 10 shows the agent trajectories, and Figure 11 shows the evolution of the coverage objective  $\mathcal{H}$  with time as a percentage of its maximum value. It can be observed that the agent final configuration differs significantly between the complete and simplified control laws, with the mean distance between final agent positions being 11.97% of the region’s diameter. Additionally, one of the agents in the final configuration of the complete control law has been stopped by the control enhancement described in section “Constraining agents inside the region” before crossing the boundary at the top of the region of interest.

However, the most important observation made from this case study is the fact that the simplified control law achieved a higher final value of 78.50% for the coverage objective  $\mathcal{H}/\mathcal{H}^{\max}$ , versus a value of 71.03% for the complete control law. It should be stressed that this is due to the nature of gradient-ascent methods leading to local optima. The complete control law leads the agents toward the gradient of  $\mathcal{H}$ . Consequently the agents move toward the direction that will result in the greatest rate of increase of  $\mathcal{H}$ . Indeed, it is observed that the complete control law reaches its final configuration sooner than the simplified law. There is no guarantee however that the agents will converge to a global optimum of  $\mathcal{H}$ , as is the case with all gradient-ascent methods. It is possible that the gradient-ascent control input (equation (10)) leads the agents to a low-value local optimum of  $\mathcal{H}$ , while a method that does not follow the gradient, such as the simplified control law (15), avoids this particular local optimum and converges to a local optimum with a higher value. Convergence to local optima is a



**Figure 12.** Case study III: initial (a) and final configuration for the complete (b) and simplified (c) control laws.

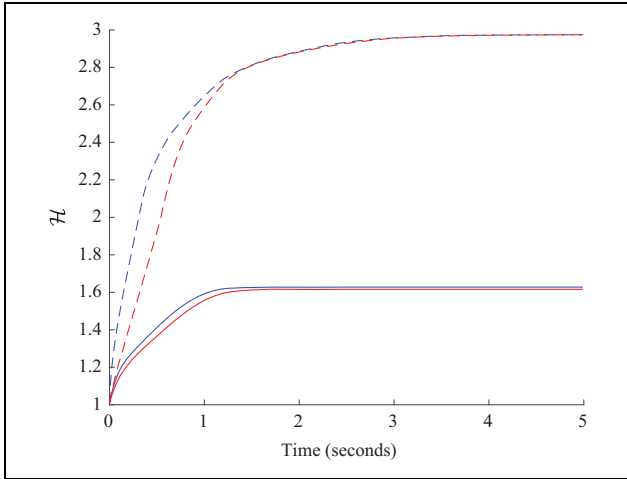


**Figure 13.** Case study III: agents’ trajectories for the complete (a) and simplified (b) control laws.

characteristic of such gradient-based distributed control schemes and this case study highlights how different control schemes may converge to different optima of the same optimization function.

### Case study III

This case study serves to highlight the need for taking heterogeneous sensing into account when designing the control law. To that extent case study I was repeated with the control law being computed as if the sensing radius of all agents was equal to the smallest sensing radius among agents in case study I. The computation of the value of the coverage objective  $\mathcal{H}$  however is made with the actual sensing radii which are the same as those in case study I. Figure 12 shows the initial and final states of the network for both control laws and Figure 13 shows the agent trajectories. It is observed that for both control laws the agents converge to almost identical final configurations, with a mean distance between final agent positions of just 0.27% of the region’s diameter. Figure 14 shows the evolution of the coverage objective  $\mathcal{H}$  with time in solid line for case study III and dashed line for case study I. Blue is used for the complete and red for the simplified control law. Both simulations of case study I converge to a value of 2.97, whereas those of Case Study III converge to final values of 1.62 and 1.63 which corresponds to a decrease in performance of 45%. It is thus shown that taking into account the agents’ heterogeneous sensing performance can lead to significantly improved sensing performance.



**Figure 14.** Case study III: evolution of the coverage objective  $\mathcal{H}$  through time for the complete (blue) and simplified (red) control laws.

## Experimental studies

Two experiments were conducted for the evaluation of the simplified control law (15) in the case of homogeneous agents, both in terms of positioning uncertainty and sensing performance. All agents have common positioning uncertainty  $r$  and sensing radii  $R$ , thus the AWGV partitioning converges to the GV partitioning. The experiments consist of three differential drive robots, an ultra-wideband (UWB) positioning tracking system, and a computer acting as a base station, communicating via a WiFi router.

The utilized robots were the differential drive *Arduino UNO DS Robots*.<sup>32</sup> Although these robots have four powered wheels, they have no steering; thus they are controlled in a manner similar to differential drive robots. These robots lack any kind of positioning system or motor encoders thus an external pose tracking system has to be used. Due to the robots' lack of omnidirectional sensors, circular sensing patterns with a common radius  $R = 1$  m were assumed for the sake of the control law implementation. The robots have a high-level command set that allows them to move forward and backward in a straight line, rotate in place and set the velocity of their motors. Throughout the experiment, their speed was set to the lowest setting of approximately 0.16 m/s. The robots can be inscribed inside a circle with radius  $R^d = 0.299$  m.

The *Pozyx* UWB positioning system<sup>33</sup> was used to measure the robots' pose. The *Pozyx* system consists of four stationary anchors and three tags, 1 affixed to each robot. Each tag communicates with the anchors and measures its relative position using time of arrival measurements and multilateration. Apart from the UWB system, the tags are also equipped with an inertial measurement unit and are thus able to measure their orientation in space in addition to their position. Since it is not possible for more than one tag to communicate with the anchors, a fourth tag located at the

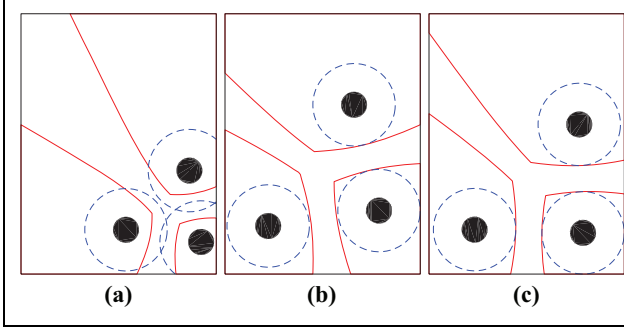
base station computer was used. This tag acts as a master and synchronizes other tags' communication with the anchors in a time-division multiple-access manner. The positioning error of our setup was measured in 38 locations, and its maximum measured value of 0.24 m was used as the robots' common positioning uncertainty radius  $r$ .

The experimental setup was as follows. The master tag synchronizes the tags located on the robots and receives their pose measurements. These measurements are sent to the computer which broadcasts them through WiFi. An ODROID-XU4<sup>34</sup> computer is affixed on each robot whose purpose is receiving the pose measurements and computing the control input for the robot. Each ODROID-XU4 implements the following loop. The robot's control input  $u_i$  is computed according to equation (15) using the current *Pozyx* measurements. If  $u_i$  is greater than some arbitrary threshold, the robot rotates to match the direction of  $u_i$  and moves a small distance in this direction using the high-level command set. Otherwise it remains immobile. The process is then repeated with the new measurements received from the base station computer through WiFi. Each iteration lasts at least 600 ms and varies according to the measure of the angle the robot needs to rotate. It should be noted that the robots were allowed to move both forward and in reverse in order to minimize the magnitude of their rotation.

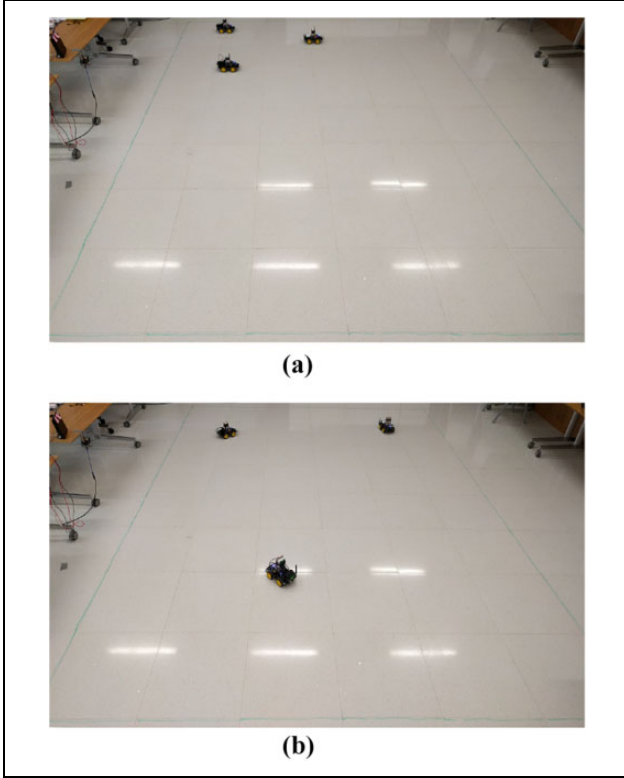
In all figures regarding the experiments, the region shown is  $\mathcal{Q}^s \subseteq \mathcal{Q}$  that is, the region inside which the centers of the positioning uncertainty disks are constrained, as described in section "Constraining agents inside the region". This region is common among all robots since they were assumed to have the same measure of positioning uncertainty. The experiments are compared with simulations with the same initial conditions, that is, robot positions, measure of positioning uncertainty and sensing radii. The region  $\mathcal{Q}^s$  was a  $3.6 \times 4.8$  m<sup>2</sup> rectangle. The robots' initial and final positions for both the experiments and the respective simulations can be seen in Figures 15 and 19, with the uncertainty disks shown as filled black circles, the guaranteed sensing disks in dashed blue and the GV cells in solid red. The evolution of the coverage objective  $\mathcal{H}/\mathcal{H}^{\max}$  can be seen in Figures 18 and 22, in solid blue for the experiment and in dashed red for the simulation.

## Experiment I

The robots' initial and final positions for both the experiment and the respective simulation can be seen in Figure 15 while photographs of the robots' initial and final configurations during the experiment can be seen on Figure 16. It is observed from the final robot configurations of both the experiment and the simulation, that all three guaranteed sensing disks are contained completely within their respective GV cells, indicating that the network has reached a globally optimal configuration. Figure 17 presents a comparison between the robot trajectories filtered by a moving average filter with a window of 49 samples and the

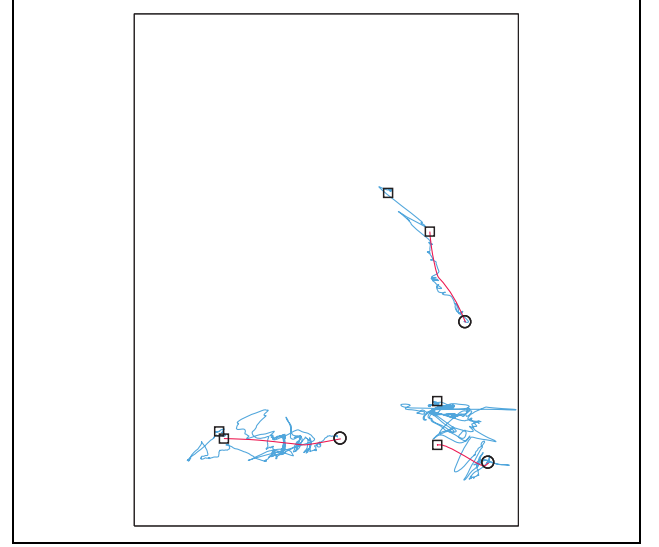


**Figure 15.** Experiment I: initial configuration (a), experiment final configuration (b) and simulation final configuration (c).

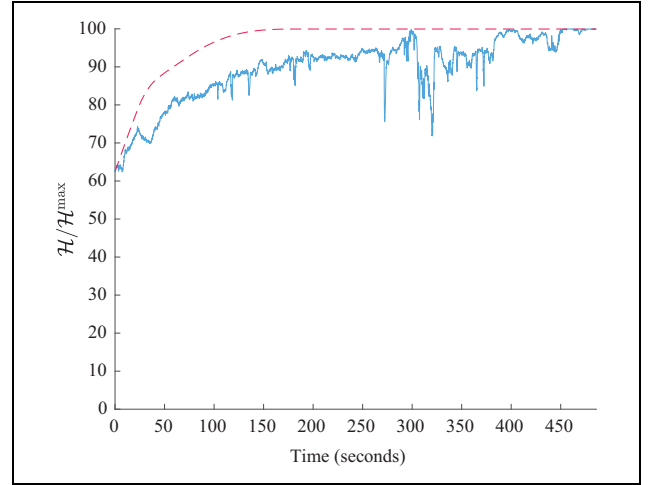


**Figure 16.** Experiment I: images from the initial (a) and final (b) robot positions during the experiment.

simulated trajectories where it can be observed that they are not significantly different. Initial and final positions marked with circles and squares respectively. When compared between the experiment and the simulation, the final positions of the agents have a mean distance of 1.34% the diameter of  $\mathcal{Q}^s$ . The difference in final configurations is small, considering that for the current number of agents and initial conditions there exist multiple equivalent feasible globally optimal configurations and the differences between the theoretical and implemented control law. However this is not the case in all experiments as will be seen in the next section. The evolution of the coverage objective  $\mathcal{H}/\mathcal{H}^{\max}$  can be seen in Figure 18. Although the



**Figure 17.** Experiment I: comparison between the experimental (blue) and simulated (red) robot trajectories.

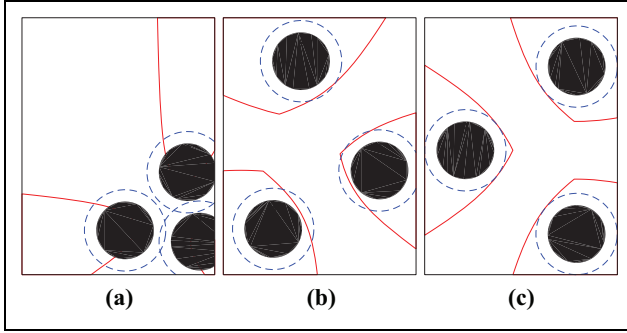


**Figure 18.** Experiment I: evolution of the coverage objective  $\mathcal{H}/\mathcal{H}^{\max}$  through time for the experiment (solid blue) and corresponding simulation (dashed red).

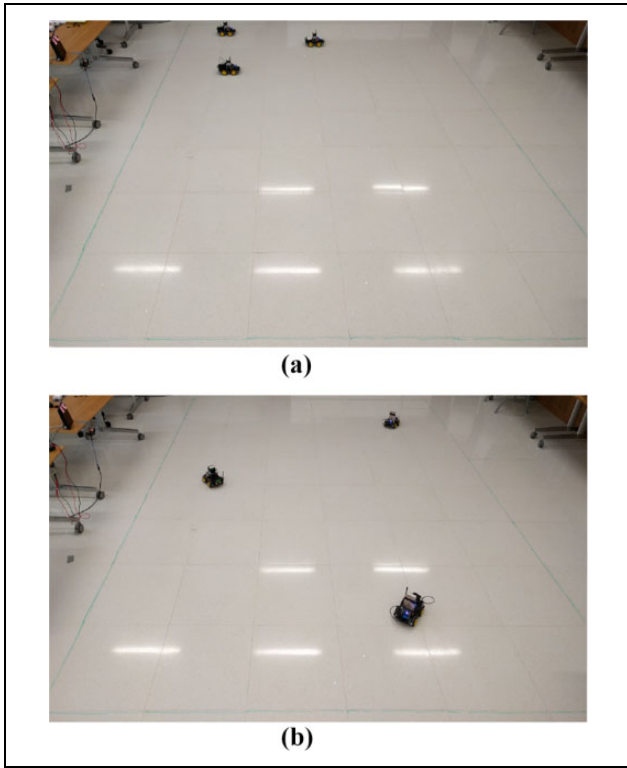
increase of the coverage objective was not monotonic as a result of the implemented control law, it increased from 62.49% to 99.95% which is the final value attained in the simulation. The spikes in the experimental values are due to measurement noise while more gradual decreases of  $\mathcal{H}/\mathcal{H}^{\max}$  are a result of the implementation details such as the robot dynamics and their finite maximum translational and rotational velocities.

### Experiment II

For this experiment, the radius  $R^d$  of the robots circle was added to their positioning uncertainty radius in order to ensure that each robot is completely contained within its respective positioning uncertainty region  $C_i^u$ .

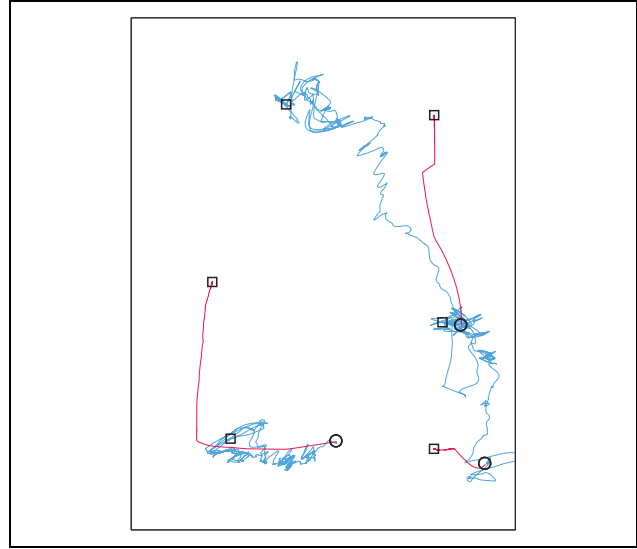


**Figure 19.** Experiment II: initial configuration (a), experiment final configuration (b) and simulation final configuration (c).

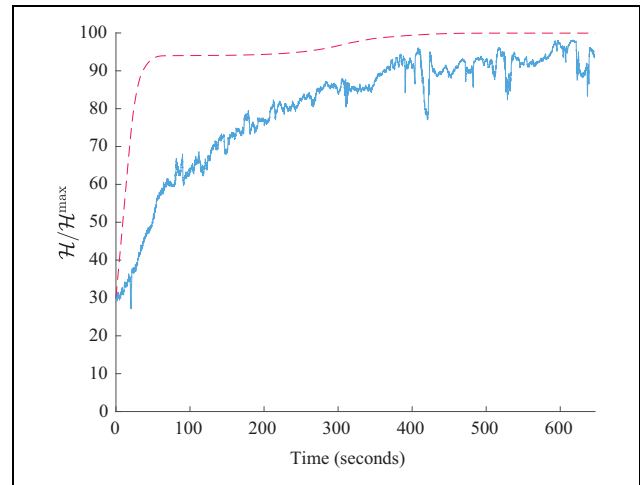


**Figure 20.** Experiment II: images from the initial (a) and final (b) robot positions during the experiment.

This way, due to the control law extension described in section “Constraining agents inside the region”, it is ensured that the whole robots and not just the positioning uncertainty regions of their centers remain inside the region of interest. Moreover, due to the control law extension described in section “Inter-agent collision avoidance” between the agents can be guaranteed. It should be noted that the guaranteed sensing disks were calculated by taking into account the initial positioning uncertainty and not the one augmented by the robots’ dimensions. The robots’ initial and final positions for both the experiment and the respective simulation can be seen in Figure 19 while photographs of the robots’ initial and final configurations during the experiment can be seen in Figure 20. It is observed from

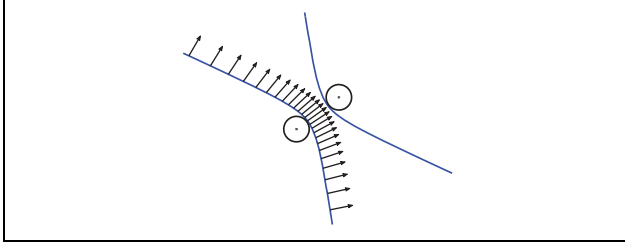


**Figure 21.** Experiment II: comparison between the experimental (blue) and simulated (red) robot trajectories.



**Figure 22.** Experiment II: evolution of the coverage objective  $\mathcal{H}/\mathcal{H}^{\max}$  through time for the experiment (solid blue) and corresponding simulation (dashed red).

the final robot configurations of both the experiment and the simulation that all three guaranteed sensing disks are contained almost completely within their respective GV cells. Figure 21 presents a comparison between the robot trajectories filtered by a moving average filter with a window of 49 samples and the simulated trajectories. Initial and final positions marked with circles and squares, respectively. When compared between the experiment and the simulation, the final positions of the agents have a mean distance of 24.65% the diameter of  $\mathcal{Q}^s$ . Due to the agents’ increased measure of positioning uncertainty compared to the previous case, the resulting GV cells are smaller, which in turn means that the number of local maxima of the objective function  $\mathcal{H}$  is smaller than in the first experiment. Although it is more probable that in this case, the



**Figure 23.** The outward unit normal vectors of one branch of the hyperbola are shown in black.

experiment and simulation will converge to the same configuration, this is not guaranteed to happen. The evolution of the coverage objective  $\mathcal{H}/\mathcal{H}^{\max}$  can be seen in Figure 22. Although the increase of the coverage objective was not monotonic as a result of the implemented control law, it increased from 30.15% to 93.40%, while the simulation attained a final value of 99.95%.

## Conclusion

The problem of area coverage of a planar region by a heterogeneous team of mobile agents under imprecise localization is examined in this article. A gradient-based distributed control scheme was designed based on an AWGV partition of the region. The control scheme is

extended to ensure the safe operation of the mobile agents and a simplified but computationally efficient control law is also proposed. The two control laws are compared in simulation studies and experiments were conducted to evaluate the efficiency of the simplified control law in a realistic scenario.

## Appendix I

### Hyperbola parametric equation

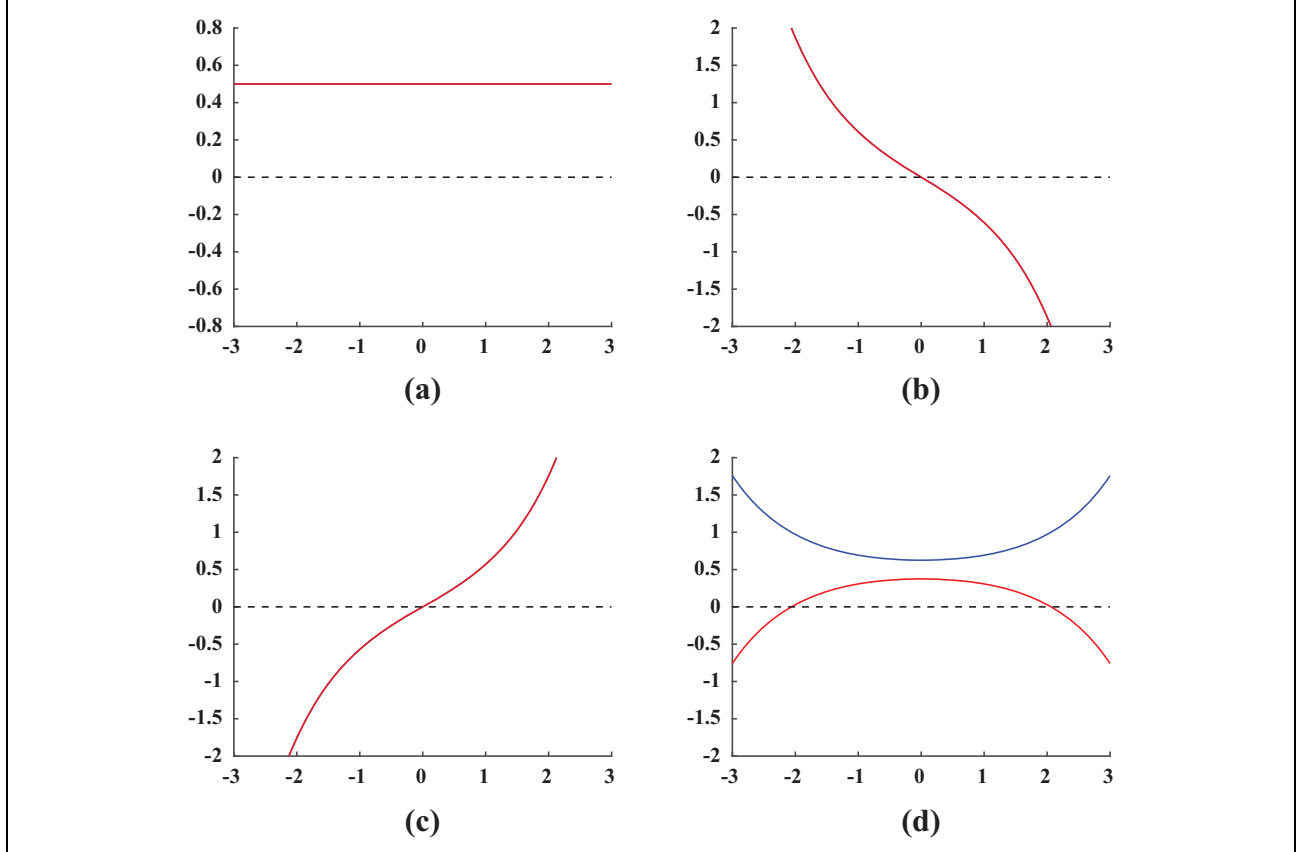
Assuming the foci of the hyperbola are  $q_i = [-c, 0]^T$  and  $q_j = [c, 0]^T$ , its parametric equation is

$$\gamma(t) = \begin{bmatrix} a \cos h(t) \\ b \sin h(t) \end{bmatrix}, t \in \mathbb{R}$$

where  $a$  and  $b$  are the semi-major and semi-minor axes, respectively, and it holds that  $a^2 + b^2 = c^2$ . The positive and negative values of  $a$  correspond to the “East” and “West” branch of the hyperbola, respectively.

By rotating the hyperbola foci around the origin by an angle  $\theta$ , the parametric equation becomes

$$\gamma(t) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} a \cos h(t) \\ b \sin h(t) \end{bmatrix}, t \in \mathbb{R}.$$



**Figure 24.** The Jacobian matrix  $v_i^j$  (solid blue) and  $v_j^j$  (solid red) elements as functions of  $t$ . (a)  $\frac{\partial \gamma_k}{\partial x_i}$ , (b)  $\frac{\partial \gamma_k}{\partial y_i}$ , (c)  $\frac{\partial \gamma_j}{\partial x_i}$ , (d)  $\frac{\partial \gamma_j}{\partial y_i}$ .



Finally, given two foci  $q_i = [x_i \ y_i]^T$  and  $q_j = [x_j \ y_j]^T$  anywhere on the plane, the general hyperbola parametric equation is

$$\gamma(t) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} a \cos h(t) \\ b \sin h(t) \end{bmatrix} + \begin{bmatrix} \frac{x_i + x_j}{2} \\ \frac{y_i + y_j}{2} \end{bmatrix}, t \in \mathbb{R}$$

where  $\theta = \arctan\left(\frac{y_j - y_i}{x_j - x_i}\right)$ .

### Outward unit normal vectors

For any curve  $\gamma(t)$ , the normal vector is defined as

$$\hat{n}(t) = \ddot{\gamma} - \left( \ddot{\gamma} \cdot \frac{\dot{\gamma}}{\|\dot{\gamma}\|} \right) \frac{\dot{\gamma}}{\|\dot{\gamma}\|}$$

The magnitude of this vector is the curvature at that particular point on the curve and it points toward the center of curvature. Since the hyperbolic branches define convex regions on the plane, the outward unit normal vector is given by

$$n(t) = -\frac{\hat{n}}{\|\hat{n}\|} \quad (16)$$

However, since the resulting expressions are too large to show here, an indicative plot of the outward unit normal vectors of one branch of a hyperbola is shown in Figure 23.

### Jacobian matrix

The Jacobian matrix  $v_j^i$  shows the change in the curve

$\gamma_j(t) = \begin{bmatrix} \gamma_{jx}(t) \\ \gamma_{jy}(t) \end{bmatrix}$  caused by the movement of agent  $i$ , thus

$$v_j^i = \begin{bmatrix} \frac{\partial \gamma_{jx}}{\partial x_i} & \frac{\partial \gamma_{jx}}{\partial y_i} \\ \frac{\partial \gamma_{jy}}{\partial x_i} & \frac{\partial \gamma_{jy}}{\partial y_i} \end{bmatrix} \quad (17)$$

The expressions for the Jacobian matrix elements are too large to show here and as such some indicative plots of them with respect to the parameter  $t$  are shown in Figure 24.

### Declaration of conflicting interests


The author(s) declared the following potential conflicts of interest with respect to the research, authorship, and/or publication of this article: This work was partially completed while the authors were with the University of Patras, Greece.

### Funding

The author(s) disclosed receipt of following financial support for the research, authorship, and/or publication of this article: This work has received partial funding from the European Union's

Horizon 2020 Research and Innovation Programme under the Grant Agreement No. 644128, AEROWORKS.

### ORCID iD

Sotiris Papatheodorou  <http://orcid.org/0000-0001-7861-6831>

### References

1. Cortés J and Egerstedt M. Coordinated control of multi-robot systems: a survey. *SICE J Control Meas Syst Integrat* 2017; 10(6): 495–503.
2. Mahboubi H and Aghdam AG. Distributed deployment algorithms for coverage improvement in a network of wireless mobile sensors: relocation by virtual force. *IEEE Trans Control Netw Syst* 2016; 61(99): 1.
3. Eeden CFV, Lee G, Katsuno S, et al. Minimised moving distance algorithm for geometric deployment of robot swarms. In: *Proceeding 55th annual conference of the society of instrument and control engineers of Japan (SICE)*, Tsukuba, Japan, 20–23 September 2016, pp. 1195–1200. New York, NY: IEEE.
4. Abbasi F, Mesbahi A, and Velni JM. A new Voronoi-based blanket coverage control method for moving sensor networks. *IEEE Trans Control Syst Technol* 2017; 99: 1–9.
5. Palacios-Gasós JM, Talebpour Z, Montijano E, et al. Optimal path planning and coverage control for multi-robot persistent coverage in environments with obstacles. In: *IEEE international conference on Robotics and automation (ICRA)*, Singapore, 29 May – 3 June 2017, pp. 1321–1327. New York, NY: IEEE.
6. Bentz W, Hoang T, Bayasgalan E, et al. Complete 3-D dynamic coverage in energy-constrained multi-UAV sensor networks. *Auto Robot* 2018; 42(4): 825–851.
7. Tzes M, Papatheodorou S, and Tzes A. Visual area coverage by heterogeneous aerial agents under imprecise localization. *IEEE Control Syst Lett* 2018; 2(4): 623–628.
8. Alitappeh RJ, Jeddisaravi K, and Guimarães FG. Multi-objective multi-robot deployment in a dynamic environment. *Soft Comput* 2017; 21: 6481–6497.
9. Stergiopoulos Y, Thanou M, and Tzes A. Distributed collaborative coverage-control schemes for non-convex domains. *IEEE Trans Autom Contr* 2015; 60(9): 2422–2427.
10. Stergiopoulos Y and Tzes A. Cooperative positioning/orientation control of mobile heterogeneous anisotropic sensor networks for area coverage. In: *Proceeding IEEE international conference on robotics & automation*, Hong Kong, China, 31 May–5 June 2014, pp. 1106–1111. New York, NY: IEEE.
11. Farzadpour F, Zhang X, Chen X, et al. On performance measurement for a heterogeneous planar field sensor network. In: *IEEE international conference on advanced intelligent mechatronics (AIM)*, Munich, Germany, 3–7 July 2017, pp. 166–171. New York, NY: IEEE.
12. Kwok A and Martinez S. A distributed deterministic annealing algorithm for limited-range sensor coverage. *IEEE Trans Control Syst Technol* 2011; 19(4): 792–804.
13. Ramaswamy V and Marden JR. A sensor coverage game with improved efficiency guarantees. In: *Proceeding American*



- control conference (ACC)*, Boston, MA, USA, 6–8 July 2016, pp. 6399–6404. New York, NY: IEEE.
14. Cortés J, Martinez S, and Bullo F. Spatially-distributed coverage optimization and control with limited-range interactions. *ESAIM: Control Optim Calc Variat* 2005; 11(4): 691–719.
  15. Mohseni F, Doustmohammadi A, and Menhaj MB. Distributed receding horizon coverage control for multiple mobile robots. *IEEE Syst J* 2016; 10(1): 198–207.
  16. Mavrommati A, Tzorakoleftherakis E, Abraham I, et al. Real-time area coverage and target localization using receding-horizon ergodic exploration. *IEEE Trans Robot* 2018; 34(1): 62–80.
  17. Nguyen MT, Rodrigues L, Maniu CS, et al. Discretized optimal control approach for dynamic multi-agent decentralized coverage. In: *Proceeding IEEE international symposium intelligent control (ISIC)*, Buenos Aires, Argentina, 19–22 September 2016, pp. 1–6. New York, NY: IEEE.
  18. Nowzari C and Cortés J. Self-triggered coordination of robotic networks for optimal deployment. *Automatica* 2012; 48(6): 1077–1087.
  19. Habibi J, Mahboubi H, and Aghdam AG. Distributed coverage control of mobile sensor networks subject to measurement error. *IEEE Trans Automat Contr* 2016; 61(11): 3330–3343.
  20. Hage JA, El Najjar ME, and Pomorski D. Fault tolerant collaborative localization for multi-robot system. In: *Proceeding 24th Mediterranean conference on control and automation (MED)*, Athens, Greece, 21–24 June 2016. pp. 907–913. New York, NY: IEEE.
  21. Davis B, Karamouzas I, and Guy SJ. C-opt: coverage-aware trajectory optimization under uncertainty. *IEEE Robot Autom Lett* 2016; 1(2): 1020–1027.
  22. Pierson A and Rus D. Distributed target tracking in cluttered environments with guaranteed collision avoidance. In: *International symposium on multi-robot and multi-agent systems (MRS)*, Los Angeles, CA, USA, 4–5 December 2017, pp. 83–89. New York, NY: IEEE.
  23. Zhou D, Wang Z, Bandyopadhyay S, et al. Fast, online collision avoidance for dynamic vehicles using buffered voronoi cells. *IEEE Robot Autom Lett* 2017; 2(2): 1047–1054.
  24. Papatheodorou S, Stergiopoulos Y, and Tzes A. Distributed area coverage control with imprecise robot localization. In: *Proceeding 24th Mediterranean conference on control and automation (MED)*, Athens, Greece, 21–24 June 2016, pp. 214–219. New York, NY: IEEE.
  25. Papatheodorou S, Tzes A, and Giannousakis K. Experimental studies on distributed control for area coverage using mobile robots. In: *Proceeding 25th Mediterranean conference on control and automation (MED)*, Valletta, Malta, 3–6 July 2017, pp. 690–695. New York, NY: IEEE.
  26. Evans W and Sember J. Guaranteed Voronoi diagrams of uncertain sites. In: *20th Canadian conference on computational geometry (CCCG)*, Montreal, Quebec, 13–15 August 2008, pp. 207–210. CCCG.
  27. Ash PF and Bolker ED. Generalized Dirichlet tessellations. *Geomet Dedi* 1986; 20(2): 209–243.
  28. Kim DS, Kim D, and Sugihara K. Voronoi diagram of a circle set from Voronoi diagram of a point set: II. Geometry. *Comput Aided Geom Des* 2001; 18(6): 563–585.
  29. Jin L, Kim D, Mu L, et al. A sweepline algorithm for Euclidean Voronoi diagram of circles. *Comput Aided Des* 2006; 38(3): 260–272.
  30. Karavelas MI and Yvinec M. Dynamic additively weighted Voronoi diagrams in 2D. *Algorithms ESA* 2002: 586–598.
  31. Flanders H. Differentiation under the integral sign. *Am Math Monthly* 1973; 80(6): 615–627.
  32. Xiao R. Technology. Arduino UNO DS Robot, [http://www.xiao-r.com/Xiaor/?lang=en\\_us](http://www.xiao-r.com/Xiaor/?lang=en_us) (accessed 15 July 2018).
  33. Pozyx Labs. Pozyx Accurate Positioning, [www.pozyx.io/](http://www.pozyx.io/) (accessed 15 July 2018).
  34. Hardkernel. ODROID-XU4, [www.hardkernel.com/](http://www.hardkernel.com/) (accessed 15 July 2018).