Abstract—This paper presents a decentralized, hybrid control approach for persistent area coverage control applications. The hybrid cooperative control framework builds on the concepts of cellular automata and awareness-based model to guarantee safe coverage of a large-scale domain by a network of mobile wireless sensors with bounded input disturbances and limited communication and sensing ranges. The framework only requires communication among nearby agents at discrete intervals of time, reducing the communication among agents. Results are validated through simulation and experiments with a team of ten differential-drive mobile robots.

I. INTRODUCTION

One of the most common applications for Mobile Wireless Sensor Networks (MWSNs) is coverage control, in which a given geographical area (or task domain) needs to be covered or monitored in a finite amount of time. Examples of applications include military tactics [1], public safety [2], and environmental sampling [3].

Methods in coverage control can be classified as static or dynamic [4]. The former refers to the use of static sensors and is considered to be a localization optimization problem. The goal in static coverage control is to determine the optimal sensor placement given a map of the area to be covered along with some density function characterizing different regions of importance within the task domain [5]. Some algorithms assume limited sensor mobility and use this property to optimize their location after initial deployment by applying gradient-based methods (such as artificial potential fields or virtual forces [6]) and Voronoi cells [7]. Changes in information about the task-domain can be addressed by considering time-varying density functions and developing distributed control laws capable to re-arrange the configuration of the MWSN as information changes [8], [9]. These methods, however, assume that the area is small-scale and can be covered by the union of the agents’ sensory domains.

If the area is large-scale and cannot be effectively covered by a fixed number of stationary sensors, the agents may exploit mobility to gradually cover the entire task domain. In dynamic coverage, sensors are able to continuously move and, therefore, can visit any point in the task domain at some instance of time. Several methods have been proposed for dynamic coverage. In [10] a rule-based algorithm is employed that guarantees coverage of a bounded area. In [11] a gradient-based approach is presented that achieves coverage of an area while avoiding collisions among agents. These methods, however, assume that the information to be collected is static, implying that a point does not need to be visited more than once. If, instead, properties of the task domain change over time, the MWSN may lose awareness. By loss of awareness it is meant that information collected about a particular point decreases in accuracy with time [12]. Different control methods have been developed to guarantee that every point is visited persistently by at least one sensor in order to maintain a desired level of awareness [13]–[16]. In [12], a coverage model with known loss of awareness rate is proposed to capture the loss of information over time. Based on this concept, [15], [17], [18] developed methodologies that guarantee a prescribed optimal level of coverage after some finite time. In [19], this concept is extended to unknown awareness loss rate. Similar to [12], the authors in [19] propose a control framework that switches between two continuous-time controllers. One controller guarantees that the agent gathers enough information within a small region, whereas the second controller drives the agent through different neighborhoods of the task-domain. Other similar examples using a switching control strategy but without loss of awareness include [20], [21].

This paper presents a novel decentralized cooperative control strategy for safe, persistent coverage control that exploits the hybrid nature of MWSN, that is, the coordination of continuous dynamical agents interconnected via discrete communication resources and planning mechanisms. The control frameworks is comprised of a decentralized discrete cooperative controller and a local continuous trajectory tracking control. The discrete cooperative controller is based on the concept of cellular automata [22], [23] and is tasked with navigating the agents through different neighborhoods (i.e., also called cells) of the task domain. In contrast to most of the aforementioned work, the proposed discrete controller is completely decentralized requiring agents to communicate at discrete intervals of time and within a bounded communication range, reducing the dependency of communication resources. The local continuous trajectory tracking controller is in charge of safely driving the agent between neighborhoods and only requires local information. The synthesis of both controllers guarantees: 1) global awareness-based coverage of a large-scale task domain despite inter-agent communication constraints and limited sensing; and 2) collision avoidance among agents at all time. The performance of the hybrid control framework is validated via simulations with teams of agents with double-integrator dynamics and via experiments with a team of ten differential-drive robots.

This work was supported in part by the U.S. Office of Naval Research Grant No. N0001420WX00651.

E. J. Rodríguez-Seda is with the Department of Weapons, Robotics, and Control Engineering at the United States Naval Academy, Annapolis, MD. Y. Diaz-Mercado is with the Department of Mechanical Engineering at the University of Maryland, College Park, MD. rodrigue@usna.edu, yancy@umd.edu
II. PROBLEM FORMULATION

Consider a MWSN comprised of $N$ mobile sensors (also termed as agents) with dynamics given by
\[
\dot{q}_i(t) = u_i(t) + v_i(t), \quad \forall \ i \in \mathbb{I} := \{1, \cdots, N\}, \ t \geq 0 \tag{1}
\]
where $q_i \in \mathbb{R}^2$, $u_i \in \mathbb{R}^2$, and $v_i \in \mathbb{R}^2$ are the Cartesian position, the control input, and the external disturbance for the $i$th agent. It is assumed that external disturbances are radially bounded by some constant $\tilde{V} \geq 0$, i.e.,
\[
\|v_i(t)\| \leq \tilde{V}, \quad \forall \ i \in \mathbb{I}, \ t \geq 0
\]
and that the size (i.e., shape) of all agents can be bounded by a circular region of radius $r_a$.

The main task of the MWSN is to cover a large-scale domain $\mathbb{W} \subseteq \mathbb{R}^2$. It is further assumed that each agent has a bounded sensory domain given by
\[
S(q_i(t)) = \{q \in \mathbb{W} : \|q_i(t) - q\| < r_s\}
\]
where $r_s \gg r_a$ is the sensing range. Similarly, it is assumed that the agents can exchange information, such as position and coverage, with any other agent within a circular communication region of radius $R > r_s$.

A. Task-Domain

It is assumed that the task domain $\mathbb{W}$ can be covered by a tessellation (i.e., partition) $\mathbb{W}_H$ of $N_p$ regular hexagons, also known as cells, as illustrated in Fig. 1. Each cell has edges of equal size $h \in (\frac{\sqrt{3}}{2}r_a, r_a]$ and the distances from the center of the cell to any vertex and any edge are $h$ and $r = \frac{\sqrt{3}}{2}h$, respectively. These dimensions imply that an agent can physically fit inside one cell and that its sensory domain can effectively cover at least one entire cell at a time. Some further limitations and assumptions on the shape of the workspace are given in Section V.

To simplify the analysis of coverage control when working with hexagonal grids, this paper adopts a normalized skewed-axis coordinate system $(a,b)$ to identify the center of the cells. The coordinate system along with the axes is illustrated in Fig. 2. The centers of the cells in the new coordinate system $p = [a, b]^T \in \mathbb{P}$, where $\mathbb{P}$ denotes the set of partition centers, can be transformed to Cartesian coordinates $x = [x, y]^T$ by the following bijective transformation
\[
x = \Lambda p, \quad \Lambda = \begin{bmatrix} \sqrt{3}h & \frac{\sqrt{3}}{2}h \\ 0 & \frac{\sqrt{3}}{2}h \end{bmatrix}.
\]

The main advantage of using a skewed-axis coordinate system over Cartesian coordinates is that the distance in a cell $d_{c}(\cdot)$ and the length of the shortest path in Cartesian coordinates $d_{xy}(\cdot)$ when traveling from center to center, between two cells $p_i$ and $p_j$, can be easily computed using the following distance formulas
\[
d_{c}(p_i, p_j) = \max\{|a_i - a_j|, |b_i - b_j|, |a_i + b_i - a_j - b_j|\}
\]
\[
d_{xy}(p_i, p_j) = 2r \cdot d_{c}(p_i, p_j).
\]

**Remark 1.** The use of hexagonal grids to partition a 2-dimensional area has several advantages over other regular tilings, including square grids. First, hexagonal tiling is the densest way to arrange the agents circular sensory domains $S(q_i(t)) \in \mathbb{R}^2$, minimizing overlapping between two sensors. Second, the distance between the centers of any two adjacent hexagons is always $2r$, which facilitates the design of motion planning algorithms [24].

B. Awareness-based Coverage Model

Herein, we consider task domains that are much larger than the union of the agents’ sensory domains (that is, $\bigcup_{i \in \mathbb{I}} S(q_i(t)) \subset \mathbb{W}$) and which properties to be measured can change over time. To account for the evolution of the measurements and to characterize coverage in a large-scale domain, this paper adopts the awareness-based coverage concept proposed in [12] and [18]
\[
\dot{z}_i(q_i(t), p, t) = -(M(q_i(t), p) - \alpha(p))z_i(q_i(t), p, t), \tag{2}
\]
\[
z_i(q_i(t), p, 0) > 0
\]
where $z_i(q_i(t), p, t)$ characterizes how aware the $i$th agent is of events occurring at partition center $p$ at time $t$. The value $z_i(q_i(t), p, t) = 0$ implies that the $i$th sensor has full awareness of events at $p$ at time $t$, whereas $z_i(q_i(t), p, t) \rightarrow \infty$
∞ implies a lack of awareness or, alternatively, a high
coverage uncertainty. The function \( \alpha(p) \geq 0 \) \( \forall p \in P \) characterizes the awareness loss rate (i.e., the uncertainty rate) and can vary based on cell (e.g., regions that require more coverage can be given a higher loss rate value). The function \( M : \mathbb{R}^2 \times P \rightarrow \mathbb{R} \) represents the agent’s sensing capacity, that is, the awareness gain rate and is assumed to satisfy the following properties:

- \( M(q_i, p) > 0 \) \( \forall p \in \mathcal{S}(q_i) \); and
- \( M(q_i, p) = 0 \) \( \forall p \in \mathcal{S}(q_i) \).

An example of a binary function \( M \) is given in Section VI, although other common functions can be assumed [11].

C. Control Objectives

The main objective is to develop decentralized control strategies \( u_i \) for a group of mobile sensors with dynamics given by (1) that can guarantee full coverage over a large-scale domain \( \mathbb{W} \) while avoiding collisions with each other. Mathematically, one wants to design \( u_i \) such that

1. \( z_i(q_i(t), p, t) \rightarrow 0 \) as \( t \rightarrow \infty \) \( \forall i \in I, p \in P \); and
2. \( ||q_i(t) - q_j(t)|| > 2r_o \) \( \forall i, j \neq i \) and \( \forall t \geq 0 \).

III. HYBRID PERSISTENT AWARENESS-BASED COVERAGE CONTROL

To provide coverage over the entire task domain, the MWSN needs to safely and persistently visit every cell. Accordingly, this paper proposes a control strategy that guarantees that every partition center is persistently visited by at least one agent. The overall idea is to have agents moving from cell to cell based on coverage while considering safety.

The control framework for each sensor is divided in two parts: a decentralized cooperative discrete desired partition center controller and a local continuous trajectory tracking controller. The discrete controller has the role of dictating the desired cells that the \( i \)th agent needs to visit and is updated at discrete intervals of time \( T \), where \( T \) is a design parameter. The goal of the \( i \)th agent’s trajectory tracking controller is to trace a safe trajectory between the current and desired partition center and to guarantee that the \( i \)th agent remains within a bounded distance from this trajectory.

A. Cooperative Discrete Desired Partition Center Controller

The \( i \)th agent’s cooperative discrete controller is built similar to a cellular automaton. A cellular automaton is a dynamical system in which time and space are discretized and which evolution depends on local interactions [25]. It is described by the dimension of the space, a set of states for each cell, the neighborhood around each cell, and a set of transition rules. Herein, the space is the tessellation \( \mathbb{W}_H \) covering \( \mathbb{W} \), where a partition center \( p \) represents a cell. The time is discretized in instances \( \{t_0, t_1, \cdots, t_{\infty}\} \), where \( t_0 = 0 \) and \( t_{k+1} = t_k + T \). The states of a partition center \( p \) are its occupancy \( \theta_i(p, t_k) \in \{0, 1\} \), where \( \theta_i(p, t_k) = 1 \) if the cell is occupied by an agent and \( \theta_i(p, t_k) = 0 \) otherwise, and the awareness coverage \( z_i(q_i(t_k), p, t_k) \in [0, \infty) \) that the \( i \)th sensor has about \( p \). Cells outside \( \mathbb{W}_H \) are considered as occupied, i.e., \( \theta_i(p, t_k) = 1 \) \( \forall p \notin \mathbb{W}_H, i \in I, t_k \geq 0 \). Each cell is also associated with two von Neumann neighborhoods of range \( m = 1 \) and \( m = 2 \) given by

\[
\mathcal{D}_m(p) = \{ p \in \mathbb{P} : d_c(p, p) \leq m \}.
\]

The state of the neighboring cells along with a set of transition rules (to be defined next) will govern the motion of the agents.

A.1. Communication and Awareness-Based Coverage Update

It is assumed that each agent can communicate its current cell location and awareness coverage with any other agent inside its communication range. Accordingly, define the current cell location of the \( i \)th agent, also known as its cellular position, as \( s_i(t_k) \in P \), where \( s_i(t_k) = \arg\min_{p \in \mathbb{P}} ||q_i(t_k) - \alpha(p)|| \). Define \( N_i(t_k) := \{ j \in I, j \neq i : s_j \in \mathbb{D}_2(s_i(t_k)) \} \) as the set of neighbors for the \( i \)th agent within a distance of two cells. Then, it is said that the \( i \)th agent can exchange its cellular position \( s_i(t_k) \) and overall awareness coverage \( z_i(q_i(t_k), p, t_k) \forall p \in P \) at time \( t_k \) with any other agent in \( N_i(t_k) \). Therefore, one can update the \( i \)th agent’s awareness coverage \( z_i(q_i(t), p, t_k) \) between discrete intervals as

\[
\dot{z}_i(q_i(t), p, t) = -(M(q_i(t), p) - \alpha(p))z_i(q_i(t), p, t),
\]

\[
\forall t \in [t_k, t_{k+1}) \forall p \in \mathbb{W}_H
\]

\[
z_i(q_i(t_k), p, t_k) = \min_{j \in N_i(t_k)} \{ z_j(q_j(t_k), p, t_k) \}.
\]

In order to guarantee that the agents can communicate with any other agent in \( N_i(t_k) \), the following assumption is made.

Assumption 1 (Communication Range). The agents communication range is lower bounded by \( R \geq 5r - 2r_o = 5\sqrt{3}h/2 - 2r_a \).

Remark 2. Note that \( h \) is in general a design parameter dividing the space \( \mathbb{W} \) and can be made smaller or larger as long as \( \frac{3}{2}r_o < h \leq r_o \). In addition, note that communication among nearby agents only occurs at discrete intervals \( T \) and that constant communication among agents is not required.

A.2. Set of Transition Rules

Having defined the states and neighborhoods for the cellular automata-like controller, one can proceed with the transition rules. Rules 1 and 2 are based on safety, while Rules 3 and 4 are based on coverage objectives.

Rule 1. An agent can only execute one of two actions within a single time interval \( T \). It can either remain in its current cell or move to an unoccupied adjacent cell, i.e., \( s_i(t_{k+1}) = p \) only if \( p \in \mathcal{D}_1(s_i(t_k)) \) and \( \theta_i(p, t_k) = 0 \).

Rule 2. An agent cannot move to a cell that is also within one cell distance from another agent of higher priority. For two agents located at \( s_i(t_k) = [a_i, b_i]^T \) and \( s_j(t_k) = [a_j, b_j]^T \in \mathcal{D}_2(q_i(k)) \), the \( j \)th agent has a higher priority if and only if \( b_j > b_i \) or \( b_j = b_i \) and \( a_j > a_i \).

Rule 2 allows only one agent to move with the aim of avoiding collisions or deadlocks. Priority was chosen arbitrarily and other priority-based rules can be applied.
The priority region used herein is represented by the darker shades of green and yellow in Fig. 2.

**Rule 3.** If it does not contradict Rules 1 or 2, an agent should move to

\[ s_i(t_{k+1}) = \arg \min_{p \in \mathbb{D}_1(s_i(t_k))} \{d_c(s, s^d_i(t_k))\} \]

\[ s^d_i(t_k) = \arg \max_{p \in \mathbb{P}} \left\{ \frac{w_i(p, t_k)}{1 + \sigma d_c(s_i(t_k), p)} z_i(q_i(t_k), p, t_k) \right\} \]

(4a, 4b)

for some tuning parameter \( \sigma \geq 0 \). The function \( w_i(p, t_k) \) assigns a weight to each partition center based on the proximity of the \( i \)-th agent to \( p \) in comparison to other neighboring agents and is given by

\[ w_i(p, t_k) = 1 - \left( 1 - w_i(p, t_k) \right) e^{-\nu t} \]

otherwise, where \( 0 < \omega_0 < 1, w_i(p, 0) = 1 \ \forall \ p \in \mathbb{P}, \) and \( \nu > 0 \) is a control tuning parameter. If there are more than one solution to (4), pick the cell with less coverage. If the cells have the same coverage, pick the first cell clockwise.

**Rule 4.** If Rule 3 contradicts Rules 1 or 2, the agent should pick a different partition center that does not increase the distance in cells to \( s^d_i(t_k) \), i.e.,

\[ s_i(t_{k+1}) = \{s \in \mathbb{D}_1(s_i(t_k)) : d_c(s, s^d_i(t_k)) = d_c(s_i(t_k), s^d_i(t_k))\} \]

If there are more than one solution, pick the cell with less coverage. If the cells have the same coverage, pick the first cell clockwise. If there is no viable solution, stay in the same cell, i.e., \( s_i(t_{k+1}) = s_i(t_k) \).

Rule 3 drives the agent toward the partition center of least awareness. In the case that Rule 3 cannot be met, Rule 4 picks a new partition center of less coverage that does not increase the distance to \( s^d_i(t_{k+1}) \). A flowchart of the control algorithm with the transition rules is given in Fig. 3.

**Remark 3.** Note that the evolution of the discrete controller does not represent a true cellular automaton in that not all cells are updated simultaneously by local rules. Herein, only cells within the proximity of the \( i \)-th agent are updated and the new state depends on information (i.e., coverage) of the entire task space.

### B. Local Continuous Trajectory Tracking Controller

Once the discrete controller has determined the next partition center \( s_i(t_{k+1}) \in \mathbb{P} \) to be visited by the \( i \)-th agent, the control task shifts to guaranteeing a safe transition from \( s_i(t_k) \) to \( s_i(t_{k+1}) \). Accordingly, let the desired trajectory for the \( i \)-th agent be a set of straight segments from the current cell to the next desired partition center computed as

\[ q_i^d(t) = c_5, i(t) \cdot g(t)^5 + c_4, i(t) \cdot g(t)^4 + c_3, i(t) \cdot g(t)^3 + c_0, i(t) \]

for \( t \in [t_k, t_{k+1}] \) and

\[ c_5, i(t) = 6\Lambda(s_i(t_{k+1}) - s_i(t_k)), \quad c_0, i(t) = \Lambda(s_i(t_k)) \]

\[ c_4, i(t) = -15\Lambda(s_i(t_{k+1}) - s_i(t_k)), \quad g(t) = \frac{t - t_k}{T} \]

\[ c_3, i(t) = 10\Lambda(s_i(t_{k+1}) - s_i(t_k)) \]

Note that the trajectory \( q_i^d(t) \) is continuous and smooth and that \( q_i^d(t_k) = q_i^d(t_{k+1}) = 0 \ \forall \ k \in \{1, 2, \cdots \} \). The control law is a feed-forward proportional-derivative control given by

\[ u_i(t) = \dot{q}_i(t) + K_D(q_i^d(t) - q_i(t)) + K_P(q_i^d(t) - q_i(t)) \]

(5)

where \( K_P > 0 \) and \( K_D > 0 \) are both positive definite matrices.

### IV. Motion and Collision Avoidance Analysis

To prove that the hybrid control approach can guarantee safe motion for all agents, one needs to first show that the continuous trajectory tracking controller can drive the agents along the desired trajectories with little or no deviation. Therefore, define the trajectory tracking error for the \( i \)-th agent as \( e_i(t) = q_i^d(t) - q_i(t) \). It is easy to show that the error dynamics are given by

\[ \dot{e}_i(t) = \left[ \begin{array}{c} \mathbf{e}_i(t) \\ \dot{\mathbf{e}}_i(t) \end{array} \right], \quad A = \left[ \begin{array}{cc} 0_{2 \times 2} & I_{2 \times 2} \\ -K_P & -K_D \end{array} \right], \quad \mathbf{v}_i(t) = \left[ \begin{array}{c} 0_{2 \times 1} \\ \mathbf{v}_i(t) \end{array} \right] \]

and \( 0_{m \times n} \) and \( I_{n \times n} \) denote the \( m \times n \) zero and \( n \times n \) identity matrices, respectively. Note that \( A \) is Hurwitz and, therefore,
there exists positive definite symmetric matrices $P$ and $Q$ satisfying the Lyapunov equation
\[ A^T P + PA = -Q. \] (6)

The next proposition will show that for any given bounded disturbance, the error dynamics are ultimately bounded.

**Proposition 1** (Bound on the Tracking Error). The error dynamics for an agent with control input (5) are uniformly ultimately bounded with ultimate bound given by $\Delta = 2\lambda_{\text{max}}(P)\sqrt[\lambda]{\lambda_{\text{min}}(Q)}$, where $\lambda_{\text{max}}(P)$ and $\lambda_{\text{min}}(Q)$ are the maximum and minimum eigenvalues of $P$ and $Q$.

**Proof.** Consider the Lyapunov function $V_i(t) = \bar{d}^T(t)P\bar{e}(t)$. Taking the time derivative and using (6), one obtains
\[
\dot{V}_i(t) = \bar{e}_i^T(t)A^T(t)P\bar{e}_i(t) + \bar{e}_i^T(t)P\Lambda\bar{e}_i(t) - 2\bar{e}_i^T(t)P\bar{v}_i(t) \\
\leq -\|\bar{e}_i(t)\|_2\lambda_{\text{min}}(Q)\|\bar{e}_i(t)\|_2 - 2\lambda_{\text{max}}(P)\|\bar{v}_i(t)\|_2.
\]
Since $\dot{V}_i(t) < 0 \forall \|\bar{e}(t')\| > \Delta$, one can conclude that the trajectories of the error dynamics are uniformly ultimately bounded with ultimate bound $\Delta$. \[\square\]

Proposition 1 implies that, if for some $t' \geq t_0$, $\|\bar{e}(t')\| \leq \Delta$; then $\|\bar{e}(t)\| \leq \Delta \forall t \geq t'$. Using this statement, one can then proceed to prove collision avoidance at all times.

**Proposition 2** (Collision Avoidance). Consider a MWSN with local control law given by (1) and desired cells chosen according to Rules 1 through 4. Suppose that $\exists K_P > 0, K_D > 0$ such that $\Delta = 2\lambda_{\text{max}}(P)\sqrt{\lambda_{\text{min}}(Q)} < \frac{a}{2}h - r_a$ for some $P$ and $Q$ satisfying (6). Furthermore, assume that $\|\bar{e}_i(t_0)\| = \Delta$ and that $s_i(t_0) \neq s_j(t_0) \forall i, j \neq i$. Then, $\|q_i(t) - q_j(t)\| > 2r_a \forall i, j, t \geq t_0.$

**Proof.** Assume that $\|\bar{e}_i(t_0)\| = \Delta$ and $s_i(t_0) \neq s_j(t_0) \forall i, j \neq i$. The former implies that $\|\bar{e}_i(t)\| \leq \Delta \forall t \geq t_0$ and the latter implies that all agents start at different cells. For simplicity, consider the interaction of any two agents. A collision can only take place if 1) an agent travels to a previously occupied cell; 2) two or more agents simultaneously travel to the same cell; or 3) two agents with non-intersecting trajectories come into contact while transition to two different adjacent cells. The first case violates Rule 1. For the second case that the agents should have been able to detect each other at least one time step $T$ before the collision, given that they can only move one cell at a time. Since for any arrangement between two agents one will always have priority (see Rule 2), the second agent should have not been able to move to the same cell. As for the third case, note that for all non-intersecting desired trajectories $q_i^d(t)$ and $q_j^d(t)$, the distance at any given time is always equal or greater than $d_{x,y}(q_i^d(t), q_j^d(t)) \geq \frac{a}{2}h$. However, given that the agents do not deviate more than $\Delta$ from their desired trajectory and that their shape is radially bounded by $r_a$, one have that for two agents to come into contact, their desired trajectories must be within a distance $d_{x,y}(q_i^d(t), q_j^d(t)) \leq 2\Delta + 2r_a < \frac{a}{2}h$ of each other. The latter is a contradiction. Because none of the three cases can occur, one can conclude that the agents will not collide, i.e., $\|q_i(t) - q_j(t)\| > 2r_a \forall i, j, t \geq t_0$. \[\square\]

**Remark 4.** The proof for collision avoidance requires each agent to start at different cells and with bounded error dynamics. Therefore, an initialization phase that drives the agents to their closer initial cell may be required. If so, it is sufficient to require each agent to start from rest and at a distance equal or larger than $2r = \sqrt{3}h$ from each other.

**V. AWARENESS-BASED COVERAGE ANALYSIS**

In order to guarantee coverage of a large-scale domain, it is necessary that the gain rate of awareness when an agent passes through a cell is larger than the loss of awareness. Therefore, based on characteristics of the sensing function in Section II-B, we formulate the following assumption.

**Assumption 2.** $\exists \mu_0 > h > \alpha \geq \alpha \geq 0$ such that $\forall p \in \mathbb{P}$, one have that $\alpha \leq \alpha(p) \leq \alpha$ and
\[
\mu T \leq \int_{t_0}^{t_{k+1}} M(q_i(t), p) d\tau \leq \mu T
\]
if $\Delta p \in \{S_2(q_i(t))\} \forall t \in [t_k, t_{k+1})$.

Assumption 2 implies that the solution to the awareness coverage model (3) can be lower and upper bounded by
\[
e^{-(\bar{\mu} - \omega)(t-t_k)}z_i(q_i(t_k), p, t_k) \leq z_i(q_i(t_k), p, t),
\]
\[
z_i(q_i(t_k), p, t) \leq e^{-(\bar{\alpha} - \omega)(t-t_k)}z_i(q_i(t_k), p, t_k)
\]
if the partition center is covered in the interval $t \in [t_k, t_{k+1})$.

Similarly, to avoid an agent trapping itself in a local minima, the following property about the task domain is assumed.

**Assumption 3.** The task domain $\mathbb{W}$ is bounded and can be entirely covered by a tessellation of $N_p$ hexagons, $\mathbb{W}_H$, with edges $h$ and centers $p \in \mathbb{P}$. Moreover, for any two partition centers $\{p_0, p_f\} \in \mathbb{P}$, there is a set of partitions centers $\{p_0, p_1, \cdots, p_{f-1}, p_f\} \subset \mathbb{P}$ such that $d_c(p_i, p_{i-1}) = 1$ and $d_c(p_{i-1}, p_f) - d_c(p_i, p_f) = 1 \forall i \in \{0, 1, \cdots, f - 1\}$.

Assumption 3 implies that between any two cells in $\mathbb{W}_H$ there is always a path for which the distance between cells decreases. This is similar to implying that the task domain is convex.

**Proposition 3** (Guaranteed Coverage with Loss of Awareness). Consider a MWSN with motion governed by control (5) and Rules 1 through 4. Assume that Assumptions 2 and 3 hold. Furthermore, assume that $\alpha(p) < \mu_0/N_p^2 \forall p \in \mathbb{P}$. Then, $z_i(q_i(t), p, t_k) \rightarrow 0$ as $k \rightarrow \infty \forall i \in \mathbb{I}, p \in \mathbb{P}$.

**Proof.** First, note that the discrete controller always drives the agent to the partition center $s_i^d(t_k) = p$ that maximizes (4). Similarly, because the space is bounded, there is a finite time $T < N_pT$ that will take an agent to go from one partition center to any other, unless it encounters another agent. In the latter case, we have that either 1) both agents continue to the same destination or 2) coverage at the desired partition center is improved without visiting the cell. In either
case, the time it takes for the agent to gain information about the cell can be conservatively bounded by $\bar{T} < N_p T$.

Now, suppose that $\alpha(p) \leq \bar{\alpha} \leq \mu/N_p^2 \forall p \in \mathbb{P}$ and that Assumptions 2 and 3 hold. Consider the following function

$$\bar{V}_i(t_k) = \max_{p \in \mathbb{P}} \left\{ \frac{w_i(p, t_k)}{1 + \sigma d_c(s_i(t_k), p)} z_i(q_i(t_k), p, t_k) \right\}$$

which is positive $\bar{V}_i(t_0) > 0$ if there is at least one $p_j$ such that $z_i(q_i(t), p_j, t) > 0$, and zero otherwise. Now, note that only positive awareness coverage values can be a solution of (4), which means, that at least every $\bar{T}$, an agent will visit a cell with non-zero awareness coverage. Accordingly, the maximum time it can take for an agent to not repeat a single cell is upper bounded by $N_p T$. Therefore, $\bar{V}_i(t_k)$ can be bounded by

$$\bar{V}_i(t_k) \leq \max_{p \in \mathbb{P}} \left\{ z_i(q_i(t_k), p, t_k) \right\}$$

which goes to zero as $t \to \infty$ for $\bar{\alpha} < \mu/N_p^2$ and the proof is complete. \qed

Proposition 3 is quite conservative in the sense that it does not take into consideration cooperation among agents, i.e., it also holds for the case of a single agent. The following Proposition provides a lower bound on the number of agents required to achieve full awareness-based coverage.

**Proposition 4 (Failed Coverage).** Consider a group of $N$ sensors with awareness model (3) and $z_i(q_i(0), p, 0) > 0 \forall i \in \mathbb{I}, p \in \mathbb{P}$. Suppose that Assumptions 2 and 3 hold. If

$$N \leq \frac{\alpha N_p}{\bar{\mu} N_s},$$

then $z_i(q_i(t), p, t) \to \infty$ as $t \to \infty$ for some $i \in \mathbb{I}, p \in \mathbb{P}$.

**Proof.** Consider the awareness model in (3) and let $z_i(p_j, t_k) = \min_{i \in \mathbb{I}} \{ z_i(q_i(t_k), p_j, t_k) \}$. Regardless of any inter-agent exchange, one can lower bound $z_i(p_j, t_k)$ as

$$\lim_{k \to \infty} z_i(p_j, t_k) \geq \sum_{k=1}^{\infty} \min_{i \in \mathbb{I}} e^{-\bar{M}(q_i(t_k), p_j)} e^{-\alpha(p_j)} T, z_i(p_j, 0)$$

$$\geq z_i(p_j, 0) \sum_{k=1}^{\infty} \min_{i \in \mathbb{I}} e^{-\bar{\mu} T} e^{-\alpha(1-\tau_j) T}$$

$$= z_i(p_j, 0) \sum_{k=1}^{\infty} \min_{i \in \mathbb{I}} e^{-\bar{\mu} T} e^{2 T}$$

where $\tau_j k = 1$ if one or more agents are covering $p_j$ at time $t = t_k$ and zero otherwise. Ideally, one wants $z_i(p_j, t_k) \to 0 \forall p_j \in \mathbb{P}$, which implies that $\sum_{k=1}^{\infty} (-\mu T_j k + \bar{\alpha}) < 0 \forall j \in N_p$. A necessary condition for the latter is that

$$\sum_{k=1}^{N_p} (-\mu T_j k + \bar{\alpha}) < 0. \quad (7)$$

Now, note that at any given instance $k$, $\sum_{j=1}^{N_p} \tau_j k \leq N N_s$, where $N_s$ denotes a bound on the maximum number of partition centers that any single agent can cover at any given time. Returning to (7) yields

$$\sum_{k=1}^{N_p} (-\mu T_j k + \bar{\alpha}) < \sum_{k=1}^{N_p} (-\mu N N_s + \bar{\alpha} N_p)$$

which is negative if and only if $N > \alpha N P / \mu N_s$. Since the latter is a necessary condition to provide full awareness-based coverage, one can conclude that $\exists i \in \mathbb{I}, p \in \mathbb{P}$ such that $N \leq \alpha N P / \mu N_s$ then $z_i(q_i(t), p_j, t_k) \to \infty$ as $k \to \infty$. \qed

**VI. Example**

To demonstrate the performance of the proposed hybrid controller, a simulation example with $N = 12$ agents with physical radius $r_a = 1.5$ m was performed. The agents are assumed to have dynamics given by (1) with disturbance vector $v_i = [0.2 \cos(i \cdot t), 0.2 \sin(i \cdot t)]^T \forall i \in \mathbb{I}$. For simplicity, a binary sensing function is assumed

$$M(q_i(t), p) = \begin{cases} 0.08, & \text{if } ||q_i(t) - p|| \leq r_s \\ 0.00, & \text{otherwise} \end{cases}$$

where $r_s = 6/\sqrt{3}$ m, while the communication radius is taken to be $R = 14.4$ m. The task domain is a regular hexagonal region of $3367.1$ m$^2$, which can be divided in $N_p = 127$ regular hexagonal cells with inner radius $r = 3$ m and edge length $h = 2r/\sqrt{3} = r_s$.

The parameters for the hybrid controller were chosen as $K_P = K_{2 \times 2}, K_D = 0.5 I_{2 \times 2}, T = 10$ s, $\omega_{0} = 0.02$, and $v = 0.05$ s$^{-1}$. The parameters for the awareness model were chosen as $\alpha(p) = 0.004 \forall p \in \mathbb{P}$. Based on these parameters one can verify that Assumptions 1 and 2 are satisfied with $\bar{\mu} = \mu = 0.08$ and $\bar{\alpha} = \alpha = 0.004$. Similarly, one can use Propositions 1 and 4 to compute the bound on the tracking error as $\Delta = 1.05$ m, which satisfies Proposition 2, as well as a lower bound on the minimum number of agents required for coverage (6.35 agents).

The agents were initialized at rest using an uniform random distribution over the entire task domain but satisfying the constraint $||q_i(0) - q_i(0)|| \geq 2r$. The initial awareness for each agent and each cell was also picked at random within the interval $[3, 4]$. Fig. 4 depicts the motion of all agents at different intervals of time as well as the awareness-based coverage map for the first agent $z_1(q_1(t_k), p, t_k)$. Darker green colors denote lack of awareness while brighter yellow colors indicate cells with high awareness (i.e., lower $z_1(q_1(t_k), p, t_k)$ values). Note that the first agent achieves good coverage of the entire task domain by $t_k = 1600$ s. In terms of collision, no pair of agents came closer than $4.83$ m at all times.

In addition to the above example, the same coverage problem was simulated with groups of 6, 7, 8, and 16 agents. Fig. 5 denotes the maximum awareness-based coverage values across all agents and partition centers $p$ as a function of time. Note that the higher the number of agents, the faster the agents are able to cover the entire task domain. Also note that, as predicted by Proposition 4, a team of 6 agents cannot provide satisfactory coverage to the entire task domain.
Fig. 4: Sequential Motion of All Agents and Awareness-Based Coverage for the First Agent. The motion and position of the first agent is given in red, while the motion and position of all other agents is given in blue. The dotted red circle delimits the communication region for the first agent.

Fig. 5: Awareness-based Coverage for Different Number of Agents.

VII. EXPERIMENTAL VALIDATION

The simulation scenario described in the previous section was experimentally validated on a team of mobile robots. The hexagonal cells were scaled down to \( r = 0.126 \) m and \( h = 0.1455 \) m, while the sensing and communication range were taken to be \( r_s = h \) and \( R = 0.6048 \) m, respectively. With a regular hexagonal configuration of \( N_p = 127 \) cells, the task domain area is approximately 6.985 m².

The robot team consists of \( N = 10 \) Khepera IV differential-drive robots by K-Team, whose diameter is \( 2r_a = 0.14 \) m. The Khepera IV are equipped with an 800MHz ARM Cortex-A8 Processor with C64x Fixed Point DSP core running a Linux core, and receive velocity commands over a 802.11 b/g WiFi network. The commands are sent from a workstation with an Intel(R) Xeon(R) W-2125 processor, as determined by a Matlab implementation of the proposed controller. Position and orientation data is obtained through a motion capture system consisting of eight Vicon Vantage V8 cameras. An overhead projector allows one to visualize the coverage information in real-time over the robot workspace, a 5.182 m × 3.658 m rectangular area.

The Khepera IV can be modeled as

\[
\begin{align*}
\dot{x}_i &= \nu_i \cos \phi_i, \\
\dot{y}_i &= \nu_i \cos \phi_i, \\
\dot{\phi}_i &= \omega_i
\end{align*}
\]

where \((x_i, y_i)\) is the center of the robot, \(\phi_i\) its heading, and \((\nu_i, \omega_i)\) are its linear and angular velocity inputs. A velocity tracking controller was used similar to the one in (5) to provide the reference velocity vector for each robot. This vector was transformed into linear and angular velocity inputs through the following expression

\[
\begin{bmatrix}
\nu_i(t) \\
\omega_i(t)
\end{bmatrix} = \begin{bmatrix}
\cos \phi_i(t) & \sin \phi_i(t) \\
-\frac{1}{\ell} \sin \phi_i(t) & \frac{1}{\ell} \cos \phi_i(t)
\end{bmatrix} u_i(t)
\]

where \(\ell = 2r_a = 0.14 \) m is a control parameter.

The experimental results are illustrated in Fig. 6 and 7. Fig. 6 shows three instances of the implementation along with a projection of the awareness-based coverage for the first agent. Fig. 7 illustrates the overall improvement in coverage over time. As it can be seen in both figures, the agents are able to increase coverage of the entire task domain. Finally, the minimum distance between any two pairs of agents was reported to be 0.1567 m.

VIII. CONCLUSIONS AND FUTURE WORK

This paper presented a decentralized, hybrid cooperative control strategy for an arbitrarily large MWSN with radially
bounded input disturbances for coverage control applications in time-varying large-scale domain. The hybrid control strategy is comprised of a discrete cooperative controller and a continuous local trajectory tracking controller. The discrete control is built on the concepts of awareness-based coverage and cellular automata to guarantee coverage of the entire task domain. The continuous local control guarantees that the agents stay close to their desired trajectories. The synthesis of both controllers guarantees the safety and satisfactory coverage of the entire task domain despite decentralization, limited inter-agent communication, and time-varying properties of the domain. Future work includes enforcing connectivity among agents and optimizing their performance based on control effort and coverage.

REFERENCES