2024 IEEE 63rd Conference on Decision and Control (CDC)
December 16-19, 2024. MiCo, Milan, Italy

A Distributed Reinforcement Learning Strategy to Maximize Coverage in a Hybrid Heterogeneous Sensor Network

Hesam Mosalli and Amir G. Aghdam

***Abstract*—This paper introduces an efficient distributed deployment strategy for a network of mobile and stationary sensors with nonidentical sensing and communication radii. A collaborative distributed multi-agent deep reinforcement learning method is proposed to find the best moving direction and step size for each sensor, considering the coverage priority. The gradient of the local coverage function is used to generate a fast-converging solution as well as a learning-inspired arbitrary input to enable the network to avoid the local optima. The sensors use their partial observation of the network and field to iteratively relocate themselves to explore the field and learn the optimal policy to increase their local coverage. The efficiency of the proposed strategy in different scenarios is demonstrated by simulations.**

I. Introduction

Wireless sensor networks (WSNs) are composed of multiple sensor nodes capable of gathering data from the environment and forwarding it through the underlying communication network to the sink node(s). They are becoming increasingly popular in various applications, including environmental monitoring and traffic surveillance [1], [2]. A major challenge in employing WSNs is ensuring adequate coverage over the region of interest (ROI) while maintaining sensors' energy consumption at a low level.

In the absence of a communication infrastructure, it is beneficial to deploy WSNs using distributed strategies that rely on locally available information to determine the movements and actions of each sensor node, maximizing coverage. These strategies typically involve partitioning the sensing field into regions, using the Voronoi diagram [3], for instance, and assigning a node to each region. Different approaches can then be employed to find a candidate moving point inside the assigned Voronoi regions to enhance the covered area by the WSN. In virtual-force-based algorithms, each sensor is driven to a new point by a combination of attractive and repulsive forces [4]–[6], modelling the interactions between sensors and their environment.

Under more realistic assumptions, there may be obstacles in the sensing field, or the sensors may be nonidentical, i.e., they may have different mobility, sensing, and communication specifications, making the WSN heterogeneous. Some of these challenges have been addressed in [7]–[9] by using the multiplicatively-weighted Voronoi diagram [10].

Another category of deployment strategies for maximizing coverage in dynamic WSNs involves leveraging the gradient of the sensing field to determine the optimal moving direction and step size for each sensor. Gradient-based approaches offer a crucial advantage by integrating environmental and operational constraints, such as obstacles, prioritized coverage areas, or energy consumption, into the optimization process. This capability enables a broader problem formulation and facilitates necessary adjustments to the strategy. For instance, a gradient descent algorithm is proposed in [11] that is tailored to a range of utility functions encoding optimal coverage and sensing policies. Moreover, [12] employs a distributed nonlinear optimization method, utilizing iterative information exchange between neighbouring sensors to identify target points to move to for maximizing local coverage. A modification of this algorithm is introduced in [13] that can address the additional challenges posed by combining the mentioned variations of the coverage problem altogether.

Although Voronoi-based partitioning of the sensing field facilitates the design and implementation of distributed deployment strategies, it may have some disadvantages. For instance, due to the iterative nature of such strategies, restricting the operational domain of each sensor to its Voronoi region may lead to a local optima in the solution and lower the WSN's overall performance. In the last decade, AI-based methods have proved effective in solving the complex problems in WSNs. Particularly, utilizing reinforcement learning (RL) and Q-learning demonstrates potential in routing, coverage optimization, and management of performance parameters. Using RL principles, sensor agents can learn and adjust their actions according to received rewards, thereby facilitating optimal decision-making within dynamic network conditions [14], [15]. Moreover, the application of multi-agent deep reinforcement learning (MADRL) can revolutionize the performance of WSN distributed deployment algorithms. There has been a notable focus on employing this approach in resource allocation and coverage optimization problems in wireless communication networks [16], [17]. The similarities between such problems and the present one motivate the development of distributed DRL-based deployment strategies for WSNs. Finally, integrating the networked communication between sensors into the learning approach enables one

This work has been supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) under grant RGPIN-2024-06367.

H. Mosalli and A. G. Aghdam are with the Department of Electrical and Computer Engineering, Concordia University, Montréal, QC, Canada. Email: hesam.mosalli@mail.concordia.ca, amir.aghdam@concordia.ca

to account for heterogeneous agents with different reward functions and reduce the coordination cost by considering neighbour-to-neighbour communication, facilitating the design of decentralized MADRL algorithms [18], [19].

In this paper, the weighted coverage optimization problem in a heterogeneous WSN is considered via a distributed MADRL strategy under a combination of operational specifications and constraints. These include heterogeneity (sensors with different sensing and communication capabilities), hybridity (both mobile and stationary sensors), the prioritized sensing field, as well as power supply management. In the proposed algorithm, the sensors exchange their current state with their neighbours over synchronous communication and use the locally available information to find the best moving direction and acceleration during each iteration interval. Due to the unsupervised learning method, each sensor acts as an agent, exploring the environment and interacting with other agents until it learns the optimal policy for achieving the maximal coverage with the least energy consumption.

This paper is organized as follows. The coverage problem formulation in a WSN and the key definitions from the RL theory are presented in the next section. In Section III, the coverage maximization problem is presented in a standard distributed multi-agent reinforcement learning (MARL) framework. The gradient-based RL-Max deployment strategy is proposed in Section IV and its performance in various scenarios is demonstrated by simulations in Section V. Finally, Section VI summarizes the results of the paper and provides suggestions for future research directions.

II. Preliminaries and problem statement

A. Network and Coverage

Consider a network of n sensors denoted by $\mathcal{S} = \{s_1(x_1, r_{s_1}, r_{c_1}), s_2(x_2, r_{s_2}, r_{c_2}), \ldots, s_n(x_n, r_{s_n}, r_{c_n})\}$, tasked with covering a 2D sensing field $\mathcal{F}$, where x_i, r_{s_i}, and r_{c_i} are, respectively, the position, sensing radius, and communication radius of sensor s_i, $i \in \mathbb{N}_n := \{1, 2, \ldots, n\}$. It is assumed that each sensor follows a deterministic sensing model, i.e., s_i has perfect sensing over the points within the radius r_{s_i}, and no sensing coverage beyond this range. Similarly, the communication of the sensors is described by a deterministic model. In other words, each sensor can broadcast its information only to the sensors within its communication radius r_{c_i}. Define the set of all neighbours of a sensor s_i, denoted by $\mathcal{N}_i$, as the set of all sensors whose communication ranges reach s_i, from which s_i can receive information. The network is assumed to be heterogeneous, i.e., the sensing and communication radii of different sensors are not necessarily identical. In addition, the WSN is hybrid too, which means sensors $s_1, s_2, \ldots, s_m$ are mobile, and the remaining $n - m$ are stationary. Let the movement of all mobile sensors in the 2D field be modelled by double-integrator dynamics described by

$$\ddot{x}_i = u_i, \tag{1}$$

where $u_i \in \mathbb{R}^2$ is the control action vector applied to the $i-$th sensor. The sensing field is prioritized with respect to coverage importance, meaning that the relative significance of coverage at any point $q = (q_1, q_2)$ in the field, is represented by a priority function $\varphi(q) : \mathcal{F} \to \mathbb{R}^+$, where $\mathbb{R}^+$ is the set of all non-negative real numbers. A point with a higher value of the priority function is more important to cover compared to points with a lower value. To formulate the weighted coverage problem, let the sensing disk of a sensor be defined as follows.

Definition 1. *The sensing disk of the sensor $S(x, r_s, r_c)$ is defined as $D(x) = \{q \in \mathcal{F} | d(x, q) \leq r_s\}$, where $d(x, q)$ is the Euclidean distance between points q and x in $\mathcal{F}$.*

Problem Definition: Given a WSN with an initial configuration and a priority function defined over the sensing field, it is desired to find sensor locations from which the weighted coverage over the ROI is maximized. The global coverage maximization problem is formulated as follows:

$$\max_{\{x_i\}_{i=1}^m} \int_{\mathcal{F} \cap \left(\bigcup_{i=1}^n D(x_i)\right)} \varphi(q) dq. \tag{2}$$

Here, the overall weighted coverage is defined as the surface integral of the priority function over regions in the field $\mathcal{F}$ that are covered by at least one sensor, either stationary or mobile. Note that in this formulation, only mobile sensors are capable of improving the overall coverage by moving to proper points.

B. Reinforcement Learning

In RL strategies, learning agents interact with an environment to solve iterative decision-making problems modelled by Markov Decision Processes (MDP) [20]. The definition of MDPs can be generalized to networked multi-agent settings with agents having partial observation of the environment [18].

Let n be the number of agents interacting in an environment, S be the state space, and A_i and O_i, respectively, denote the action space and observation space of the $i-$th agent for $i \in \mathbb{N}_n$. Define $\mathbf{A} = A_1 \times \cdots \times A_n$ and $\mathbf{O} = O_1 \times \cdots \times O_n$ as the joint action and observation spaces of all agents, respectively. Then, networked partially observable MDPs (NPO-MDP) over a network of sensors $\mathcal{S}$ can be defined as $(\mathcal{S}, S, \mathbf{A}, P, \mathbf{R}, \gamma)$, where γ is the discount factor and $P : S \times \mathbf{A} \times \mathbf{O} \to [0, 1]$ is the observation function providing the probability $P(\boldsymbol{o}|\boldsymbol{a}, s')$ of the agents observing $\boldsymbol{o} = \{o_i\}_{i \in \mathbb{N}_n} \in \mathbf{O}$ after executing a joint action $\boldsymbol{a} = \{a_i\}_{i \in \mathbb{N}_n} \in \mathbf{A}$ and moving to the new state $s' \in S$. Also, $\mathbf{R} = \{R_i\}_{i \in \mathbb{N}_n}$ where the $i-$th agent's reward function $R_i : S \times \mathbf{A} \times S \to \mathbb{R}$ specifies the immediate reward it receives. In a distributed NPO-MDP, each agent is supposed to take action solely based on its local observation o_i, which may include part of the observations of its neighbouring agents. This enables the agents to team up with their neighbours and collaborate to maximize the overall reward.

Given the observation o_i, the next action to be taken (or the probability distribution over the agent's action space) is determined by the agent's policy denoted by $\pi_i : O_i \to A_i$.

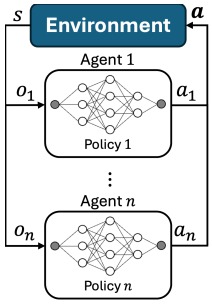

Fig. 1. General framework of a distributed multi-agent DRL problem with partial observability

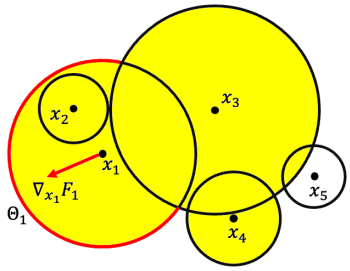

Fig. 2. Local coverage area of s_1 where $\mathcal{N}_1 = \{2, 3, 4\}$.

Each agent tends to find its optimal policy π_i^* that maximizes its benefit in the long term. Then, the joint policy of all agents, on the other hand, can be defined as $\boldsymbol{\pi} = \{\pi_i\}_{i\in\mathbb{N}_n}$. The value function of the $i-$th agent is then given by:

$$V_i^{\boldsymbol{\pi}} = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R_i(s^t, \boldsymbol{a}^t, s^{t+1}) \middle| \boldsymbol{a}^t \sim \boldsymbol{\pi}\left(\boldsymbol{o}^t\right), s^0 = s\right]. \tag{3}$$

RL methods like Q-learning are impractical where the action space and the states are continuous. In such cases, the policy and/or the returned values can be approximated by deep neural networks (DNN). The advancements in the DNN models enable deep reinforcement learning (DRL) algorithms, including proximal policy optimization (PPO) [21] and actor-critic methods [22], to handle more complex state spaces and environments. PPO, in particular, is known for its stability and reliability during training, which are crucial in complex and dynamic environments including WSNs. Moreover, in continuous action spaces, gathering data can be especially costly. PPO, being an on-policy algorithm, is relatively sample-efficient for such a method. In other words, it can learn effective policies with fewer interactions with the environment than most of the existing on-policy algorithms, resulting in lower power consumption during the training. Finally, its approach to policy updates allows for a more gradual adaptation of the agents to not only the environment but also the behaviours of other agents in a MARL framework. Fig. 1 demonstrates the general framework of a multi-agent DRL problem in which each agent learns the optimal policy of its own.

III. Network Coverage Problem in Multi-Agent Reinforcement Learning Framework

In this section, the local coverage maximization problem is reformulated in a distributed MARL framework. To do so, let $\mathcal{S}$ be a WSN with n sensors, including m mobile sensors, as described previously. The environment $\mathcal{E} = (\mathcal{F}, \mathcal{S})$ is defined as the combination of the sensor field ($\mathcal{F}$) and sensors ($\mathcal{S}$) operating as a network. The exact state of $\mathcal{E}$ at any time instant can be described by the configuration of the WSN in the sensing field. The mobile sensors are the agents acting based on their partial observations of the environment states, and the extent of their access to the states is determined by the flow of information between them. Thus, partial observation of $\mathcal{E}$ by the $i-$th agent is $o_i = \left\{s_j(x_j, r_{s_j}, r_{c_j})\right\}_{j\in\mathcal{N}_i}$, for $i \in \mathbb{N}_m$. On the other hand, the action set of an agent consists of any modifiable variable for the control strategy generating the vector u_i.

Lastly, defining the reward function is the key step to achieving a high-performance collaboration between the agents. Since having a central control unit is not tractable in most real-world applications where WSNs consist of large numbers of sensors with limited-range communication capabilities scattered over a wide area, it is desirable to find a distributed strategy that can maximize the overall coverage to be as close as possible to the globally-optimal solution.

In an alternative distributed cooperative approach, the mobile sensors move step-by-step under an iterative algorithm until the network coverage reaches a steady state. As the first step in reformulating the global optimization problem (2), the local coverage of a sensor $s(x, r_s, r_c)$ sensor with its set of neighbouring sensors denoted by $\mathcal{N}$ can be defined as:

$$F = \int_{\mathcal{F}\cap\left(D(x)\cup\bigcup\limits_{j\in\mathcal{N}} D(x_j)\right)} \varphi(q)dq. \tag{4}$$

The above integral provides the weighted coverage over the areas that are covered either by the sensor itself or by any of its neighbours. As a simple illustrative example, in Fig. 2, the region locally covered by sensor s_1 in a network of five sensors is depicted in yellow, as this sensor is only within the communication range of sensors s_2, s_3, and s_4.

In addition to the coverage, every sensor is required to manage its limited power supply during the deployment. To this end, the reward function must be intricately designed to promote energy conservation and maximize the coverage simultaneously. A positive reward is attributed to a sensor node when its local coverage given by (4) increases compared to the previous iteration. Conversely, penalties are introduced when nodes fail to increase their coverage. Also, sensors are penalized proportionally to the energy they consume during the time between consecutive iterations. This incentivizes the network to self-organize and adapt to changing conditions to prolong its operational lifespan while maintaining high performance. The major source of energy consumption in hybrid sensor networks is mobility. This includes the energy a mobile sensor consumes during each iteration, i.e., starting to move from a stationary position, continuing to move, and returning to a stationary position. This implies that the total

energy consumption of the $i-$th sensor's movement from iteration t to iteration $t+1$ can be expressed as:

$$E_i^t = E_{move} d\left(x_i^t, x_i^{t+1}\right) + E_{i,stop}^t + E_{i,start}^t, \quad (5)$$

for all $i \in \mathbb{N}_n$, where $E_{i,start}^t$ is the energy required for sensor $i \in \mathbb{N}_m$ to start moving from a complete stop at the beginning of the iteration (at time t), which is assumed to be a fixed value E_{start}, and E_{move} is the energy required for the sensors to move for one meter. Similarly, $E_{i,stop}^t$ is the energy required for the sensor to make a complete stop at the end of the iteration (at time $t+1$), which is also a fixed value E_{stop}. Note that the constants values E_{start}, E_{move}, and E_{stop} are assumed to be the same for all sensors.

To consider the two objectives together, given the joint action $\boldsymbol{a}^t$ applied to the agents, resulting in the transmission of the states from s^t to s^{t+1}, the reward received by the $i-$th sensor is defined as a weighted linear combination of local coverage and energy as below:

$$R_i(s^t, \boldsymbol{a}^t, s^{t+1})|\boldsymbol{a}^t = \lambda(F_i^{t+1} - F_i^t) - (1-\lambda)\alpha E_i^t, \quad (6)$$

where α is a normalization coefficient that brings the reward and penalty to roughly the same scale, preventing the one with the larger scale from dominating the other. Moreover, the weight λ determines the relative importance of the coverage versus energy consumption. This formulation facilitates the exploration of different trade-offs between the two objectives in a continuous and flexible manner.

IV. Distributed RL-Max Coverage Deployment Strategy

With the required foundation introduced in the previous section, we develop a distributed RL gradient-based coverage maximization deployment strategy for a heterogeneous hybrid WSN. Sensors' movements are to be appropriately coordinated to maximize the weighted coverage over the sensing field. It is shown in [13] and [12] that moving the sensors in the direction of the gradient of the weighted coverage function can efficiently increase the coverage. However, in the existing approaches, the local coverage of the sensor inside a Voronoi region is considered the objective function, which may trap the WSN in a local optimum. One can use the local coverage function in (4) instead of the Voronoi-based coverage area to address this hurdle. The following lemma provides the gradient of the above-mentioned objective function.

Lemma 1. *Let the local coverage of a sensor $s(x, r_s, r_c)$ be given by* (4)*. Then, the gradient of the local coverage w.r.t. the location of the sensor can be approximated by*

$$\nabla_x F = \frac{2\pi r_s}{N} \sum_{\substack{k=1 \\ q_k \in \Theta}}^{N} \begin{bmatrix} \cos\theta_k \\ \sin\theta_k \end{bmatrix} \varphi(q_k), \quad (7)$$

for a sufficiently large integer N. Here, $\theta_k = 2(k-1)\pi/N$ ($k \in \mathbb{N}_N$), $q_k = x + r_s\left[cos\theta_k, sin\theta_k\right]^T$, and Θ is part of the boundary of $D(x)$ not inside $D(x_j)$ for any $j \in \mathcal{N}$.

Proof. The proof follows the same reasoning in [12]. ■

Lemma 1 states that the gradient of a sensor's local coverage can be approximated by a finite summation and that it only depends on the points not covered by any neighbouring sensors. This is illustrated by an example in Fig. 2. Note also that the result of the above lemma only applies when the neighbours of a mobile sensor remain unchanged and do not move. Such an assumption holds from the viewpoint of a sensor when its displacement is sufficiently small.

A candidate moving direction is to be introduced for each sensor to increase its local coverage. Finding the best moving step is essential in this process. On the other hand, by simply following the gradient direction, the WSN may get stuck in a local optimum or in a resonating state in which sensors move periodically between a set of points, causing power depletion. To overcome this problem, an additional perturbation input vector $u_p \in \mathbb{R}^2$ is introduced for each sensor. Moreover, a braking force is modelled for the sensor, proportional to but in the opposite direction of the sensor's velocity. As a result, the control input u_i at time t is designed as follows:

$$u_i^t = k_{1,i}^t \nabla_{x_i} F_i^t + u_{p,i}^t + k_{2,i}^t \dot{x}_{s_i}, \quad (8)$$

where $k_{1,i}^t > 0$ determines the relative importance of the gradient direction compared to the other terms at each iteration. As a result, the set of actions for the $i-$th sensor agent at time step t is $a_i^t = (k_{1,i}^t, k_{2,i}^t, u_{p,i}^t)$.

According to (8) and on noting the formulation in Section III, the actions and observations of the agents are all continuous variables. Thus, we use DRL methods to learn the optimal policy for each agent. Due to its advantages, PPO is implemented along with DNNs to approximate the optimal policy and the value function. Although using DNN provides the means to address the problem's complexities, it is crucial to avoid high-dimensional models unsuitable for the limited processing and power capabilities of typical sensor nodes in WSNs, introducing another trade-off in the strategy design.

The policy and value functions of each agent are modelled by convolutional neural network (CNN) architectures consisting of initial convolutional layers that efficiently extract local patterns from observation matrices. This choice of architecture is crucial given the spatial interdependencies inherent to WSN environments. Subsequent fully connected layers integrate the extracted features, translating them into a comprehensive representation that informs the action probabilities. The architecture's depth and the dimensionality of each layer are carefully calibrated to balance the model's expressiveness with computational tractability, keeping the model sufficiently complex to capture the environment's dynamics without being computationally prohibitive. Moreover, the missing observations corresponding to sensors absent in the neighbourhood of a given sensor are compensated by zero padding.

Finally, since the proposed RL-Max strategy is an iterative algorithm in the learning and test stages, the choice of sampling time and stopping criteria significantly affects the final results. Choosing a small sampling time results in smooth trajectories for the mobile sensors, letting them interact with their neighbours in a timely manner. A choice

of large sampling time, on the other hand, causes large steps in the sensors' trajectories, which may lead to non-smooth and periodic movements and consequently, longer convergence time. As for the termination condition, a sensor stops moving when the absolute value of the change in the local coverage between two consecutive time steps is less than a prescribed threshold $\epsilon > 0$. Eventually, the learning episodes and the deployment procedure end if all sensors satisfy the termination condition. Similar to the sampling time, the choice of ϵ is made based on a trade-off between the steady-state coverage precision and the convergence speed of the network. The distributed RL-Max deployment strategy is presented in Algorithm 1.

Algorithm 1 The distributed RL-Max coverage deployment strategy for a hybrid heterogeneous WSN

Inputs: A WSN $\mathcal{S}$ with an initial configuration, a trained joint policy $\boldsymbol{\pi}$, sampling time T_s, and coverage increase threshold ϵ.
Initialize $t = 0$.
Initialize $\Delta F = \epsilon$ and calculate F_i^t from (4) for $i \in \mathbb{N}_m$.
while $\Delta F > \epsilon$ **do**
  Update the states s^t and the partial observations $\boldsymbol{o}^t$ according to the WSN configuration.
  Sample a joint action $\boldsymbol{a}^t \sim \boldsymbol{\pi}(\boldsymbol{o}^t)$.
  Calculate $\nabla_{x_i} F_i^t$ and u_i^t for $i \in \mathbb{N}_m$ using (7) and (8).
  Move the sensors according to (1) given T_s.
  Calculate F_i^{t+1} using (4) for $i \in \mathbb{N}_m$.
  Update $\Delta F = \max\{F_i^{t+1} - F_i^t\}_{i \in \mathbb{N}_m}$.
  Update $t = t + 1$.
end while

V. Simulation Results

In this section, the performance of the RL-Max strategy is investigated in different scenarios. Given the maximum coverage problem's complexity and nonlinearity, identifying the globally optimal sensor arrangement is infeasible using some of the existing methods. Therefore, the strategy's performance is assessed through Monte Carlo simulations. The communication radius of each sensor is assumed to be four times its sensing radius and the sensing field is a square region of 10m×10m in all examples. The parameter values and bounds used in this section are presented in Table I.

Example 1. In this example, the performance of the proposed strategy is evaluated for a hybrid heterogeneous WSN in a sensing field with a uniform coverage priority function. A WSN with 20 mobile and 10 stationary sensors is randomly dispersed in the sensing field. The sensing

TABLE I
Parameter values used in the examples

Parameter	Value	Parameter	Value (interval)
T_s	0.1 second	α	0.1
E_{start}	33.072 J [23]	k_1	[0,1]
E_{move}	8.268 J/m [23]	k_2	[-1,0]
E_{stop}	8.268 J [23]	u_p	[-1,1]×[-1,1]

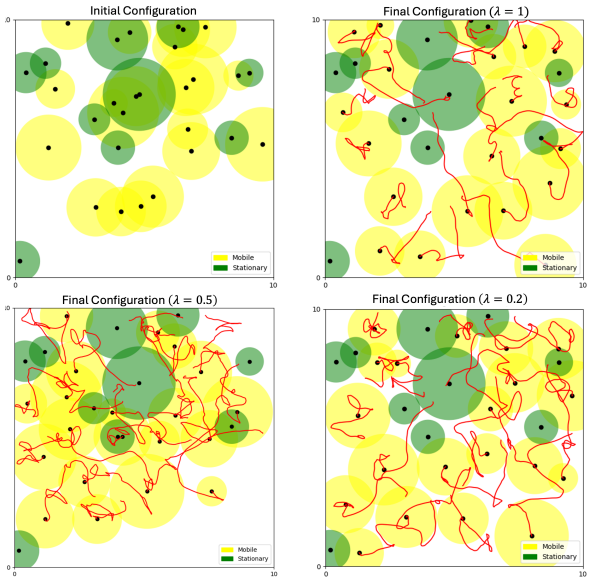

Fig. 3. The initial and final configurations of a WSN under RL-Max strategy for different values of λ

radii of the sensors are chosen randomly between 1.5m and 2.5m. For this scenario, the optimal policy is trained with a discount factor of $\gamma = 0.86$ to balance each agent's priority between the immediate reward and the long-term goal. The termination threshold is chosen as $\epsilon = 0.01\text{m}^2$. A sample implementation of this case is shown in Fig. 3. The figure demonstrates the strategy's ability to relocate the mobile sensors to a configuration in which the overall coverage of the network is increased by avoiding unnecessary overlap between the sensing disks of the adjacent sensors. The efficiency of the method in maximizing the overall coverage is investigated through Monte Carlo simulations run on 50 random initial configurations. The coverage efficiency is measured by the coverage factor, defined as the ratio of the network's overall (weighted) coverage to the weighted area of the sensing field.

To investigate the performance of the strategy in sensor power supply management, the test is repeated for different values of λ. The average final coverage factors, energy consumptions, and termination times are summarized in Table II. The results demonstrate that as λ decreases, the sensors become more hesitant to move to improve the local coverage, resulting in lower energy consumption. In contrast, a larger value of λ results in a higher final coverage factor achieved at the cost of more energy consumption. Also, the test results for high and low values of λ show a faster convergence time due to smoother and shorter travelling distances in such cases compared to those with medium λ values, which is also confirmed by Fig. 3.

Example 2. In this example, the performance of the proposed strategy is studied in a sensing field with a non-uniform priority function. The priority function is given by $\varphi(q) = \exp(-0.1d^2(q,(7,7))$ which has a peak value at the focal point $q = (7,7)$, and exponentially decays as moving farther from it. The darker spots in Fig. 4 indicate

TABLE II
AVERAGE PERFORMANCE OF THE RL-MAX STRATEGY OVER 50 TESTS

λ	0.2	0.4	0.6	0.8	1
Final coverage factor (%)	74.72	75.17	77.31	78.09	78.54
Energy usage by a mobile sensor (J)	185.7	214.2	228.1	276.9	311.1
Termination time (sec)	5.9	6.5	11.7	10.12	8.4

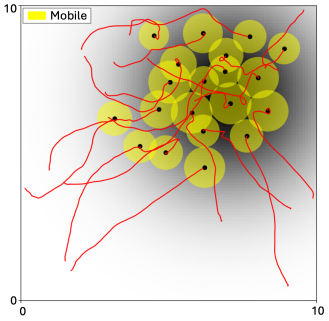

Fig. 4. Performance of the RL-Max strategy in a sensing field with exponential priority function

more important points to cover. The network consists of 20 mobile sensors with sensing radii randomly chosen between 0.25m and 0.75m. In this scenario, the sensors need to keep moving to maximize the weighted coverage collaboratively. In prioritized sensing fields, the WSN is more prone to getting stuck in a local optimum, which is a result of not considering the long-term rewards. This issue can be addressed by setting the discount factor to 0.95, promoting long-term planning.

As shown in Fig. 4, the sensors are initially deployed randomly and move towards the focal point by tracking the direction of the weighted coverage gradient vector. In such cases, the perturbation force helps the network get out of the local optimum point where the gradient of the local coverage is almost zero. This is observed from the trajectories of the sensors which are smooth until they get closer to other sensors, at which point they exhibit minor and abrupt shifts in their direction. Note that due to the small values of the priority function in areas far from the focal point, the termination threshold is reduced to $\epsilon = 0.001$ so that the sensors can detect smaller changes in the coverage reward.

VI. CONCLUSIONS

A distributed reinforcement learning strategy is presented in this work to optimize coverage in hybrid heterogeneous sensor networks. By integrating MADRL with a gradient-based deployment strategy, the proposed approach effectively utilizes the individual capabilities of mobile and stationary sensors, adjusted to their distinct sensing and communication radii. The framework demonstrates improvements in network coverage and exhibits adaptability and efficiency across various deployment scenarios. The sensors autonomously evolve their strategies through iterative learning and interaction, optimizing coverage while conservatively managing their energy resources. Simulations underscore the efficacy of the method compared to existing results.

REFERENCES

[1] H.-C. Lin, Y.-C. Kan, and Y.-M. Hong, "The comprehensive gateway model for diverse environmental monitoring upon wireless sensor network," *IEEE Sensors J.*, vol. 11, no. 5, pp. 1293–1303, 2011.

[2] M. Tubaishat, P. Zhuang, Q. Qi, and Y. Shang, "Wireless sensor networks in intelligent transportation systems," *Wirel. Commun. Mob. Comput.*, vol. 9, no. 3, pp. 287–302, 2009.

[3] A. Okabe, B. Boots, K. Sugihara, and S. N. Chiu, *Spatial tessellations: concepts and applications of Voronoi diagrams*. John Wiley & Sons, 2009.

[4] H. Mahboubi and A. G. Aghdam, "Distributed deployment algorithms for coverage improvement in a network of wireless mobile sensors: Relocation by virtual force," *IEEE Trans. Control Netw. Syst.*, vol. 4, no. 4, pp. 736–748, 2017.

[5] G. Wang, G. Cao, and T. La Porta, "Movement-assisted sensor deployment," *IEEE Trans. Mobile Comput.*, vol. 5, no. 6, pp. 640–652, 2006.

[6] Y. Zou and K. Chakrabarty, "Sensor deployment and target localization based on virtual forces," in *22nd IEEE INFOCOM*, vol. 2, pp. 1293–1303 vol.2, 2003.

[7] W.-T. Wang and K.-F. Ssu, "Obstacle detection and estimation in wireless sensor networks," *Computer Networks*, vol. 57, no. 4, pp. 858–868, 2013.

[8] 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 *IEEE GLOBECOM*, pp. 1–6, 2010.

[9] H. Mahboubi, F. Sharifi, A. G. Aghdam, and Y. Zhang, "Distributed control of multi-agent systems with limited communication range in the fixed obstacle environments," *IEEE Access*, vol. 7, pp. 118259–118268, 2019.

[10] E. Deza, M. M. Deza, M. M. Deza, and E. Deza, *Encyclopedia of distances*. Springer, 2009.

[11] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Trans. Robot. Autom*, vol. 20, no. 2, pp. 243–255, 2004.

[12] J. Habibi, H. Mahboubi, and A. G. Aghdam, "A gradient-based coverage optimization strategy for mobile sensor networks," *IEEE Trans. Control Netw. Syst.*, vol. 4, no. 3, pp. 477–488, 2017.

[13] H. Mosalli and A. G. Aghdam, "A distributed strategy to maximize coverage in a heterogeneous sensor network in the presence of obstacles," in *62nd IEEE CDC*, pp. 8426–8431, 2023.

[14] O. S. Egwuche, A. Singh, A. E. Ezugwu, J. Greeff, M. O. Olusanya, and L. Abualigah, "Machine learning for coverage optimization in wireless sensor networks: a comprehensive review," *Annals of Operations Research*, pp. 1–67, 2023.

[15] X. Luo, C. Chen, C. Zeng, C. Li, J. Xu, and S. Gong, "Deep reinforcement learning for joint trajectory planning, transmission scheduling, and access control in uav-assisted wireless sensor networks," *Sensors*, vol. 23, no. 10, 2023.

[16] S. Baghdady, S. M. Mirrezaei, and R. Mirzavand, "Reinforcement learning placement algorithm for optimization of uav network in wireless communication," *IEEE Access*, pp. 1–1, 2024.

[17] X. Liu, C. Xu, H. Yu, and P. Zeng, "Multi-agent deep reinforcement learning for end—edge orchestrated resource allocation in industrial wireless networks," *Front. Inf. Technol. Electron. Eng.*, vol. 23, no. 1, pp. 47–60, 2022.

[18] A. Feriani and E. Hossain, "Single and multi-agent deep reinforcement learning for ai-enabled wireless networks: A tutorial," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 2, pp. 1226–1252, 2021.

[19] J. Parras, M. Hüttenrauch, S. Zazo, and G. Neumann, "Deep reinforcement learning for attacking wireless sensor networks," *Sensors*, vol. 21, no. 12, p. 4060, 2021.

[20] S. Thrun, *Probabilistic robotics*, vol. 45. ACM New York, NY, USA, 2002.

[21] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017.

[22] I. Grondman, L. Busoniu, G. A. D. Lopes, and R. Babuska, "A survey of actor-critic reinforcement learning: Standard and natural policy gradients," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev*, vol. 42, no. 6, pp. 1291–1307, 2012.

[23] S. Yoon, O. Soysal, M. Demirbas, and C. Qiao, "Coordinated locomotion and monitoring using autonomous mobile sensor nodes," *IEEE Trans. Parallel Distrib. Syst.*, vol. 22, no. 10, pp. 1742–1756, 2011.