Coverage Control of Mobile Robots with Different Maximum Speeds for Time-Sensitive Applications

Soobum Kim[1], María Santos[2], Luis Guerrero-Bonilla[1], Anthony Yezzi[1], and Magnus Egerstedt[3]

***Abstract*—This paper presents a coverage control algorithm for robots with different maximum speeds to effectively cover an area in terms of time of travel. In order to utilize the different speeds of the robots, the region of dominance of a robot is approximated by the set of points where the robot can arrive faster than any other robots in the team by cruising at its maximum speed. Multiplicatively weighted Voronoi diagrams are used to express these speed weighted regions of dominance of the robots. The new cost function of the system takes time instead of distance as its basis, and it is interpreted as the summation of time for robots to reach all weighted points in the coverage domain. We use a controller that drives the robots towards the weighted centroids of their cells, which locally minimizes the new cost function. The resulting final configuration of the robots is shown to provide a temporally optimal coverage instead of a spatially optimal coverage over a domain.**

***Index Terms*—Multi-Robot Systems, Networked Robots, Cooperating Robots**

I. INTRODUCTION

THE problem of coverage control deals with the spatial distribution of robots across a domain of interest in such a way that the domain is effectively "covered", [1]. The standard interpretation of what such coverage entails is that the robots should be able to detect events/phenomena in the domain, with each mobile sensor node (robot) being responsible for all events inside its particular region of dominance. The basic idea is that the robots should take up a static position somewhere in their region of dominance, from which the region is well-covered by the robot's sensor.

In [2], this problem was formulated in terms of the minimization of a locational cost that encodes how well the domain is being covered spatially. Then, the problem was solved by flowing against the gradient of the locational cost in order to find a minimizer, resulting in a continuous-time variation of Lloyd's algorithm [3]. The technical novelty in this paper is the replacement of the locational cost by a temporal cost that reflects the fact that time rather than space is now of primary importance.

If the robots are equipped with different sensor suites, their ability to cover a given region is not uniform across the team, and such heterogeneous sensing capabilities have been considered—both for quantitatively heterogeneous sensors (same events can be detected but the sensors have different effective ranges) [4], [5], [6], [7] and for qualitatively different sensors (different types of phenomena can be detected by the sensor suites) [8], [9], [10]. Similarly, [11] considered robots changing their specializations for sensing specific types of events over time. Additional variations on these themes include robots with different dynamics [5], [12] or energy consumption considerations [13].

If, however, the purpose with the coverage is not detection but rapid servicing of events, the formulation changes. For example, consider the problem of placing fire trucks across a domain. Then it is not enough to simply place them so that the fires are detected, but the trucks also need to be able to get to the fires as quickly as possible. What is at play here are two different mobility phases in that the fire trucks (robots) first need to distribute themselves effectively. This does not have to be done quickly—no sirens need to be blaring—but it should be done so that a rapid response is possible if a fire does erupt. The second phase, in contrast, should have the fire trucks move at maximum speed in order to put out the fires as quickly as possible. Additionally, if the robots have different maximum speeds, the regions of dominance must be modified correspondingly; faster fire trucks should be responsible for a larger area as compared to their slower counterparts.

To date, relatively little has been done when it comes to examining the role of heterogeneous mobility for coverage control. Embracing the idea that faster robots should be in charge of a larger area than slower robots, [5], [12], adjusted the size of the Voronoi cells (the robots' regions of dominance) through an additively weighted Voronoi diagram, which is also referred to as a Power Diagram [14]. Although properly reflecting the idea that faster robots should control larger regions, such a construction is not exact in the sense that all points belonging to a particular region of dominance should be the ones that can be reached faster by the robot in that region than by any other robots.

Based on this observation that a robot should be assigned exactly those points that it can reach faster than any other robot, this paper proposes a coverage control scheme for robots with different maximum speeds which minimizes the time for each agent to reach any points in its region of dominance. This is achieved by using the so-called multiplicatively weighted

Manuscript received: September, 9, 2021; Revised December, 9, 2021; Accepted January, 3, 2022.

This paper was recommended for publication by Editor M. Ani Hsieh upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by the Army Research Lab through Grant No. DCIST CRA W911NF-17-2-0181.

[1]S. Kim, L. Guerrero-Bonilla, and A. Yezzi are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA {skim743, lgb7, ay10}@gatech.edu

[2]M. Santos is with the Department of Mechanical and Aerospace Engineering, Princeton University, Princeton, NJ, USA maria.santos@princeton.edu

[3]M. Egerstedt is with the Samueli School of Engineering, University of California, Irvine, Irvine, CA, USA magnus@uci.edu

Digital Object Identifier (DOI): see top of this page.

(MW) Voronoi diagrams which have been previously introduced to the multi-robot domain in a few different settings. In [6] and [7], MW Voronoi diagrams were used to find and minimize coverage "holes" for a group of mobile sensors with different sensor ranges. In [15], the diagrams were similarly used to minimize the overall cost over a multi-agent system where each agent has a different, individual operating cost. By drawing from these past ideas, this paper utilizes MW Voronoi diagram as a characterization of the areas that can be reached by the robots in each cell in the shortest amount of time. Accordingly, the first mobility phase, i.e., the effective distribution of robots, should result from the minimization of an appropriate cost function, and, accordingly, this cost function takes as its basis time instead rather than distance. The corresponding controller subsequently drives the robots to the configuration that minimizes this new cost function; achieving an effective temporal covering rather than a spatial covering.

The organization of this paper is as follows. Section II introduces MW Voronoi cells by replacing the distance function in the definition of the regular Voronoi cell with the time it takes for a robot to get to a point when driving at a constant, maximum speed. The resulting bisector geometry delineating the cells belonging to neighboring robots in a MW Voronoi diagram is discussed. In Section III, a controller that drives the robots to a locally optimal configuration for minimizing the new cost function is derived. Section IV demonstrates the experimental results, while Section V concludes the paper.

II. MULTIPLICATIVELY WEIGHTED VORONOI DIAGRAMS

Consider a convex domain $\mathcal{D} \in \mathbb{R}^2$ that is to be covered by a team of N robots. Let $p_i \in \mathcal{D}$ be the position of the i^{th} robot and $p = [p_1^\top, p_2^\top, \cdots, p_N^\top]^\top \in \mathbb{R}^{2N}$ be the position vector for the N robots in the system. Also, let us define $\mathcal{N}$ as the set of indices of robots, i.e., $\{1, \cdots, N\}$, and denote $i, j \in \mathcal{N}$. The standard Voronoi cell of the i^{th} robot is defined as

$$V_i(p) = \{q \in \mathcal{D} \mid d(q, p_i) \leq d(q, p_j), \forall j \neq i\} \qquad (1)$$

where $d(q, p_i)$ is the distance between the point $q \in \mathcal{D}$ and p_i. Typically, this is just the Euclidean distance, which is also what will be adhered to for the remainder of the paper. Correspondingly, the Voronoi cell $V_i(p)$ represents all points q that are closer to Robot i than to all other robots. An example of the Voronoi diagram consisting of these Voronoi cells is shown in Fig. 1.

Now, if the point q is a point that may need to be visited by the robot as quickly as possible, e.g., this might be the location of a fire that must be put out, one can define a different Voronoi cell

$$V_{t,i}(p) = \{q \in \mathcal{D} \mid t(q, p_i) \leq t(q, p_j), \forall j \neq i\}, \qquad (2)$$

where $t(q, p_i)$ is the time of travel for the i^{th} robot to get to point q. As prescribed, (2) represents all points q that can be reached faster by Robot i than by all other robots.

Additionally, as the events must be serviced urgently, as is the case in the fire scenario, the travel to that point should happen at maximum speed. Assuming the robots are covering relatively large areas compared to their mobility, e.g., similar to a fire station covering a whole town or district, which leads to a sparse formation of the robots, the region that can be reached by a robot in the shortest amount of time can be accurately approximated only with the maximum speed of the robot, i.e., other mobility aspects such as acceleration have virtually no influence in determining the region of dominance. Therefore we can let

$$t(q, p_i) = \frac{d(q, p_i)}{v_i},$$

where $v_i > 0$ is the maximum speed of the i^{th} robot, i.e.,

$$V_{\text{MW},i}(p, v) = \left\{ q \in \mathcal{D} \;\middle|\; \frac{d(q, p_i)}{v_i} \leq \frac{d(q, p_j)}{v_j}, \forall j \neq i \right\} \qquad (3)$$

which, by definition, is a multiplicatively weighted (MW) Voronoi cell. Here, $v = [\frac{1}{v_1}, \frac{1}{v_2}, \cdots, \frac{1}{v_N}]^\top$ is the vector containing the inverse of the maximum speeds of the robots. An example of the Voronoi diagram consisting of these cells is shown in Fig. 2. From the definition (3), if all robots have the same maximum speed, i.e., $v_i = v_j, \forall i, j$, we get a standard Voronoi diagram as per (1).

Unlike the straight line bisectors that delineate regions in regular Voronoi diagrams, the bisector between two neighboring robots in a multiplicatively weighted Voronoi diagram is a circle, and we state this fact as a lemma.

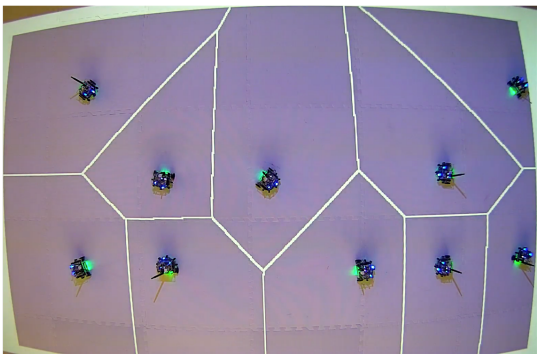


Fig. 1. A standard Voronoi diagram for a team of 10 robots obtained with the Euclidean distance metric as defined in (1).

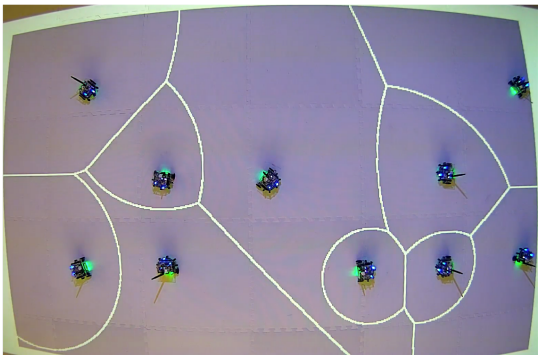


Fig. 2. A multiplicatively weighted Voronoi diagram for a team of 10 robots obtained with the time metric as defined in (3). For comparison purposes, the placement of the robots in the domain is identical to the one presented in Fig. 1 for the standard Voronoi diagram. Here, $v_{i,max} \neq v_{j,max}$, for some robots i and j, which results in a different geometry for the Voronoi cells.

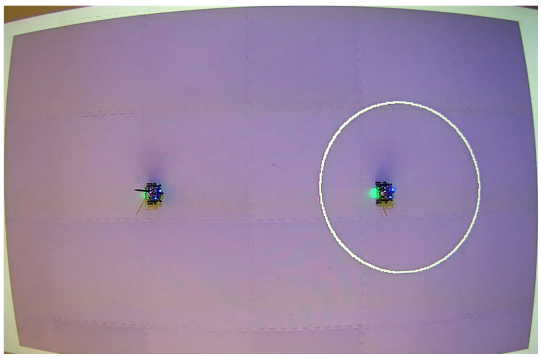


Fig. 3. A multiplicatively weighted Voronoi diagram calculated for 2 robots. Here, the ratio of the maximum speeds between the robot on the left and the robot on the right is 3:1. From the figure, it can be identified that the area enclosed by the circular boundary belongs to the slower robot on the right.

Lemma 1 (Bisector of a MW Voronoi diagram)**.** *The bisector between two robots with different maximum speeds is a circular segment, given by*

$$b_{ij} = \left\{ q \in \mathcal{D} \;\middle|\; \left\| q - \frac{v_j^2 p_i - v_i^2 p_j}{v_j^2 - v_i^2} \right\| = \frac{v_i v_j \|p_i - p_j\|}{v_j^2 - v_i^2} \right\}, \tag{4}$$

where $\frac{v_j^2 p_i - v_i^2 p_j}{v_j^2 - v_i^2}$ *is the center of the circle and* $\frac{v_i v_j \|p_i - p_j\|}{v_j^2 - v_i^2}$*, its radius [16]. It is assumed that* $v_j \geq v_i$ *without loss of generality.*

The lemma can be proved by manipulating the boundary definition of the multiplicatively weighted Voronoi cells between i and j, given by the points $q \in \mathcal{D}$ that satisfy

$$\frac{\|q - p_i\|}{v_i} = \frac{\|q - p_j\|}{v_j}.$$

Note that in case of $v_i = v_j$, the boundary between the two robots becomes a circle with infinite radius, which is a line, as shown in [16].

In a multiplicatively weighted Voronoi diagram with two robots with different maximum speeds, the slower robot is always inside the circular bisector defined in (4), and therefore, the region inside the circular boundary is the region of dominance belonging to the slower robot. We also state this as a lemma.

Lemma 2 (The MW Voronoi cell of the slower robot)**.** *In a MW Voronoi diagram constructed with two robots that have different maximum speeds, the MW Voronoi cell of the slower robot is the area enclosed by a circular boundary and possibly the boundaries of the coverage domain as shown in Figure 3.*

Proof. Note the right hand side of (4) is the radius of the circular boundary between the two robots. In order to prove that the circular region belongs to the slower robot, we need to prove the distance from the center of the circle to the position of the slower robot, p_i, is always less than or equal to the radius of the circle as described in Fig. 4. This condition can be represented with the inequality

$$\left\| p_i - \frac{v_j^2 p_i - v_i^2 p_j}{v_j^2 - v_i^2} \right\| \leq \frac{v_i v_j \|p_i - p_j\|}{v_j^2 - v_i^2}. \tag{5}$$

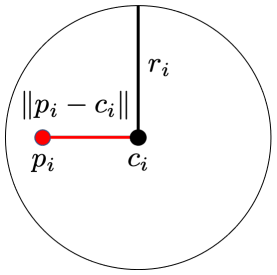


Fig. 4. Visualization of the distance from the center of the circle, c_i, to the position of Robot i, p_i, and the radius of the circle, r_i. Here, $c_i = \frac{v_j^2 p_i - v_i^2 p_j}{v_j^2 - v_i^2}$ and $r_i = \frac{v_i v_j \|p_i - p_j\|}{v_j^2 - v_i^2}$.

Simplifying the expression inside the norm on the left hand side of (5) yields

$$\frac{v_i^2 \|p_i - p_j\|}{v_j^2 - v_i^2} \leq \frac{v_i v_j \|p_i - p_j\|}{v_j^2 - v_i^2}. \tag{6}$$

Since we previously defined $v_i \leq v_j$ without loss of generality, (6) holds true. Therefore, the area enclosed by the circular boundary is the multiplicatively weighted Voronoi cell of the slower robot. □

III. TEMPORAL COST FUNCTIONS AND GRADIENT DESCENT ALGORITHMS

The main objective of most coverage control problems is to spread out robots across a domain so as to make them able to effectively detect events/phenomena that are expected to occur. Therefore, in such scenarios, the robots are distributed such that the spatial coverage over the domain is maximized given the probability density of the events/phenomena over the domain. Accordingly, the cost function of the multi-robot system to be minimized is typically given by the locational cost

$$\mathcal{H}(p) = \sum_{i=1}^{N} \int_{V_i(p)} \|q - p_i\|^2 \phi(q) dq, \tag{7}$$

where $\phi : \mathcal{D} \to \mathbb{R}_{\geq 0}$ is the density function which describes the probability of an event/phenomenon occurring at the point q, [2].

On the other hand, if the objective of the coverage is rapid servicing of events, e.g., for the fire truck placing problem mentioned in the introduction, the robots need to be distributed over the domain such that the temporal coverage over the domain is maximized. In other words, the time for robots to reach any weighted points in their region of dominance needs to be minimized. The new cost function that reflects this ambition needs to take the different maximum speeds into account as

$$\mathcal{H}_v(p) = \sum_{i=1}^{N} \int_{V_{\mathrm{MW},i}(p,v)} \frac{\|q - p_i\|^2}{v_i^2} \phi(q) dq, \tag{8}$$

where v is the vector containing the inverse of maximum speeds of the robots, and $V_{\mathrm{MW},i}(p, v)$ is the multiplicatively weighted Voronoi cell of the Robot i with gain $\frac{1}{v_i}$. This cost can be interpreted as the sum (in an integral sense) of all

times squared needed for the robots to get to every point with a weighted distance in their Voronoi cells. We denote this new cost as a *temporal cost* (as opposed to a locational cost).

In order to make the robots maximize the temporal coverage over the domain, the temporal cost in (8) needs to be minimized. This can be achieved by letting the robots follow against the gradient of the cost with respect to their positions, which eventually makes the robots converge to local minimizers. To this end, the gradient of the cost with respect to an arbitrary i^{th} robot needs to be computed. Since the temporal cost can be seen as the locational cost in (7) with MW distance functions and MW Voronoi cells, we use a generalized MW locational cost function for computing the gradient to provide a more generalized gradient analysis. The function is defined as

$$\mathcal{H}_g(p) = \sum_{i=1}^{N} \int_{V_{\text{MW},i}(p,\alpha)} \beta_i \|q - p_i\|^2 \phi_i(q) dq, \tag{9}$$

where $\alpha = [\alpha_1, \cdots, \alpha_n]^\top$ is the vector containing the multiplivative weights of the robots for computing their Voronoi cells with $\alpha_i \in \mathbb{R}_{>0}\ \forall i$, and $\beta_i \in \mathbb{R}_{>0}\ \forall i$ is the multiplicative weight on the distance function of the Robot i. Also, each robot may be equipped with different types of sensors, which leads to the robot specific density function, $\phi_i(q)$.

The computation of the gradient of (9) with respect to p_i involves derivatives of integrals. Therefore, Leibniz integral rule must be applied.

Lemma 3. *(Leibniz Integral Rule [17]) Let $\Omega(p)$ be a region that depends smoothly on p such that the unit outward normal vector $n(p)$ is uniquely defined almost everywhere on the boundary $\partial\Omega(p)$. Let*

$$F = \int_{\Omega(p)} f(p,q) dq.$$

Then,

$$\frac{\partial F}{\partial p} = \int_{\Omega(p)} \frac{\partial}{\partial p} f(p,q) dq + \int_{\partial\Omega(p)} f(p,q) n(q)^\top \frac{\partial q}{\partial p} dq.$$

Using Lemma 3, we obtain the following theorem.

Theorem 1. *Given the cost function in the form of* (9)*, the gradient of the cost with respect to the position of Robot i is given as*

$$\begin{aligned}\frac{\partial \mathcal{H}_g(p)}{\partial p_i} &= 2\beta_i m_{\alpha,i}(p)(p_i - c_{\alpha,i}(p))^\top \\ &+ \sum_{j\in\mathcal{N}_i} \int_{\partial V_{MW,ij}} \frac{\alpha_i^2 (q-p_i)^\top}{\|\alpha_i^2(q-p_i) - \alpha_j^2(q-p_j)\|} \\ &\quad (\beta_i \|q-p_i\|^2 \phi_i(q) - \beta_j \|q-p_j\|^2 \phi_j(q)) dq\end{aligned} \tag{10}$$

where $\partial V_{MW,ij}$ is the boundary between robots i and j, and

$$m_{\alpha,i}(p) = \int_{V_{MW,i}} \phi_i(p) dq, \quad c_{\alpha,i}(p) = \frac{\int_{V_{MW,i}} q\phi_i(p) dq}{m_{\alpha,i}(p)}.$$

are the mass and the center of mass of the MW Voronoi cell of the i^{th} robot, respectively. Also, the dependence of p and α on the MW Voronoi cells and the boundaries between them was omitted for notational convenience.

Proof. The cost in Equation (9) can be divided into three terms as

$$\begin{aligned}\mathcal{H}_g(p) &= \int_{V_{\text{MW},i}} \beta_i \|q-p_i\|^2 \phi_i(q) dq \\ &+ \sum_{j\in\mathcal{N}_i} \int_{V_{\text{MW},j}} \beta_j \|q-p_j\|^2 \phi_j(q) dq \\ &+ \sum_{j\notin\mathcal{N}_i\cup\{i\}} \int_{V_{\text{MW},j}} \beta_j \|q-p_j\|^2 \phi_j(q) dq\end{aligned} \tag{11}$$

where $\mathcal{N}_i$ is the set containing the indices of the Delaunay neighbors of Robot i. By Delaunay neighbors, we refer to those robots that share a boundary in the Voronoi partition. Then, taking partial derivative of the above expression with respect to p_i gives

$$\begin{aligned}\frac{\partial \mathcal{H}_g(p)}{\partial p_i} &= \frac{\partial}{\partial p_i}\left(\int_{V_{\text{MW},i}} \beta_i \|q-p_i\|^2 \phi_i(q) dq\right) \\ &+ \frac{\partial}{\partial p_i}\left(\sum_{j\in\mathcal{N}_i} \int_{V_{\text{MW},j}} \beta_j \|q-p_j\|^2 \phi_j(q) dq\right)\end{aligned} \tag{12}$$

where the partial derivative of the last term in (11) with respect to p_i is 0 because the term does not depend on p_i. Applying Lemma 3 to (12), and using the relationship that $n_i(q) = -n_j(q)$ along the boundary between neighboring robots i and j, we get

$$\begin{aligned}\frac{\partial \mathcal{H}_g(p)}{\partial p_i} &= 2\beta_i \int_{V_{\text{MW},i}} (p_i - q)^\top \phi_i(q) dq \\ &+ \sum_{j\in\mathcal{N}_i} \int_{\partial V_{\text{MW},ij}} \beta_i \|q-p_i\|^2 n_i(q)^\top \frac{\partial q}{\partial p_i} \phi_i(q) dq \\ &- \sum_{j\in\mathcal{N}_i} \int_{\partial V_{\text{MW},ij}} \beta_j \|q-p_j\|^2 n_i(q)^\top \frac{\partial q}{\partial p_i} \phi_j(q) dq.\end{aligned} \tag{13}$$

where $\partial V_{\text{MW},ij}$ is the boundary between robots i and j.

Now, we need to simplify the term, $n_i(q)^\top \frac{\partial q}{\partial p_i}$. First, let q on the boundary between Robot i and j, $\partial V_{\text{MW},ij}$, be parameterized with respect to the arclength parameter s, i.e., $q(s, p_i, p_j)$. From the definition of the multiplicatively weighted Voronoi cell in (3), the boundary between two MW Voronoi cells can be represented by all points q satisfying the condition,

$$\alpha_i \|q-p_i\| = \alpha_j \|q-p_j\|. \tag{14}$$

Squaring both sides of (14) and taking partial derivative of both sides with respect to the arclength parameter s gives

$$(\alpha_i^2(q-p_i) - \alpha_j^2(q-p_j))^\top \frac{\partial q}{\partial s} = 0.$$

Note that $\frac{\partial q}{\partial s}$ is the tangential vector of the boundary. Therefore,

$$\alpha_i^2(q-p_i) - \alpha_j^2(q-p_j) \tag{15}$$

is the normal vector of the boundary.

In order to utilize Lemma 3, we need to know if the vector (15) is outward normal vector or inward normal vector. To

check this, first let $\alpha_i = \frac{1}{v_i}$ and $\alpha_j = \frac{1}{v_j}$. Substituting α_i and α_j into (15) and simplifying the expression yield

$$\frac{v_j^2 - v_i^2}{v_i^2 v_j^2}\left(q - \frac{v_j^2 p_i - v_i^2 p_j}{v_j^2 - v_i^2}\right). \tag{16}$$

Note that $\frac{v_j^2 p_i - v_i^2 p_j}{v_j^2 - v_i^2}$ is the center of the circular boundary between Robot i and j in (4). Therefore, the vector term inside the parentheses in (16) is pointing towards any point q on the circular boundary from the center of the circle. If $v_i < v_j$, this direction does not change. According to Lemma 2, the MW Voronoi cell of the slower robot i is enclosed by the circular boundary. Therefore, in this case, (16) is the outward normal vector of Robot i on the boundary between Robots i and j. On the other hand, if $v_i > v_j$, the direction of the vector (16) is from the boundary of the circle towards its center. However, since Robot i is the faster robot in this case, the normal vector is again the outward normal vector of Robot i along the boundaries between Robot i and j. Therefore, (15), which is identical to (16), is always the outward normal vector of the MW Voronoi cell of Robot i. Accordingly, the unit outward normal vector of the MW Voronoi cell of Robot i on the boundary, $n_i(q)$, is given as

$$n_i(q) = \frac{\alpha_i^2(q - p_i) - \alpha_j^2(q - p_j)}{\|\alpha_i^2(q - p_i) - \alpha_j^2(q - p_j)\|}. \tag{17}$$

Next, taking the partial derivative of both sides of the squared terms of (14) with respect to p_i yields

$$(\alpha_i^2(q - p_i) - \alpha_j^2(q - p_j))^\top \frac{\partial q}{\partial p_i} = \alpha_i^2(q - p_i)^\top. \tag{18}$$

Finally, dividing both sides of (18) by $\|\alpha_i^2(q-p_i)-\alpha_j^2(q-p_j)\|$ and substituting (17) into the equation gives

$$n_i(q)^\top \frac{\partial q}{\partial p_i} = \frac{\alpha_i^2(q - p_i)^\top}{\|\alpha_i^2(q - p_i) - \alpha_j^2(q - p_j)\|}. \tag{19}$$

In case of $\alpha_i = \alpha_j$, which implies the boundary between Robot i and j is a straight line, (17) and (19) still hold, and the results are consistent to what were shown in [8]. Plugging (19) back into (13) and simplifying the boundary terms give the generalized gradient expression in (10). □

Corollary 1. *In case of the cost function in* (9) *becoming a temporal cost as in* (8) *with* $\alpha_i^2 = \frac{1}{\beta_i} = v_i^2,\ \forall i \in \mathcal{N}$, *the generalized gradient expression in* (10) *is simplified to*

$$\frac{\partial \mathcal{H}_v(p)}{\partial p_i} = \frac{2}{v_i^2} m_{v,i}(p)(p_i - c_{v,i}(p))^\top.$$

where $m_{v,i}(p)$ *and* $c_{v,i}(p)$ *are the mass and the center of mass of the* i^{th} *cell in the MW Voronoi diagram with weights* $v = [\frac{1}{v_1}, \cdots, \frac{1}{v_N}]^\top$.

Corollary 2. *In case of the cost function in* (9) *becoming a locational cost with* $\alpha_i = \beta_i = 1$ *and* $\phi_i(q) = \phi(q),\ \forall i \in \mathcal{N}$, *the gradient expression in* (10) *is reduced to*

$$\frac{\partial \mathcal{H}(p)}{\partial p_i} = 2m_{\vec{1},i}(p)(p_i - c_{\vec{1},i}(p))^\top$$

as derived in [2], where $\vec{1}$ *is a column vector of ones. Note that* $m_{\vec{1},i}(p)$ *and* $c_{\vec{1},i}(p)$ *are the mass and the center of mass of the standard Voronoi cell of Robot* i.

Corollary 3. *In case of the cost function in* (9) *with* $\alpha_i = \beta_i = 1, \forall i \in \mathcal{N}$, *and if each robot has a different density function,* $\phi_i(q)$, (10) *simplifies to*

$$\frac{\partial \mathcal{H}_\mathcal{C}(p)}{\partial p_i} = 2m_{\vec{1},i}(p)(p_i - c_{\vec{1},i}(p))^\top + \sum_{j \in \mathcal{N}_i} \int_{\partial V_{ij}} (q - p_i)^\top \frac{\|q - p_i\|^2}{\|p_j - p_i\|}(\phi_i(q) - \phi_j(q))dq$$

where $\vec{1}$ *is a column vector of ones, and* $m_{\vec{1},i}(p)$ *and* $c_{\vec{1},i}(p)$ *are the mass and the center of mass of the standard Voronoi cell of Robot* i. *Also,* ∂V_{ij} *is the boundary between standard Voronoi cells of Robot* i *and* j. *This result corresponds to the gradient of the heterogeneous locational cost for coordination derived in [8].*

Since our goal in this paper is to minimize a temporal cost, we focus on Corollary 1. Assuming single integrator dynamics that $\dot{p}_i = u_i$, the gradient descent controller of the i^{th} robot for the temporal cost, $\mathcal{H}_v(p)$, is given as

$$\dot{p}_i = -\frac{2}{v_i^2} m_{v,i}(p)(p_i - c_{v,i}(p)). \tag{20}$$

We can now scale this with any positive scalar to get a controller that is simpler to express and manage. At the same time, although we are not interested in having the robots move at maximum speed at this point—the fire trucks are spreading out waiting for a fire as opposed to moving at full speed to put out a fire—we do need to make sure that no robot is required to go faster than its maximum speed during this phase of the deployment.

To this end, we can multiply the controller in (20) by a gain, $\gamma_i(p, v_i) = \frac{v_i^2}{2m_{v,i}(p)}\hat{k}_i > 0$, which gives the new update law

$$\dot{p}_i = -\hat{k}_i(p_i - c_{v,i}(p)), \tag{21}$$

which is akin to Lloyd's algorithm. Here, $\hat{k}_i$ is selected to be a saturation function that limits the speed of each robot to its maximum value, i.e.,

$$\hat{k}_i(p, v_i, k_i) = \min\left(\frac{v_i}{k_i\|p_i - c_{v,i}(p)\|}, 1\right) k_i, \tag{22}$$

where $k_i > 0$ is the gain of the i^{th} robot before saturation.

Proposition 1. *Under the controller in* (21), *robots are asymptotically stable and converge to a stationary configuration.*

Proof. The time derivative of the temporal cost is

$$\frac{d\mathcal{H}_v(p)}{dt} = -\sum_{i=1}^{N} \gamma_i(p, v_i) \left\| \frac{\partial \mathcal{H}_v(p)}{\partial p_i}^\top \right\|^2$$

which is negative definite. Since the set of p that makes the time derivative be 0 is equal to the equilibrium of the system, $p_i = c_{v,i}(p),\ \forall i \in \mathcal{N}$, according to LaSalle's invariance principle, p_i converges to $c_{v,i}$ for all $i \in \mathcal{N}$ as $t \to \infty$. □

The new update law is active during the first mobility phase where robots (or fire trucks, for example) distribute themselves over a domain, and this does not have to be done quickly. Therefore, the asymptotic convergence property does not impede the rapid servicing of events, which will take place in the second mobility phase.

IV. EXPERIMENTAL RESULTS

In the previous section, we derived the controller for the multi-robot team for the first mobility phase. The proposed controller allows the robots to achieve a configuration that minimizes the temporal cost of the system. In order to demonstrate the effectiveness of this configuration for rapidly servicing events which take place in the second mobility phase (e.g., a fire truck moving at its maximum speed to put out a fire), we evaluate the performance of a multi-robot team consisting of 10 robots under three different coverage scenarios at the Robotarium [18].

To this end, the total number of 100, 1,000, 10,000, and 100,000 target points are randomly generated from a density function $\phi(q)$. Then, the average time for the robot team to capture a target (i.e., the average time for a robot in the team to reach a target within its region of dominance) is calculated using the maximum speeds of the robots for each scenario. In the first scenario, the robots achieve a coverage configuration using Lloyd's algorithm as in [2]. In the second scenario, the robots use the temporal cost gradient descent controller in (21). In the third scenario, the robots use the power cost gradient descent controller in [4] where the cost function is defined as

$$\mathcal{H}_{\mathrm{p}}(p) = \sum_{i=1}^{N} \int_{V_{\mathrm{p},i}(p)} \left(\|q - p_i\|^2 - v_i^2 \right) \phi(q) dq$$

where $V_{\mathrm{p},i}(p)$ is (1) with the distance function, $d_{\mathrm{p},i}(q, p_i) = \|q - p_i\|^2 - v_i^2$. The final spatial configuration obtained in each scenario for the same initial condition is depicted in Fig. 5. In the same figure, the density function used for the experiments is described as gradation of colors. In order to prevent possible inter-agent collisions, the built-in control barrier function in the Robotarium [18] was used during the first mobility phase.

The ratios of maximum speed between robots used in the experiments are described by a vector, $r = [2, 3, 1, 1, 2, 2, 3, 2, 2, 3]$ where r_i represents the i^{th} element in the vector. The actual maximum speed of a robot was set to $v_i = 0.15 r_i \min(r')$ where $r' = \{\frac{1}{r_1}, \cdots, \frac{1}{r_{10}}\}$. In order to simulate an environment where the robots are covering a relatively large area compared to their mobility as mentioned in Section II, the gains of the robots before saturation in (22) were set to $k_i = r_i \min(r'), \forall i$ so that they are proportional to the maximum speeds of the robots.

The rectangular boundaries of the Robotarium are defined as [-1.6, 1.6] for the X-coordinate and [-1, 1] for the Y-coordinate (all the dimensions in meters), where the origin is located at the center of the rectangle. The MW Voronoi cells of the robots were computed numerically by applying (3) to a 400×640 pixel grid over this rectangular domain. The density function used for the experiment is the sum of two bivariate normal distributions defined as

$$\phi(q) = \frac{1}{4\pi\sqrt{|\Sigma|}} \sum_{i=1}^{2} \exp\left(-\frac{1}{2}(q - \mu_i)^\top \Sigma^{-1} (q - \mu_i) \right) \tag{23}$$

where μ_i is the mean of the distribution and Σ, the covariance matrix. Specifically, the mean and the covariance are chosen as $\mu_1 = [0.8, 0.8]^\top$, $\mu_2 = [-0.8, -0.8]^\top$ and $\Sigma = 0.3I$ where I is the 2×2 identity matrix. From this distribution, 100, 1,000, 10,000, and 100,000 target position vectors were randomly generated. Any target positions obtained from the distribution that exceeded the Robotarium boundary were discarded and redrawn.

TABLE I
AVERAGE TIME TO REACH A TARGET FOR 100 TARGETS RANDOMLY GENERATED FROM (23)

Case	Lloyd	Temporal Cost G.D.	Power Cost G.D.
1	1.8500(s)	1.5213(s)	1.7347(s)
2	1.6520(s)	1.5193(s)	1.4247(s)
3	1.7073(s)	1.5047(s)	1.5327(s)

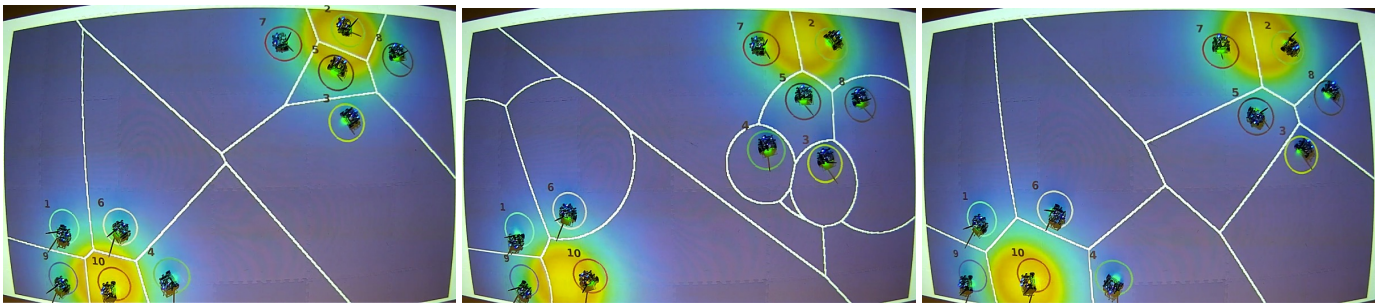
TABLE II
AVERAGE TIME TO REACH A TARGET FOR DIFFERENT NUMBER OF TARGETS RANDOMLY GENERATED FROM (23)

nTargets	Lloyd	Temporal Cost G.D.	Power Cost G.D.
1,000	1.7767(s)	1.4973(s)	1.5353(s)
10,000	1.7567(s)	1.5227(s)	1.5593(s)
100,000	1.7687(s)	1.5333(s)	1.5747(s)

In order to check that the configuration obtained using the proposed controller yields the lowest average time to reach a target, the average times to reach a target for different number of target points were computed for each scenario. For the 100 targets case, since 100 is a relatively low number to accurately describe a density function, 3 different sets of samples were used as shown in Fig. 6 to check for cases where the targets do not accurately describe the density function $\phi(q)$, which may allow other coverage formations perform better than the proposed one. For Case 2 in Fig. 6, power cost gradient descent performed better than the temporal cost gradient descent as can be confirmed in TABLE I. However, from TABLE II, it can be identified that the coverage formation achieved by the proposed temporal cost gradient descent controller consistently outperforms others with larger number of targets which well describe the density.

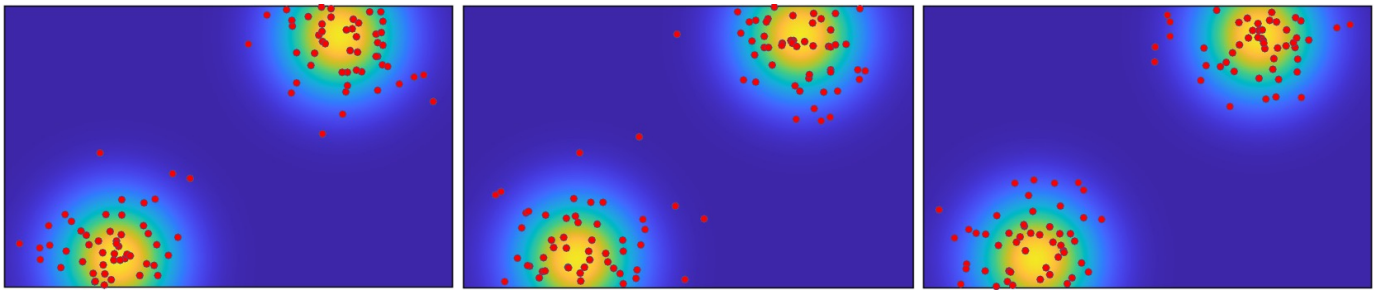
V. CONCLUSIONS

In this paper, we presented a coverage control strategy for a team of robots with different maximum speeds. Based on the idea that robots with different mobility travel different distances within the same amount of time, the region of dominance of a robot is approximated to the set of points where the robot can arrive faster than any other robots in the team by cruising at its maximum speed. Also, the cost for a robot to cover a point is defined as the time needed for the robot to reach the point. In order to minimize the resulting temporal cost function of the system, a gradient



(a) Lloyd (b) Temporal Cost Gradient Descent (c) Power Cost Gradient Descent

Fig. 5. The final coverage formations of the robots using 3 different algorithms. The density function is represented as gradation of colors where colors close to yellow indicate higher density, and colors close to blue indicate lower density. The initial configuration of the robots for all cases is described in Fig. 1 and Fig 2. The maximum speed ratio between robots are $r = [2, 3, 1, 1, 2, 2, 3, 2, 2, 3]$.



(a) Case 1 (b) Case 2 (c) Case 3

Fig. 6. Three different sets of 100 target points randomly generated from the density function in (23). Each target point is shown as a red dot.

descent controller was used to drive robots to local minima. For the gradient computation, we derived a theorem that defines the gradient of the generalized MW locational cost. According to the theorem, the gradient of the temporal cost in this paper and the gradient of the locational cost in two different forms in the literature can be considered as special cases of the theorem. The theoretical and experimental results suggest that the coverage strategy proposed in this work can be effective in scenarios where a group of robots is collaborating with each other to rapidly respond to an expected event over a domain.

REFERENCES

[1] J. Cortes and M. Egerstedt, "Coordinated control of multi-robot systems: A survey," *SICE Journal of Control, Measurement, and System Integration*, vol. 10, no. 6, pp. 495–503, 2017.

[2] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, pp. 243–255, 2004.

[3] S. Lloyd, "Least squares quantization in pcm," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, 1982.

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

[5] A. Pierson, L. C. Figueiredo, L. C. Pimenta, and M. Schwager, "Adapting to sensing and actuation variations in multi-robot coverage," *The International Journal of Robotics Research*, vol. 36, no. 3, pp. 337–354, 2017.

[6] H. Mahboubi, K. Moezzi, A. G. Aghdam, K. Sayrafian-Pour, and V. Marbukh, "Self-deployment algorithms for coverage problem in a network of mobile sensors with unidentical sensing ranges," in *2010 IEEE Global Telecommunications Conference GLOBECOM 2010*, 2010, pp. 1–6.

[7] H. Mahboubi, J. Habibi, A. G. Aghdam, and K. Sayrafian-Pour, "Distributed coverage optimization in a network of static and mobile sensors," in *2013 American Control Conference*, 2013, pp. 6877–6881.

[8] M. Santos, Y. Diaz-Mercado, and M. Egerstedt, "Coverage control for multirobot teams with heterogeneous sensing capabilities," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 919–925, 2018.

[9] M. Santos and M. Egerstedt, "Coverage control for multi-robot teams with heterogeneous sensing capabilities using limited communications," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 5313–5319.

[10] A. Sadeghi and S. L. Smith, "Coverage control for multiple event types with heterogeneous robots," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 3377–3383.

[11] B. Reily, T. Mott, and H. Zhang, "Adaptation to team composition changes for heterogeneous multi-robot sensor coverage," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 9051–9057.

[12] A. Pierson, L. C. Figueiredo, L. C. A. Pimenta, and M. Schwager, "Adapting to performance variations in multi-robot coverage," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 415–420.

[13] A. Kwok and S. Martinez, "Deployment algorithms for a power-constrained mobile sensor network," in *2008 IEEE International Conference on Robotics and Automation*, 2008, pp. 140–145.

[14] F. Aurenhammer, "Power diagrams: Properties, algorithms and applications," *SIAM Journal on Computing*, vol. 16, no. 1, pp. 78–96, 1987.

[15] H. Mahboubi, F. Sharifi, A. G. Aghdam, and Y. Zhang, "Distributed coordination of multi-agent systems for coverage problem in presence of obstacles," in *2012 American Control Conference (ACC)*, 2012, pp. 5252–5257.

[16] A. Okabe, B. Boots, K. Sugihara, and S. N. Chiu, *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*. New York, NY, USA: John Wiley & Sons, Inc., 2000.

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

[18] S. Wilson, P. Glotfelter, L. Wang, S. Mayya, G. Notomista, M. Mote, and M. Egerstedt, "The robotarium: Globally impactful opportunities, challenges, and lessons learned in remote-access, distributed control of multirobot systems," *IEEE Control Systems Magazine*, vol. 40, no. 1, pp. 26–44, 2020.