A Novel Spatio-Topological Embeddings for Efficient & Model-free Redundant Node Placement in 6G IoT Networks

Keywords: 6g IoT, topological features, GCN, DDPG, Reinforcement Learning

Moving towards a ubiquitous network with every device speaking with other, 6G era has gained tremendous growth. One important expectation in the 6G era is that machines and things will be the main consumers of mobile data traffic, and thus there will be more than 55 billion devices connected to the Internet by the end of 2025. These devices will continuously sense, process, act, and communicate with the surrounding environment, generating more than 73 zettabytes of data per year [1]. With the advancement of communication and AI, the 6G is supposed to handle massive communication, hyper-large AI data handling, reliable and low-latency communication, and ubiquitous connectivity [2]. The primary challenge and the basic requirement in this new era are continuous connectivity with devices like self driving cars, drones, IoT, robot operated factories etc. Traditional IoT applications have been advanced with the 6G communication capabilities which lead to more processing at IoT nodes using AI. The more the processing is done at local or edge devices or cloud, more is the battery consumption. The massive transmission in 6G imposes a challenge of continuous connectivity and longer battery life. This creates coverage holes in the network [3]. On the other side, the selfish behaviour of a few nodes also leads to connectivity issues. These coverage holes pose the challenge of connectivity and quality of service in the 6G powered intelligent IoT services, so the solution to the criticality of the coverage holes is the primary motive of this research article.

Many researchers have focused on the coverage hole issues. It involves the study into two phases: hole detection and hole recovery. The detection of holes is primarily focused on Voronoi diagram and Delaunay triangulation, which requires the exact geo locations of the nodes, which is impractical and expensive for the large applications of IoT in 6G [4]. Another topologically based hole detection is distance-based schemes [5][6][7], which also require the precise coordinates of the sensor nodes, which is again impractical and expensive in case of mobile IoT nodes. Another category of research in the hole detection is the confident information coverage and disk coverage model. The disk coverage model is too idealistic for 6G IoT as shown in simulated figure 1, whereas confident information coverage defines the coverage from the estimation perspective and is suitable for 6G IoT applications [3][8]. Another approach considering the estimation perspective in the hole detection and estimating the boundary nodes to detect the hole and its irregular shape is using the improved Gaussian mixture model [4]. This present work considers our previous work on hole detection [4] and primarily focuses on the hole recovery mechanism.

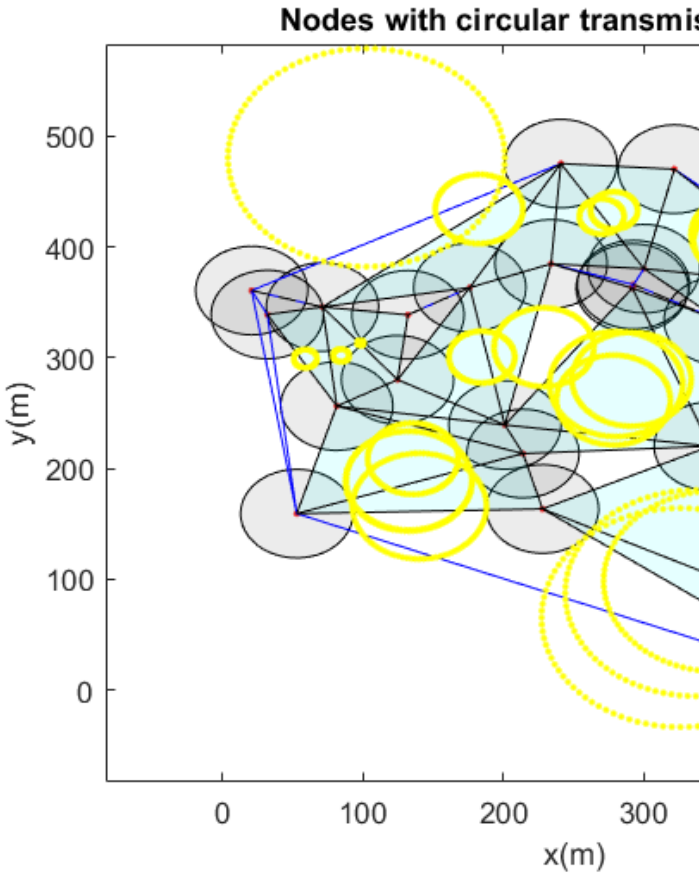

Figure 1: Coverage holes' detection with disk coverage model using circular transmission range

The coverage hole recovery algorithms either use the static nodes, which are in sleep mode during the idle phase, or mobile/redundant nodes, which are deployed in the network as per requirement. Previous studies have either used the optimization-based node deployment schemes [9][10] [11] or reinforcement learning based schemes for hole recovery [3][8][12]. The reinforcement learning (RL) scheme is more adaptive and intelligent for 6G IoT and can adapt more decision variables in the objective of redundant nodes deployment. The scalability is also better handled by the RL. The redundant node deployment is, although with a primary objective of maximum coverage, it should also solve the energy consumption and the long lifetime of neighboring nodes too. The previous research had used the states in RL as the decision parameters. These are coverage area in [3][8], reduced hole area and overlapped coverage in [12], the cell number, sensing range, total lifetime of the node [13], state of node as redundancy or isolated based upon the reward of coverage area to sensing range [14]. In these previous studies, the action is the location of the redundant node for maximum coverage area. Although there are a few other notable characteristics too that are affected by the redundant node's location and if ignored, may lead to the generation of

new coverage holes in the IoT in a short span of time. These are location criticality [15], load balancing [16] and failed device priority value [16]. Out of these, the first two are dependent upon the redundant node's location, and the third one depends upon the failed node's location. In the proposed work in this paper, the RL algorithm states that variables are a set of unnoticed variables for a longer network lifetime, and for the first time, this research suggests it along with minimum energy consumption and maximum coverage area objectives.

In the context of decision variables, the lack of load balancing, location criticality for longer lifetime is ignored in the previous studies, whereas failed device priority is another important parameter for the quick restoration of services in the emergency areas in 6G IoT. However, the IoT is a connected network and it involves many spatial and topological attributes to look for the redundant nodes' deployment. Spatial features refer to the physical arrangement and distribution of nodes in the network, while topological features encompass the interconnections between these nodes and their functional relationships. Ignoring these critical aspects can lead to suboptimal deployment of redundant nodes, resulting in inefficient coverage hole recovery strategies. The lack of a holistic view that integrates both spatial and topological information may also contribute to the rapid emergence of new coverage holes, undermining the overall network reliability and performance. Furthermore, existing reinforcement learning frameworks often fail to account for interactions between neighboring nodes, which can significantly influence the effectiveness of coverage strategies. By incorporating spatial and topological dimensions into the state representation, RL algorithms could improve adaptability and responsiveness to dynamic network conditions, ultimately enhancing coverage reliability and energy efficiency.

To deal with challenges in the redundant nodes placement in the hole healing, the following is the contribution in this paper:

- Failed device priority, location criticality, load balancing and energy consumption will be considered as the primary decision variables in the 6G powered IoT network
- The graphical convolution network (GCN) is first time to be introduced in this work to extract the spatial and topological attributes from the above set of decision variables for the multiple nodes which makes it generalize for different network sizes.
- Model free Deep Reinforcement Learning (DRL) is used to select the optimal location and number of redundant nodes to be placed in the network, considering the graphical spatial-topological attributes from the GCN.

The novel solution to address coverage hole healing involves two steps: using a new set of decision variables for placing redundant nodes to extend network lifetime, and extracting spatio-topological embeddings from IoT node connections to train DRL for optimal node placement.

Extending the work from [4], the IoT is presented as an undirected graph $G = (V,\ E)$. The IoT network in 6G is not a static network as it is a ubiquitous device connection, and the mobility of devices makes it dynamic. The solution of coverage hole healing in this proposed scheme involves the placement of redundant nodes, which are also moving nodes, either deployed on any vehicle or any robotic sensor node. Although in the present work, once the optimal location is identified for the redundant nodes, these become static. The operational chrematistics of redundant IoT nodes are similar to on-demand RIS in 6G to provide coverage to places with line-of-sight obstruction to reduce energy consumption and cost [17]. So, the network graph will be a continuously evolving structure in 6G IoT with respect to time.

The set of nodes are vertices $v \in V$ and the set of edges are communication links $e \in E$. E is the set of edges $(u,\ v)$ generates when nodes u and v are in the communication range of each other. The neighbor of each node is defined during the generation of the graph. A distance matrix $d(u_i, v_i)$ for each node is created where $i \in N$ is the number of nodes present in the network. The node is said to be in the neighborhood if the distance is less than the sensing radius R_s. An adjacency matrix A is created here for the cell element as 1 for nodes in range.

$$A\left(d(u_i, v_i)\right) = \begin{cases} 1, & if \quad d\left(u_i, v_i\right) \leq R_s,\ i \in N \\ 0, & otherwise \end{cases} \tag{1}$$

This matrix A creates an IoT-connected graph. The elements entries with element value 1 are considered to be connected. For example $A_{1,2} = 1$, it suggests that vertex v_1 and v_2 are connected with an edge length $e_{1,2} = d_{1,2}$.

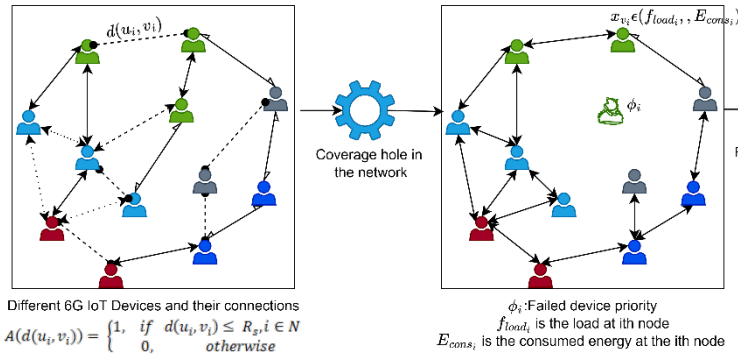

Figure 2: Graphical Network connection in 6G IoT with coverage hole and redundant node location spatial and topological features

The graph construction on the basis of the distance connection only gives the spatial relationship but lacks the topological and functional relationships with the other nodes.

The location criticality (C_{critical}), load balancing (f_{load}) and energy consumption (E_{cons}) are the topological relationships attributes. The value of these attributes changes with the simulation time as these are dependent on the neighbouring nodes. A node $x_{v_1} \epsilon \{C_{\text{critical}1}, f_{\text{load}_1}, \ E_{\text{cons}1}\}$ is a set of these attributes to generate the topological relationships with other sensor nodes. Figure 2 shows the graph $\mathcal{G}$ and its topological and spatial connections between the vertices. The embeddings as topological attributes are extracted using Graphical convolutional network (GCN) which uses a layer propagation rule for topological features as in equation 2 [18]:

$$f\left(x_{v_i}\left(t\right),\ A_t\right) = H_t^{(l+1)} = \sigma(D^{-\frac{1}{2}}\ \widetilde{A}_t\ x_{v_i}\left(t\right)\ \widetilde{D}^{-\frac{1}{2}}\ H_t^l\ W_t^l) \quad (2)$$

Here $H_t^{(l+1)}$ represents the embeddings of the next layer $l+1$ and $x_{v_i}\left(t\right)$ are the features of the ith sensor node at $t = 0$. The GCN in [18] is developed for the semi-supervised classification problem, though the work in the current manuscript is not the classification work, yet it is using the strength of GCN layers' embeddings which considers the spectral convolution on the IoT network graph with the defined nodes' features $x_{v_i}\left(t\right)$.

The GCN updates the nodes' features iteratively by aggregating information from the neighbouring nodes using graph convolution operators. This process reflects how sensor nodes influence each other's positions and states within the network over time. This helps in capturing the interactions between nodes which falls in the category of topological attributes. The graph convolution operators not only update the connections but also nodes' features $x_{v_i}\left(t\right)$, The learned adaptive feature representations are low-dimensional vectors of nodes at a time, referred to as $H_t^{(l+1)}$ at the time t and at layer l, which is 2 in this paper. It gets dynamically updated during the iterations. Figure 3 shows the change in the coverage area in the network with respect to simulation period. To validate and visualise the problem formulation, we have clustered the embeddings in t-SNE plot at few time simulation instant using k-means. However, in the core of this work, only trained embeddings from the GCN are used as state variables in the deep reinforcement learning as described in section 3.2. The figure 3 represents the network topology change after the random placement of redundant node. The silhouette score indicates the better coverage after the placement of redundant nodes.

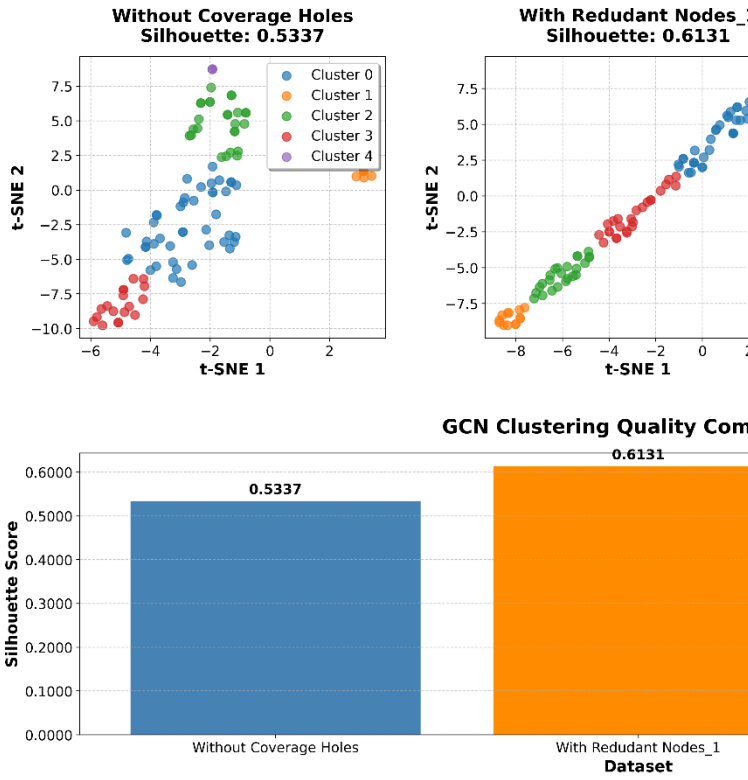

Figure 3: GCN embeddings t-SNE plot for 6G IoT network without holes, with different placement of redundant nodes at different positions

The decision variables formulation is discussed in the next sub-section which is used to calculate the nodes features for embeddings extraction to be used as state variables in model free redundant nodes placement.

The decision variables in the placement of redundant nodes are location criticality, load balancing, energy consumption which are directly related to location of the redundant node and failed device priority as the fourth one which is calculated from the location of the failed node. These are described as following:

As is validated from figure 3, the coverage area changes with the deployment of the redundant node and the information processing load also varies which may lead to non-uniform energy consumption and generation of coverage hole into another area after few simulation period [16]. The failure of the new node forces the neighbouring node to follow the longer path for data transmission which results in the probability of new coverage hole generation. Hence, the load balancing consideration in the redundant node's deployment is important.

To address this problem, it is to assume that the balanced clusters can have the uniform residual energy distribution and number of member nodes. So, the optimal position of the redundant node is considered with uniform load on the cluster head. Additionally, nodes' proximity can also be

considered to ensure less energy consumption. Where f_{load} is defined as the effective fitness function for load balancing of cluster head and can be calculated using equation below:

$$f_{\text{load}} = \left(1 - \frac{\mu_{\text{load}}}{\text{CH}_{\max}}\right) + \left(\frac{\text{loaded cluster head}}{\text{total cluster head}}\right) \quad (3)$$

Where, μ_{load} defines the mean of load can be calculated by equation 4, $\text{CH}_{\max}$ is the CH with maximum load.

$$\mu_{\text{load}} = \frac{\sum_{i=1}^{m} \text{Load } (\text{CH}_q)}{\text{cluster heads}} \quad (4)$$

$$\text{Load } (\text{CH}_q) = N \times \frac{E_{\text{remain}} \ (\text{CH}_q)}{E_{\text{initial}} \ (\text{CH}_q)} \quad (5)$$

Where, q is the number of CH and N denotes number of sensor nodes, E_{initial} (CH_q) and E_{remain} (CH_q) are the initial and remaining energy of cluster head CH_q respectively. The load on the cluster head increases with the number of sensor nodes and packet size. Therefore, the remaining energy is calculated by subtracting the consumed energy given in equation 6 from the initial energy given as follows [19][24]:

$$E_{\text{cons}}(N,d) = \begin{cases} N \times E_{\text{elec}} + N \times \in_{\text{fs}} \times d^2, & d < d_0 \\ N \times E_{\text{elec}} + N \times \in_{\text{mp}} \times d^4, & d \geq d_0 \end{cases} \quad (6)$$

Where, d_0 defines the distance between sender and receiver, N denotes the nodes, and E_{elec} is the energy required by electronic circuitry. $\in_{\text{mp}}$, is Multipath channel energy and $\in_{\text{fs}}$, is free space energy.

The failed device priority is also location and energy dependent. The work in [16] calculates the failed device priority on both, however, in this current work, the energy consumption is to be involved in reward calculation in the DRL, so the priority is calculated solely on the basis of neighbourhood degree. It is calculated as follows:

$$\phi_i = \sum A\left(d(u_i, v_i)\right) \quad (7)$$

Higher the ϕ_i, more is the priority of the failed device. The location criticality (C_{critical}) for the redundant node is also calculated based on the neighbourhood degree with adaptive graph construction with every new position of the redundant node. The equation 7 is modified and calculated for the redundant node's neighbourhood degree.

These network topological features are fed into the GCN to extract the spatio-topological embeddings to be used in the DRL as discussed in the next section of the complete proposed solution of coverage hole healing. The next section of the article discusses the proposed solution using these embeddings into DRL for the hole coverage.

Previously in this paper, the 6G IoT network is presented as network graph to extract the spatio-topological features. The decision variables are discussed in section 2.2. these decision variables are objectives of the hole recovery. With the placement of redundant nodes, the maximum hole area (A_{hole}) should be covered with lesser number of redundant nodes (N_r) with the consideration of the minimization of the E_{cons}. The objective function of the proposed methodology can be defined as in equation 8:

$$f_{\text{obj}} = argmin\left(w_1 \times N_r + w_2 \times \sum_{i\in(1,N)} A_{hole,i} + w_3 \times \sum_{i\in(1,N)} E_{\text{cons}}\right) \quad (8)$$

Here w_1, w_2, w_3 are random weights $\in (0,1)$ with condition of $(w_1 + w_2 + w_3 = 1)$. Here the maximization of coverage area is termed as minimization of the hole area $(A_{hole,i})$ to bring the objective function uniformity as rest other two contributing functions tend to be minimized. With the objective of equation 8, the graphical deep reinforcement learning (GDRL) has been proposed to place the N_r optimally in the network. It is termed as GDR now as the saptio-topological features from the Graph convolutional network (GCN) are extracted using the decision variables in equations 3,6,7 and these are to be used by DRL to generate the optimal action space. This methodology is also termed as model free approach as it doesn't require the understanding of the environment and takes the action as per the change and feedback from the 6G IoT network environment [20]. The overview of the complete methodology is shown in figure 4. The set of decision variables $x_{v_i}(t)$ from the network are fed into the GCN and trained for every iteration of DRL. The new embeddings thus generated are input to the DRL and corresponding action by the agent is decided as the redundant nodes' positions $[x_j, y_j]$, $j \epsilon 1..N_r$. The reward is calculated on the measured node's energy consumption in few network simulations and a new policy of the agent decides the new action as in figure 4.

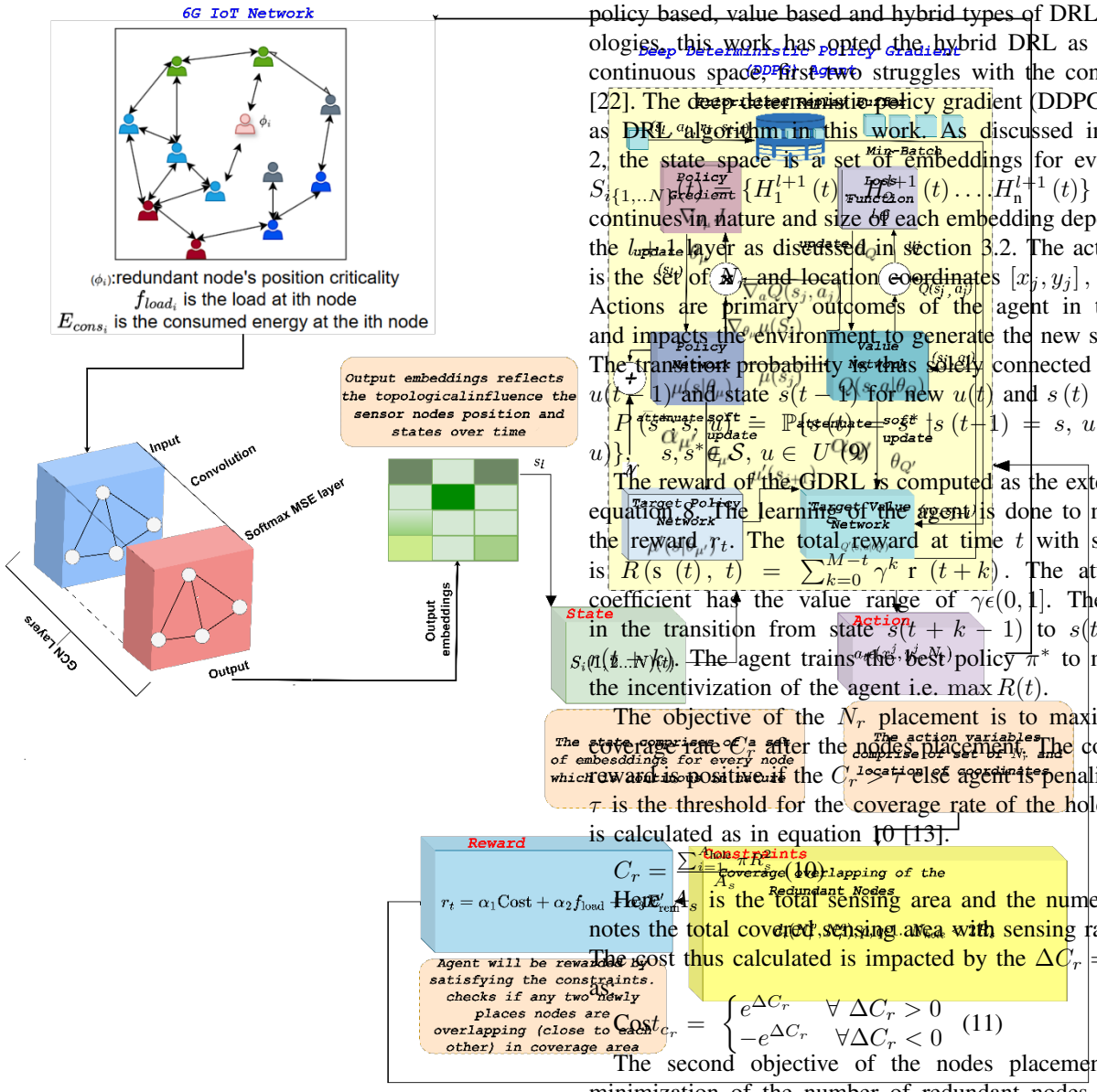

Figure 4: Methodology Diagram of the proposed spatio-topological embeddings consideration for Redundant nodes placement by DRL

The DRL is based upon the Markov decision process with 5 tuple (S, U, P, γ, R), where S is the state space variables which is the feedback from the environment, U is the set of actions of environment variable to be tuned as input to change the states, P is the transition of one state s to new state $s^{'}$ at next time interval using action A, R is the reward which motivates or penalize the agent for the right direction learning. γ is the reward discount parameter which is usually kept fixed at 0.95 and considered as the sensitive variable [21]. Skipping the more on MDP formulation, we suggest reader to explore more in [21].

Various DRL networks are existing in the literature and the selection of those is defined by the type and size of action space. The action space in the proposed work is continuous as it's the location of redundant node. Out of policy based, value based and hybrid types of DRL methodologies, this work has opted the hybrid DRL as for large continuous space, first two struggles with the convergence [22]. The deep deterministic policy gradient (DDPG) is used as DRL algorithm in this work. As discussed in section 2, the state space is a set of embeddings for every node $S_{i\{1,..N\}} = \{H_1^{l+1}(t), H_2^{l+1}(t) \ldots H_n^{l+1}(t)\}$ which is continues in nature and size of each embedding depend upon the l_{th} layer as discussed in section 3.2. The action $u(t)$ is the set of N_r and location coordinates $[x_j, y_j]$, $j \epsilon 1..N_r$. Actions are primary outcomes of the agent in the DRL and impacts the environment to generate the new state $s(t)$. The transition probability is thus solely connected to action $u(t-1)$ and state $s(t-1)$ for new $u(t)$ and $s(t)$ as:

$$P(s, s', u) = \mathbb{P}\{s(t) = s' \mid s(t-1) = s, u(t-1) = u\}, \quad s, s^* \in \mathcal{S}, u \in U(s) \quad (9)$$

The reward of the GDRL is computed as the extension of equation 8. The learning of the agent is done to maximize the reward r_t. The total reward at time t with state $s(t)$ is $R(s(t), t) = \sum_{k=0}^{M-t} \gamma^k r(t+k)$. The attenuation coefficient has the value range of $\gamma \epsilon (0, 1]$. The reward in the transition from state $s(t+k-1)$ to $s(t+k)$ is $s_i(1,2...N_r(t))$. The agent trains the best policy π^* to maximize the incentivization of the agent i.e. $\max R(t)$.

The objective of the N_r placement is to maximize the coverage rate C_r after the nodes placement. The cost of the reward is positive if the $C_r > \tau$ else agent is penalized. The τ is the threshold for the coverage rate of the hole area. It is calculated as in equation 10 [13].

$$C_r = \frac{\sum_{i=1}^{N_r} \pi R_s^2}{A_s} \quad (10)$$

Here, A_s is the total sensing area and the numerator denotes the total covered sensing area with sensing radius R_s. The cost thus calculated is impacted by the $\Delta C_r = C_r - \tau$ as

$$\text{Cost}_{c_r} = \begin{cases} e^{\Delta C_r} & \forall\ \Delta C_r > 0 \\ -e^{\Delta C_r} & \forall \Delta C_r < 0 \end{cases} \quad (11)$$

The second objective of the nodes placement is the minimization of the number of redundant nodes which is improved for the incentivization of the DDPG agent. The network is divided into clusters based on the Eigen Heuristics method [23], and the uniform load distribution on CH_q is considered as the positive or negative cost calculation for the redundant nodes' placement. The normalized f_{load} from equation 3 is termed as second objective in the reward calculation. Similarly, the remaining energy maximization $E_{\text{rem}} = 1 - E_{\text{cons}}$ from equation 6 is used as the maximization of reward. The reward at any iteration is calculated as in equation 12

$$r_t = \alpha_1 \text{Cost}_{c_r} + \alpha_2 f_{\text{load}} + \alpha_3 E_{\text{rem}} \quad (12)$$

The r_t is calculated in every iteration of the agent training and accordingly agent takes the decision for the u_{t+1} update.

The pseudocode of the proposed solution is given in algorithm 1. The present methodology aims to locate the optimal placement of the redundant nodes (x_r, y_r) using spatio-topological features using GDRL. The hybrid of the

value network $Q(s,\ u|\theta_Q)$ and policy network $\mu(s|\theta_\mu)$ in the DDPG works together to derive the optimal location of the node. The input to the algorithm 1 is the coverage hole and network parameters whereas output of it is the position of redundant nodes and volume of them. After a few simulation seconds run of the network, the set of topological features are updated. The two layered GCN as in equation 2 extracts the spatio-topological features considering all four decision variables discussed in section 2.2. Step 12 in algorithm 1 defines the stopping criteria of the GDRL. If the condition of overlapping of two redundant nodes is violated, the a_t at that iteration of the agent is discarded and reset.

Algorithm 1: Pseudocode for the proposed Spatio-topological extraction with GDRL optimized redundant nodes placement

[]@ ¿p() * 1.00@ ***Input:*** N_{hole}, A_{hole}, R_s, N, R_c
Output: $x_r^{j=1..N_{\text{hole}}}$, $y_r^{j=1..N_{\text{hole}}}$, N_r

Assuming the hole locations and hole area A_{hole} with number of holes N_{holes} are already identified in the hole detection phase (out of context of this paper, as is published in [4]). The next section focuses on the layered architecture of the DDPG.

The proposed DDPG model comprises an actor network and a critic network, which form its architecture. The actor network determines the redundant nodes' positions, known as the action a_t. The critic network produces the action-value $Q^\pi(s, a)$.

i. *Actor Network*

It is important to identify valuable characteristics from the input data, specifically the states of the others nodes when a new redundant node is placed in hole area. The actor-network is responsible for selecting the optimal course of action based on a specific situation. The actor network consists of an input layer, fully connected layers, and activation functions. The objective of this activity is to optimize the anticipated outcome (reward) by engagement with the environment, namely an 6G IoT network. The actor network architecture can be seen in Figure 5. The network begins with a feature input layer designated as "state" to represent the current state. The dimensions are 3×1, denoting three distinct features that represent aspects of the state as given in section 3.1. Then, the first fully connected layer processes the input layer with weights 400×3 and bias 400×1. ReLU activations are utilized subsequent to the first fully connected layer in order to induce non-linearity, as can be seen in Figure 5. ReLU facilitates the acquisition of intricate patterns by implementing a straightforward yet efficient non-linear transformation. Again, a second fully connected layer with 300×3 activations, further processing the output of the previous layer. The learnable size is 300×400 weights and 300×1 biases. Lastly, the third fully connected layer following the ReLU layer has the size of 7×1. A hyperbolic tangent activation function is used after this third layer. This function squashes the output between -1 and 1, which can be useful for actions that need to be within a specific range. At the end, A scaling layer is applied to adjust the range of the output actions. This can be critical for normalizing actions to the scale expected by the energy management environment.

Table 1. Description of different layers of Actor network

[]@ ¿p() * 0.25 ¿p() * 0.25 ¿p() * 0.25 ¿p() * 0.25@
Sizes
State (three features) Input layer 3×1 -
FC_1 (400 fully connected layer) fully connected 400×1 weights 400×3 bias 400×1
ReLU1 ReLU layer 400×1 -
FC_2 (300 fully connected layer) fully connected layer 300×1 Weights 300×400 Bias 300×1
ReLU2 ReLU layer 300×1
FC_3 (7 fully connected layer) fully connected layer 7×1 Weights 7×300 Bias 7×1
Tanh1 hyperbolic tangent 7×1 -
Scale scaling layer 7×1 -

ii. *Critic network*

The objective of this study is to train the critic network to accurately predict the Q-values so that the actor network can learn to make optimal decisions. The critic network assesses the efficacy of the activities proposed by the actor-network by considering the reward, which is determined by factors as in equation 12. The critic network structure comprises several input layers and fully connected layers tailored to process state and action inputs, ultimately predicting a Q-value. The input layers consist of a State Input layer, which has dimensions of 3×1 and represents the current state of the energy system, including variables like power loss, cost, and power deficit as given in section 3.1. Additionally, there is an Action Input layer with dimensions of 7×1, capturing the actions taken by the actor network, such as decisions regarding $(x_r,\ y_r)$. The fully connected layers begin with FC_1, which features 400×1 activations, processing the state input with a size of $400 \times (3 + 1)$. The weights and biases can be seen table 3. The next layer, FC_2, also has $400(C) \times 1(B)$ activations and integrates processed state and action information. Its size is $400 \times (400 + 7)$, indicating it combines features from both the state and action. The final fully connected layer, FC_3, has 1(C)×1(B) activation, reducing the output to a single value that represents the Q-value. The critic network architecture can be seen in Figure 5. To introduce non-linearity as we discussed in above given actor network, ReLU layers are used as activation functions between layers, known for their simplicity and efficiency in handling gradients. An addition layer sums the inputs, likely combining state and action inputs after their respective processing through earlier layers.

Table 2. Description of different layers of Critic network

[]@ ¿p() * 0.25 ¿p() * 0.25 ¿p() * 0.25 ¿p() * 0.25@ Sizes

State (three features) Input layer 3×1 -

FC_1 (400 fully connected layer) fully connected 400×1 weights 400×3 bias 400×1

ReLU1 ReLU layer 400×1 -

Action (7 features) Features input 7×1 -

Add (element-wise addition of 2 inputs) Addition 400×1

FC_2 (400 fully connected layer) fully connected layer 400×1 Weights 400×7 Bias 400×1

ReLU2 ReLU layer 400×1

FC_3 (7 fully connected layer) fully connected layer 1×1 Weights 1×400 Bias 1×1

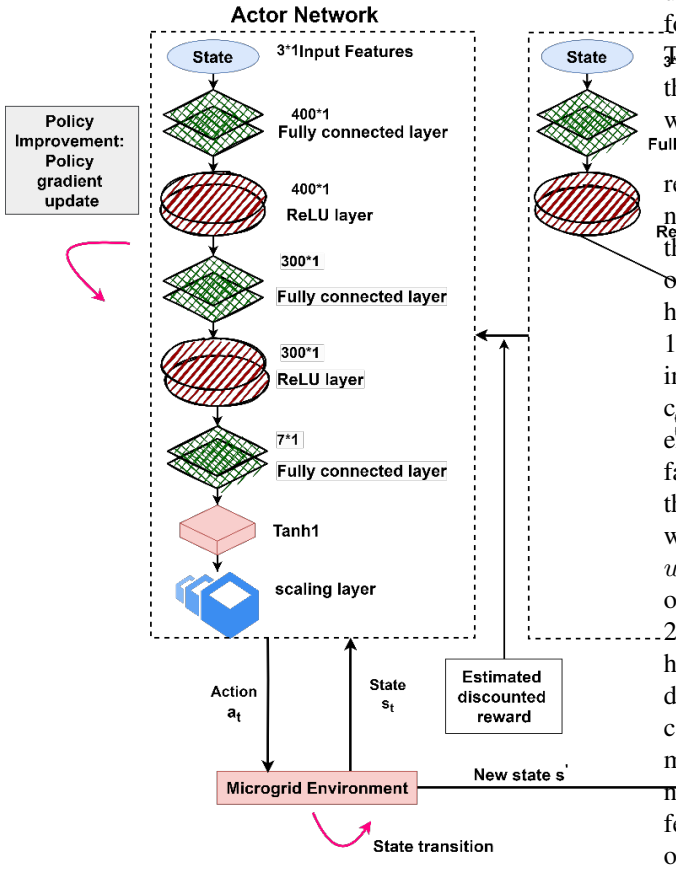

Figure 5. The structure of actor critic network in DDPG for 6G IoT hole healing

The proposed algorithm starts with the random deployment of sensor nodes in 6G IoT area. The validation of the proposed algorithm is done at different R_s and network area. It is assumed that the coverage area and location is already identified using [4]. The primary objective to maximise the coverage area with least energy consumption and uniform loading factor. The directed placement of N_r by spatio-topological features learning change the network topology and new clusters generation is done by GCN following the section 2.1. The residual energy and loading factor are calculated on the CH after the 100 simulation rounds. The implementation of the proposed work is done in MATLAB as well as pytorch. The node's features are generated in MATLAB and fed into pytorch module of GCN for the embeddings $H_t^{(l+1)}$ generations. The DDPG agent is trained using MATLAB toolbox. Whenever a new action u_{t+1} for a state s_t is generated by the trained agent policy $\mu(s|\theta^{\mu})$, the new embeddings are calculated and new CH features $x_{q\epsilon 1,2..N_q}$ are used for reward calculation form equation 12. In the previous work, the hole healing problems has been dealt with either by heuristic optimization algorithms [9][11] or with reinforcement learning [3][8][13]. The distinction between the proposed work and the state-of-the-art is the use of the topological features embedding from the GCN followed by DDPG agent training for model free approach. The uniform loading after the N_r placements constrained by the overlapping area of redundant nodes make the present work unique and novel.

Since the failed device priority and location criticality are related terminologies and dependent upon the number of neighbouring connections, the embeddings as the input to the GCN are E_{cons}, f_{load} and ϕ_i. Further in the discussion of the results, the validation and difference between the hole healing performance with maximization of equation 12 is shown in figure 6. The performance improvement in consideration of spatio-topological features as a novel contribution is justified in figure 6 itself. The use of GCN to extract the relationship of the nodes with other in the network facilitates the decision making more reliable and accurate in the redundant node placement, whereas the common DDPG with same hyperparameters a in table 1 is used for the u_t. The 6G IoT network is simulated for 100 rounds. The optimal number of clusters calculation following the step 2 for 100 nodes in the network comes to be $q = 5$. It has been noticed in various iterations with different nodes density and respective x_{v_i}, the optml number of clusters comes to be approximate $q = 5$. So, to reduce the steps in the methodology, it is now fixed for different nodes density in the network. The use of spatial features and spatio-topological features are separately trained with DDPG for a network size of $200 \times 200m^2$ and with $R_s = 30\ m$. The bar chart shown in figure 6 indicates the effect of graphical connections between the nodes in hole healing.

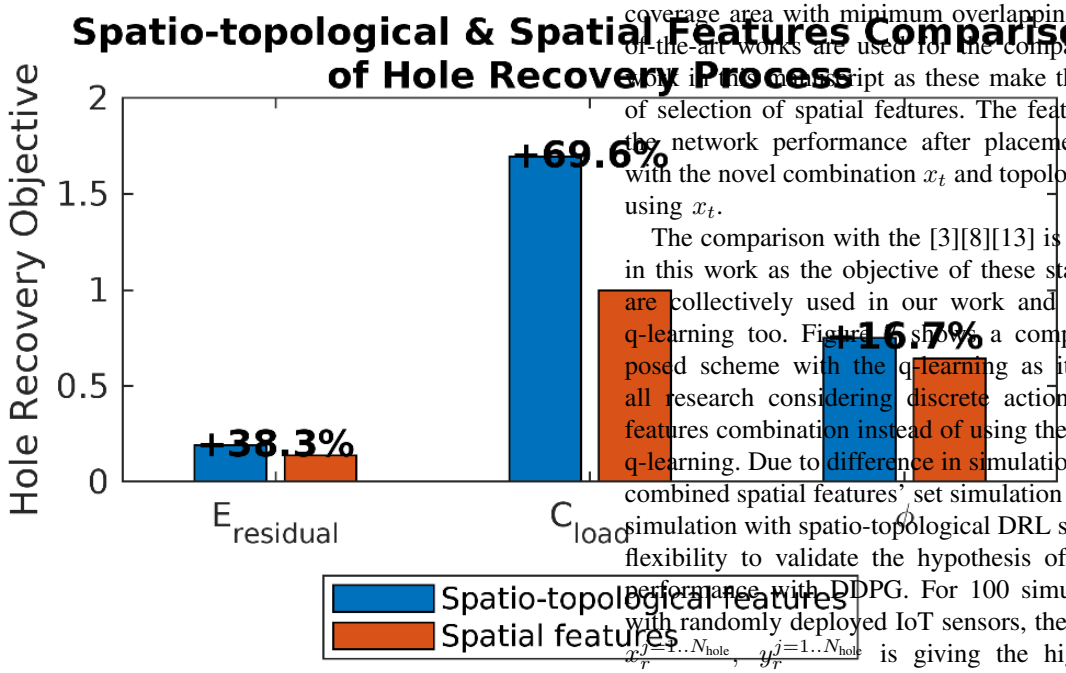

Figure 6: Comparison of spatial and spatio-topological features impact on the hole coverage performance

Table 1. Training hyperparameters of 6G IoT DDPG network

[]@ll@ Sample Time	1
Target Smooth Factor	10^{-3}
Discount Factor	0.99
Mini Batch Size	64
Experience Buffer Length	10^{-6}
Actor learning rate	0.01
Critic learning rate	0.01

The proposed algorithm is evaluated on different number of network size, transmission range and number of holes. The state-of-the-art works as discussed earlier have used either reinforcement leaning and heuristic optimization algorithms. In the work of [3][8][13], the action variables are redundant nodes' location however these differ in the s_t and r_t definition. In [3], the reward is being calculated with maximum coverage area and minimum energy consumption in the mobile redundant node. Though, this work has ignored the overall performance of the network which is defined by the energy residual of the relay node (cluster head) and dependent on the uniform load distribution. The discrete action space makes it more prone to fall in local minima as q-learning has been used in the work. Q-learning defines a discrete action space within a Q-table, which limits its adaptability to the dynamic network topology of 6G. The objective of the hole healing in [8] is to maximize the working time of the covered subject which is again possible by minimum energy consumption and fulfilled by the q-learning itself. The work in [14] is targeting to maximize the coverage area with minimum overlapping area. These state-of-the-art works are used for the comparison of suggested work in this manuscript as these make the founding ground of selection of spatial features. The feature gap to increase the network performance after placement of N_r is filled with the novel combination x_t and topological characteristics using x_t.

The comparison with the [3][8][13] is shown qualitatively in this work as the objective of these state-of-the-art works are collectively used in our work and simulated with the q-learning too. Figure 6 shows a comparison of the proposed scheme with the q-learning as it has been used in all research considering discrete action space and spatial features combination instead of using them individually with q-learning. Due to difference in simulation environments, the combined spatial features' set simulation with q-learning and simulation with spatio-topological DRL simulation, gives the flexibility to validate the hypothesis of novel features set performance with DDPG. For 100 simulation time stamps with randomly deployed IoT sensors, the proposed scheme's $x_r^{j=1..N_{\text{hole}}}$, $y_r^{j=1..N_{\text{hole}}}$ is giving the highest E_{residual} over different number of holes in the area.

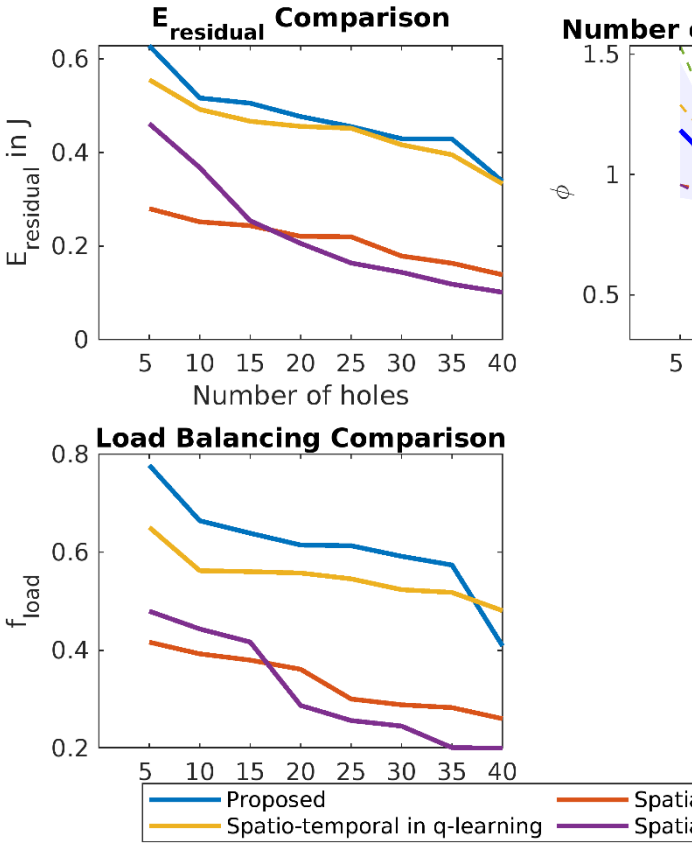

Figure 7: Performance Comparison of Proposed methodology with the spatial features with DRL placement, and Q-learning as the training agent with both spatial and spatio-topological features with varying number of holes

In the figure 7 analysis with varying number of holes, the reward r_t in equation 12 is maximized. The residual energy

in the proposed spatio-topological features and DDPG training is highest, however due to probabilistic RL and random deployment of the sensors, the improvement is non-uniform with varying number of holes. The second-best residual energy is achieved when the same proposed features set are trained by q-learning considering the discrete action space limiting the exploration of the search area by the agent. The maximum improvement of E_{res} using proposed scheme is 13.1% and minimum is 0.8%. The E_{res} is not monotonically decreasing due to different spatial distribution of holes and constant N_r irrespective of the number of holes. The C_{critical} and ϕ are correlated defined by equation 7. This equation is the sum of all nodes in the sensing range or in other terms, it is giving the number of neighbours. Higher the number of neighbours, better the location to place the recovery node. The subplot 2 in the figure 7 shows the reduction in neighbourhood degree with the increase in holes in the network; however, no single approach is showing the consistency and variance is also very less, so the range of the change in the neighbouring connections w.r.t. holes is observed by the envelope and mean line indicates the expected behaviour. On the contrary of this parameter performance, the f_{load} has shown the significant improvement in the proposed spatio-topological features with DDPG training, than the q training. It has been validated earlier in figure 6 too that the spatial features only are less performing than the spatio-topological features, so is validated again in figure 7 too. A maximum improvement of f_{load} in figure 7 is 19.4% and minimum is 10.62%. This significant loading factor improvement is achieved with the topological characteristics of nodes' spatial features with GCN.

This behaviour of the methods used in comparison can be better explained within the context of our multi-objective framework. As demonstrated in the residual energy and load balancing comparisons, the proposed method achieves superior performance in these metrics while maintaining competitive connectivity. The envelope representation (dashed lines) captures the inherent variability due to topological transformations as hole density increases. Particularly noteworthy is proposed method's ability to maintain connectivity comparable to Spatio-temporal in q-learning' while delivering 13.1% better energy efficiency and 19.4% improved load balancing at moderate hole densities (15-25 holes). This represents a favourable trade-off in practical WSN deployments where network longevity is often prioritized over maximum connectivity.

The simulation of figure 7 is done with 100 nodes distributed in the area of $200 \times 200m^2$. The smaller search area gives the advantage to discrete action space too; however, the proposed model free training advantage is clearly validated in figure 8 in higher distributed area with varying number of sensor density.

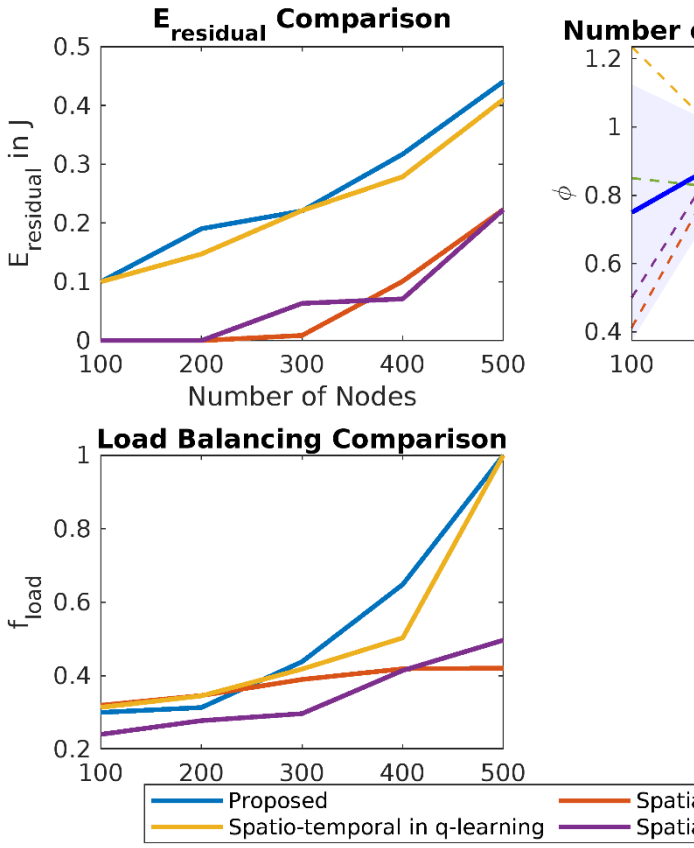

Figure 8: Performance comparison of proposed methodology with the q learning and spatial features trained on DDPG and Q learning for varying sensor density

The analysis is done considering the number of holes where the results in figure 7 are under performing. Figure 8 keeps the $N_{\text{hole}} = 40$ for varying sensor densities. The E_{res} is increasing with the increase in number of nodes due to stable connectivity as validated from the neighbouring curve. The neighbours of the cluster heads are stable due to spatio-topological features. It indicates the uniform distribution of load too. The f_{load} is increasing due to equation 5 which is using energy in the calculation. The energy residual has increased with number of nodes due to fixed deployment area. With the increase in node density, the distance between nodes reduces and hence energy consumption as in equation 6. The proposed scheme has shown the nonlinear improvement from 200-400 nodes' densities and this range can be considered as the optimal density for the stable connectivity and performance in 6G IoT. The shortcoming of the proposed spatio-topological features with DDPG placement of redundant nodes is validated from figure 8 on a larger scale with 40 failed nodes in the network creating the connectivity holes.

The coverage hole healing in 6G IoT is proposed in this work. A novel approach using spatio-topological features have been proposed to make the dependent decision on

neighbouring sensors to the redundant nodes. The continuous action space as the location of redundant nodes in the DDPG training algorithm is defined which makes it model free and adaptive to higher sensor nodes' density too. Topological characteristics of GCN have been used as state space variables. The advantage of the proposed methodology is validated by simulating under various environmental conditions of varying number of holes and varying nodes density. A significant improvement of 13.1% in the proposed scheme than the state-of-the-art q-learning scheme for the holes density of 5 is achieved and 0.81% in the residual energy for the higher hole density of 35 holes in the network. Similarly, the 19.4% improvement in the load balancing is noticed with stable connectivity. Furthermore, the simulation experiment with varying sensor density for 40 holes have shown the similar behaviour of performance improvements. Considering the comparison of proposed saptio-topological features and spatial features, the location criticality also shown an improvement of 16.7%.

Although the significant improvement with the usage of DDPG in spatio-topological scenario has been observed yet there is still scope for the improvement for 6G scalability. Present work has shown the stable performance between the 200-400 nodes density which can be improved if topological features with more than single neighbour is considered. Furthermore, the multiagent DRL can also be helpful in handling higher hole area in the network.

[1] Lee, Howon, Byungju Lee, Heecheol Yang, Junghyun Kim, Seungnyun Kim, Wonjae Shin, Byonghyo Shim, and H. Vincent Poor. "Towards 6G hyper-connectivity: Vision, challenges, and key enabling technologies." Journal of Communications and Networks 25, no. 3 (2023): 344-354.

[2] Cui, Qimei, Xiaohu You, Ni Wei, Guoshun Nan, Xuefei Zhang, Jianhua Zhang, Xinchen Lyu et al. "Overview of AI and communication for 6G network: fundamentals, challenges, and future research opportunities." Science China Information Sciences 68, no. 7 (2025): 171301.

[3] Xia, Yunzhi, Xianjun Deng, Lingzhi Yi, Laurence T. Yang, Xiao Tang, Chenlu Zhu, and Zhongping Tian. "AI-driven and MEC-empowered confident information coverage hole recovery in 6G-enabled IoT." IEEE Transactions on Network Science and Engineering 10, no. 3 (2022): 1256-1269.

[4]. Gupta, Abhishek, Somesh Kumar, and Manisha Pattanaik. "Coverage hole detection using social spider optimized Gaussian Mixture Model." Journal of King Saud University-Computer and Information Sciences 34, no. 10 (2022): 9814-9821.

[5]. Masoud, Mohammad Z., Yousef Jaradat, Ismael Jannoud, and Mustafa A. Al Sibahee. "A hybrid clustering routing protocol based on machine learning and graph theory for energy conservation and hole detection in wireless sensor network." International Journal of Distributed Sensor Networks 15, no. 6 (2019): 1550147719858231.

[6]. Koriem, Samir M., and Mohamed A. Bayoumi. "Detecting and measuring holes in wireless sensor network." Journal of King Saud University-Computer and Information Sciences 32, no. 8 (2020): 909-916.

[7]. Beghdad, Rachid, and Amar Lamraoui. "Boundary and holes recognition in wireless sensor networks." Journal of Innovation in Digital Ecosystems 3, no. 1 (2016): 1-14.

[8]. Tang, Yong, Xianjun Deng, Lingzhi Yi, Yunzhi Xia, Laurence T. Yang, and Xiao Tang. "Collaborative intelligent confident information coverage node sleep scheduling for 6G-empowered green IoT." IEEE Transactions on Green Communications and Networking 7, no. 2 (2022): 1066-1077.

[9]. Hallafi, Ali, Ali Barati, and Hamid Barati. "A distributed energy-efficient coverage holes detection and recovery method in wireless sensor networks using the grasshopper optimization algorithm." Journal of Ambient Intelligence and Humanized Computing 14, no. 10 (2023): 13697-13711.

[10]. Ma, Donghui, and Qianqian Duan. "A hybrid-strategy-improved butterfly optimization algorithm applied to the node coverage problem of wireless sensor networks." Math. Biosci. Eng 19, no. 4 (2022): 3928-3952.

[11]. Chen, Xinyi, Mengjian Zhang, Ming Yang, and Deguang Wang. "NHBBWO: A novel hybrid butterfly-beluga whale optimization algorithm with the dynamic strategy for WSN coverage optimization." Peer-to-Peer Networking and Applications 18, no. 2 (2025): 1-25.

[12]. Hajjej, Faten, Monia Hamdi, Ridha Ejbali, and Mourad Zaied. "A distributed coverage hole recovery approach based on reinforcement learning for Wireless Sensor Networks." Ad Hoc Networks 101 (2020): 102082.

[13]. Chauhan, Nilanshi, Piyush Rawat, and Siddhartha Chauhan. "Reinforcement learning-based technique to restore coverage holes with minimal coverage overlap in wireless sensor networks." Arabian Journal for Science and Engineering 47, no. 8 (2022): 10847-10863.

[14]. Sharma, Anamika, and Siddhartha Chauhan. "A distributed reinforcement learning based sensor node scheduling algorithm for coverage and connectivity maintenance in wireless sensor network." Wireless Networks 26, no. 6 (2020): 4411-4429.

[15]. Gou, Pingzhang, Gang Mao, Fen Zhang, and Xiangdong Jia. "Reconstruction of coverage hole model and cooperative repair optimization algorithm in heterogeneous wireless sensor networks." Computer Communications 153 (2020): 614-625.

[16]. Yang, Chen, Shengchao Su, Xiang Ju, and Jiayan Song. "A mobile sensors dispatch scheme based on improved SOM algorithm for coverage hole healing." IEEE Sensors Journal 21, no. 18 (2021): 21080-21089.

[17]. Ndjiongue, Alain R., Octavia A. Dobre, and Hyundong Shin. "On-Demand RIS-Assisted Free Space Optical

Access System for 6 G Networks." IEEE Transactions on Vehicular Technology (2025).

[18]. Jiang, Bo, Ziyan Zhang, Doudou Lin, Jin Tang, and Bin Luo. "Semi-supervised learning with graph learning-convolutional networks." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 11313-11320. 2019.

[19]. Mehta, Shalu, and Amita Malik. "A Swarm Intelligence Based Coverage Hole Healing Approach for Wireless Sensor Networks." EAI Endorsed Transactions on Scalable Information Systems 7, no. 26 (2020).

[20]. Huang, Liping, Jianbin Zheng, Yifan Gao, Qiuzhi Song, and Yali Liu. "A Lower Limb Exoskeleton Adaptive Control Method Based on Model-free Reinforcement Learning and Improved Dynamic Movement Primitives." Journal of Intelligent & Robotic Systems 111, no. 1 (2025): 24.

[21]. Maan, Ujjawal, and Yogesh Chaba. "Deep Q-network based fog node offloading strategy for 5 G vehicular Adhoc Network." Ad Hoc Networks 120 (2021): 102565.

[22]. Saleem, Rabbia, Wei Ni, Muhammad Ikram, and Abbas Jamalipour. "Deep-reinforcement-learning-driven secrecy design for intelligent-reflecting-surface-based 6G-IoT networks." IEEE Internet of Things Journal 10, no. 10 (2022): 8812-8824.

[23]. Kumari, Ashish, Shailender Kumar, and Ram Shringar Raw. "Modified clustering and incentivized stable CH selection for reliable VANET communication." *Cluster Computing* 27, no. 9 (2024): 11983-12005.

[24]. Fofana, Adama Hawa Rahima, and Jianjun Lei. "Reinforcement Learning Based Cluster Formation Algorithm in Wireless Sensor Networks." In Proceedings of the 2nd International Conference on Signal Processing, Computer Networks and Communications, pp. 367-371. 2023.