Graph Neural Network Assisted Beam-Aware UAV Topology Optimization

Ali Yilmaz Yildirim
Airbus Defence and Space
Chair of Network Architectures and Services
Technical University of Munich
Munich, Germany
ali-yilmaz.yildirim@airbus.com

Fabien Geyer
Airbus Defence and Space
Munich, Germany
fabien.geyer@airbus.com

Georg Carle
Chair of Network Architectures and Services
Technical University of Munich
Munich, Germany
carle@net.in.tum.de

***Abstract*—Directional antennas offer substantial advantages in wireless communications by enhancing link quality and boosting achievable end-to-end (E2E) data rates. Extending their use to Unmanned Aerial Vehicles (UAVs) can significantly elevate airborne network capacity. However, deploying directional tracking in dynamic aerial environments introduces a critical design challenge in interference-limited regimes: determining the optimal 3D antenna orientation to maximize network performance. Motivated by the severe impact of bottleneck links on overall throughput, this study focuses on maximizing the bottleneck Signal-to-Interference-plus-Noise Ratio (SINR) while satisfying constraints for path redundancy and strong connectivity. Given its combinatorial nature, this problem is formulated as a Mixed-Integer Nonlinear Programming (MINLP). While an optimal solution can be found using the branch-and-bound mechanism, this approach incurs prohibitive computational overhead, rendering it infeasible for real-time applications. To address this limitation, we developed a Graph Neural Network (GNN) to predict optimal edges and proposed two GNN-augmented topology-creation frameworks that produce near-optimal topologies with drastically reduced execution time. We demonstrate the GNN's capability to predict optimal edges using standard classification metrics. We then compare the GNN-augmented topology-creation frameworks against a Mixed-Integer Linear Programming (MILP) bisection search derived from the original MINLP formulation, as well as standalone heuristic and metaheuristic methods.**

***Index Terms*—UAVs, Directional Antennas, Mixed-Integer Linear Programming, Mixed-Integer Nonlinear Programming, Graph Neural Networks, Topology Optimization**

I. Introduction

The use of Unmanned Aerial Vehicles (UAVs) has expanded across civilian and military domains, where they are increasingly deployed for monitoring, rescue operations, or in war zones. To maximize spatial reuse, these platforms are integrated with high-gain directional antennas operating in the Millimeter wave (mmWave) band. By employing narrow-beam transmissions, these antennas facilitate multiple simultaneous communication links while effectively suppressing inter-link interference.

While UAV network studies [1] have expanded, challenges persist regarding specific mission-driven deployments. Current research is predominantly focused on sub-6 GHz schemes that utilize UAVs as aerial base stations. However, multi-hop airborne networks for specialized applications such as data collection or rescue operations have received less attention, with existing work often limited to beam tracking for high-speed nodes [2].

In this work, we propose a framework to optimize the topology of an mmWave-enabled UAV network, aiming to maximize the bottleneck Signal-to-Interference-plus-Noise Ratio (SINR) while satisfying connectivity and path redundancy constraints. Since the lowest-capacity link caps the achievable end-to-end (E2E) throughput of any flow [3], maximizing the minimum SINR maximizes worst-case E2E capacity. In contrast to research centered on trajectory optimization, our framework assumes that UAV movement is strictly dictated by mission profiles—such as firefighting or environmental monitoring. This shifts the optimization challenge toward the communication layer, where we determine the optimal topology by aligning directional antennas to establish links within an interference-limited environment. The resulting problem is an NP-hard Mixed-Integer Nonlinear Programming (MINLP) (Section IV): branch-and-bound runtime grows exponentially with network size, which rules it out when antenna directions must be re-optimized at the seconds level, while the greedy heuristics used in its place frequently converge to suboptimal solutions.

Machine Learning (ML) is now widely applied to combinatorial optimization in wireless systems, yielding usable solutions within practical timeframes. We follow this direction with a Graph Neural Network (GNN)-based algorithm for link establishment in interference-limited UAV networks, validated against MINLP and heuristic baselines in terms of E2E capacity and computational overhead.

The primary contributions of this study are as follows:

- We formulate the topology optimization as a combinatorial problem and derive benchmark solutions using Mixed-Integer Linear Programming (MILP), which is the transformed version of MINLP.
- We develop a GNN architecture designed to approximate optimal solutions at the sub-second scale. For training, we generate a comprehensive dataset using MILP solutions, and we validate the model's predictive performance using standard classification metrics.

979-8-3195-0167-7/26/$31.00 ©2026 IEEE

- Recognizing that model accuracy may not reach 100%, we propose frameworks that utilize the model's probabilistic outputs as inputs for topology creation. We evaluate two distinct methodologies for leveraging these insights to construct topologies.

The remainder of this paper is structured as follows: Section II surveys existing literature on topology creation across various applications. Section III describes the network architecture and the communication model, while Section IV provides the formal mathematical problem definition. The proposed GNN for edge estimation and the corresponding topology construction frameworks are detailed in Section V. Section VI outlines the simulation environment, underlying assumptions, and the data generation process. Section VII offers an in-depth analysis of the performance results and key findings. Finally, Section VIII summarizes the study and suggests potential avenues for future research.

II. Related Work

Topology Control (TC) was first studied for omnidirectional networks, where topologies are shaped by tuning per-node transmission power [4]. As directional antennas matured and their interference-suppression benefits became apparent, attention shifted to directional-beam deployments, and the growing interest in Airborne Networks (ANs) extended this to aerial vehicles in scenarios such as airborne-sea ad hoc networks [5] and aeronautical networks over the Northeast Atlantic Corridor [6], [7]. The latter two solve the same problem with different tools, a MILP solver [6] and a heuristic that trades solution quality for scalability [7], but both differ from our objective: they extend links to a ground station via aerial-to-aerial hops and maximize the number of aircraft meeting a data-rate threshold, leaving global connectivity unconstrained. TC in ANs has also been studied for other media, e.g. Free-Space Optical Communications (FSOC) links connecting commercial aircraft to ground gateways, with hop count minimized and node degree maximized for robustness [8].

ML-driven topology optimization for nodes equipped with directional antennas has been investigated in [9] and [10]. These works aim to derive network topologies while satisfying constraints such as maximum node degree and global connectivity. Methodologically, both studies utilize the attention mechanism, with the latter integrating it into a GNN architecture to learn optimal edge configurations. However, these studies neglect realistic RF phenomena such as interference and path loss, limit their objectives to basic connectivity, and lack comprehensive evaluations of classification metrics and resulting physical network performance.

While existing literature has addressed topology optimization for UAVs equipped with directional antennas, the application of ML—and specifically GNNs—remains largely underexplored. To address these deficiencies, we propose a framework for multi-hop UAV ad hoc networks using directional mmWave antennas. Our approach leverages a trained GNN to estimate optimal edges, integrated into a solution framework that constructs near-optimal topologies within computationally feasible timeframes.

III. System Model

In our study, each UAV is equipped with four steerable directional mmWave antennas—one mounted on each side—capable of 45° steering. Directional antennas are primarily used for data transmission and multi-hop relaying between UAVs. The number of bidirectional links from each UAV is represented as the nodal degree d_i, constrained via maximum and minimum degree as $d_{min} \leq d_i \leq d_{max}$.

A. Network Model

We model the UAV network as a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ denotes the set of UAVs and $\mathcal{E}$ represents the set of active communication links. For any pair of nodes $(i, j) \in \mathcal{V} \times \mathcal{V}$ where $i \neq j$, the relative position vector from i to j is characterized by the triplet $(\theta_{i,j}, \phi_{i,j}, d_{i,j})$, representing the azimuth angle, elevation angle, and Euclidean distance with respect to the UAV body frames. We define the antenna boresight direction of a transmitter (e_{tx}) toward a target receiver (e_{rx}) using the angular coordinates $[\theta_e, \phi_e]$. Based on this boresight alignment, we calculate the horizontal and vertical deviation angles, $\Delta\theta_{e,j}$ and $\Delta\phi_{e,j}$, for any other node $j \in \mathcal{V} \setminus \{e_{tx}\}$.

B. Communication Model

As a result of the absence of physical obstructions, the inter-UAV links are modeled as line of sight (LOS) conditions. According to that, the communication channel is modeled as Rician fading with LOS dominance [11]. For any link $i \to j$, the path loss ($PL_{i,j}$) and the instantaneous received power ($P_{rx_{i,j}}$) are determined with the following equations:

$$PL_{i,j} = \left(\frac{4\pi \cdot d_{i,j} \cdot f_c}{c}\right)^2 \tag{1}$$

$$P_{rx_{i,j}} = \frac{P_{tx_i} \cdot g_{tx,i} \cdot g_{rx,j} \cdot |h_{i,j}|^2}{PL_{i,j}} \tag{2}$$

$|h_{i,j}|^2$ represents the power gain of the Rician fading channel. The fading characteristic is defined by the K-factor, which represents the ratio between the power of the LOS component and the scattered multi-path components. In this study, we assume a highly dominant LOS path with $K = 15$ dB [12].

$d_{i,j}$, f_c and c are the distance between the two UAVs, operating frequency, and the speed of light respectively. P_{tx_i} denotes the transmission power, while g_{tx} and g_{rx} represent the linear transmit and receive antenna gains, respectively. We determine the directional antenna gain in dB (G_{tx} and G_{rx}), according to the 3GPP radiation pattern for directional antennas operating above 6 GHz, by considering both horizontal (A_H) and vertical (A_V) components [13] as below:

$$A_H(\Delta\theta_{tx,rx}) = \min\left\{12\left(\frac{\Delta\theta_{tx,rx}}{\theta_{3dB}}\right)^2, A_m\right\}, \tag{3}$$

$$A_V(\phi_{tx,rx}) = \min\left\{12\left(\frac{\Delta\phi_{tx,rx}}{\phi_{3dB}}\right)^2, A_m\right\}, \quad (4)$$

$$G_{tx} = G_{\max} - (A_H(\theta_{tx,rx}) + A_V(\phi_{tx,rx})), \quad (5)$$

where A_m denotes the sidelobe attenuation, θ_{3dB} and ϕ_{3dB} represent the horizontal and vertical 3dB beamwidths, respectively, and $G_{\max}$ is the peak antenna gain in dB. Similarly, G_{rx} is also calculated from the receiver's perspective using Equation (5). We assume a UAV's own transmission toward a receiver does not interfere with other concurrent receptions at that same receiver.

Using the criteria defined in Equations (3) to (5), the SINR for a link $i \to j$ is now dependent on the spatial filtering of both the transmitter and the receiver. It is expressed as:

$$SINR_{i,j} = \frac{P_{rx_{i,j}}}{N + \sum_{e_{k,g}\in\mathcal{E}\ k\notin\{i,j\}\ g\neq j} I_{k,g,i,j}} \quad (6)$$

We define $I_{k,g,i,j}$ as the interference power at the reception of the transmission from node i to node j, caused by the simultaneous transmission from node k to node g. The total interference power is then calculated as:

$$I_{k,g,i,j} = \frac{P_{tx_k} \cdot g_{tx,k}(k,g,j) \cdot g_{rx,j}(i,j,k)}{PL_{k,j}} \quad (7)$$

$g_{tx,k}(\cdot)$ denotes the antenna gain of the interfering transmitter k in the direction of the victim receiver j, relative to the transmitter boresight oriented toward node g. Similarly, $g_{rx,j}(\cdot)$ denotes the antenna gain of the victim receiver j in the direction of the interfering transmitter k, relative to the receiver boresight oriented toward its intended transmitter i. Both $g_{tx,k}(\cdot)$ and $g_{rx,j}(\cdot)$ are computed according to Equations (3) to (5).

IV. Problem Formulation

In multi-hop wireless routing, the maximum achievable end-to-end (E2E) throughput of a path P is fundamentally governed by its bottleneck link [3]. Consequently, our primary objective is to maximize the SINR at this critical bottleneck.

Subject to this primary objective, the optimization problem is further formulated under the following constraints:

- **Strongly Connected Network:** An E2E path must exist between any pair of nodes.
- **Node Degree:** Each UAV is subject to a degree constraint ($d_{min} \le d_i \le d_{max}$), to ensure both path redundancy and hardware feasibility.
- **Bi-directionality:** All established communication links must be bi-directional.

We formulate a mathematical model to determine the optimal assignment of bidirectional links using directional antennas. In our formulation, the model components—sets, indices, decision variables, and parameters—are defined as follows:

Sets and Indices

- $V = \{1, \ldots, n\}$ represents the UAV set
- $D = \{(i,j) \in V \times V : i \neq j\}$ represents the transmission node pairs
- $\mathcal{Y} = \{(i,j) \in V \times V : i \neq j\}$ represents the possible directional links.
- $\mathcal{T}_{\text{quad}} = \{(k,g,i,j) \in V^4 : (k,g) \in D,\ (i,j) \in D,\ k \notin \{i,j\},\ g \neq j\}$ represents the interference quadruples, denoting the interference from link $k \to g$ onto the receiver j, which is oriented towards i.
- $\mathcal{T}_{\text{tri}} = \{(k,g,j) \in V^3 : \exists\, i \in V \text{ s.t. } (k,g,i,j) \in \mathcal{T}_{\text{quad}}\}$ represents the unique interference triples used to define the topological state of the interfering transmitter.
- $r \in V$ represents the root node, which will be used to realize the strongly connected constraint via network flows. It is only an arbitrary node used for the strongly connected constraint.

Decision Variables

- $y_{ij} \in \{0,1\}, \forall (i,j) \in Y$: Decision variable for the link $i \to j$ to be active or inactive.
- $f_{ij} \ge 0, \forall (i,j) \in D$: Flow variable used for the strongly connected graph constraint.
- $w_{k,g,j} \in \{0,1\}$: A binary indicator variable representing whether the transmission link $k \to g$ can interfere with receiver j. This variable is introduced to enforce the constraint that an actively transmitting UAV cannot simultaneously generate interference toward another concurrent reception associated with the same receiver.
- $\tau \ge 0$: SINR threshold in linear scale.

Parameters

- S_{ij}: Received signal power of link $i \to j$.
- $I_{k,g,i,j}$: Interference power received at link $i \to j$ from transmission $k \to g$.
- N: Noise power.
- d_{min}, d_{max}: Minimum and maximum node degrees.
- M: A sufficiently large constant (Big-M) used to relax the SINR constraint for inactive links.

$$\underset{y,f,w,\tau}{\text{maximize}} \quad \tau$$

subject to

$$\textbf{(C-1)} \quad y_{ij} = y_{ji}, \quad \forall (i,j) \in Y$$

$$\textbf{(C-2)} \quad d_{min} \leq \sum_{j \in V \setminus \{i\}} y_{ij} \leq d_{max}, \quad \forall i \in V$$

$$\textbf{(C-3)} \quad f_{ij} \leq (n-1) y_{ij}, \quad \forall (i,j) \in Y$$

$$\textbf{(C-4)} \quad \sum_{j \in V \setminus \{r\}} f_{rj} - \sum_{j \in V \setminus \{r\}} f_{jr} = n - 1$$

$$\textbf{(C-5)} \quad \sum_{j \in V \setminus \{i\}} f_{ij} - \sum_{j \in V \setminus \{i\}} f_{ji} = -1, \quad \forall i \in V \setminus \{r\}$$

$$\textbf{(C-6a)} \quad w_{k,g,j} \leq y_{k,g}, \quad \forall (k,g,j) \in \mathcal{T}_{tri}$$

$$\textbf{(C-6b)} \quad w_{k,g,j} \leq 1 - y_{k,j}, \quad \forall (k,g,j) \in \mathcal{T}_{tri}$$

$$\textbf{(C-6c)} \quad w_{k,g,j} \geq y_{k,g} - y_{k,j}, \quad \forall (k,g,j) \in \mathcal{T}_{tri}$$

$$\textbf{(C-7)} S_{ij}\, y_{ij} \;\geq\; \tau \cdot \left(N + \sum_{(k,g,i,j) \in \mathcal{T}_{quad}} (I_{k,g,i,j} \cdot w_{k,g,j}) \right) \cdots - M(1 - y_{ij}), \forall (i,j) \in Y \tag{8}$$

The primary objective is the maximization of τ. The formula is subject to constraints of bi-directionality (C-1), node degree, a strongly connected network via flow conservation from the root, interference rule, and SINR threshold of active links. The model in Equation (8) is an MINLP problem, because C-7 contains the product of the decision variables τ and $w_{k,g,j}$. To ensure computational feasibility and achieve global optimality, we convert the optimization problem into an iterative MILP framework utilizing a bisection search. This transformation is achieved by treating the decision variable τ as a constant parameter. Consequently, the objective function reduces to a feasibility check, since the maximization parameter remains constant within a single execution. The bisection search is demonstrated in Algorithm 1 and explained below.

- **Initialization (Lines 1–2):** The search boundaries, $\hat{\tau}_{low,db}$ and $\hat{\tau}_{up,db}$, are set to their predefined minimum and maximum limits.
- **Bisection Loop (Lines 3–17):** The search iterates until the gap between the upper and lower bounds falls below $\hat{\tau}_{precision,db}$. A smaller precision threshold increases the likelihood of finding the global optimum, but at the cost of higher computational overhead.
- **MILP Transformation (Lines 4–6):** During each iteration, the midpoint SINR threshold $\hat{\tau}_{mid,db}$ is computed and converted to a linear scale ($\hat{\tau}_{mid,lin}$). Treating τ as a constant ($\tau = \hat{\tau}_{mid,lin}$) rather than a decision variable relaxes the original MINLP formulation (Equation (8)) into a computationally feasible MILP.
- **Dynamic Big-M Calculation (Line 6):** Rather than using a generic heuristic, the Big-M constant M is dynamically derived in each iteration. It represents the worst-case scenario in which all candidate interferers are active for Constraint C-7.

Algorithm 1: Bisection Search with MILP

Input: $\mathcal{Y}$, D, $\mathcal{T}_{quad}$, $\mathcal{T}_{tri}$, S_{ij}, $I_{k,g,i,j}$, N, d_{min}, d_{max}, $\hat{\tau}_{min,db}$, $\hat{\tau}_{max,db}$, $\hat{\tau}_{precision,db}$

Output: Optimized Network Topology $G^* = (\mathcal{V}, \mathcal{Y}_{best})$, Optimal Threshold $\hat{\tau}_{best,db}$

1 $\hat{\tau}_{low,db} \leftarrow \hat{\tau}_{min,db}$
2 $\hat{\tau}_{up,db} \leftarrow \hat{\tau}_{max,db}$
3 **while** $(\hat{\tau}_{up,db} - \hat{\tau}_{low,db}) > \hat{\tau}_{precision,db}$ **do**
4 $\quad \hat{\tau}_{mid,db} \leftarrow 0.5 \cdot (\hat{\tau}_{low,db} + \hat{\tau}_{up,db})$
5 $\quad \hat{\tau}_{mid,lin} \leftarrow 10^{\hat{\tau}_{mid,db}/10.0}$
6 $\quad M \leftarrow \hat{\tau}_{mid,lin} \cdot \left(N + \sum_{(k,g,i,j) \in \mathcal{T}_{quad}} (I_{k,g,i,j}) \right)$
7 $\quad \mathcal{Y}_{temp,sol} \leftarrow$ Solve MILP with $\hat{\tau}_{mid,lin}$
8 $\quad$ **if** $\mathcal{Y}_{temp,sol}$ *is feasible* **then**
9 $\quad\quad \hat{\tau}_{low,db} \leftarrow \hat{\tau}_{mid,db}$
10 $\quad\quad \hat{\tau}_{best,db} \leftarrow \hat{\tau}_{mid,db}$
11 $\quad\quad \hat{\tau}_{best,lin} \leftarrow \hat{\tau}_{mid,lin}$
12 $\quad\quad \mathcal{Y}_{best} \leftarrow \mathcal{Y}_{temp,sol}$
13 $\quad$ **end**
14 $\quad$ **else**
15 $\quad\quad \hat{\tau}_{up,db} \leftarrow \hat{\tau}_{mid,db}$
16 $\quad$ **end**
17 **end**
18 **return** $\mathcal{Y}_{best}, \hat{\tau}_{best,db}$

- **Feasibility Check & Update (Lines 7–16):** The solver attempts to find a feasible network topology for the current $\hat{\tau}_{mid,lin}$:
  - If a feasible solution exists, the topology $\mathcal{Y}_{temp,sol}$ is saved as the new optimum, and the lower bound is raised to search for an even higher threshold.
  - If the solution is infeasible, the upper bound is lowered to constrain the search space.

V. Solution Model

A. Graph Neural Network

Our GNN framework adopts a line GNN approach to model the network topology. In a line GNN, each node in the GNN corresponds to a candidate bidirectional link (i, j) between a pair of UAVs from the original set. To characterize the nodes in line GNN, we construct a feature vector consisting of the following link features:

- The Euclidean distance $d_{i,j}$ between the UAVs.
- The relative azimuth and elevation angles $(\theta_{i,j}, \phi_{i,j})$.
- Link betweenness centrality: The sum of the fraction of all-pairs shortest paths that pass through the link.
- Distance-Based Spatial Closeness Centrality: Defined as the inverse of a node's cumulative physical distance to all other nodes, this metric quantifies its structural centrality. We included the spatial closeness values of both endpoints to the link's feature vector.
- The transmission power P_{tx} (mW) and antenna beamwidth (degrees). Since the transmission power and

antenna beamwidth are identical for the bidirectional link node pairs, we have used these values as a link feature.

Adjacency within the GNN is established based on mutual interference or shared endpoints (transmitters or receivers) between the candidate links. To capture the nature of wireless interference, we define these relationships as directed edges in the GNN, pointing from the interfering link ($k \to g$) to the interfered-with link ($i \to j$). We employ multi-dimensional edge attributes to distinguish between these interaction types, incorporating two primary features:

- Endpoint Conflict: A binary indicator specifying whether the two links share a common transmitter or receiver.
- Interference Magnitude: The linear interference power from link $k \to g$ incident on link $i \to j$, calculated according to Equation (7).

To enhance computational efficiency, we apply a threshold to the GNN adjacency matrix. An edge between nodes is established only if the linear interference caused by link $k \to g$ on link $i \to j$ exceeds 1% of the intended signal power, defined by the condition $\frac{I_{k,g,i,j}}{S_{ij}} \geq 0.01$. Adjacencies below this threshold are neglected to prevent the model from being overwhelmed by insignificant interference and to keep the message-passing process from slowing down.

The developed GNN architecture consists of the three components:

- **Feature Embedding**: Both node and edge features are mapped into a high-dimensional latent space using an initial embedding layer.
- **Graph Transformer Message Passing**: We adopt a Graph Transformer [14] for message aggregation, in which attention weights are computed jointly from node representations and edge attributes, integrating self-attention into the line-graph framework. During message passing, the attention mechanism dynamically assigns weights to messages from adjacent nodes, enabling the model to generate incoming messages driven by node and edge features. To ensure stable training and mitigate the vanishing gradient problem, residual connections are integrated into each message-passing block.
- **Node-level Readout**: The message-passing phase is followed by a readout module applied to each GNN node, mapping the refined latent representations to the final output values, which are the logits that represent the probability of each link being included in the final topology.

A deterministic threshold on the predicted scores ($p > 0.5$) does not reliably yield valid configurations: even a small number of false positives or false negatives can violate the constraints or collapse the bottleneck SINR. We therefore propose two frameworks that consume the probabilities directly.

B. Probability-Aware Greedy Edge Pruning

The GNN output is turned into a topology in two stages (Algorithm 2). A baseline graph is formed from the top three candidate edges per node, then pruned in ascending order of predicted probability, with an edge retained whenever its removal would partition the network or violate d_{min}. Link selection therefore inherits the GNN's global interference awareness while structural feasibility is enforced explicitly.

Algorithm 2: Greedy Edge Pruning for Topology Optimization

Input: $G = (\mathcal{V}, \mathcal{E})$, $P = \{p_e \mid e \in \mathcal{E}\}$, d_{min}, $k = 3$
Output: Optimized Network Topology $G^* = (\mathcal{V}, \mathcal{E}^*)$

foreach *node* $v \in \mathcal{V}$ **do**
  $\mathcal{E}_{\text{top,k}} \leftarrow \mathcal{E}_{\text{top,k}} + \mathcal{E}^v_{\text{top,k}}$
end
Sort ($\mathcal{E}_{\text{top,k}}$) in ascending order of predicted probability
$\mathcal{E}^* \leftarrow \mathcal{E}_{\text{top,k}}$
foreach *edge* $e = (u, v) \in \mathcal{E}_{top,k}$ **do**
  Calculate SINR_{min};
  $\mathcal{E}' \leftarrow \mathcal{E}^* \setminus \{e\}$
  Calculate new $\text{SINR}'_{\text{min}}$
  is_safe $\leftarrow (\deg(u) > d_{min}) \wedge (\deg(v) > d_{min}) \wedge$ is_connected
  if $(SINR'_{min} > SINR_{min}) \wedge$ *is_safe* **then**
    $\mathcal{E}^* \leftarrow \mathcal{E}'$
  end
end
return $G^* = (\mathcal{V}, \mathcal{E}^*)$

C. GNN Augmented Grey Wolf Optimizer

Grey Wolf Optimizer (GWO) [15] steers a population of candidate solutions using its three fittest members (α, β, δ). Each wolf is encoded as a binary vector $\mathbf{x} \in \{0,1\}^{|E_{cand}|}$ over the restricted candidate set, where $x_d = 1$ marks link d as active, and fitness is the minimum SINR subject to the network constraints. We modify the standard algorithm in two ways (Algorithm 3): the search space is restricted to the top three GNN-ranked edges per node, and the population is probabilistically seeded from the predicted edge probabilities rather than randomly. Position updates are computed in the continuous domain using the standard GWO rules. The resulting real-valued position is mapped back to a link set by a hard threshold of 0.5. The algorithm details are demonstrated in Algorithm 3.

VI. Methodology

In this section, we explain the methodology we followed to create the dataset, followed by the evaluation methods.

A. Dataset Creation

Due to the lack of publicly available UAV swarm datasets, we created the trajectories with the following characteristics:

- UAVs are placed on a 3D grid with 6 km nominal spacing, perturbed by zero-mean Gaussian jitter with $\sigma = 1.5km$.
- The UAVs follow a Gaussian-Markov mobility pattern, which mathematically simulates realistic flight behavior rather than moving in perfectly straight lines [16]. The velocity of the UAVs ranges from 100 m/s to 200 m/s.

Algorithm 3: GNN-Driven Grey Wolf Optimizer (GWO) for Topology Optimization

Input: $G = (\mathcal{V}, \mathcal{E})$, $P = \{p_e \mid e \in \mathcal{E}\}$, $k = 3$, N_{pop} (Population Size), $T_{patience}$ (Early Stop Threshold)

Output: Optimized Network Topology $G^* = (\mathcal{V}, X_\alpha)$

foreach *node* $v \in \mathcal{V}$ **do**
  $\mathcal{E}_{\text{top,k}} \leftarrow \mathcal{E}_{\text{top,k}} + \mathcal{E}^v_{\text{top,k}}$
end
Initialize population $X = \{X_1, X_2, \ldots, X_{N_{pop}}\}$;
Evaluate fitness $f(X_i)$ for all $X_i \in X$;
  $X_\alpha \leftarrow \arg\max_{X_i} f(X_i)$
  $X_\beta \leftarrow \arg\max_{X_i \setminus \{X_\alpha\}} f(X_i)$
  $X_\delta \leftarrow \arg\max_{X_i \setminus \{X_\alpha, X_\beta\}} f(X_i)$
$t \leftarrow 0$
$t_{\text{last_improve}} \leftarrow 0$
while $(t - t_{last_improve}) < T_{patience}$ **do**
  foreach *wolf* $X_i \in X$ **do**
    Update the position of X_i ;
  end
  Evaluate fitness $f(X_i)$ for all $X_i \in X$;
  if $\max_{X_i} f(X_i) > f(X_\alpha)$ **then**
    $t_{\text{last_improve}} \leftarrow t$
  end
  $X_\alpha \leftarrow \arg\max_{X_i} f(X_i)$
   $X_\beta \leftarrow \arg\max_{X_i \setminus \{X_\alpha\}} f(X_i)$
   $X_\delta \leftarrow \arg\max_{X_i \setminus \{X_\alpha, X_\beta\}} f(X_i)$
  $t \leftarrow t + 1$
end
return $G^* = (\mathcal{V}, X_\alpha)$

Labels were obtained by solving the MINLP with the bisection search of Algorithm 1 in the SCIP Optimization Suite [17], with $\hat{\tau}_{\min,\text{dB}}$, $\hat{\tau}_{\max,\text{dB}}$, and $\hat{\tau}_{\text{precision,dB}}$ set to -5 dB, 10 dB, and 0.05 dB from preliminary runs. Using the communication parameters of Table I, we generated 8000 training and 2000 test samples over disjoint node sets, spanning 10 to 60 nodes with diverse beamwidths and transmit powers. Training is capped at 32 nodes, so that performance above this size measures extrapolation to unseen scales. The dataset and its documentation will be made available at 10.14459/2026mp181516.

B. Training Procedure

In our topology optimization framework, treating edge prediction as a binary classification task introduces severe class imbalance, as rejected candidate links vastly outnumber the optimal links that form the final topology. To prevent the model from being overwhelmed by abundant, easily classified non-optimal edges, we employ Focal Loss [18]. By introducing a modulating factor that dynamically down-weights the loss contribution of highly confident predictions, Focal Loss modifies the standard cross-entropy objective.

The GNN model was implemented in PyTorch, and all training and evaluation were performed on an NVIDIA Ampere A2 GPU. Model optimization was performed using the AdamW optimizer. As described in Section VI-A, the dataset was split into training and test sets to enable rigorous evaluation of model generalization performance. Furthermore, 20% of the training set was allocated as a validation set for performance monitoring during each training epoch. To mitigate overfitting, an early-stopping strategy was employed, terminating training when the validation metric did not improve for 30 consecutive epochs. The model hyperparameters and training configurations are summarized in Table II.

TABLE I: Communication Parameters for 5G URLLC Simulation

Parameter	Value
Minimum number of the connections (d_{min})	$\{1, 2, 3\}$
Number of the antennas (d_{max})	4
Maximum Antenna Gain (G_{max})	20 dB
Side Lobe Attenuation (A_m)	30 dB
Antenna Beamwidth (θ_{3dB}, ϕ_{3dB})	$\{15°, 30°, 45°\}$
Rician K-factor	15 dB
Transmit Power (P_{tx})	$\{14, 17, 20\}$ dBm
Channel Bandwidth (BW)	100 MHz
Carrier Frequency (f_c)	31 GHz
Subcarrier Spacing (Δf)	120 kHz
Number of Resource Blocks (N_{PRB})	66
Symbol Duration (T_s)	8.93 μs
Scaling Factor (f)	0.8
Overhead Percentage (OH)	0.2
MIMO Layers (v_{Layers})	1
Target BLER	10^{-8}

C. Evaluation

The first part of the evaluation focuses on the predictive accuracy of the GNN. To ensure robust assessment across varying network sizes and layouts, we calculate all metrics per graph and report graph-wise means.

We evaluate the practical utility of our GNN-augmented topology creation framework by benchmarking it against an optimal MILP solution, distance-based greedy pruning, and pure GWO baseline. Beyond SINR and data rate metrics, computation time is analyzed.

Link data rates are derived in two stages:

- Instantaneous SINR is mapped to a Modulation and Coding Scheme (MCS) maintaining a Block Error Rate

TABLE II: Model & Training Parameters

Parameter	*Value*
State Dimension	36
Number of GNN layers	4
Number of Readout Layers	4
DropOut Rate	0.15
Activation Function	GeLU
Training and validation ratio	0.80, 0.20
Early stop patience	30
Batch Size	64
Learning Rate	0.0001
Weight Decay	0.005

TABLE III: Per Graph Mean Edge Prediction Performance of the GNN Framework

Category	Metric	Value
Classification Metrics	F1 Score	0.83
	Precision	0.77
	Recall	0.9
	Brier Score	0.07
Confusion Matrix (Ratios)	True Positives (TP)	0.25
	True Negatives (TN)	0.64
	False Positives (FP)	0.07
	False Negatives (FN)	0.02
Class-Specific Accuracy	Accuracy ($y = 1$)	0.91
	Accuracy ($y = 0$)	0.89
Confusion Matrix Probabilities	TP ($\mu \pm \sigma$)	0.91 ± 0.09
	TN ($\mu \pm \sigma$)	0.04 ± 0.10
	FP ($\mu \pm \sigma$)	0.74 ± 0.10
	FN ($\mu \pm \sigma$)	0.21 ± 0.06

[a]All metrics are computed per graph and averaged over graphs; ratios therefore need not sum exactly to one.

(BLER) of 10^{-8} for Ultra-Reliable Low-Latency Communications (URLLC) compliance [19].

- Physical Layer Rate (R) is calculated per 3GPP TS 38.306 [20] as follows:

$$R = v_{Layers} \cdot Q_m \cdot f \cdot R_{code} \cdot \frac{N_{PRB} \cdot 12}{T_s} \cdot (1 - OH) \quad (9)$$

where v_{Layers} is the number of MIMO layers, Q_m is the modulation order, f is the scaling factor, R_{code} is the code rate of the selected MCS, N_{PRB} is the number of allocated resource blocks, OH is the signaling overhead, and T_s is the average OFDM symbol duration.

VII. Results and Discussion

As explained in Section VI-C, we first evaluated the optimal edge estimation capabilities of the developed model and its ability to differentiate between optimal and non-optimal edges.

The edge prediction performance of the GNN is summarized in Table III. The model achieves a high recall (0.9), indicating that it successfully identifies the majority of optimal link candidates while maintaining a very low probability of missing critical edges. While the precision (0.77) is slightly lower due to a moderate false-positive rate, suggesting an optimistic estimate of optimal links, the resulting F1 score of 0.83 demonstrates a robust balance between link discovery and selection accuracy.

TABLE IV: Mean Performance Summary of Topology Optimization Algorithms

Algorithm	Mean Min SINR (dB)	Mean Min Data Rate (Mbps)	Mean Execution Time (s)
MILP	0.057	27.359	430.560
Pure GWO	-1.039	15.191	91.491
Distance-Based Greedy Pruning	-4.078	9.597	0.138
GNN + Greedy Pruning	-0.540	23.025	0.160
GNN + GWO	-0.324	24.057	1.839

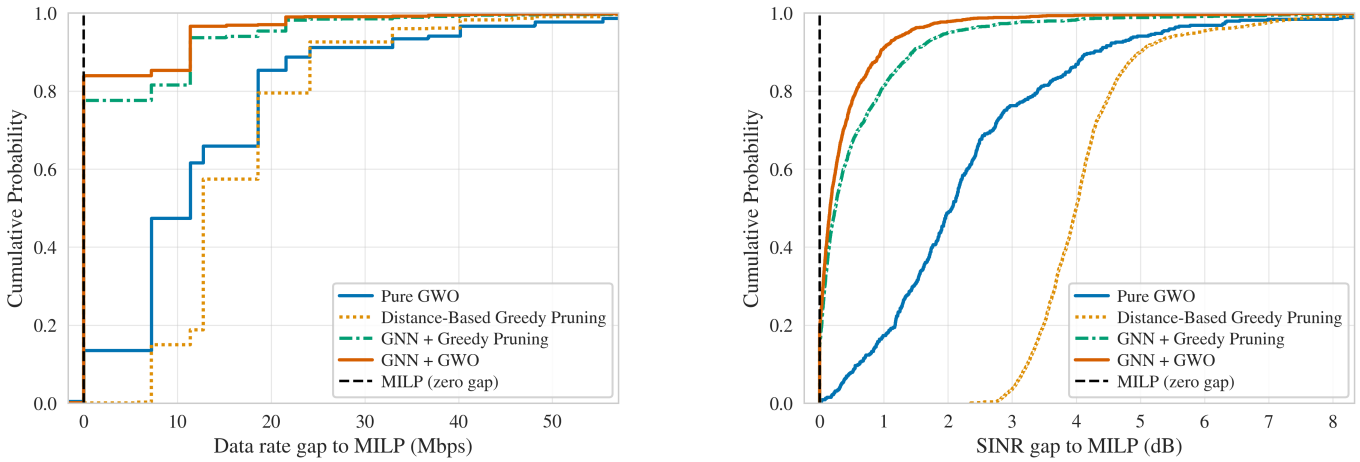
Furthermore, the low Brier score (0.07) confirms that the model is well-calibrated, meaning its predicted probabilities closely reflect the true likelihood of link optimality. This high degree of calibration is further supported by the mean predicted probabilities across the classification outcomes: the exceptionally high mean probability for True Positives (0.91) and the very low mean probability for True Negatives (0.04) indicate that the model's highly confident estimations are correct. Conversely, the mean probabilities for False Positives (0.74) and False Negatives (0.21) indicate that misclassifications primarily occur when the model's certainty decreases, and predictions move closer to the decision boundary.

In the next phase of our evaluation, we benchmark the GNN-augmented methods against the exact MILP solver and baselines. Table IV presents the mean bottleneck link SINR, minimum E2E data rate, and execution time for each algorithm. The bottleneck link SINR and minimum E2E data rate are the main points of our performance analysis. Furthermore, we analyze execution time to quantify the computational speedup over the exact MILP solver. As shown in the table, the GNN-augmented algorithms consistently outperform their pure GWO and distance-based Greedy-Pruning counterparts. Additionally, integrating the GNN with GWO yields substantial computational advantages; the GNN-augmented GWO operates approximately 49.8 times faster than pure GWO. This significant acceleration is primarily due to reducing the search space and probabilistically seeding the population rather than randomly generating it. Among the augmented approaches, GNN + GWO outperforms GNN + Greedy Pruning. This superior performance results from the evolutionary nature of GWO, which thoroughly explores the solution space.

To further evaluate the efficacy of the proposed framework, we analyze the CDF of the optimality gap for both bottleneck SINR and minimum E2E data rate. The optimality gap represents the absolute performance difference between the topologies generated by the heuristic algorithms and the optimal solutions derived from the MILP. As illustrated in Figure 1, the GNN-augmented algorithms outperform the pure baselines. In terms of data rate, GNN + GWO and GNN + Greedy Pruning match the MILP solution in approximately 85% and 78% of the evaluated scenarios, respectively. Without GNN guidance, pure GWO and the distance-based baseline exhibit wider optimality gaps, frequently deviating from the upper bound by three to five dB or more.

In addition to the aggregated performance metrics, we analyze the algorithms' performance as a function of network size, as illustrated in Figure 2. For networks with fewer than 35 nodes, the GNN-augmented algorithms generate topologies that closely approximate the optimal MILP solutions. Beyond 35 nodes, the optimality gap between the proposed methods and the exact solver increases.

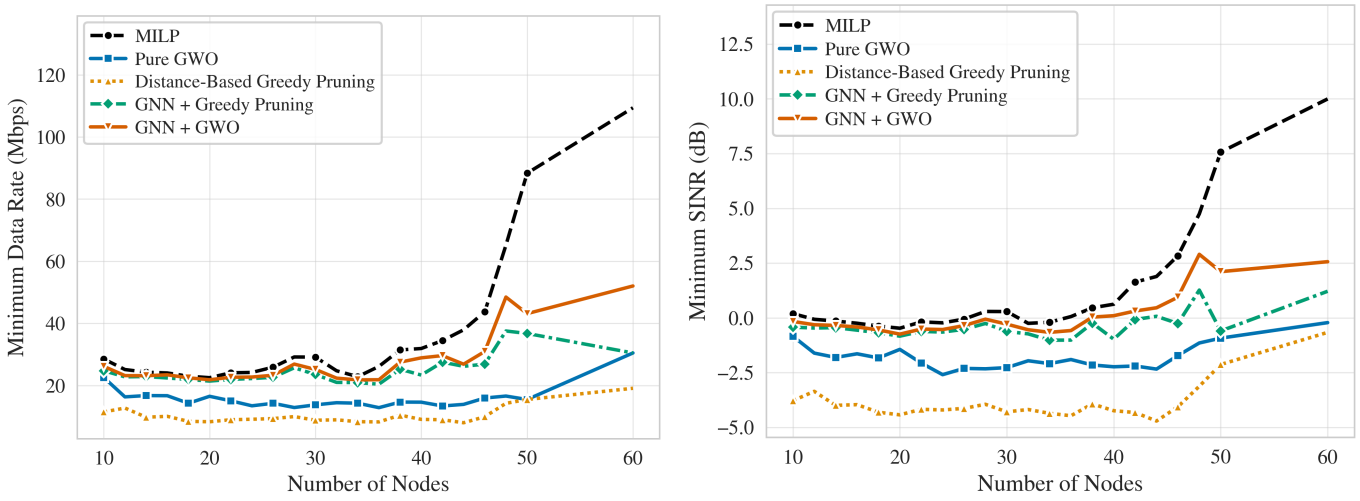
We attribute this to three primary factors: centrality features computed over the full node set shift out of their training distribution as network size grows; the interference-induced line-graph adjacency densifies, causing a fixed four-layer receptive field to aggregate over statistically different neighborhoods;



(a) CDF Minimum Data Rate Gap

(b) CDF Minimum SINR Gap

Fig. 1: Cumulative Distribution Function (CDF) of the optimality gap of the GNN augmented and baseline algorithms compared to MILP



(a) Bottleneck Link Data Rate

(b) Bottleneck Link SINR

Fig. 2: The impact of network size on the bottleneck link data rate and bottleneck link SINR

and the bottleneck objective is governed by a minimum, meaning a single misclassified critical link caps the overall result while the total link count grows linearly with network size. The fixed candidate budget $k = 3$ compounds this limitation, as it covers a shrinking fraction of the potential candidates. A potential mitigation is adaptive candidate selection instead of a fixed budget, but we leave its formal evaluation to future work.

Nevertheless, the value added by the GNN becomes increasingly pronounced at scale: the performance margin between the GNN-augmented solutions and their pure heuristic baselines expands significantly as the network size grows. This scaling behavior remains consistent across both the bottleneck link SINR and its corresponding data rate.

Finally, Figure 3 illustrates the computational time of the algorithms, demonstrating the primary advantage of the GNN-augmented solutions over MILP. As network size increases, the execution time of the MILP solver explodes, making it practically unfeasible for deployment. Conversely, GNN-augmented methods maintain low latency compared to both MILP and pure GWO. The GNN + Greedy Pruning algorithm executes in under a second for all network sizes. Meanwhile, the GNN + GWO approach scales moderately from 1 to 8 seconds. Because GNN + Greedy Pruning achieves topological performance nearly identical to that of GNN + GWO, it offers an excellent trade-off for highly time-sensitive applications.

VIII. Conclusion and Future Work

In this study, we investigate the application of a GNN for topology creation in an Airborne Network consisting of UAVs

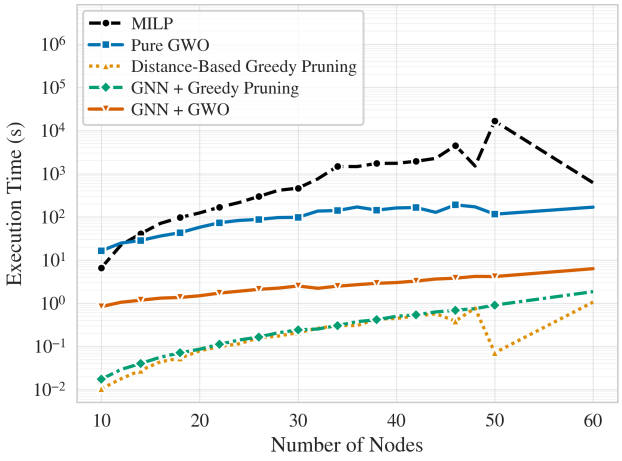


Fig. 3: Computation Time vs Number of Nodes

equipped with directional mmWave antennas. The primary objective is the maximization of the bottleneck link SINR. To address the NP-hard complexity of this problem, we develop and train a GNN to predict optimal edges within a practical time frame, offering a computationally efficient alternative to exact MILP solvers. Building on the developed GNN, we introduce two frameworks for integrating these predictions into the topology synthesis process: probability-aware greedy edge pruning and GNN-GWO.

We assess the system performance in terms of computation time, bottleneck link SINR, and achievable data rate. Our results demonstrate that the GNN-augmented algorithms significantly outperform traditional distance-based Greedy-Pruning and pure GWO, closely approaching the optimal topology SINR while drastically reducing computational overhead.

The framework applies to mission profiles in which a swarm transits to an area of interest and then holds a quasi-static formation for an interval. Surveillance and disaster response are the clearest use cases: a swarm monitoring a wildfire front or relaying for ground teams occupies mission-dictated station-keeping positions. In coverage tasks flown as parallel-track sweeps, the geometry is constant along each leg and changes only at the turns, so re-optimization is driven by formation changes and residual drift rather than by continuous motion. What these profiles share is a short window in which the topology must be recomputed once the geometry does change, and it is here that our GNN-augmented frameworks are applicable, returning a near-optimal topology within that window, whereas an exact MILP solver would still be running once the formation has moved on.

References

[1] H. Shakhatreh, A. H. Sawalmeh, A. Al-Fuqaha, Z. Dou, E. Almaita, I. Khalil, N. S. Othman, A. Khreishah, and M. Guizani, "Unmanned aerial vehicles (uavs): A survey on civil applications and key research challenges," *IEEE access*, vol. 7, pp. 48 572–48 634, 2019.

[2] W. Zhang, W. Zhang, and J. Wu, "Uav beam alignment for highly mobile millimeter wave communications," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 8, pp. 8577–8585, 2020.

[3] K. Jain, J. Padhye, V. N. Padmanabhan, and L. Qiu, "Impact of interference on multi-hop wireless network capacity," in *Proceedings of the 9th annual international conference on Mobile computing and networking*, 2003, pp. 60–73.

[4] R. Ramanathan and R. Rosales-Hain, "Topology control of multihop wireless networks using transmit power adjustment," in *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No. 00CH37064)*, vol. 2. IEEE, 2000, pp. 404–413.

[5] W. Zhang, L. Peng, R. Xu, and L. Zhang, "Topology control in wireless mobile ad hoc networks with directional antennas," in *2016 2nd IEEE international conference on computer and communications (ICCC)*. IEEE, 2016, pp. 1701–1705.

[6] S. Hofmann, V. Megas, M. Ozger, D. Schupke, F. H. Fitzek, and C. Cavdar, "Combined optimal topology formation and rate allocation for aircraft to aircraft communications," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. IEEE, 2019, pp. 1–6.

[7] V. Megas, S. Hoppe, M. Ozger, D. Schupke, and C. Cavdar, "A combined topology formation and rate allocation algorithm for aeronautical ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 23, no. 1, pp. 12–28, 2022.

[8] B. Newton, J. Aikat, and K. Jeffay, "Explicit topology management for continental-scale airborne networks," in *2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2017, pp. 72–77.

[9] F. Marcoccia, C. Adjih, and P. Mühlethaler, "Attentive, permutation invariant, one-shot node-conditioned graph generation for wireless networks topology optimization," in *International Conference on Machine Learning for Networking*. Springer, 2023, pp. 14–31.

[10] ——, "A deep learning approach to topology configuration in multi-hop wireless networks with directional antennas: nodes2net," in *2023 12th IFIP/IEEE International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks (PEMWN)*. IEEE, 2023, pp. 1–6.

[11] E. Haas, "Aeronautical channel modeling," *IEEE transactions on vehicular technology*, vol. 51, no. 2, pp. 254–264, 2002.

[12] X. Cao, P. Yang, M. Alzenad, X. Xi, D. Wu, and H. Yanikomeroglu, "Airborne communication networks: A survey," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 9, pp. 1907–1926, 2018.

[13] 3rd Generation Partnership Project (3GPP), "Study on channel model for frequency spectrum above 6 GHz," 3GPP Technical Specification Group Radio Access Network, Technical Report (TR) 38.900, 6 2018.

[14] Y. Shi, Z. Huang, S. Feng, H. Zhong, W. Wang, and Y. Sun, "Masked label prediction: Unified message passing model for semi-supervised classification," *arXiv preprint arXiv:2009.03509*, 2020.

[15] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in engineering software*, vol. 69, pp. 46–61, 2014.

[16] J.-D. M. M. Biomo, T. Kunz, and M. St-Hilaire, "An enhanced gauss-markov mobility model for simulations of unmanned aerial ad hoc networks," in *2014 7th IFIP Wireless and Mobile Networking Conference (WMNC)*. IEEE, 2014, pp. 1–8.

[17] T. Achterberg, "Scip: solving constraint integer programs," *Mathematical Programming Computation*, vol. 1, no. 1, pp. 1–41, Jul 2009. [Online]. Available: https://doi.org/10.1007/s12532-008-0001-1

[18] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[19] L. Li, Q. Jiang, and W. Luo, "A unified non-cqi-based amc scheme for 5g nr downlink and uplink transmissions," in *2021 IEEE 6th International Conference on Computer and Communication Systems (ICCCS)*. IEEE, 2021, pp. 881–886.

[20] 3rd Generation Partnership Project (3GPP), "Technical Specification Group Radio Access Network; NR; User Equipment (UE) radio access capabilities (Release 18)," 3GPP, TS 38.306, 2024.