# A Deep Learning Clustering Beamforming Approach for Future 6G Mobile Ad Hoc Networks

**Vincenzo Inzillo**

vincenzo.inzillo@tecnologicovibo.edu.it

IIS-ITG-ITI VV

**David Garompolo**

IIS-ITG-ITI VV

**Research Article**

**Keywords:** Clustering, MANET, 802.11ay, Beamforming, Massive MIMO

**Posted Date:** August 2nd, 2023

**DOI:** https://doi.org/10.21203/rs.3.rs-3205106/v1



**Additional Declarations:** No competing interests reported.

# A Deep Learning Clustering Beamforming Approach for Future 6G Mobile Ad Hoc Networks

Dr Vincenzo Inzillo PhD[1*] and Dr David Garompolo PhD[1†]

[1*]Department of Computer Science, IIS-ITG-ITI VV, Via G. Fortunato, Vibo Valentia, 89900, Italy.

*Corresponding author(s). E-mail(s): vincenzo.inzillo@tecnologicovibo.edu.it;
Contributing authors: david.garompolo@tecnologicovibo.edu.it;
†These authors contributed equally to this work.

**Abstract**

The rapid advancement of wireless technologies and the increasing demand for seamless, high-speed connectivity has spurred the need for efficient routing protocols in 6G Mobile Ad Hoc Networks (MANETs). This paper proposes a modern approach, the Deep Learning Clustering Beamforming Massive MIMO Routing Protocol (DLCB), that is designed to achieve high level throughput by exploiting beamforming advantages and for enhancing communication efficiency for future next generation 6G MANETs. The DLCB algorithm establishes robust and efficient routes for data transmission by adopting a clustering strategy based on deep learning allowing to select cluster heads by taking advantage of Graph Neural Networks (GNN) features and creating effective cluster paths. Furthermore, this approach integrates IEEE 802.11ay and Massive MIMO technologies to exploit their benefits in terms of high-speed data transmission, improved network capacity, and better signal quality. DLCB aims to facilitate seamless data transmission in dynamic network environments, paving the way for enhanced network performance in 6G MANETs. Experimental evaluations and simulations are conducted to validate the performance of the proposed approach through the use of Omnet++ Network Simulator.

**Keywords:** Clustering, MANET, 802.11ay, Beamforming, Massive MIMO

# 1 Introduction

Over recent years, the research interest in Mobile Ad Hoc Networks (MANETs) has been escalating due to their self-configuring capacities and versatile network topologies. MANETs are composed of mobile nodes, dynamically forming a network that fosters communication without necessitating a fixed infrastructure. They have become instrumental in various areas such as disaster relief, military operations, vehicular communications, among others. However, the development of an efficient, robust routing protocol for MANETs poses a challenge, given their highly dynamic and decentralized character.

Fifth-generation (5G) wireless networks have instigated substantial advancements regarding data rates, latency, and connection density, paving the way for new applications and services. Still, the burgeoning demand for high-speed wireless communications and the advent of new technologies, like the Internet of Things (IoT) and autonomous vehicles, necessitates further advancements in wireless networks beyond 5G [1–4].

Anticipated to offer unmatched performance levels, the sixth-generation (6G) wireless networks promise higher data rates, lower latency, and enhanced reliability [5, 6]. To meet the next generation wireless networks' standards, particularly regarding data rate and Signal To Noise Ratio (SNR), the IEEE 802.11ac and IEEE 802.11ax standards have been widely adopted in MANET environments [7]. These modifications, however, cap the maximum data rate at roughly 10 Gbps, potentially making them less suitable for evaluating 6G scenarios [8]. The IEEE 802.11ay standard, also known as Next Generation 60 GHz (NG60) [9, 10], is the on of the latest amendment to the IEEE 802.11 standard, specifically designed for high-speed wireless communication. 802.11ay offers a theoretical maximum data rate up to 100 Gbps in LoS (Line Of Sight) conditions, channels weekly affected from noise [11], and improved quality of service (QoS), making it a promising technology for supporting high-throughput applications in MANET [9, 10]. This enhancement in data rates enables us to meet the increasing demand for high-speed wireless communication in 6G networks [12, 13].

Although a large number of approaches exploit benefit of clustering for the most recent MANET contexts because of its advantages resulting in high level network managing [14] it provides several issues to be addressed; for example, hierarchical strategies could involve significant routing overhead [15] while deterministic approaches could lead to scalability troubles and use of specific cluster shapes [16, 17]. Advancements in deep learning technologies, specifically Graph Neural Networks (GNN), offer a robust alternative to traditional clustering techniques. Unlike deterministic algorithms that impose rigid rules, GNNs model the complex relationships in data by exploiting the power of neural networks, which enable them to adaptively learn from data, yielding more flexible and accurate clustering results [18]. GNNs are particularly adept at handling relational or graph-structured data, making them an ideal choice for applications where data entities have interconnected relationships. Therefore, In the context of MANET GNN can be used as effective mobility prediction strategy [19]. Furthermore, GNNs exhibit resilience to the curse of dimensionality, a common problem faced by traditional clustering techniques when dealing with high-dimensional data. By exploiting their inherent capability to capture complex, high-level feature representations, GNNs

can navigate high-dimensional data spaces with ease, generating meaningful clusters that effectively capture the intrinsic structure of the data. Another prominent advantage is the ability of GNNs to handle noisy data and outliers. Their learning capability allows them to detect and ignore irrelevant noise or anomalies that might otherwise distort the clusters in traditional techniques. This results in more reliable and robust clusters, even in the presence of imperfect data. Additionally, GNN-based techniques naturally lend themselves to scalability, a trait that is crucial for handling actual vast and ever-growing data volumes. Overall, by harnessing the power of deep learning and the inherent structure of graph data, GNN-based clustering techniques offer a flexible, scalable, and robust approach to data clustering. As such, they hold great promise for a range of applications, from social network analysis and bioinformatics to communication networks and beyond.

In this context, we propose a Deep Learning Clustering Beamforming approach (DLCB) as an innovative solution aiming to satisfy next generation MANET communications requirements. DLCB exploits Neural Networks fundamentals to form efficient intra-cluster and inter-cluster communication paths, thereby optimizing the network routing decisions. In order to satisfy next generation MANET requirements, our approach also involves the use of Massive MIMO (Multiple Input Multiple Output) technology, which provides for a large number of antennas at the base station or access point [20, 21]. This technology enables simultaneous communication with multiple users and allow to achieve efficient performance at very high data rates if it is employed in combination with 802.11ay [22, 23], thereby increasing the network capacity and spectral efficiency [24]. By utilizing Massive MIMO, our DLCB algorithm could optimize the utilization of the available resources and enhances the overall network performance in terms of data rates, throughput, and reliability. DLCB also incorporates a beamforming optimization process aiming to maximize the probability that a CH beamforms a large number of nodes in order to achieve high throughput performance. Overall network performance of DLCB are evaluated by using extensive simulations in OMNeT++ that is one of the most common and used discrete event network simulator supporting a large suite of network protocols and standards [25]. Finally, he integration of these advanced technologies could facilitate the demanding requirements for 6G networks and support a wide range of applications and services to be integrated with very common employed techniques such as Terahertz communications [26, 27].

The remainder of this paper is organized as follows. Section II provides an overview of related works by emphasizing the most recent and novel cluster-based approaches proposed for future MANET communications. In Section III main features of IEEE 802.11ay standard are highlighted. The Section IV illustrates an overall comparison between DLCB and CBPR approaches; Section V describes the proposed DLCB protocol by detailing all phases of which it is composed; Section VI presents the simulation setup and experimental results by considering network performance. Finally, Section VII concludes the paper, summarizes the contributions, and discusses future research directions. Table 1 includes the list of main symbols used in the paper.

**Table 1**: List of main Symbols used in the paper

| Parameter | Value |
|---|---|
| $M$ | number of antenna elements in the Massive MIMO |
| $d$ | CH desired beam direction |
| $k$ | minimum number of nodes to be covered by a CH |
| $\alpha$ | scaling factor |
| $N$ | total number of nodes |
| $C$ | total number of clusters |
| $n_c$ | number of nodes in a given cluster |
| $N_x$ | number of antenna elements in the URPA x-axis |
| $N_y$ | number of antenna elements in the URPA y-axis |
| $\theta$ | azimuth angle defining the beamforming direction |
| $\phi$ | elevation angle defining the beamforming direction |
| $\mathbf{h}_{m,n}(\theta, \phi)$ | Channel from the $(m, n)$th antenna element |
| $m, n$ | Position of an antenna element in the array |
| $ndist$ | Normalized distance between antenna elements (relative to wavelength) |
| $\mathbf{y}m, n$ | Received signal at the $(m, n)$th antenna element |
| $\mathbf{x}$ | Transmitted signal |
| $\mathbf{n}_{m,n}$ | Noise at the $(m, n)$th antenna element |
| $\mathbf{w}m, n$ | Weights applied to each antenna element |
| $P(d)$ | probability that the Massive MIMO beamforms towards $d$ |
| $P(\theta, \phi)$ | power pattern of the URPA |
| $\Omega(d)$ | solid angle corresponding to the direction $d$ |
| $\Omega_{\text{total}}$ | total solid angle covered by the array |
| $P_{\text{total}}$ | total power radiated by the array |
| $P_{\max}$ | maximum allowed power radiation |
| $E_i$ | current energy level of node $i$ |
| $N_i$ | number of neighboring nodes of node $i$ |
| $S_i$ | average received power measured by node $i$ |
| $F_i$ | feature vectors for each node $i$ |
| $P_i$ | predicted probability that node $i$ should be selected as a CH |
| $y_i$ | true label for node $i$ (1 if node $i$ is a CH, 0 otherwise) |
| $L$ | binary cross-entropy loss function |
| $h_i$ | node embedding for node $i$ |
| $f_i$ | feature vector of node $i$ |
| $W, b, W_o, b_o$ | weights and biases of the GNN |
| $msg_i$ | message that node $i$ receives from its neighbors |
| $P(GNN)_i$ | output probabilities of the GNN for node $i$ |
| $t$ | probability threshold for CH election |

# 2 Related Works

Many clustering-based routing protocols have been proposed for MANETs, with the purpose of addressing issues such as scalability, network lifetime, and energy efficiency. In this section, we review some of the most relevant works in the literature. In [28], Giordano et al. propose a hierarchical routing protocol for MANETs that employs a cluster-based approach. The protocol creates a hierarchical structure of clusters with a cluster head (CH) selected for each cluster. The CH is responsible for collecting and aggregating data from its cluster members and forwarding it to the base station. The authors evaluated the performance of the protocol using simulation and found

that it outperformed other non-hierarchical protocols in terms of energy efficiency and network lifetime.

Another approach presented in [29] is a novel hierarchical routing protocol for large-scale MANETs, which is based on a grid-based clustering approach. The protocol divides the network into a set of grids and selects a cluster head for each grid. The cluster heads communicate with each other to maintain the network topology and exchange control messages. Simulation results demonstrated that this protocol outperforms other clustering-based protocols in terms of scalability and network lifetime.

In [30], Wang et al. propose a hierarchical routing protocol that is designed to be energy-efficient and adaptive to changes in network topology. The protocol employs a multi-level clustering approach, where clusters are formed at different levels of the network hierarchy. Simulation results showed that this protocol outperforms other hierarchical and non-hierarchical protocols in terms of energy efficiency and network lifetime.

Another hierarchical routing protocol proposed in [31] is based on a hybrid clustering approach. The protocol divides the network into a set of clusters, each with a cluster head, and further divides each cluster into sub-clusters, each with a sub-cluster head. The sub-cluster heads communicate with each other to maintain the network topology and exchange control messages. Simulation results demonstrated that this protocol outperforms other clustering-based protocols in terms of scalability and network lifetime.

In [32] author propose a method in which cluster nodes and optimal routes are elected based on low-energy data transfer. The method employs the concept of SCH (Super Cluster Head). In particular, the approach is based on two main algorithms TSR (Trees Social Relation) and LDHA (Leader Dolphins Herd Algorithm) along with a multi-objective function to address the issue of MANET high energy usage. However, author in this case aim to focus on network energy saving rather than maximizing throughput and end to end delay.

Cluster-Based Backpressure Routing (CBPR) is a well-known clustering-based protocol in the literature [33], widely acknowledged for its efficient utilization of clustering techniques in Wireless Sensor Networks (WSNs) and the Internet of Things (IoT) applications. The purpose of CBPR is to address the unique requirements and challenges of WSNs and IoT by utilising clustering techniques to improve network scalability, resource utilization, and energy management. CBPR aims to dynamically adapt to changes in network topology and cluster formations through cluster merging and splitting mechanisms, ensuring efficient routing and effective resource allocation. It incorporates load balancing, fault tolerance, and energy management mechanisms to enhance the overall performance of the network in terms of scalability and lifetime. Although CBPR is primarily designed for WSNs and IoT, because of its clustering-based approach and its focus on efficient resource management, make it comparable to our Clustering Beamforming Massive MIMO Routing Protocol (DLCB) in the context of MANETs, enabling valuable insights into the transferability and adaptability of clustering-based approaches across different network types.

# 3 802.11ay Communications and Massive MIMO for DLCB in 6G Networks

The DLCB approach targets optimizing routing and communication for next generation of MANET, which are projected to function at higher frequencies and offer unparalleled data rates. The choice of suitable communication technology is crucial to achieving efficient and reliable performance in such networks. While Terahertz (THz) communication models present potential in delivering ultra-high data rates, they carry restrictions like propagation losses and limited coverage range. In this scenario, 802.11ay, the successor of 802.11ad, emerges as a potential alternative, especially when augmented with Massive MIMO technology.

802.11ay is an evolved version of the Wi-Fi standard, introducing considerable enhancements over its predecessors, including substantial improvements in the physical layer. These improvements facilitate 802.11ay to deliver higher data rates, superior spectral efficiency, and better performance in high-density environments, even in MANET scenarios where devices are moving and changing their position frequently. However, these high data rates can be achievable under certain conditions, primarily when there is LoS between the devices. Some of the key features and specifications of 802.11ay include:

- **Maximum Data Rate:** 802.11ay supports a maximum data rate of up to 100 Gbps in ideal conditions, providing a tremendous leap compared to the maximum data rate of 802.11ax. This high data rate enables faster and more efficient transmission of large volumes of data, aligning with the evolving applications in 6G networks.
- **Channel Bonding and MIMO:** 802.11ay utilizes channel bonding to combine four 2.16 GHz channels into a single channel, and MIMO techniques to increase the capacity and reliability of wireless links. However, the effective use of MIMO is challenged by the mobility and lack of stable LoS connections in MANETs.
- **Beamforming:** 802.11ay employs advanced beamforming techniques to overcome the high signal attenuation at 60 GHz. Beamforming focuses the wireless signal towards the intended receiver, improving signal quality, range, and overall network performance.

When augmented with Massive MIMO technology, 802.11ay capabilities can be significantly amplified, providing further benefits for DLCB in 6G networks. Massive MIMO involves the deployment of a large number of antennas at the base station or access point, allowing for simultaneous communication with multiple users, thereby increasing the network capacity and spectral efficiency. However, the implementation of Massive MIMO with 802.11ay requires careful consideration and presents its own set of challenges:

- **Spatial Multiplexing:** Massive MIMO enables spatial multiplexing of multiple data streams, significantly increasing the data rate of 802.11ay by permitting simultaneous transmission and reception of multiple data streams. However, this requires careful alignment and coordination of the antennas.
- **Beamforming:** Massive MIMO facilitates enhanced beamforming capabilities. Improved signal quality, increased range, and mitigation of the high atmospheric

absorption experienced by 802.11ay signals in the 60 GHz frequency band can be achieved. However, acquiring precise knowledge of the channel state information, which is necessary for beamforming, can be challenging in a dynamic MANET environment.

- **Interference Management:** The deployment of multiple antennas in Massive MIMO can help manage and reduce interference. Techniques like null steering can improve overall network performance of 802.11ay in high-density MANETs.
- **Network Capacity:** Massive MIMO significantly increases the network capacity of 802.11ay by serving multiple users simultaneously in the same frequency band, enhancing the scalability of DLCB in 6G networks.

To summarize, the advanced features and improvements in the physical layer of 802.11ay, especially when augmented with Massive MIMO, enable faster data rates, better spectral efficiency, and enhanced performance, making it a suitable choice for supporting the communication needs of DLCB in next-generation wireless networks. However, the practical implementation of Massive MIMO with 802.11ay requires careful planning, calibration, and synchronization of the antennas. The performance gains will also largely depend on the ability to maintain stable wireless links and accurately estimate the channel state information in a dynamic MANET environment.

# 4 Comparison of DLCB and CBPR

While DLCB is primarily designed for Mobile Ad Hoc Networks (MANETs), and CBPR is specifically designed for the Internet of Things (IoT), it is valuable to compare these two protocols to understand their respective strengths, weaknesses, and potential improvements. CBPR is a routing protocol specifically developed for IoT environments, aiming to address the unique challenges and requirements of IoT networks. Although DLCB and CBPR are designed for different network types, a comparative analysis can provide valuable insights into their routing mechanisms, cluster-based approaches, and potential transferability of ideas.

Here, we highlight the key points of comparison between DLCB and CBPR:

1. Routing Objectives: DLCB and CBPR aim to optimize routing decisions to improve network performance and efficiency. While DLCB focuses on maximizing the total throughput of the MANET, CBPR adopts a backpressure-based approach to balance traffic across clusters and minimize delays in IoT networks. By comparing these objectives, we can assess the applicability of backpressure-based techniques in MANETs and evaluate the trade-offs between throughput optimization and delay minimization.
2. Cluster Formation: Both DLCB and CBPR employ a cluster-based approach to enhance routing efficiency. DLCB promote smart clustering formation and CH election strategies. On the other hand, CBPR utilizes clusters in IoT networks to enable distributed coordination and load balancing. By comparing the cluster formation mechanisms, we can identify similarities and differences in cluster construction, cluster head selection, and cluster management techniques.
3. Resource Constraints: MANETs and IoT networks often operate under resource-constrained environments. DLCB and CBPR consider these constraints in their

routing mechanisms but with different emphases. DLCB aims to optimize the utilization of network resources such as link capacities, transmission powers, and channel gains by exploiting benefits of beamforming and Spatial Division Multiple Access (SDMA). CBPR, on the other hand, focuses on balancing traffic and reducing congestion in resource-constrained IoT environments. By comparing these approaches, we can gain insights into resource optimization techniques tailored for different network types.

Because CBPR is mainly designed for WSN and not for MANET is important to clarify a possible comparison scenario for evaluating main strengths and weakness over our DLCB approach. However, especially relating to Routing Protocol WSN and MANET purposes are easily overlapping. The reference scenario we considered includes the following main features:

- Node Mobility: In this scenario, we assume low node mobility, where the nodes are relatively stationary or have limited movement within the network. This assumption is crucial for evaluating the performance of routing protocols in stable network conditions. By focusing on stable network scenarios, we can assess the efficiency and effectiveness of the routing protocols without the confounding factors introduced by highly dynamic network topologies.
- Communication Links: The communication links between nodes are wireless links, representing the wireless connectivity in a multi-hop network. These links are subject to factors such as signal strength, interference, and path loss, which can significantly impact the quality and reliability of the communication. Both DLCB and CBPR operate in this wireless communication environment, and their respective mechanisms must contend with these challenges.
- Data Rate: Both DLCB and CBPR are designed to support high data rate requirements. However, CBPR does not support Very High Troughput (VHT) or Enhanced Directional Multi Gigabit (EDMG) communications; on the other hand DLCB is perfectly suitable to satisfy 5G and 6G MANET PHY specifications.

However, it is important to note that the comparison results are specific to the considered scenario and may vary in different network conditions, topologies, and application requirements. Finally in Table 2, we summarize possible improvements of DLCB over CBPR.

# 5 Proposed approach

The DLCB approach consists of 5 distinct phases each one one having a specific task. The first two phases exploit benefit of GNN for CH election and clusters formation; other phases involve the use of beamforming in order to achieve high level of efficiency relating to routing process. In this section, we explain DLCB phases in detail.

**Table 2**: DLCB and CBPR comparison features

| DLCB | CBPR |
|---|---|
| • Deep Learning based Clustering | • Hierarchical based Clustering |
| • Supports Dynamic Network Topologies and Cluster Formations | • Limited Adaptability to Changing Topologies and Cluster Structures |
| • Potential for Integration with Advanced Technologies | • Primarily Designed for WSNs and IoT, Limited Integration Potential |
| • Hybrid and dynamic beamforming (Massive MIMO + omni) | • Static beamforming (omni) |
| • Beam optimization | • No radiation pattern optimization |
| • Dynamic routing depending on radiation pattern | • Static routing |

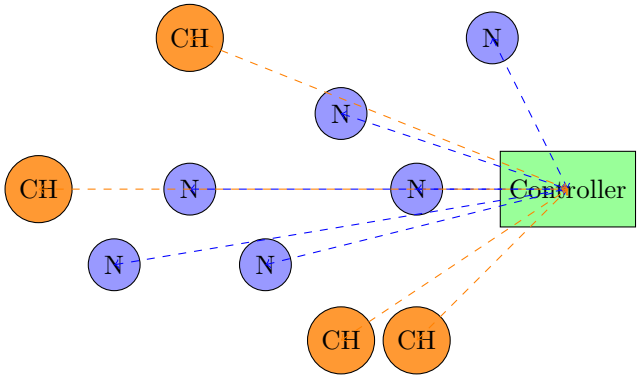


**Fig. 1**: DLCB operation principle

In the Figure 1, we can observe a detailed representation of the operation principle of a Distributed Load Cooperative Beamforming (DLCB) system. The scenario consists of multiple nodes (N), cluster heads (CH), and a centralized Controller. The nodes, denoted by the blue circles, represent individual devices or elements within the network. Their functionality includes data communication with the controller as well as other nodes. The network also consists of specific nodes known as cluster heads, represented by orange circles. These cluster heads are essentially nodes with additional responsibilities such as coordinating the activities of the nodes within their respective clusters. The central entity in the diagram is the Controller, represented by the green rectangle. Its primary role is to control and manage the entire network operation. In the context of a DLCB, after the cluster heads are elected and the clusters are formed, the controller enables the use of Massive MIMO URPA for each cluster head. The dashed lines connecting the controller to the nodes and the cluster heads depict the radiation pattern or communication links in the network. The blue and orange dashed lines represent the communication from the regular nodes and cluster heads to the controller, respectively. The cluster heads, employing Massive MIMO URPA facilitated by the controller, can then effectively manage the nodes within their clusters, ensuring efficient and cooperative beamforming.

## 5.1 Phase 1: Selection of Cluster Heads

The selection of Cluster Heads (CHs) is a crucial step in clustering-based routing protocols in MANETs. It significantly influences the network overall efficiency. To optimize this process, we utilize a Graph Neural Network (GNN). A GNN is an artificial neural network specifically designed to operate on graph structures. Unlike traditional neural networks, a GNN can handle relational data, effectively capturing complex interactions between nodes (devices in a MANET). The GNN employs a process called 'message passing,' in which each node aggregates feature information from its neighbors and updates its features iteratively.

For the purpose of training our GNN model, we utilize a representative dataset of a MANET. This dataset comprises of 10,000 network states, each represented as a graph. In each graph, nodes are characterized by their features: current energy level ($E_i$), the number of neighboring nodes ($N_i$), and the average received power measured ($S_i$), obtained by considering the Received Signal Strength (RSS). These features are evaluated during the initial 'hello' and discovery packet exchange phase. Based on the energy efficiency of the configuration, we label each node as either a cluster head (1) or a regular node (0).

```
[
    {
        "state_id": 1,
        "nodes": [
            {"id": 1, "E_i": 0.75, "N_i": 5, "S_i": 0.66, "label": 1},
            {"id": 2, "E_i": 0.70, "N_i": 3, "S_i": 0.60, "label": 0},
            {"id": 3, "E_i": 0.80, "N_i": 4, "S_i": 0.72, "label": 1},
            {"id": 4, "E_i": 0.65, "N_i": 3, "S_i": 0.65, "label": 0},
            {"id": 5, "E_i": 0.68, "N_i": 6, "S_i": 0.70, "label": 0}
        ]
    },
    {
        "state_id": 2,
        "nodes": [
            {"id": 1, "E_i": 0.72, "N_i": 4, "S_i": 0.62, "label": 1},
            {"id": 2, "E_i": 0.70, "N_i": 3, "S_i": 0.61, "label": 0},
            {"id": 3, "E_i": 0.74, "N_i": 6, "S_i": 0.68, "label": 1},
            {"id": 4, "E_i": 0.68, "N_i": 2, "S_i": 0.60, "label": 0},
            {"id": 5, "E_i": 0.69, "N_i": 5, "S_i": 0.71, "label": 0}
        ]
    },
    {
        "state_id": 3,
        "nodes": [
            {"id": 1, "E_i": 0.70, "N_i": 5, "S_i": 0.64, "label": 0},
            {"id": 2, "E_i": 0.76, "N_i": 4, "S_i": 0.65, "label": 1},
            {"id": 3, "E_i": 0.75, "N_i": 3, "S_i": 0.66, "label": 1},
            {"id": 4, "E_i": 0.67, "N_i": 2, "S_i": 0.63, "label": 0},
            {"id": 5, "E_i": 0.71, "N_i": 6, "S_i": 0.69, "label": 1}
        ]
    },
    ...
]
```

Listing 1: Portion of dataset used CH selection

The Listing 1 illustrates an extract of the dataset used in our training process. It presents multiple network states (3 in this sample, but can be extended up to 10,000 in the full dataset), each with the data for several nodes. For each node, features $E_i$, $N_i$, and $S_i$ are provided along with the label indicating whether the node should be a

cluster head or not. For the training process, the feature vectors $F_i = [E_i, N_i, S_i]$ for each node are fed into the GNN. The GNN performs its message-passing operations, updating the node features based on their neighboring nodes' features. After a set number of these updates, a final layer (often a linear or fully-connected layer) maps the updated features to an output value between 0 and 1 for each node. This output represents the predicted probability, $P_i$, that the node should be selected as a CH.

The training objective is to adjust the parameters of the GNN so that its predicted probabilities closely match the true labels in the training data. We achieve this using a binary cross-entropy loss function, which is the standard for binary classification problems. This loss function compares the GNN output probabilities $P_i$ against the true labels $y_i$ (1 if node $i$ is a CH, 0 otherwise) over all nodes. The loss function is given by $L = -\sum_i y_i \log(P_i) + (1 - y_i) \log(1 - P_i)$. The GNN has been trained to predict the likelihood of a node being an optimal CH based on three critical node features: current energy level, the number of neighboring nodes, and average received signal strength. The computation is done through multiple steps, where the GNN uses the mathematical model proposed by Kipf in [34]:

1. Node Embedding: $h_i = \text{ReLU}(W * f_i + b)$

   This equation is the initial step in which each node $i$ is represented by a vector $h_i$, called its embedding. The node embedding is created by taking the feature vector $f_i$ of the node (this could include various types of information about the node, depending on the specific application), multiplying it by a weight matrix $W$, adding a bias term $b$, and passing the result through a Rectified Linear Unit (ReLU) activation function. The purpose of the ReLU function is to introduce non-linearity into the model. This process effectively transforms the initial features into a new representation that can better capture the patterns in the data.
2. Message Passing: $m_i = \text{aggregate}h_j : j \in \text{Neighbors}(i)$

   In this step, the GNN aggregates information from the neighborhood of each node to capture the local graph structure. Specifically, for each node $i$, the model collects the embeddings $h_j$ of all its neighbors, and combines them using an aggregation function. This function could be a simple one like sum, average or max, or a more complex one like an LSTM. The result $msg_i$ can be seen as a message that node $i$ receives from its neighbors.
3. Embedding Update: $h_i = \text{combine}(h_i, m_i)$

   Once the messages $msg_i$ have been computed, the next step is to update the embeddings $h_i$ of the nodes. This is done by combining the old embedding of the node with the message it received from its neighbors. The combine function could be a simple concatenation followed by a linear layer, or a more complex operation like a gating mechanism.
4. Output Probabilities: $p_i = \text{sigmoid}(W_o * h_i + b_o)$

   Finally, the updated embeddings are used to compute the output of the GNN. In our case, the goal is to predict a probability for each node, so the embeddings are transformed using another weight matrix $W_o$ and bias $b_o$, and the result is passed through a sigmoid activation function. The sigmoid function ensures that the output is a number between 0 and 1, which makes it suitable for interpreting as a probability.

The weights and biases $(W, b, W_o, b_o)$ are parameters that the GNN learns during training. They are updated in a way that minimizes the difference between the model predictions and the true values (according to some loss function), given the training data. The ReLU and sigmoid functions are commonly used activation functions in neural networks that help to capture non-linear relationships in the data.

For instance, consider a MANET with 10 nodes. We input the features of these nodes into our trained GNN, which gives the following probabilities:

| $P(GNN)_1$ | $P(GNN)_2$ | $P(GNN)_3$ | $P(GNN)_4$ | $P(GNN)_5$ |
|---|---|---|---|---|
| 0.45 | 0.70 | 0.30 | 0.60 | 0.55 |
| $P(GNN)_6$ | $P(GNN)_7$ | $P(GNN)_8$ | $P(GNN)_9$ | $P(GNN)_{10}$ |
| 0.20 | 0.80 | 0.35 | 0.50 | 0.65 |

**Table 3**: GNN output probabilities

We set a probability threshold, $t$, above which nodes elect themselves as CHs. Here, we assume the threshold $t = 0.5$. Consequently, nodes with a GNN output above $t$, namely nodes 2, 4, 5, 7, and 10, elect themselves as CHs.

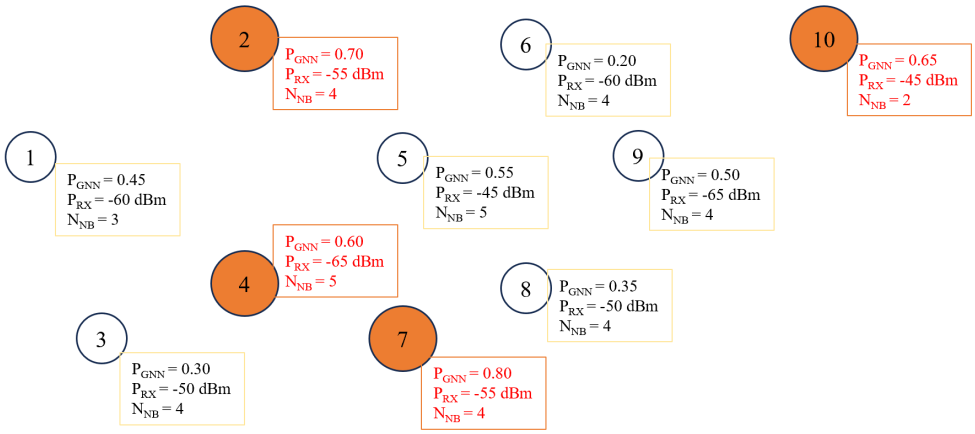


**Fig. 2**: Cluster Head election by using GNN

In Figure 2, we showcase an advanced representation of a MANET composed of ten nodes. Each node, depicted as a circle, is accompanied by a rectangle displaying three essential parameters the probability output by the Graph Neural Network (GNN) $P(GNN)$, the received power $P_{RX}$, and the number of neighbors $N_{nei}$. The $P(GNN)$ values are the outcomes of the trained GNN, which were calculated by considering the features of the nodes. These probabilities are used to determine the selection of Cluster Heads (CHs) in the network. If the probability of a node exceeds a predetermined threshold in this case, $t = 0.5$ the node elects itself as a CH. Consequently, nodes 2, 4, 7, and 10, which have $P(GNN)$ values above the threshold, are denoted as CHs

and filled with an orange color. All other nodes, with $P(GNN)$ values below the threshold, are depicted in white, indicating they are regular nodes. The $P_{RX}$ parameter indicates the received power of each node. This is an important feature as it can significantly influence the selection of CHs. Nodes with higher received power levels are often preferable for the CH role as they can maintain robust and reliable connections with other nodes. The third parameter, $N_{nei}$, denotes the number of neighbors for each node. This feature provides information about the node connectivity within the network. Nodes with a higher number of neighbors can often serve as more effective CHs because they can directly communicate with a larger number of nodes. Overall, this figure provides a clear visual depiction of how the GNN-based selection process operates within the MANET. By considering specific node features such as the GNN output probabilities, received power levels, and the number of neighbors, the model can effectively select CHs, helping to improve the overall performance and efficiency of the network.

## 5.2 Phase 2: Formation of Clusters

Once the selection of Cluster Heads process is completed it is possible to perform to the next step: the formation of clusters and the association of non-CH nodes to these clusters. This stage is crucial in establishing a well-structured network and ensuring efficient data transmission with minimum energy expenditure. In this context, it is important to clarify the criterion for node association to a particular cluster. Although a multitude of factors can potentially influence this decision, our methodology prioritizes the received signal strength from the CHs. This choice is grounded in the aim to minimize energy consumption during data exchange, both within and between clusters.

The signal strength received by a node from a CH generally has an inverse relationship with their distance apart. Hence, a node is likely to receive the strongest signal from a CH that is closest to it. Consequentially, a node will naturally gravitate towards the cluster led by a CH from which it receives the strongest signals, thereby reducing energy consumption associated with communication over longer distances. To denote the signal strength received from a CH by a node n, we use the notation RSS(CH, n).

Let us delve into our previously discussed ten-node network from Subsection 5.1, where nodes 2, 4, 7, and 10 were appointed as CHs, to illustrate this decision-making process more vividly:

- Node 1 and Node 3 register the strongest signal from Node 2. Consequently, both these nodes join Cluster 2.
- Nodes 5 and 6 find the strongest signal from Node 4, therefore they become part of Cluster 4.
- Node 8 receives the strongest signal from Node 7, leading it to align with Cluster 7.
- Node 9 registers the highest signal from Node 10, resulting in its association with Cluster 10.

This signal strength evaluation and the consequent decisions by the nodes lead to the formation of the following clusters:

- Cluster 2: Nodes 1, 3

- Cluster 4: Nodes 5, 6
- Cluster 7: Node 8
- Cluster 10: Node 9

Nodes 2, 4, 7, and 10, while functioning as CHs, also constitute their own individual clusters. This is due to the fact that no other CHs are located closer to them than they are to themselves.

After the formation of clusters, the next step is data transmission. Nodes that are not CHs transmit their data to their corresponding CHs. These CHs then collate this data and transmit it either to the base station or other CHs, depending on the specificities of the network communication protocol.

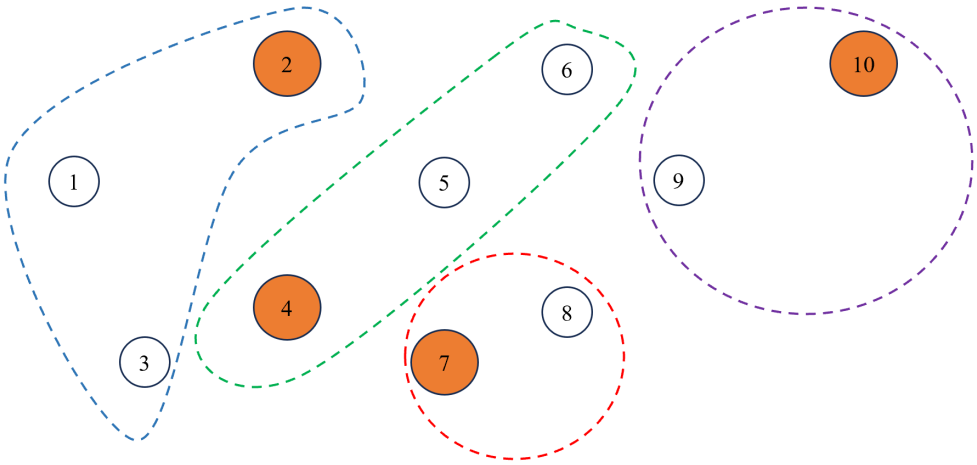


**Fig. 3**: Formation of clusters

Figure 3 offers an enhanced visualization of the cluster formation phase. The larger circles with orange filling represent the Cluster Heads (CHs), labeled by their respective numbers. The larger circles, drawn only with colored borders representing the formed clusters, are centered around the corresponding CHs. The smaller white circles represent the non-CH nodes, labeled by their numbers. This figure illustrates how nodes affiliate with the clusters from which they receive the strongest signal, with the absence of directed lines emphasizing the wireless nature of the MANET. Observe that formed clusters, denoted by dashed colored lines, are not necessarily circle shaped, but are tracked based on the average RSS of nodes for each cluster. Therefore, it is important to highlight that they are dynamically shaped based on updating resulting by GNN training, for example due to mobility node events. Finally, cluster formation process does not involve a fixed shape neither a fixed number of nodes and does not depend on the total number of nodes in the network but they only rely on GNN parameters.

### 5.3 Phase 3: Beamforming optimization

After cluster formation and CH election the Controller assigns the Massive MIMO System to each CH; at this point a beamforming optimization process is required in order to maximize network performance. More specifically, this phase aims to maximize the number of nodes that can be beamformed by the cluster heads in the direction of maximum gain. In particular, a mathematical model related to the selection of CH can be synthetized as follows:

Let $M$ be the number of antenna elements in the Massive MIMO system. The beamforming capability of a node depends on the number of antenna elements it possesses. Now, we can define the direction $d$ as the angle in radians at which a given node should beamform to cover at least $k$ nodes where $k$ is evaluated as follows:

$$k \geq \left\lceil \alpha \times \frac{C \cdot M}{2} \right\rceil \tag{1}$$

Where, the term $\alpha$ is a scaling factor that is chosen in order that the CH beamforms at least the 75% of nodes in a given cluster. For example, for $N = 20$, $C = 4$, $n_c = 5$, and assuming 100 radiating elements for Massive MIMO antenna we obtain $\alpha = 0.01875$ resulting in $k = 4$. If we consider larger networks for $N = 100$, $C = 10$, $n_c = 10$ and keeping the same levels of number of radiating elements of Massive MIMO we obtain $\alpha = 0.015$ and $k = 8$ that means that a potential CH is selected if it is able to cover at least 8 nodes over 10 with its main beam. In case none of the nodes in the cluster can cover at least 75% of nodes in a given cluster, the elected CH is the node that is surrounded by the most number of neighbors compared to all other. We assume that the optimal beamforming direction depends on the density of nodes around CH; in order to discover the direction corresponding to the maximum density of nodes we consider the total received power at CH as a measure of the estimation of the density of nodes. In particular, the higher is the received CH power at a given direction $d$ the higher is the number of nodes in that direction due that nodes that surround CH have the same omnidirectional pattern. In our approach an URPA structure is employed as Massive MIMO System, so the received power of an antenna array system is primarily governed by the beamforming vector which concentrates the signal in a particular direction (refer to Appendix A). In this case, the beamforming vector is usually chosen as the conjugate of the channel vector, a strategy known as Maximum Ratio Combining (MRC). Assuming a channel model of the form:

$$\mathbf{h}_{m,n}(\theta, \phi) = e^{-j2\pi(m-1)ndist\,\sin(\theta)\cos(\phi)} e^{-j2\pi(n-1)ndist\,\sin(\theta)\sin(\phi)}, \tag{2}$$

where $\mathbf{h}_{m,n}(\theta, \phi)$ represents the channel from the $(m, n)$th antenna element to the direction of $(\theta, \phi)$, $ndist$ is the normalized distance between antenna elements (relative to wavelength), and $(m, n)$ denotes the position of an antenna element in the array. The received signal can be modeled as:

$$\mathbf{y}m, n = \mathbf{h}m, n(\theta, \phi)\mathbf{x} + \mathbf{n}_{m,n}, \tag{3}$$

where $\mathbf{x}$ is the transmitted signal and $\mathbf{n}_{m,n}$ is the noise at the $(m, n)$th antenna element. In the case of MRC, the weights applied to each antenna element are proportional to the complex conjugate of the channel gains:

$$\mathbf{w}m, n = \mathbf{h}m, n^*(\theta, \phi). \tag{4}$$

This way, the signal received at each antenna is coherently combined to maximize the received signal power. The optimal direction $d$ corresponding to the maximum received power is then found by searching over all possible directions to maximize the array output:

$$d = \arg\max_{\theta,\phi}, \left| \sum_{m=1}^{N_y} \sum_{n=1}^{N_x} \mathbf{h}m, n^*(\theta, \phi)\mathbf{y}m, n \right|^2 . \tag{5}$$

This equation assumes knowledge of the channel gains for all directions. Instead, channel estimation techniques are employed to estimate the channel gains, and the optimal direction is determined based on these estimates.

## 5.4 Phase 4: Formation of Intra-Cluster Paths

In Phase 4, the DLCB approach try to detect robust paths for intra-cluster communication within a single cluster. Primarily, we assume that CH is necessarily included in the path for avoiding that the SNR decreases overly for increasing values of data rate. The path selection process operates by considering one of the following principles:

**For a given computed path:**

*if all nodes included in the path lie within the main beam:*

- **CH nearness and minimum number of hops**: for determining the optimal path the highest priority is assigned to paths that are composed by nodes that are in the surrounding of CH; indeed, the closer is a given node to a CH the higher is the the likelihood of being within the main coverage lobe of the CH. In this case, to maximize the network throughput the best selected path is the path having the minimum number of hops between source and destination closest to CH.

*otherwise:*

- **Secondary lobes contribute**: the choice of the path can be affected from both source and destination positions; in particular, in case at least a node in a path between a source and a destination is outside from main beam it is possible to consider the impact of secondary lobe gains by ensuring that the path also includes nodes within secondary lobes rather than considering the hop count between source and destination. In this case the quality of the link is ensured and the probability of miss-synchronization between communicating beams is minimized.

The Figure 4 outlines an example of path selection with a given cluster. The node 1 is the CH and it is filled in orange in a way that is clearly distinguished among other nodes. Node 1 beamforms in the direction $d$ according to phase 2 procedure. Let assume that node 1 wants to perform a communication toward node 9, firstly the intra-cluster path selection procedure evaluates all possible paths between source and destination. Observe that in this case CH is included in the path because it represents the source node. Let suppose to consider two of the whole possible paths computed by the procedure that are $PT_1 : \{node1, node2, node4, node9\}$ and $PT_2 :$

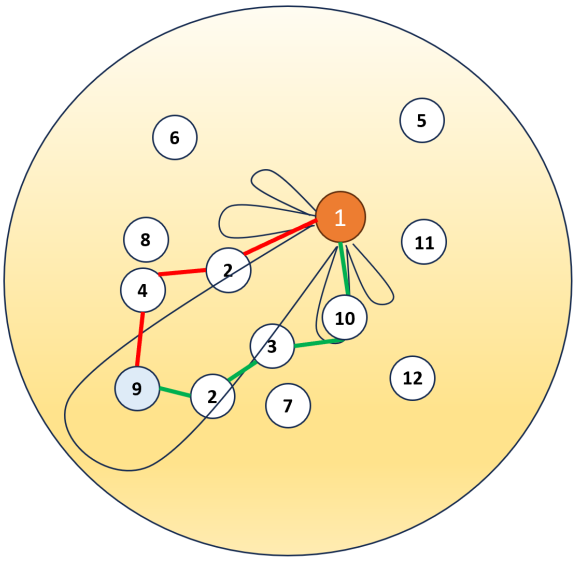


**Fig. 4**: Selection of intra-cluster path

$\{node1, node10, node3, node2, node9\}$. Although the total length of $PT_1$ in terms of number of hops is 3 that is lower with respect to the length of $PT_2$, the procedure selects the $PT_2$ path (green filled path) instead of $PT_1$ (red filled path) as the optimal path between source and destination because it notices that $PT_1$ includes node 4 that is outside the overall source beam; in the same context, while considering $PT_2$ all nodes of the path are included in the main beam except for node 10 which however lies within a secondary lobe; finally by considering this last aspect the procedure elects $PT_1$ as the best path from source to destination. Thereby, the reliability of the link is granted as well as packet network performance as long as node mobility is low. The Formation of Intra-Cluster Paths phase plays a crucial role in establishing optimized communication routes within each cluster. It enables the cluster head to select paths that maximize the benefits of beamforming and ensure efficient data transmission among the cluster members.

## 5.5 Phase 5: Inter-Cluster Communications

In the latest phase, the DLCB algorithm focuses on establishing efficient inter-cluster communication paths. Essentially, in this phase, path are selected by exploiting main benefits of beamforming process. Observe that this is a very critical phase because we assumed that CH use very high gain directional Massive MIMO Systems; in this context it is really complex to address issues implied by these kind of antennas such as the synchronization problem related to main beams. For selecting the best path between a source and destination. inter-cluster election process operates based on the following criteria:

- **Selection of pseudo-edge nodes:** in the worst case, a couple of CH that want to attempt a communication could be very far between themselves; therefore, in the unlikely event that their beam are targeting their own, the power of the beams is not enough to grant the communication because of the remoteness. Based on the BGP (Border Gateway Protocol) literature [35, 36], as a solution, for a given cluster the CH can exploit the power of its main beam for reaching one of the node that lie nearby edge of cluster (the pseudo-edge node) in order to maximize the probability to be beamformed by the closest CH.
- **Evaluation of the Received Power:** The decision about the next hop while forming the best path, is accomplished by considering the amount of the received power of the neighbor nodes. The higher is the received power for selecting the next hop in the path, the higher is the probability that a given node reaches a CH.

To illustrate the process of inter-cluster path selection, consider the following scenario:

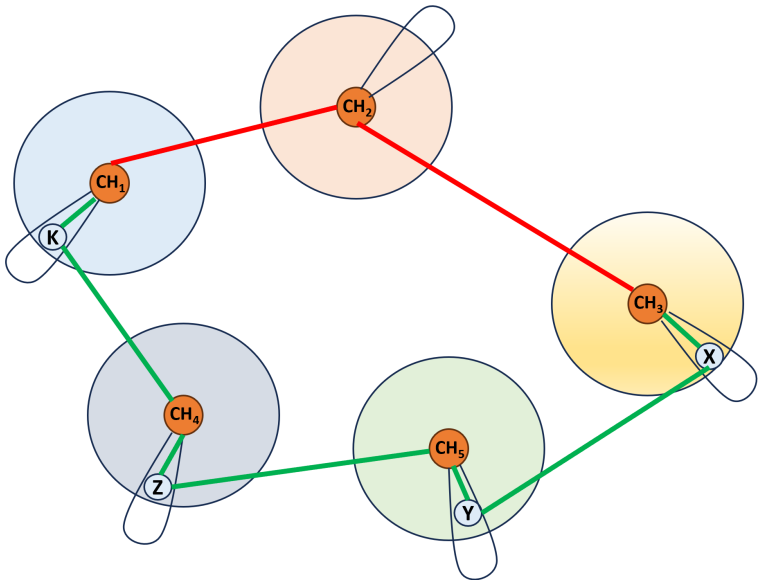


**Fig. 5**: Selection of inter-cluster path

The Figure 5 depicts an expository scenario in which there are 5 clusters each one having a CH elected based on previous phases of our approach; each CH is denoted by $CH_i$ where $i$ indicates the index of the $i_{th}$ cluster. Observe that the Figure 5 illustrates one of the worst possible communication case among clusters that is beam of all CH not pointing themselves. In particular, let us assume that $CH_3$ wants to attempt a communication toward $CH_1$; obviously, the path $PT_1 : \{CH_3, CH_2, CH_1\}$ is not absolutely possible because of different beam directions resulting in a significant beam miss-synchronization; although there exists the possibility to adapt beams

dynamically, this process is very expensive in terms of protocol overhead and energy consumption. As a workaround, $CH_3$, by exploiting its main beam (pointing toward the direction $d$) can sense and select one of the closest nodes in proximity of the cluster edge that is the *nodex*; the selection of the pseudo-edge node is performed based on the amount of the sensed received power $P(X)_{RX}$. Since cluster 5 is closer to cluster 3 more than cluster 2, it is very likely that $nodeX$ reaches $nodeY$ that is the pseudo-edge node belonging to cluster 5 due to the high sensed received power of $nodeY$ lying in the main beam of $CH_5$. At this point, it is very easy to reach the $CH_5$. By replicating the last two steps until the destination cluster has reached, the best evaluated optimal path results in $PT_2 : \{CH_3, nodeX, nodeY, CH_5, nodeZ, CH_4, nodeK, CH_1\}$ that is the green filled path in Figure 5.

## 5.6 Implementation of DLCB in OMNeT++

In this subsection, we provide an overview of the implementation details of the DLCB algorithm in the OMNeT++ simulation framework. OMNeT++ offers a wide range of classes and modules that facilitate the development of network simulations. DLCB is basically implemented by modifying such of Omnet++ modules belonging to PHY layer. For Medium Access Control (MAC) level we used part of features designed in [37].

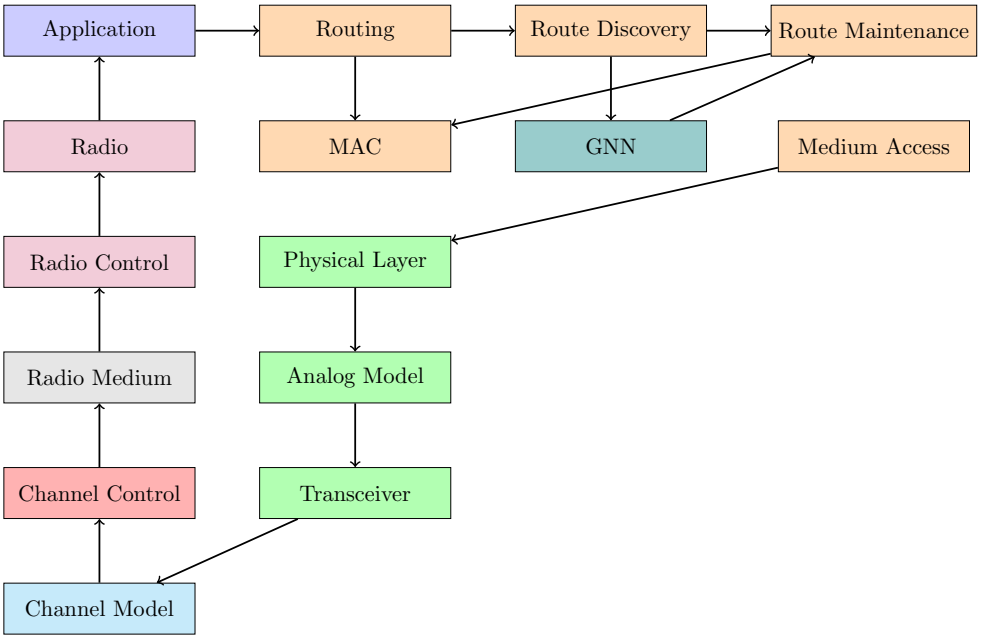


**Fig. 6**: Block Diagram of Omnet++ Modules Involved in DLCB implementation

The block diagram shown in Figure 6 illustrates the various modules involved in the implementation of DLCB. Each module represents a distinct component responsible for specific functionalities within the DLCB system. At the application layer, we have the *Application* module, which handles higher-level processes and interactions with the DLCB protocol. This module interfaces with the routing module to exchange information and request route discoveries. The DLCB sub-modules consist of the *Routing* module, responsible for managing the routing table and selecting optimal paths, the *Route Discovery* module, which initiates the process of finding routes, and the *Route Maintenance* module, which ensures the stability and reliability of established routes. The MAC (Medium Access Control) sub-module handles the medium access operations and communication between different nodes. It interacts with the routing module to exchange routing information and ensure efficient data transmission. The physical layer components include the *PhysicalLayer* module, responsible for managing the physical aspects of communication, and the *Analog Model* module, which simulates the analog characteristics of the wireless channel. The *Transceiver* module handles the transmission and reception of data packets.

The *Radio* module represents the radio interface for wireless communication. It interacts with the application layer to receive and transmit data packets. The *Radio Control* module manages the radio operational parameters and configuration. The *Radio Medium* module simulates the wireless medium and provides a communication channel between different nodes. It interacts with the radio control module to propagate signals and emulate wireless transmissions. The *Channel Control* module manages the channel access and control mechanisms, ensuring fair and efficient utilization of the wireless medium. It interacts with the radio medium and the channel model. The *Channel Model* module simulates the wireless channel characteristics, including signal propagation, interference, and noise. It interacts with the transceiver module to model realistic wireless communication scenarios.

The GNN was introduced into our implementation to enable more effective route selection and network optimization. In the context of our OMNeT++ simulation, the GNN is implemented as an additional module that interfaces with the existing DLCB modules. Specifically, the GNN module is tightly integrated with the Route Discovery and Route Maintenance modules to enhance their functionality. In order to implement the GNN within OMNeT++, we constructed a new module class, named *GNNModule*. The *GNNModule* uses the PyTorch library, which is linked with OMNeT++ via the Python/C++ API. We adopted PyTorch as it offers comprehensive support for graph neural networks, making it suitable for our purposes.

Here is how the GNN is incorporated into our OMNeT++ implementation:

- The GNN input is the adjacency matrix of the network topology, which is calculated at runtime by the *Route Discovery* module. This adjacency matrix is updated continuously as nodes move and the topology changes.
- The GNN processes the adjacency matrix and outputs a vector for each node in the network. This output vector represents the node feature vector, which includes information about the node optimal role in the network, such as whether it should act as a cluster head or a regular node.

- The feature vectors are then passed to the *Route Discovery* and *Route Maintenance* modules, which use the information to optimize the selection of routes and maintain the network stability.
- The *GNNModule* is updated iteratively throughout the simulation, using reinforcement learning techniques to adapt the GNN parameters based on the network performance. The learning rate, defined by the expression $\alpha = \frac{1}{\sqrt{t+1}}$, regulates this adaptation process. As the simulation progresses, the GNN learns to better optimize the network topology, leading to improved performance and efficiency.

In order to make a meaningful evaluation comparison with approach proposed in [33] we modified our basic *DLCBRouting* module by implementing main expressions of CBPR related to CH selection phases and route selection phase routing; obviously antenna configuration for each CBPR node is omnidirectional combined with our implemented 802.11ay module.

# 6 Simulation Results

To evaluate the performance of the proposed DLCB approach we conducted simulations using the OMNeT++ network simulator. The simulation environment consisted of a 1000 x 1000 meter square area, with a randomly generated set of 50 nodes uniformly distributed throughout the area. The DLCB algorithm was implemented as a custom module in OMNeT++, and was configured by using main parameters of GNN presented in Section 5. Basically, we compared our proposal with two widely used routing protocols: CBPR and Optimized Link State Routing (OLSR). We chose OLSR as comparison protocols due to their popularity and suitability for comparison with our approach. We varied the number of traffic flows in the simulation from 1 to 10, by evaluating performance of each routing protocol at different configuration.

## 6.1 Performance Metrics

We evaluated the performance of DLCB using the following metrics:

- Throughput: The total amount of data successfully delivered from the source to the destination per unit time.
- End-to-End Delay: The average delay experienced by packets from the source to the destination.
- Packet Delivery Ratio: The ratio of successfully delivered packets to the total number of sent packets.

## 6.2 Simulation Setup

We simulated a MANET environment using OMNeT 6 preview 10 version. Note that the part related 802.11ay extends the code presented in [38] and is referred to a private github repository branch that is in progress and also contains support for 802.11ax standard (in case of interest for contribution in the development of the code please refer to one author of the present proposal). The network consists of a certain number of uniformly distributed mobile nodes that vary from 20 to 100 (according the simulation

seed). Each simulation run had a duration of 1000 seconds, and the experiments were repeated 10 times for statistical significance. In order to maximize network throughput we equipped CH nodes with Massive MIMO URPA antennas 10x10 while other nodes use omnidirectional antenna for avoiding synchronization beamforming issues. The size of the network was set to $1000m \times 1000m$, and the maximum communication range of each node is $100m$ meters according to 802.11ay specifications. The link quality between nodes was modeled using the shadowing fading model. We generated traffic flows between randomly selected source-destination pairs in the network.

### 6.2.1 Simulation Parameters

Several parameters were considered in the simulations to assess their impact on the performance of DLCB:

**Table 4**: Simulation Parameters

| **Parameter** | **Value** |
|---|---|
| Simulation Time | 1000 seconds |
| Network Size | 1000 m x 1000 m |
| Number of Nodes | from 20 to 100 |
| Communication Range | 100 meters |
| Data Rate | up to 130 Gbps |
| Number of Massive MIMO URPA elements | 100 |
| Network standard | IEEE802.11ay |
| Routing Protocol | DLCB, OLSR annd CBPR |
| Mobility Model | Slow Mobility (maximum 2 mps) |
| Traffic Type | UDP |
| Packet Size | 1500 bytes |
| Nodes Sensitivity | -120 dBm |
| Simulation Repetitions | 10 |

The Table 4 synthesizes main simulation parameters; it is important to observe that we set the maximum data rate to 130 Gbps due that we noted in our experiments that by increasing this value, SNR (Signal to Noise Ratio) tends to dramatically reduce along with throughput; in fact, 802.11ay is normally designed to achieve average peak of 50-60 Gbps with a maximum range of about 300 meters; in order to increase this theoretical peak we use a 10x10 MIMO that reduces the coverage area to about 100 meters but provides a very large width of the main directional beam resulting in high throughput values.

The Figure 7 illustrates an explanatory example of simulation setup according to parameter set included in 4. In this case, the network scenario consists of 20 randomly distributed nodes divided into 4 clusters; CH nodes are $host5, host8, host16$ and $host4$ that are elected based on procedure explained in Subsection 5.1. CH beamform their neighbors by pointing the main beam toward $d_5 = d_4 = 120°$ and $d_8 = d_{16} = 150°$ where $d_i$ denotes the desired direction related to $i_{th}$ host; observe that desired direction of each CH is chosen in order that the beam maximizes the coverage area of neighbor nodes; dashed concentric circles around each CH delimits the maximum theoretical

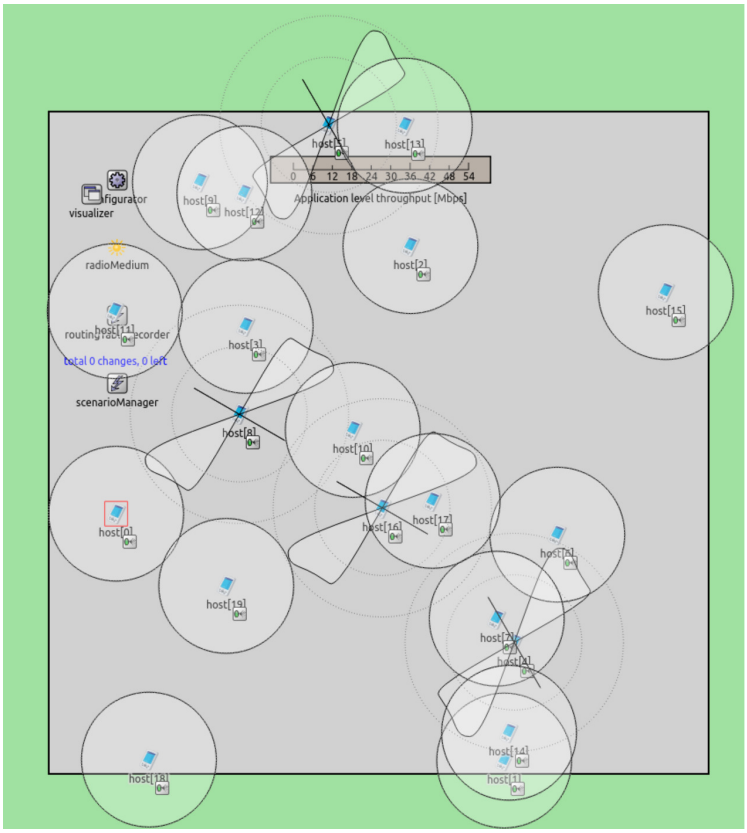


**Fig. 7**: Simulation environment example in Omnet++

coverage limit of the Massive MIMO depending on secondary lobes contribution. In this specific scenario only *host*15 and *host*18 are not potentially beamed because their position.

## 6.3 Results and Analysis

### 6.3.1 Throughput

The throughput is a crucial metric that reflects the data delivery capacity of the network. Throughput comparison compares our approach with CBPR and OLSR.

In Figure 8, the x-axis represents the data rate in gigabits per second (Gbps), and the y-axis represents the throughput in Gbps. DLCB is represented by the solid line with circular markers. CBPR is represented by the solid line with square markers. OLSR is represented by the solid line with triangular markers. We can see that DLCB, bolstered by the use of GNN and beamforming technologies, consistently outperforms both CBPR and OLSR in terms of throughput across all data rates. This superior performance is probably due to the fact that DLCB uses a smart routing approach enabled by GNN, allowing it to distribute traffic across multiple paths, hence taking

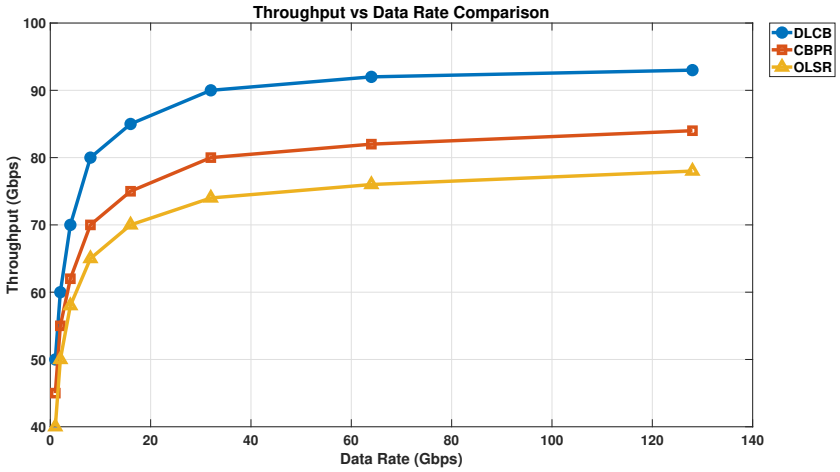


**Fig. 8**: Throughput comparison in function of data rate

full advantage of network redundancy. The GNN also aids in minimizing congestion and improving overall throughput by providing intelligent routing decisions based on its learned network representations. This results in more efficient resource utilization and optimized routing decisions. For instance, at a data rate of 64 Gbps, DLCB achieves a throughput of 92 Gbps, while CBPR and OLSR achieve throughputs of 82 Gbps and 76 Gbps, respectively. Additionally, the use of beamforming technology further enhances the performance of DLCB. With beamforming, DLCB can direct the signal towards intended devices, thereby increasing the quality of the received signal, reducing interference, and improving throughput. This feature, combined with the hierarchical structure of DLCB, which efficiently manages network resources by organizing nodes into clusters or tiers, results in reduced overhead associated with global routing information exchange. This ultimately leads to faster routing decisions and further improvements in throughput.

In the Figure 9, we have compared the throughput performance in relation to the number of nodes in the network. The x-axis represents the number of nodes, while the y-axis represents the throughput measured in Gbps (Gigabits per second). We observe that DLCB consistently achieves the highest throughput values across all the node counts, reaching up to 98 Gbps with 100 nodes. CBPR closely follows with throughput values up to 92 Gbps, and OLSR achieves throughput values up to 84 Gbps. As the number of nodes increases, DLCB demonstrates remarkable improvement in throughput, showcasing its superior performance over CBPR and OLSR. For instance, with 20 nodes, DLCB achieves a throughput of 55 Gbps, which progressively increases to 98 Gbps with 100 nodes. This significant enhancement can be attributed to DLCB efficient routing strategy, which effectively utilizes available network resources and optimizes data transmission paths. CBPR also exhibits competitive throughput performance, indicating its suitability for various network sizes. It achieves a throughput of 50 Gbps with 20 nodes, steadily increasing to 92 Gbps with 100 nodes. OLSR, on

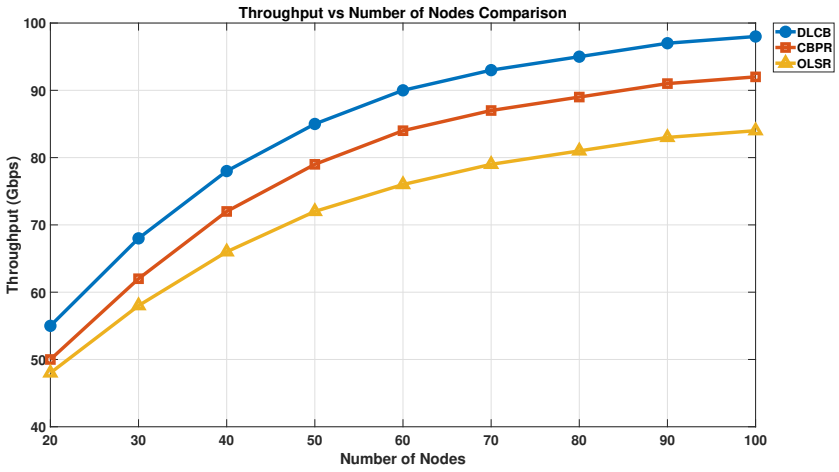


**Fig. 9**: Throughput vs. Number of nodes

the other hand, demonstrates relatively lower throughput values compared to DLCB and CBPR, reaching up to 84 Gbps with 100 nodes. These observations highlight the effectiveness of DLCB and CBPR in achieving high throughput in large-scale networks. Their utilization of efficient routing strategies contributes to enhanced data delivery and network efficiency. For a further investigation we evaluate the throughput in function of the learning rate. More specifically In the context of GNN, the learning rate is a crucial hyperparameter that determines the step size at each iteration while moving towards a minimum of the loss function. The learning rate decides by how much the weights and biases of the GNN should be adjusted in response to the estimated error during the backpropagation and optimization step. A smaller learning rate could make the model learn slower but it may provide more accurate results as it takes smaller steps towards the minimum of the loss function, potentially allowing the model to find a better minimum. Conversely, a larger learning rate makes the model learn faster, but at the risk of overshooting the minimum of the loss function. Tuning the learning rate is an important aspect of training a GNN, as it significantly impacts the accuracy, efficiency, and convergence of the training process. For the sake of clarity we can provide a possible definition of learning rate as follows [39]:

$$lr = \frac{lr_0}{1 + decay \cdot epoch} \tag{6}$$

where $lr$ is the learning rate for the current epoch, $lr_0$ is the initial learning rate, $decay$ is a pre-specified decay rate and $epoch$ is the current epoch number.

The curves depicted in Figure 10 compare the throughput performance of DLCB, CBPR, and OLSR, based on the learning rate applied during the network operation. The x-axis represents the learning rate, while the y-axis represents the throughput measured in Gbps (Gigabits per second). As we can observe, there is a gradual improvement in the throughput for all three protocols as the learning rate increases. DLCB

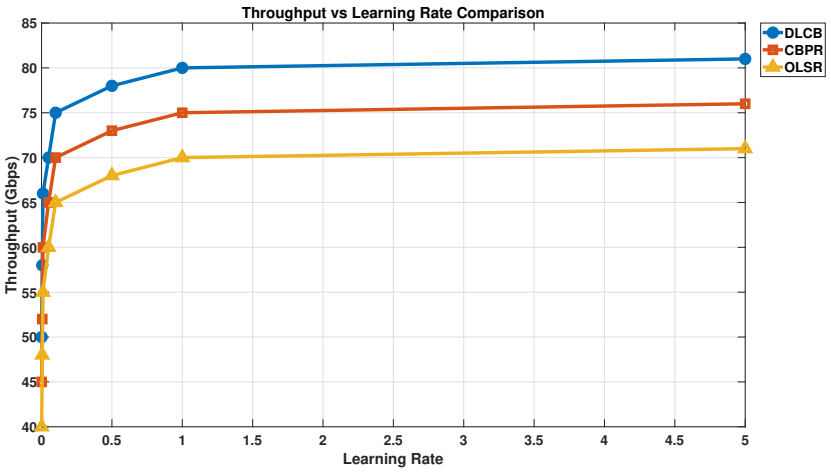


**Fig. 10**: Throughput vs. Learning Rate

achieves the highest throughput values across all learning rates, reaching up to 81 Gbps at the highest learning rate (5). The increase in throughput with higher learning rates can be attributed to the improved ability of the protocols to adjust and optimize their operation based on the learning process. As the protocols become better tuned and more adaptive, the routing decisions become more optimized, leading to enhanced data transmission and higher throughput. It is worth noting that while DLCB consistently outperforms CBPR and OLSR in terms of throughput, all three protocols showcase an upward trend in performance as the learning rate increases. This suggests that the use of learning-based strategies, regardless of the specific protocol, offers advantages in terms of improved throughput in dynamic network environments. These observations highlight the importance of machine learning-enhanced protocols, such as DLCB, CBPR, and OLSR, in achieving high throughput in networks where conditions may change over time. The utilization of learning rates enables efficient adjustment and optimization of protocol behavior, contributing to enhanced data delivery and overall network performance.

In Figure 11, we can observe that as the number of clusters increases, there is a notable improvement in the throughput for all three protocols. The application of GNN principles in DLCB allows it to consistently achieve the highest throughput values across all cluster sizes, reaching up to 93 Gbps with 50 clusters. CBPR closely follows with throughput values up to 89 Gbps, and OLSR achieves throughput values up to 83 Gbps. The increase in throughput with a higher number of clusters can be attributed to the inherent capabilities of GNNs to capture and exploit the structural information of the network. In the case of DLCB, the GNN allows for enhanced network organization, robust load balancing, and more effective data propagation through its ability to learn and adapt to the topology of the network and the interactions between nodes. As the network becomes more distributed into multiple clusters, DLCB can efficiently manage data transmission, leading to improved throughput. Moreover,

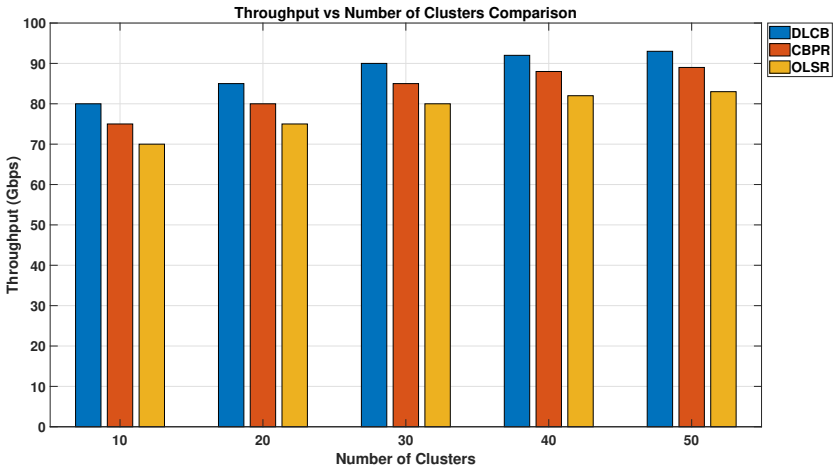


**Fig. 11**: Throughput vs. number of clusters

GNNs provide DLCB with a dynamic edge over CBPR and OLSR as they allow the protocol to learn and adjust to changes in the network environment over time, thereby maintaining optimal performance even under varying conditions. While DLCB consistently outperforms CBPR and OLSR in terms of throughput, all three protocols exhibit an upward trend in performance as the number of clusters increases. This suggests that clustering strategies, regardless of the specific protocol or the application of machine learning, offer advantages in terms of improved throughput by facilitating better resource utilization and traffic management. These observations highlight the significance of the number of clusters and the incorporation of GNN principles in influencing the throughput performance of routing protocols.

### 6.3.2 End-to-End Delay

In this section, we delve into a detailed comparison of the end-to-end delay performance of DLCB, CBPR, and OLSR under varying scenarios. These scenarios encompass the impact of data rate, number of nodes, learning rate, and the number of clusters on the delay performance of the three protocols. Figure 12 provides a comprehensive visualization of the results.

Referring to Subplot 12a, we observe that DLCB end-to-end delay tends to be lower compared to CBPR and OLSR as the data rate increases. One potential explanation for this superior performance could be attributed to the integration of GNNs in the DLCB protocol. GNNs excel at processing relational data, such as that found in networks, thereby enabling efficient packet management under high data rates. Additionally, the use of Massive MIMO could provide a significant boost here. This technology allows for the simultaneous transmission of multiple data streams, resulting in higher data rates and spectral efficiency, which consequently may result in a lower end-to-end delay.

As seen in Figure 12b, even when the network complexity increases with the number of nodes, DLCB maintains lower delay times than CBPR and OLSR. This performance

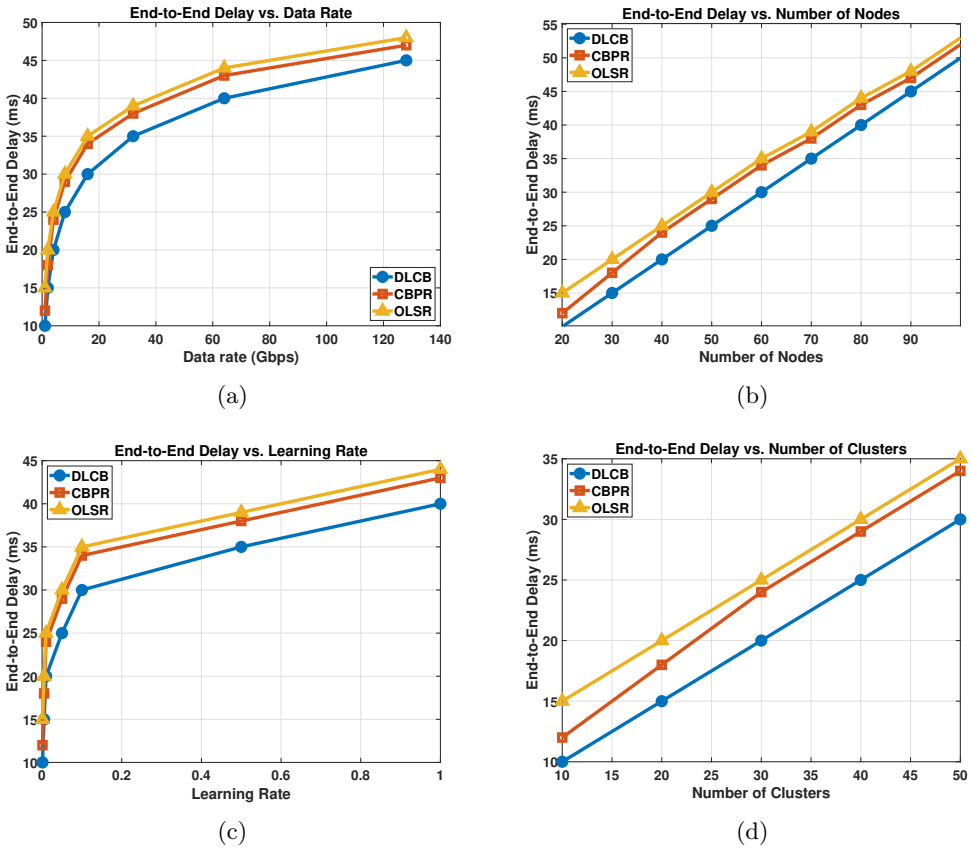


**Fig. 12**: End-to-End delay comparison

gain could potentially be due to the advantages of the dynamic clustering feature of DLCB. Dynamic clustering enables efficient resource management and load balancing across the network. The GNN within DLCB can capture and model the complex dependencies between the various nodes and clusters in the network, improving routing decisions in more complex networks.

In Subplot 12c, DLCB showcases a lower end-to-end delay at various learning rates compared to the other two protocols. This could be because of the adaptive nature of GNNs that are part of the DLCB protocol. As the learning rate changes, GNNs adjust their parameters to better model the network structure and update routing decisions. This adaptability to the learning rate changes is key in maintaining the performance of the network and, thus, results in lower delays.

The Figure 12d demonstrates the effect of increasing the number of clusters on the end-to-end delay. Even with an increase in the number of clusters, DLCB registers a lower delay compared to CBPR and OLSR. This is likely due to the efficient use of

beamforming and Massive MIMO technologies in DLCB. Beamforming focuses the signal towards intended nodes, reducing interference and improving signal quality, even in a complex network with many clusters. Massive MIMO, on the other hand, supports multiple simultaneous transmissions, making the network more robust against increasing clusters. Moreover, the GNN models embedded in DLCB can accurately capture the increased complexity brought about by the multiple clusters, providing optimized routing decisions.

### 6.3.3 Packet Delivery Ratio

In addition we compared the PDR performance of the three protocols in function of main considered parameters. This kind of analysis is very significant to prove and confirm throughput results.

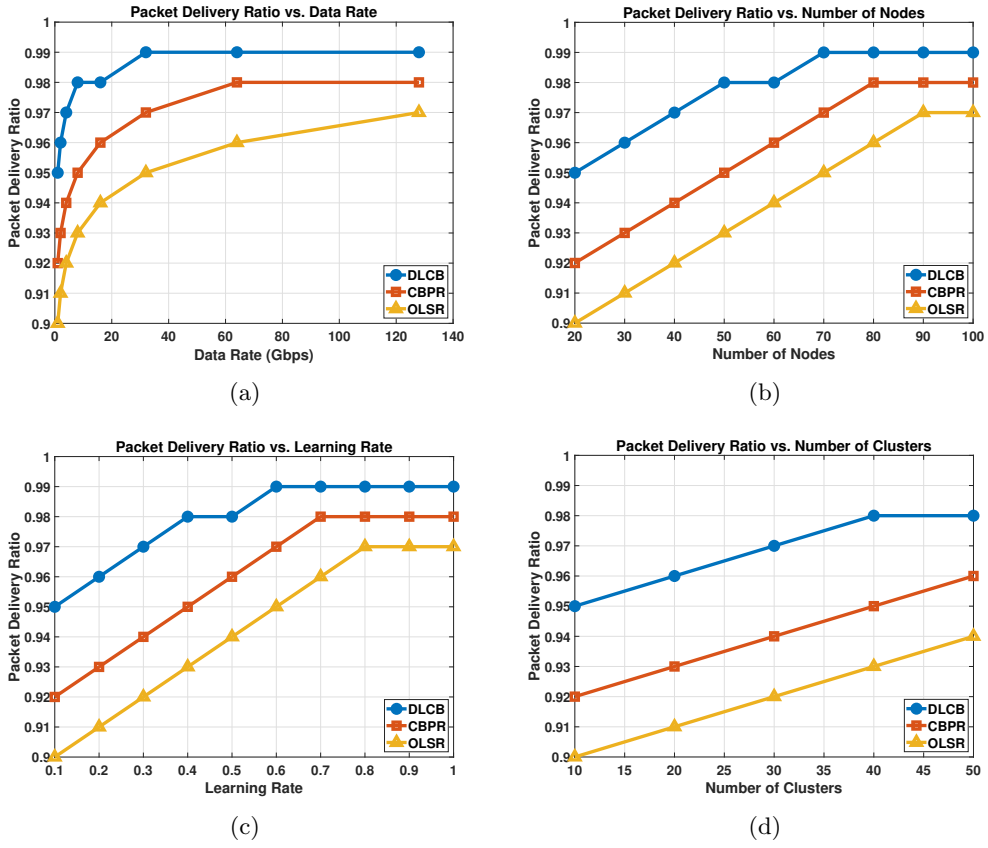


**Fig. 13**: PDR comparison

As depicted in Figure 13, our approach, showcases higher PDR performance across diverse parameters compared to the competing protocols. It is noteworthy that in the initial three subplots, the performance of DLCB and CBPR are strikingly similar due to several reasons. Firstly, during the early stages of a network setup, the topology is still being defined, and routing paths are being established. In this phase, both DLCB and CBPR are engaged in a comparable strategy of cluster formation and link-state dissemination which results in their similar performance.

Subplot 13a indicates that as data rate increases, DLCB manages to maintain a higher PDR in comparison to CBPR and OLSR. This can be traced back to the application of GNNs and Massive MIMO technologies in the DLCB protocol. GNNs can efficiently model the intricate network structure under escalated data rates, facilitating more effective packet management. Meanwhile, Massive MIMO enables multiple simultaneous transmissions, contributing to the observed improved packet delivery. This is evidenced by DLCB PDR of 0.99 at a high data rate of 128 Gbps. In Figure 13b illustrates that as the number of nodes increases, DLCB PDR performance remains steady, and in some cases, even shows improvement. The scalability of GNNs, which effectively captures the network complexity introduced by additional nodes, plays a key role in this performance. Thus, DLCB manages to uphold its packet delivery capabilities even in larger networks, as shown in a scenario with 100 nodes where it maintains a PDR of 0.99, exceeding both CBPR and OLSR.

In Subplot 13c, the adaptability of GNNs to varied learning rates seems to benefit DLCB performance. This flexibility allows DLCB to adjust its routing decisions efficiently under different learning conditions, thereby ensuring consistently high PDR performances. Even at the maximum learning rate of 1.0, DLCB maintains a better PDR, compared to both CBPR and OLSR. As demonstrated in Subplot 13d, with an increase in the number of clusters, DLCB continues to maintain a relatively high PDR. This can be attributed to the use of beamforming and Massive MIMO technologies in DLCB. Beamforming improves the signal-to-noise ratio by focusing the signal in a specific direction, thereby enhancing the reliability of packet transmission. Massive MIMO supports multiple simultaneous transmissions, improving the network robustness against an increasing number of clusters. Further, the use of GNN allows DLCB to efficiently handle the complexity introduced by multiple clusters, ensuring effective routing decisions and reliable packet delivery.

Overall, the consistent PDR performance of DLCB under a variety of conditions can be traced back to the implementation of features such as GNNs for effective and adaptive routing, Massive MIMO for enhanced network capacity, and beamforming for focused transmission. The combination of these technologies allows DLCB to uphold its performance under various circumstances, thereby indicating a beneficial direction for network protocol development.

# 7 Conclusions

In this paper, we presented a Clustering Beamforming Massive MIMO Routing Protocol bases on Deep Learning that it was designed for achieving 6G MANET requirements. DLCB incorporates basic deep learning concepts for clusters formation

and support to recent IEEE standards exploiting beamforming features to provide high level of throughput and low delay for addressing challenges in future wireless networks. To evaluate the performance and effectiveness of DLCB, we conducted a comprehensive comparison with CBPR approach, which served as our benchmark. Through the use of a Graph Neural Network underpinning the decision-making process, our approach creates an efficient and optimized network hierarchy, resulting in judicious resource usage and significant energy savings. This strategy, in turn, prolongs the network operational lifespan.

The evaluation focused on key performance metrics including throughput, end-to-end delay, and packet delivery ratio. Our simulation results clearly demonstrated that DLCB outperforms CBPR in terms of throughput, achieving higher data transfer rates across a range of data rates. This highlights DLCB superior ability to efficiently utilize network resources and deliver improved throughput in 6G MANETs. Additionally, DLCB exhibited lower end-to-end delay compared to CBPR, effectively minimizing packet transmission delays and ensuring timely delivery of data in networks of varying sizes and cluster configurations. This showcases DLCB capability to optimize routing decisions and reduce end-to-end delays in dynamic and evolving network topologies.

Furthermore, DLCB achieved higher packet delivery ratios compared to CBPR, demonstrating better reliability in packet delivery, even in scenarios involving higher data rates, larger networks, and more complex hierarchical structures. This emphasizes the robustness of DLCB in maintaining network connectivity and ensuring successful packet delivery in challenging MANET conditions. The comparison with CBPR provides valuable insights into the performance advantages offered by DLCB. By outperforming CBPR in terms of throughput, end-to-end delay, and packet delivery ratio, DLCB showcases its potential as a promising routing protocol for 6G MANETs.

By addressing these research challenges, we can unlock the full potential of DLCB and pave the way for more efficient and reliable communication in 6G MANETs. In conclusion, DLCB presents a compelling routing protocol solution for 6G MANETs, exploiting the benefits of GNN concepts combined with beamforming. With its superior throughput, lower end-to-end delay, and higher packet delivery ratio compared to CBPR, DLCB demonstrates its potential as a key enabler for efficient and scalable communication in future wireless networks.

## 7.1 Future Goals

While our study has provided valuable insights into the performance and advantages of combining Massive MIMO beamforming techniques together with a clustering strategy based on deep learning, there are several areas for future exploration and improvement. In this section, we outline some potential future goals for further research and development:

- **Enhancing Energy Efficiency**: One important future goal is to enhance the energy efficiency of DLCB. Energy consumption is a critical aspect of wireless networks, and optimizing the energy utilization of nodes in MANETs can significantly prolong network lifetime and improve overall performance. Future research can focus on developing energy-aware mechanisms and algorithms within DLCB to minimize

energy consumption during routing and data transmission. This can include techniques such as adaptive power management, node sleep/wake strategies, and efficient routing protocols that consider energy constraints.

- **Strengthening Security Mechanisms** As wireless networks continue to evolve, ensuring the security and privacy of data transmission becomes crucial. Future research can focus on integrating robust security mechanisms into DLCB, such as authentication, encryption, and intrusion detection systems. Strengthening the security aspects of DLCB will help mitigate potential threats and vulnerabilities in 6G MANETs. This can include the development of secure key management protocols, trust-based routing mechanisms, and secure data exchange protocols to protect against various security attacks and ensure the integrity and confidentiality of network communications.
- **Enabling Quality of Service Support** The support of Quality of Service (QoS) is essential for diverse applications and services in 6G MANETs. Future work can explore the integration of QoS mechanisms within DLCB to prioritize and allocate network resources based on different application requirements, such as real-time multimedia streaming, mission-critical communications, or delay-sensitive applications. This will enhance the overall user experience and enable a wide range of advanced services in MANETs. Research efforts can focus on QoS-aware routing algorithms, adaptive resource allocation techniques, and traffic engineering mechanisms to ensure the timely and efficient delivery of data with the desired QoS guarantees.

By addressing these future goals, we can further refine and advance DLCB to meet the evolving demands of 6G MANETs. The continued exploration and development of DLCB in these areas will contribute to the realization of efficient, reliable, and scalable wireless networks for future generations. It will also pave the way for innovative applications and services that take advantage of the capabilities of 6G MANETs, such as smart cities, autonomous vehicles, and immersive virtual reality experiences.

# Appendix A Massive MIMO URPA concepts

In this appendix main theoretical concept of Massive MIMO URPA are discussed. MIMO has risen to prominence as an integral part of modern wireless communication systems, significantly enhancing both spectral and energy efficiency by employing a large number of antennas at both transmitter and receiver ends. A key component in such systems is the antenna array, with the Uniform Rectangular Planar Array (URPA) being one of the key configurations.

As depicted in Figure A1 an URPA array consists of antenna elements that are positioned uniformly in a rectangular grid, which can be denoted as $Nx \times N_y$, where $N_x$ and $N_y$ are the number of antenna elements along each axis of the grid [40]. The antenna array configuration in a URPA array has a significant impact on the beam pattern. For a given antenna placement, the array factor, often denoted as $A(\theta, \phi)$, describes its radiation pattern. It can be mathematically expressed as:

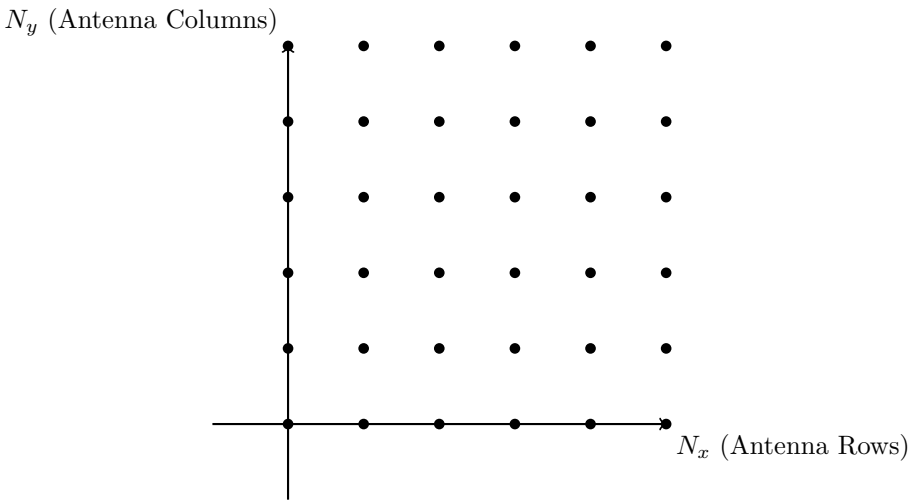


**Fig. A1**: $5 \times 5$ URPA example

$$A(\theta, \phi) = \frac{1}{NxNy} \left| \sum_{m=1}^{N_x} \sum_{n=1}^{N_y} e^{j[(m-1)\pi \sin\theta \cos\phi + (n-1)\pi \sin\theta \sin\phi]} \right|^2 \tag{A1}$$

Here, $\theta$ and $\phi$ denote the elevation and azimuth angles respectively. This formula gives a measure of the cumulative effect of individual antenna radiation patterns and thus characterizes the overall antenna array pattern.

Beamforming is the technique used to shape the radiation pattern of the antenna array to concentrate the signal in certain directions. It is implemented by controlling the phase and amplitude of the signal at each antenna in the array.

The weight vector for an URPA array is given by:

$$\mathbf{w}(\theta, \phi) = \frac{1}{\sqrt{NxNy}} \left[ e^{j[(m-1)\pi \sin\theta \cos\phi + (n-1)\pi \sin\theta \sin\phi]} \right]_{m=1,n=1}^{Nx,Ny} \tag{A2}$$

The signal at the array output is then the weighted sum of all the antennas:

$$y = \mathbf{w}^H \mathbf{x} \tag{A3}$$

where $\mathbf{w}^H$ is the conjugate transpose of the weight vector and $\mathbf{x}$ is the vector of signals received by each antenna [41]. The ability to form beams towards different directions enables a URPA array to serve multiple users in both azimuth and elevation domains, thereby significantly increasing the capacity of the communication system.

Implementing URPA arrays in Massive MIMO systems, while beneficial, also brings about significant challenges. One such challenge is the increased complexity of signal processing with a large number of antenna elements. Advanced digital signal processing techniques and machine learning algorithms can be utilized to effectively deal with this complexity.

# Declarations

- **Funding**: The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.
- **Competing interests**: The authors have no relevant financial or non-financial interests to disclose.
- **Availability of data and materials**: Enquiries about data availability should be directed to the authors.
- **Code availability**: The whole Omnet++ code belongs to a private github repository; for that reason, enquiries about code availability should be directed to the authors.
- **Authors' contributions**: All authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by Vincenzo Inzillo and David Garompolo. All authors read and approved the final manuscript.

# References


[1] Agiwal, M., Roy, A., Saxena, N.: Next generation 5g wireless networks: A comprehensive survey. IEEE communications surveys & tutorials **18**(3), 1617–1655 (2016)

[2] Boccardi, F., al.: Five disruptive technology directions for 5g. IEEE Communications Magazine **52**(2), 74–80 (2014)

[3] Atzori, L., al.: The internet of things: A survey. Computer networks **54**(15), 2787–2805 (2010)

[4] Hakak, S., Gadekallu, T.R., Maddikunta, P.K.R., Ramu, S.P., Parimala, M., De Alwis, C., Liyanage, M.: Autonomous vehicles in 5g and beyond: A survey. Vehicular Communications, 100551 (2022)

[5] Saad, W., al.: A vision of 6g wireless systems: Applications, trends, technologies, and open research problems. IEEE Network **34**(3), 134–142 (2019)

[6] Dang, S., al.: What should 6g be? Nature Electronics **3**(1), 20–29 (2020)

[7] Khorov, E., Kiryanov, A., Lyakhov, A., Bianchi, G.: A tutorial on ieee 802.11 ax high efficiency wlans. IEEE Communications Surveys & Tutorials **21**(1), 197–216 (2018)

[8] Ali, R., Kim, S.W., Kim, B.-S., Park, Y.: Design of mac layer resource allocation schemes for ieee 802.11 ax: Future directions. IETE Technical Review **35**(1), 28–52 (2018)

[9] Zhou, P., Cheng, K., Han, X., Fang, X., Fang, Y., He, R., Long, Y., Liu, Y.: Ieee 802.11 ay-based mmwave wlans: Design challenges and solutions. IEEE

Communications Surveys & Tutorials **20**(3), 1654–1681 (2018)

[10] Ghasempour, Y., Da Silva, C.R., Cordeiro, C., Knightly, E.W.: Ieee 802.11 ay: Next-generation 60 ghz communication for 100 gb/s wi-fi. IEEE Communications Magazine **55**(12), 186–192 (2017)

[11] Aldubaikhy, K., Wu, W., Zhang, N., Cheng, N., Shen, X.: mmwave ieee 802.11 ay for 5g fixed wireless access. IEEE Wireless Communications **27**(2), 88–95 (2020)

[12] Giordani, M., Polese, M., Mezzavilla, M., Rangan, S., Zorzi, M.: Toward 6g networks: Use cases and technologies. IEEE Communications Magazine **58**(3), 55–61 (2020)

[13] Choi, J., Va, V., Gonzalez-Prelcic, N., Daniels, R., Bhat, C.R., Heath, R.W.: Millimeter-wave vehicular communication to support massive automotive sensing. IEEE Communications Magazine **54**(12), 160–167 (2016)

[14] Pathak, S., Jain, S., Borah, S.: Clustering algorithms for manets: a review on design and development. In: Soft Computing Techniques and Applications: Proceeding of the International Conference on Computing and Communication (IC3 2020), pp. 563–578 (2021). Springer

[15] Nam, D.: Comparison studies of hierarchical cluster-based routing protocols in wireless sensor networks. In: CATA, vol. 69, pp. 334–344 (2020)

[16] Silva, M.M.D., Mello, R.F.D., Rios, R.A.: Time series clustering using stochastic and deterministic influences. International Journal of Computational Science and Engineering **21**(3), 394–417 (2020)

[17] Osamy, W., Aziz, A., Khedr, A.M.: Deterministic clustering based compressive sensing scheme for fog-supported heterogeneous wireless sensor networks. PeerJ Computer Science **7**, 463 (2021)

[18] Suárez-Varela, J., Almasan, P., Ferriol-Galmés, M., Rusek, K., Geyer, F., Cheng, X., Shi, X., Xiao, S., Scarselli, F., Cabellos-Aparicio, A., et al.: Graph neural networks for communication networks: Context, use cases and opportunities. IEEE network (2022)

[19] Cao, D., Li, J., Ma, H., Tomizuka, M.: Spectral temporal graph neural network for trajectory prediction. In: 2021 IEEE International Conference on Robotics and Automation (ICRA), pp. 1839–1845 (2021). IEEE

[20] Larsson, E.G., Edfors, O., Tufvesson, F., Marzetta, T.L.: Massive mimo for next generation wireless systems. IEEE communications magazine **52**(2), 186–195 (2014)

[21] Björnson, E., Larsson, E.G., Marzetta, T.L.: Massive mimo: Ten myths and one

critical question. IEEE Communications Magazine **54**(2), 114–123 (2016)

[22] Zhang, L., Roy, S.: Optimal beam training for mmwave massive mimo using 802.11 ay. arXiv preprint arXiv:2211.15990 (2022)

[23] Kim, M.-S.: Deep learning based multiple-beam transmission for ieee 802.11 ay beamforming training. In: 2022 27th Asia Pacific Conference on Communications (APCC), pp. 494–495 (2022). IEEE

[24] Chen, L., Zhang, L.: Spectral efficiency analysis for massive mimo system under qos constraint: an effective capacity perspective. Mobile Networks and Applications **26**, 691–699 (2021)

[25] Varga, A.: A practical introduction to the omnet++ simulation framework. Recent Advances in Network Simulation: The OMNeT++ Environment and its Ecosystem, 3–51 (2019)

[26] Shafie, A., Yang, N., Han, C., Jornet, J.M., Juntti, M., Kurner, T.: Terahertz communications for 6g and beyond wireless networks: Challenges, key advancements, and opportunities. IEEE Network (2022)

[27] Tataria, H., Shafi, M., Dohler, M., Sun, S.: Six critical challenges for 6g wireless systems: A summary and some solutions. IEEE Vehicular Technology Magazine **17**(1), 16–26 (2022)

[28] Giordano, A., Iera, A., Morabito, G.: Hierarchical routing protocol for mobile ad hoc networks. IEEE Communications Letters **19**(8), 1364–1367 (2015)

[29] Zhou, H., Renshaw, E.: Grid-based clustering approach for large-scale mobile ad hoc networks. In: 2016 International Conference on Wireless Communications and Signal Processing (WCSP), pp. 1–6 (2016). IEEE

[30] Wang, L., Li, J., Lu, X., Wang, Y.: Energy-efficient multi-level hierarchical routing protocol for mobile ad hoc networks. Sensors **18**(11), 3702 (2018)

[31] Singh, P., Choudhary, A., Sharma, A.: Hybrid clustering algorithm for hierarchical routing in mobile ad hoc networks. In: 2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU), pp. 1–5 (2019). IEEE

[32] Venkatasubramanian, S., Suhasini, A., Hariprasath, S.: Maximization of network lifetime using energy efficient super clustering protocol based on ldha-tsro in manet. Journal of Data Acquisition and Processing **38**(3), 523 (2023)

[33] Maheswar, R., Jayarajan, P., Sampathkumar, A., Kanagachidambaresan, G., Hindia, M.N., Tilwari, V., Dimyati, K., Ojukwu, H., Amiri, I.S.: Cbpr: A cluster-based backpressure routing for the internet of things. Wireless Personal

Communications **118**, 3167–3185 (2021)

[34] Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)

[35] Zhao, X., Band, S.S., Elnaffar, S., Sookhak, M., Mosavi, A., Salwana, E.: The implementation of border gateway protocol using software-defined networks: A systematic literature review. IEEE Access **9**, 112596–112606 (2021)

[36] Alotaibi, H.S., Gregory, M.A., Li, S., et al.: Multidomain sdn-based gateways and border gateway protocol. Journal of Computer Networks and Communications **2022** (2022)

[37] De Rango, F., Inzillo, V., Quintana, A.A.: Exploiting frame aggregation and weighted round robin with beamforming smart antennas for directional mac in manet environments. Ad Hoc Networks **89**, 186–203 (2019)

[38] Inzillo, V., De Rango, F., Ariza-Quintana, A.: Supporting 5g wireless networks through ieee802. 11ac standard with new massive mimo antenna system module design in omnet++ simulator. In: SIMULTECH, pp. 62–72 (2018)

[39] Shi, M., Tang, Y., Zhu, X., Huang, Y., Wilson, D., Zhuang, Y., Liu, J.: Genetic-gnn: Evolutionary architecture search for graph neural networks. Knowledge-Based Systems **247**, 108752 (2022)

[40] Ding, Q., Deng, Y., Gao, X., Liu, M.: Hybrid precoding for mmwave massive mimo systems with different antenna arrays. China Communications **16**(10), 45–55 (2019)

[41] Ning, B., Tian, Z., Chen, Z., Han, C., Yuan, J., Li, S.: Prospective beamforming technologies for ultra-massive mimo in terahertz communications: A tutorial. arXiv preprint arXiv:2107.03032 (2021)