User Association in Wireless Networks with Distributed GNN-Based Reinforcement Learning

Martín Randall*, Santiago Paternain‡, Pedro Casas†, Federico Larroca*, Pablo Belzarena*
*Instituto de Ingeniería Eléctrica, Facultad de Ingeniería, Universidad de la República, Uruguay
‡ Electrical Computer and Systems Engineering, Renssealaer Polytechnic Institute, USA
†AIT, Austrian Institute of Technology, Austria
{mrandall, belza, flarroca}@fing.edu.uy, paters@rpi.edu, pedro.casas@ait.ac.at

***Abstract*—User association is crucial for optimizing the performance and utility of wireless networks, enhancing key aspects such as load balancing, spectrum efficiency, energy efficiency, and overall network performance. In this paper we tackle the user association challenge in wireless networks, particularly in resource-constrained connectivity scenarios. Our proposed approach, GROWTh (Graph Representation of Wireless systems Throughput fair), introduces a graph-based reinforcement learning framework that optimizes resource utilization through a fully decentralized algorithm. We validate GROWTh across diverse scenarios, including a 5G deployment in densely populated areas characterized by high user density and traffic load, where it demonstrates significant improvements in various performance metrics. Notably, GROWTh achieves a substantial increase in system utility compared to traditional methods while simultaneously reducing user rejection rates. These findings highlight the effectiveness of GROWTh in managing user association in high-density environments and underscore its potential for real-world deployment.**

***Index Terms*—User Association; Mobile Networks; Reinforcement Learning; Graph Neural Networks**

I. Introduction

As wireless networks – encompassing FANETs, satellite constellations, 5G, and beyond – evolve at a rapid pace, the challenge of user association with the infrastructure has grown increasingly complex. The diverse range of services, high traffic demand, user mobility, and the infrastructure topology have rendered traditional user association techniques ineffective in many scenarios. Even in the case of 5G, the conventional techniques fall short, particularly during high-traffic events, underscoring the pressing need for advanced solutions. Performance degradation ranges from low throughput [1] to even occasional outages [2]. Not surprisingly, user density is a key aspect of service downgrade [3]. User association has been a key focus since the early days of wireless systems, traditionally managed through simple policies like maximizing the signal-to-noise ratio, yet only recently have the limitations of classic over-provisioning for infrastructure deployment become evident.

The user association problem can be formulated as an optimization problem: to which network provider's node (e.g., UAV, satellite, base station) a user should be connected to maximize a global system utility function, generally throughput-related. Finding the optimal policy is typically intractable, as the assignment problem is NP-hard and the number of possible states becomes too large to explore exhaustively. However, by combining state-of-the-art artificial intelligence with simple and robust system models, there is great potential for advancing user association techniques.

User association challenges can be framed as a sequential decision-making problem, making deep reinforcement learning an ideal approach for modeling wireless systems. To enhance convergence, we approximate the value function using a Graph Neural Network (GNN). By leveraging the system's underlying graph structure, this novel machine-learning technique enables a decentralized algorithm that not only scales but also generalizes to unseen scenarios, underscoring its adaptability. We address user association with a focus on the distributed nature of our approach. Our main contributions are as follows:

User modeling: we consider a non-traditional approach to users' modeling as they come and leave the system, being either served or not (and satisfied or unsatisfied). The departure of users is particularly challenging, as a superficial modeling could violate the memoryless condition (the markovian property in which the current state is enough and resumes past history). To address this issue, we propose a system's state representation respecting the Markovian properties, a key aspect enabling the application of reinforcement learning with guarantees.

Distributed decision making: Several important scenarios, either in next generation wireless systems or in FANETs deployments, call for a fully-distributed user association algorithm. This is precisely our goal in this work, where we leverage some of GNN's properties to design a fair and user-centric algorithm, which performs significantly better than other distributed baselines.

Validation results and code reproducibility: we share both our code and our experiments for the sake of reproducibility and as a contribution to the field. Code is available at https://gitlab.fing.edu.uy/mrandall/growth.

The rest of the paper is organized as follows. Sec. II presents an overview on related work. In Sec. III we introduce the problem statement, as well as the system modeling and learning formulation proposed. The distributed algorithm is presented in Sec. IV. In Sec. V we summarize implementation details and benchmark results in different scenarios. Concluding remarks and future lines of work are discussed in Sec. VI.

II. Related Work

User association (UA) plays a central role in numerous surveys on wireless system radio access networks [4], [5]. To summarize, solutions to user association challenges in the literature can be broadly categorized into three approaches: classical optimization techniques, machine learning methods, and other heuristic strategies.

In classical optimization, the need for convexity results in simpler modeling [6], [7]. A typical approach is solving the dual Lagrangian optimization problem, but to avoid a duality gap, additional constraints must be introduced, which simplify both the problem's modeling and dynamics [8]. Other proposals jointly optimize user allocation and providers placement, for example in the event of a UAV deployment or when choosing relay base stations [9]–[11]. As a turnaround for mixed-integer and no-convex programming problems risen from the discrete arrivals and fairness utilities (i.e. log sum or similar), many articles extend the primary optimization problem through relaxations, achieving suboptimal yet working solutions [9], [12], [13]. Other proposals involve jointly optimizing user association and power allocation, in order to minimize the system's energy consumption, which can be tackled through dividing the optimization problem into two (or more) sub-problems [9], [10], [14].

The machine learning boom has promoted data-centered approaches, where user association policies are not derived from a mathematical setting but are learned instead [15]. A particularly fruitful paradigm for resource allocation is reinforcement learning, as an agent makes decisions (following a certain policy) and learns from experience (improving such policy). In [16], authors propose the use of multi-agent reinforcement learning to first choose the providing base station, and afterwards dealing with power allocation, yet they minimize time delays with no considerations on fairness. Reinforcement learning (RL) suffers from *the curse of dimensionality*, and the mainstream approach to overcome this curse is to approximate some function of interest of our problem by using (mostly) supervised learning. As communications systems are prone to graph representation, these RL formulations have been pioneers in using GNNs to approximate the desired function. An important feature for allocation policies is the possibility for distributed resource assignment, avoiding the bottlenecks introduced by information gathering and/or centralized algorithms. This property makes of graph representations an interesting choice, but their application to the UA problem has usually fallen short of decentralized solutions [17]–[19].

There are interesting heuristics that are driven by the maximization of the allocated users, and have the virtue of proposing online fast-adapted algorithms for user association and frequency selection as in [20], [21]. These works do not take into account fairness in the utility achieved, which in our opinion is a must when dealing with resource allocation, and has to be included in the objective functions. A recent work optimizing network resources, [22] proposes the use of bipartite graphs to model user and base stations but focusing on the coverage maximization problem through manipulation of the base stations' transmission power and antenna alignment, not tackling the user association scheme. They state the virtues of the permutation equivariance property, which will also be a part of our current proposal.

Most similar to our proposal [23], tackles the user association problem by using a multi-agent reinforcement learning algorithm. To simplify the problem, they consider that each base station can serve only one user at each time and use the *max SINR* policy by default, only choosing which user to serve and at what time. On their proposal, [24] optimize the user association and transmit beamforming vectors in order to mitigate interference by using safe reinforcement learning, and iteratively update both decisions: user association and beamforming. Although not all, many works insist on the fairness of the proposed solutions, as in [11], [25].

This paper builds on our previous work on AI-driven user association [26], [27], where we introduced an initial version of GROWS under significantly simpler conditions. In contrast, the new version of GROWTh

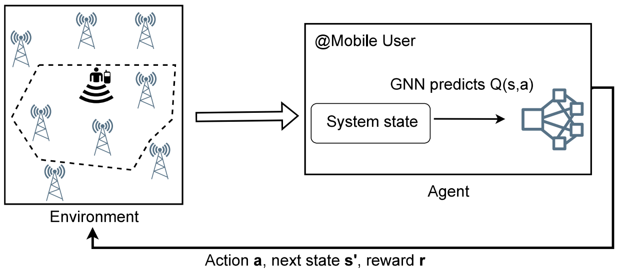


Figure 1. System model. We consider at most one arrival at each time-step. The combination of past decisions and the present arrival constitutes the state of the system. The choice of which base station associates with the currently arrived user is the action. After executing an action, a new state is observed and a reward is obtained.

presented here features a fully distributed, user-centric algorithm that eliminates the need for a central decision-making agent. Additionally, we propose a novel graph representation that incorporates the possibility of user rejection, enabling more efficient utilization of system resources. We emphasize that our new proposal takes the fairness of the resource distribution into account, and rename our algorithm Growth (Graph Representation Of Wireless systems THroughput fair). This enhanced representation allows for the modeling of more complex and realistic scenarios, extending GROWTh applicability to a wide range of wireless networks, including next-generation mobile systems, FANETs, and beyond.

III. Problem Statement

We consider the problem of user association in a system with N base stations – see Figure 1. Let us denote the time by a discrete index $t \in \{1, \dots, T\}$. At each timeslot t, a user may arrive according to a Bernoulli distribution of parameter $p \in [0, 1]$. Let the index $u \in \mathbb{N}$ represent the order of arrival of the users, and t_u^i and t_u^f the arriving and departing times of user u. The u-th user has a random discrete demand $d_u \in \mathbb{N}$, and an SNR with base station n denoted by SNR_u^n with $n = 1, \dots, N$. Base stations have a set of time-frequency resources (resource blocks, according to 5G terminology) $RB \in \mathbb{N}$ to distribute among connected users.

When a candidate user u arrives, the system selects **action** $a_u \in \{1, \dots, N\}$ and the user is served by base station a_u, or the system selects $a_u = 0$ and the user is rejected – the system having no available resources or choosing to reject.

Let us define the system's **state** s_u as the combination of the user's and base stations' features, observed only on decision times corresponding to arrival times $t_u^i \forall u \in U$. We summarize the base station's state through the number of connected users and time statistics related to the service given to the users: mean, variance, and minimum of the estimated connection times with active users. We then have the system's state $s_u \in \mathbb{R}^{N \times 6}$, where each row is the state of each base station:

$$\begin{aligned} s_u = s(t_u^i) = [&users(t_u^i, n), SNR_{u,n}, d_u, \\ &\widehat{t_\mu^f - t_u^i}, \min t_\mu^f - t_u^i, var(t_\mu^f - t_u^i)]^{k \in \mathbb{N}} \\ &\forall \mu \in \mathcal{U}/t_\mu^f > t_u^i \end{aligned} \quad (1)$$

with $users(t_u^i, n)$ denoting the number of users served by base station n on decision instance u, while $\hat{t}_\mu^f, \min t_\mu^f, var(t_\mu^f) \forall \mu \in \mathcal{U}/t_\mu^f > t_u^i$ denotes the mean, minimum, and variance of the estimated ending times for connected users, respectively.

Transition to the next state s_{u+1} results from a combination of the system's dynamics (the stochastic arrival of users) and the actions taken (the number of users connected to a base station, which is deterministic). Let us study the potential next states more closely.

Starting from state s_u, the number of users can increase based on the action $users(t_u^i, n) = users(t_u^i, n) + 1(a_u = n)$; the number of users may also decrease, as users leave after their demand has been satisfied between arrivals.

The next state time estimates for connected users to base stations n are in direct relationship to different factors, including: the peak rate that users have, the number of users being served, and the remaining demand for each user. These factors are entirely determined by the current state and the action taken. As noted, the users' features are stochastic.

We can now define the instantaneous **reward** of the system as the utility achieved for connected users between this decision instant and the following, after taking action a_u.

$$R_{u+1} = \sum_{\mu \in \mathcal{U}/t_\mu^f > t_u} \sum_{t \in [t_u^i, t_{u+1}^i]} \mathcal{F}(th_{\mu,t})$$
$$th_{u,t} = \Phi(u, t) r_u(t)$$

where $\mathcal{F}$ is an utility function evaluated on the active user's throughput over a timeslot. Furthermore, throughput for a user is proportional to the assigned resources and the rate achieved ($r_u(t)$). By Φ we denote the internal distribution criterion that base stations follow; for example, a simple Φ policy could be to distribute resources evenly between connected users [28]. The utility function used is $\log(1+th_{u,t})$, and we base this utility and reward formulation on [26], [28].

Actions are selected according to a random policy parameterized by a vector $\theta \in \mathbb{R}^n$, i.e., $a_u \sim \pi(s_u;\theta)$. We wish to find the policy π^* that maximizes the accumulated utility applied to the users' throughput over time. Finally, we define the discount factor $\gamma \in [0,1]$, which serves as a measure of the importance of the future, with which we state the RL problem as follows:

$$\max_{\theta\in\mathbb{R}^n} \mathbf{E}_\pi \Big[\sum_{\nu=\nu_0}^{\infty} \gamma^{\nu-\nu_0+1} R_\nu \Big| a_u \sim \pi(s_u;\theta) \Big]$$

Using the Bellman equations, our algorithm will look for the parameterized policy that maximizes the expected discounted reward:

$$\max_{\pi(\theta)} v_\pi(s) = \max_{\pi(\theta)} \mathbf{E}_\pi [G_u | S_u = s_u]$$

$$G_v = \lim_{V\to\infty} \sum_{\nu=0}^{V} \gamma^\nu R_{v+\nu+1}$$

As the action and state spaces grow, the convergence of traditional reinforcement learning algorithms becomes increasingly challenging, and in some cases, computationally infeasible. A common approach is to approximate the policy ($\pi(s,\theta)$) or the value function ($Q(s,a,\theta)$), by learning the parameters θ that best represent the desired target function.

IV. GROWTH: A DISTRIBUTED ALGORITHM

We address the approximation of the Q-value function by using graph neural networks, which in its more general form can be seen as a convolutional neural network applied to a graph structure [29], [30]. Assume we have a certain graph $G = (V, E)$, which we represent through a so-called Graph Shift Operator (GSO). That is to say, a matrix $\mathbf{S} \in \mathbb{R}^{|V|\times|V|}$ which respects the graph sparsity ($\mathbf{S}_{i,j} \neq 0$ whenever an edge between nodes i and j exists); e.g. the adjacency matrix. To each node we associate a certain d-dimensional vector, resulting in the graph signal matrix $\mathbf{X} \in \mathbb{R}^{|V|\times d}$.

By computing the product $\mathbf{SX}$, we generate a new graph signal where each node aggregates information from its neighboring nodes. By expressing $\mathbf{S}^K\mathbf{X} = \mathbf{S}^{K-1}(\mathbf{SX})$, the resulting operation represents the aggregation of information from nodes that are K hops away.

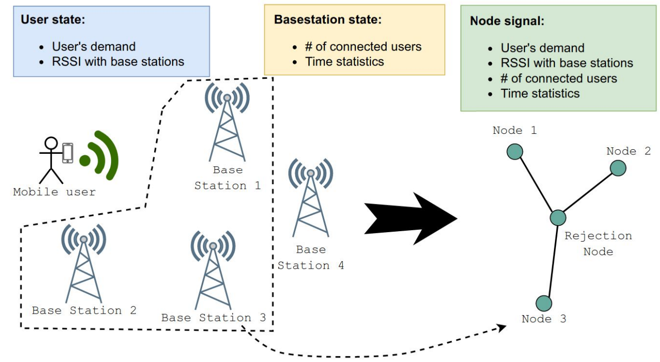


Figure 2. Graph representation of the system, where the star graph is constructed with the rejection node at the center. The rest of nodes represent base stations and the signal of each node is a composition of the base station and the user's state.

In its most basic implementation, we obtain a single-layer GNN (or graph perceptron) by applying a point-to-point non-linear function $\sigma(\cdot)$ to a linear combination of these K signals:

$$\mathbf{X}' = \sigma\left(\sum \mathbf{S}^k \mathbf{X} \mathbf{H}_k\right), \tag{2}$$

where $\mathbf{H}_k \in \mathbb{R}^{d\times d'}$ (for $k = 0,\ldots,K$) act as the graph filter taps (and change the signal's dimension from d to d'). By stacking L of these operations we obtain an L-layered GNN.

A key feature of the proposed user association and access control mechanism is its fully distributed nature. This means that the algorithm is designed to operate without relying on a centralized decision-making entity for each incoming user. Instead, each user independently acquires the state of all base stations, represented by the matrix $\mathbf{X}$, and decides which base station, if any, connecting to. If the policy is approximated using a function with a vector input, such as a Fully-Connected Neural Network, certain limitations arise. This approach requires a fixed ordering of the nodes and becomes infeasible if the vector's dimensionality changes, for example, when the number of nodes in the network increases.

Firstly, note that a GNN may be executed in a graph with any given number of nodes (since it is characterized by its filter taps $\mathbf{H}_k$). Secondly, in the construction of the GSO $\mathbf{S}$ we have also arbitrarily chosen an order for the nodes.

However, unlike a vector-based representation of the problem, a GNN is independent of the node ordering. This is due to its *permutation equivariant* property, which ensures the output remains consistent regardless of how the nodes are arranged. To understand this, consider a permutation matrix $\mathbf{P}$, which is a binary matrix satisfying $\mathbf{P1} = \mathbf{1}$ and $\mathbf{P}^\top\mathbf{1} = \mathbf{1}$. Reordering the nodes and then computing the filter's output, represented as $(\mathbf{P}^\top\mathbf{SP})(\mathbf{P}^\top\mathbf{X})$, yields the same result as first computing the output $(\mathbf{SX})$ and then reordering it, represented as $\mathbf{P}^\top(\mathbf{SX})$. This permutation-equivariant property is naturally inherited by a GNN, as it performs pointwise operations on the output of the filter.

As explained before, the algorithm is executed by the incoming user. To build the graph representation, each incoming user constructs a graph where the base stations are nodes and the signal is the base stations' state. Instead of considering the whole network, the user only includes those base stations it receives with the highest quality. Note that the involved base stations' connections are represented through the GSO operator, making our system capable to adapt to different wireless systems (i.e. FANETs, 5G, etc.), and that the base stations' state includes information about the SINR but also about their current utilization (number of connected users, time statistics). In wireless systems (as 5G), mobile users exchange information with surrounding base stations - with different system information as the CQI (Channel Quality Indicator) and other- and then negotiate the connection with their preferred choice, meaning our proposal does not involve any communication overhead. Finally, and to allow the policy rejecting users, we also include an extra node representing this decision. In this case, the graph is constructed as a star, with this "reject node" at the center, see Fig. 2. In order to avoid bottlenecks at the rejection node, self-loops are added to the shift operator, a common approach in graph convolutional networks.

This construction, which we shall refer to as GROWTh (Graph Representation of Wireless systems Throughput fair), enables a fully distributed algorithm without requiring any communication between base stations. Instead, the incoming user independently gathers all necessary information. Notably, the time and energy required for the user to compute the optimal policy are minimal, as this process involves only collecting the states of the base stations and performing a forward pass through a pre-trained GNN. Furthermore, by leveraging a GNN to define the policy, the algorithm becomes permutation equivariant, ensuring that decisions remain consistent regardless of how the nodes are reordered. The advantages of this property will be highlighted and discussed in the following section.

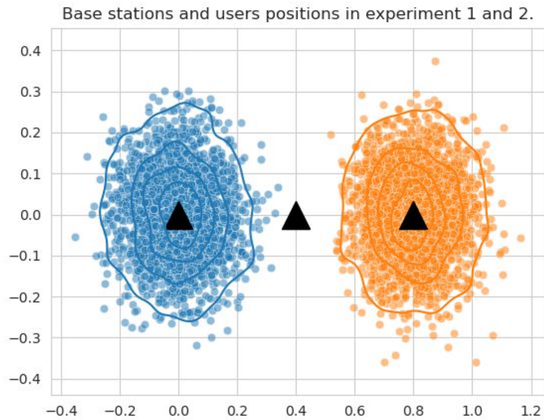
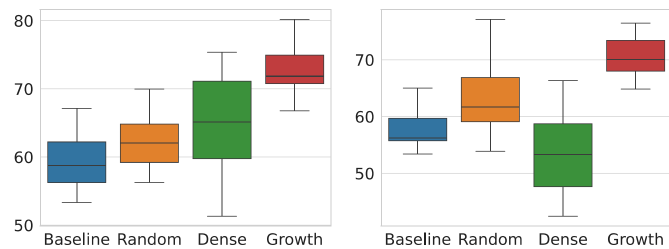


Figure 3. During training, users arrival is centered on the leftmost base station, whereas during test, users' arrival has shifted towards the base station to the right.



(a) Utility achieved, 1st scenario (Exp 1). (b) Utility achieved, 2nd scenario (Exp 2).

Figure 4. Comparison of both scenarios in the permutation experiment. When tested in scenarios similar to those encountered during training, both versions of the DDQN perform well (left figure). However, when the arrival patterns change and users come close to the other base station, as GROWTh maintains its performance, the dense version of our algorithm struggles to adapt and performs poorly (right figure). Unsurprisingly, both random and baseline achieve the same results for both scenarios.

V. Evaluation and Results

To develop a deep reinforcement learning implementation, we begin with the Double Deep Q-Learning Network (DDQN) algorithm due to its simplicity and stable convergence properties. While more advanced DRL algorithms exist, our primary objective is to validate that the implemented (Markov) decision process is well-defined and to demonstrate that the GNN outperforms a traditional fully connected neural network. In a nutshell, DDQN enhances stability by using two separate deep networks: a policy network for action selection, and a target network for action evaluation.

We develop two DDQN algorithms, one in which the Q-value function is approximated by a GNN, and another in which we use a classic three-layers fully connected neural network. The GNN architecture we take is implemented by using the Graph Convolutional Network (GCN) as defined in [29], and its Pytorch implementation – GCNConv.

We conduct a series of experiments, focusing on two specific settings that highlight the strengths of our proposal. First, we aim to demonstrate a key aspect of our algorithm: its ability to adapt to unseen yet similar scenarios, leveraging the permutation equivariance of the GCN. Second, we evaluate the algorithm's performance in a real-world 5G deployment in the city of Paris. All examples and their selected parameters and hyper parameters can be found at https://gitlab.fing.edu.uy/mrandall/growth.

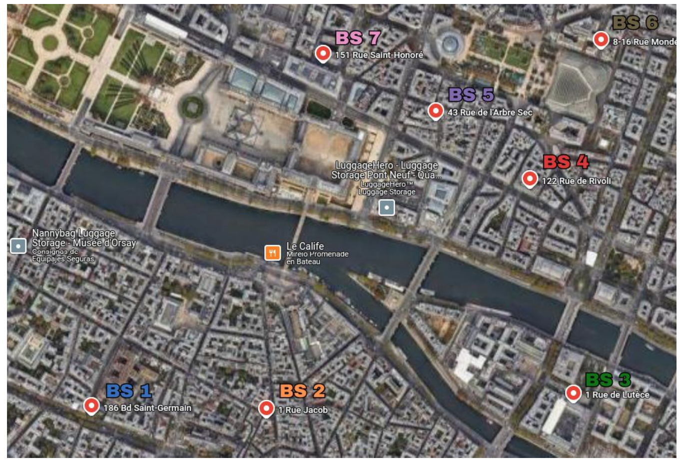


Figure 5. We select a densely populated area where large crowds gather, and select the 5G base stations deployed by a specific mobile operator. We randomize mobile users with higher density around the center of the figure, and random arrival times and demands.

Table I
SUMMARY OF THE RESULTS ACHIEVED FOR THE TWO SCENARIOS DESCRIBING THE PERMUTATION EXPERIMENT, AND FOR THE PARIS EXPERIMENT.

	Mean Utility			**Mean # of Rejections**		
	Exp 1	**Exp 2**	**Paris**	**Exp 1**	**Exp 2**	**Paris**
Baseline	59.8	58.5	47.1	76	72	97.5
Random	61.9	63	44.3	61	**56.2**	90
Dense	64.9	53.4	40.9	**59**	60.6	107.4
GROWTh	**73.2**	**70.5**	**51.4**	62	57.5	**84.5**

A. *Permutation Equivariance*

To evaluate our proposal, we compare it against three different approaches: (a) a **baseline** algorithm that selects the base station with the strongest signal-to-interference-plus-noise ratio (SINR), (b) a **random** policy, and (c) a modified version of our algorithm that employs a **dense** network instead of the GNN. We refer to our proposal as **GROWTh**.

We first consider a simple setting of three base stations in a straight line, as shown in Figure 3. During training, users arrive according to a normal distribution centered around the leftmost base station. We subsequently test two scenarios, referred to as Experiments 1 and 2: one based on the initial setup and another where users cluster around the rightmost base station. Numerical results from these experiments are presented in Table I, in the columns marked as **Exp 1** and **Exp 2**.

Results for throughput utility are reported in Figures 4(a) and 4(b). As expected, the fully connected network performs well in scenarios similar to those it was trained on; however, it struggles to adapt to even minor changes. In contrast, the permutation equivariance property of the GNN allows GROWTh to adjust effectively, yielding results in experiment 2 comparable to those achieved with the training realized in the first experiment. Naturally, both the random and the baseline policy achieve similar results in both scenarios, but systematically inferior to GROWTh.

An interesting conclusion from this first evaluation is that our proposal not only enhances utility – defined as a metric of user throughput (see Section III) – but also reduces user rejection rates. Unlike the Baseline and Random policies, which cannot treat user rejection as a valid action, GROWTh algorithms have the possibility to reject users. Although this presents a challenge when comparing rejection rates, our algorithm intelligently selects which users to accept, resulting in fewer overall rejections. In particular, this improvement is achieved without specific reward tuning for this goal; instead, it comes from a more effective allocation policy that creates additional capacity for new users.

B. *The Paris Scenario*

Next, we consider an evaluation scenario characterized by high user density and traffic load, which can significantly degrade system performance, particularly during large gatherings. To illustrate this, we focus on Paris, the host city for the 2024 Olympic Games, as a case study for managing user association in crowded environments. We utilize the 5G deployment data from a selected operator[1] to determine the positions of the base stations, as shown in Figure 5, and we distribute users following a gaussian distribution around the center of

[1]Data was sourced from nperf.com, which provided information on the operator's 5G deployment, last visited on December 2024.

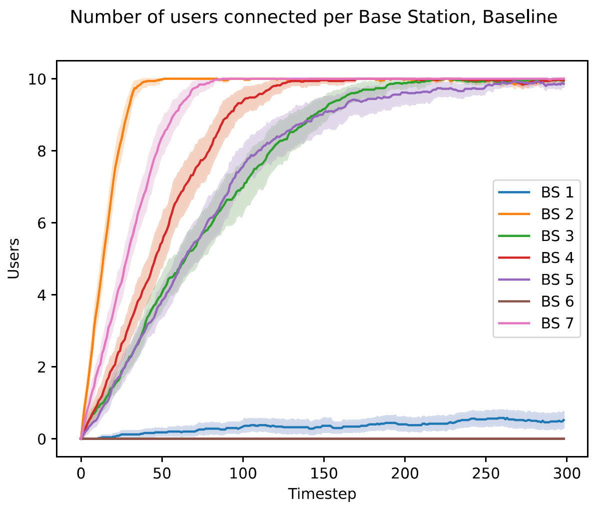


Figure 6. Users connected per base station under the Baseline policy. It is evident that this algorithm prioritizes base stations with stronger SINR, filling them sequentially. In extreme cases, the resources of the farthest base stations remain entirely unused.

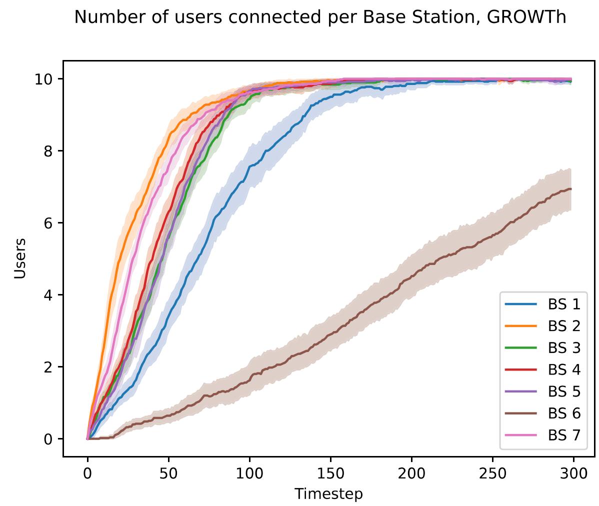


Figure 7. Users connected per base station under the GROWTh policy. In this scenario, all base stations actively serve users, reflecting a more evenly distributed approach that optimizes resource utilization across the entire network.

the selected area to simulate a crowd gathering. As in other simulations, users arrive at random times and with randomized demands. The seven different base stations are individually identified by an ID and a color, matching the results depicted in Figures 6, 7, 10, and 11. The results of this analysis are summarized in Table I, in the columns marked as **Paris**.

As expected, the baseline approach in Figure 6 leads to saturation of the base station capacities, starting with the most centrally located stations and then filling them sequentially. In contrast, Figure 7 illustrates how the base stations are populated with connected users throughout an episode, effectively distributing them among all available candidates.

The users' connection to each base station during an episode is presented for the baseline and GROWTh on figures 8 and 9. Our proposal is able to use the farthest base stations when the closer ones are saturated, meaning our problem formulation is able to address both objectives: to optimize fair rate utility and to minimize rejections, whereas the baseline is very similar to a "closest base station" heuristic.

As we analyze the delivered throughput for each base station in Figures 10 and 11, the advantages of the GROWTh policy become evident. Unlike the Baseline approach, which tends to saturate the nearest base stations – typically those positioned centrally with higher SINR ratio – the GROWTh policy ensures that all base stations are actively utilized. This broader engagement not only enhances overall system performance but also

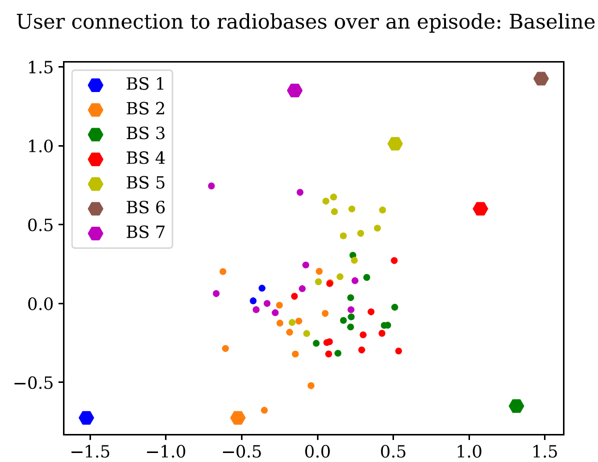


Figure 8. For the Paris experiment, distribution of users connected to each base station under the Baseline policy. The shape is closer to a "closest neighbor" policy, with some differences due to random fading and base stations saturation.

mitigates the risk of congestion in high-demand areas. By distributing user connections more evenly across the network, the GROWTh policy maximizes throughput efficiency and better accommodates varying user densities.

Similar to the previous experiment, the GROWTh policy not only achieves higher utility, but also results in fewer user rejections compared to all alternative approaches – cf. Table I. It is important to note that while our algorithm can explicitly take the 'rejection' action, the baseline and random policies do not have this capability, which could skew the comparison of rejection rates in their favor. This discrepancy is par-

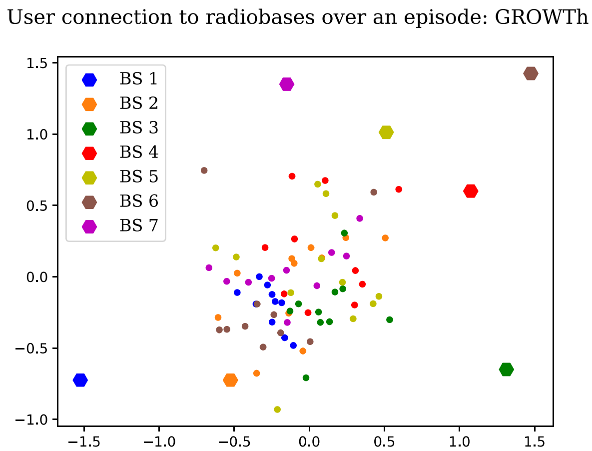


Figure 9. For the Paris setting, distribution of users connected to each base station under the GROWTh policy. The proposed policy is able to redirect incoming users to farther base stations when close ones are already serving a number of connections.

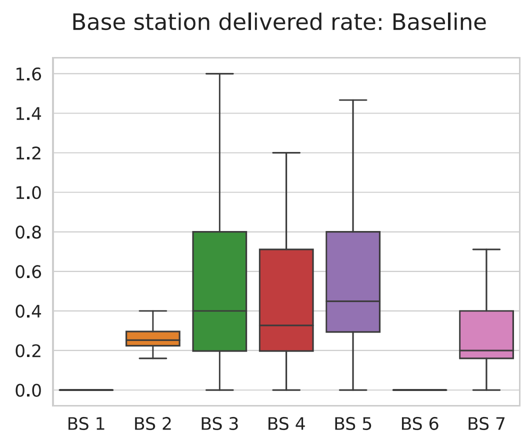


Figure 10. For the Paris experiment, we compare the rates delivered per base station. In this case, the baseline's saturation effect makes it so that mainly BS 3-5 are used, achieving higher rates for those but leaving unused resources in the rest of BS.

ticularly evident when users receive strong signals from all base stations; in such cases, a random policy tends to distribute connections uniformly across the available stations. While this uniformity may lead to a lower rejection rate, it often results in sub-optimal utility due to poor resource allocation.

VI. Concluding Remarks

We presented GROWTh, a distributed algorithm for user association in wireless networks. We comprehensively defined the essential components of a reinforcement learning agent and developed a graph representation that effectively captures the nuances of our problem. By leveraging a graph neural network, we exploited the

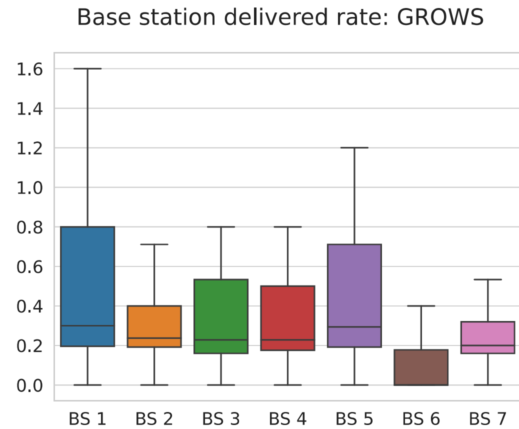


Figure 11. For the Paris experiment, delivered rates per Base Station. We can see that in contrast to the baseline, GROWTh is able to distribute resource through the whole system. Notice that for the closest BS (3-5), GROWTh delivered rates are slightly lower than for the baseline, but it still manages to achieve higher rewards through distribution of users and better usage of resources.

underlying graph structure to enhance decision-making in user association.

Our experimental results validated the permutation equivariance property of the graph neural network, demonstrating that our approach offers a significant advantage in terms of user throughput as compared to existing algorithms, including commonly used heuristics. Additionally, our algorithm is capable of serving more users without requiring explicit reward tuning to minimize rejections. Moreover, the graph representation provides a pathway for generalizing to unseen scenarios, which we have yet to explore.

This capability represents a promising direction for future work, in which we wish to demonstrate GROWTh potential for generalization. There are also pending modifications to the reward structure to account for fairness in the resource distribution – i.e., reward engineering, to include handover scenarios – e.g., by using time-centered neural networks as LSTM or Transformer, or to minimize energy consumption by adding restrictions to our problem's formulation (as in constrained reinforcement learning).

Acknowledgment

The AI-driven modeling presented in this work has been partially supported by the Austrian FFG *AI4SIMPROD* project – AI-Assisted Simulation and Digital Twining for Efficient Industrial Production – project ID 909824.

References

[1] Xinlei Yang, Hao Lin, Zhenhua Li, Feng Qian, Xingyao Li, Zhiming He, Xudong Wu, Xianlong Wang, Yunhao Liu, Zhi Liao, Daqiang Hu, and Tianyin Xu. Mobile Access Bandwidth in Practice: Measurement, Analysis, and Implications. In *Proc. ACM SIGCOMM Conf.*, 2022. ACM.

[2] Ahmad Hassan, Arvind Narayanan, Anlan Zhang, Wei Ye, Ruiyang Zhu, Shuowei Jin, Jason Carpenter, Z. Morley Mao, Feng Qian, and Zhi-Li Zhang. Vivisecting Mobility Management in 5G Cellular Networks. In *Proc. ACM SIGCOMM Conf.*, 2022. ACM.

[3] Xinjie Yuan, Mingzhou Wu, Zhi Wang, Yifei Zhu, Ming Ma, Junjian Guo, Zhi-Li Zhang, and Wenwu Zhu. Understanding 5G Performance for Real-World Services: A Content Provider's Perspective. In *Proc. ACM SIGCOMM Conf.*, 2022. ACM.

[4] Hawar Ramazanali, Agapi Mesodiakaki, Alexey Vinel, and Christos Verikoukis. Survey of User Association in 5G HetNets. In *8th IEEE Latin-American Conference on Communications (LATINCOM)*, pages 1–6. IEEE, 2016.

[5] Xiaohu Ge, Hui Cheng, Mohsen Guizani, and Tao Han. 5G Wireless Backhaul Networks: Challenges and Research Advances. *IEEE Network*, 28(6):6–11, 2014.

[6] Hongseok Kim, Gustavo De Veciana, Xiangying Yang, and Muthaiah Venkatachalam. Distributed α-Optimal User Association and Cell Load Balancing in Wireless Networks. *IEEE/ACM Trans. on Net.*, 20(1):177–190, 2011.

[7] Qiaoyang Ye, Beiyu Rong, Yudong Chen, Mazin Al-Shalash, Constantine Caramanis, and Jeffrey G Andrews. User Association for Load Balancing in Heterogeneous Cellular Networks. *IEEE Trans. on Wireless Comm.*, 12(6):2706–2716, 2013.

[8] Mark Eisen and Alejandro Ribeiro. Optimal Wireless Resource Allocation with Random Edge Graph Neural Networks. *IEEE Trans. on Signal Processing*, 68:2977–2991, 2020.

[9] Daosen Zhai, Huan Li, Xiao Tang, Ruonan Zhang, and Haotong Cao. Joint Position Optimization, User Association, and Resource Allocation for Load Balancing in UAV-Assisted Wireless Networks. *Digital Comm. and Networks*, 10(1):25–37, 2024.

[10] Asad Mahmood, Thang Xuan Vu, Symeon Chatzinotas, and Björn Ottersten. Joint Optimization of 3D Placement and Radio Resource Allocation for Per-UAV Sum Rate Maximization. *IEEE Trans. on Vehicular Tech.*, 72(10):13094–13105, 2023.

[11] Mohammad Alnakhli, Ehab Mahmoud Mohamed, Wazie M Abdulkawi, and Sherief Hashima. Joint User Association and Power Control in UAV Network: A Graph Theoretic Approach. *Electronics*, 13(4):779, 2024.

[12] Qiaoyang Ye, Beiyu Rong, Yudong Chen, Mazin Al-Shalash, Constantine Caramanis, and Jeffrey G. Andrews. User Association for Load Balancing in Heterogeneous Cellular Networks. *IEEE Trans. on Wireless Comm.*, 12(6):2706–2716, 2013.

[13] Enric Pardo, Juan Carlos Estrada-Jimenez, and Sébastien Faye. EMF-Aware User Association Optimization in 5G Networks. *IEEE Access*, 12:15946–15956, 2024.

[14] Nabila Sehito, Yang Shouyi, Haya Mesfer Alshahrani, Mohammad Alamgeer, Ashit Kumar Dutta, Shtwai Alsubai, Lewis Nkenyereye, and Rajesh Kumar Dhanaraj. Optimizing User Association, Power Control and Beamforming for 6G Multi-IRS Multi-UAV NOMA Communications in Smart Cities. *IEEE Trans. on Consumer Electronics*, 2024.

[15] Nan Zhao, Ying-Chang Liang, Dusit Niyato, Yiyang Pei, Minghu Wu, and Yunhao Jiang. Deep Reinforcement Learning for User Association and Resource Allocation in Heterogeneous Cellular Networks. *IEEE Trans. on Wireless Communications*, 18(11):5141–5152, 2019.

[16] Ali Nauman, Haya Mesfer Alshahrani, Nadhem Nemri, Kamal M Othman, Nojood O Aljehane, Mashael Maashi, Ashit Kumar Dutta, Mohammed Assiri, and Wali Ullah Khan. Dynamic Resource Management in Integrated NOMA Terrestrial–Satellite Networks Using Multi-Agent Reinforcement Learning. *Journal of Network and Computer Applications*, 221:103770, 2024.

[17] Yifei Shen, Yuanming Shi, Jun Zhang, and Khaled B Letaief. A Graph Neural Network Approach for Scalable Wireless Power Control. In *IEEE GLOBECOM Workshops*, 2019. IEEE.

[18] Yifei Shen, Yuanming Shi, Jun Zhang, and Khaled B Letaief. Graph Neural Networks for Scalable Radio Resource Management: Architecture Design and Theoretical Analysis. *IEEE Journal on Selected Areas in Comm.*, 39(1):101–115, 2020.

[19] Ziyan He, Liang Wang, Hao Ye, Geoffrey Ye Li, and Biing-Hwang Fred Juang. Resource Allocation Based on Graph Neural Networks in Vehicular Communications. In *IEEE Global Communications Conference*, 2020. IEEE.

[20] Guangming Cui, Qiang He, Xiaoyu Xia, Feifei Chen, Fang Dong, Hai Jin, and Yun Yang. OL-EUA: Online User Allocation for NOMA-Based Mobile Edge Computing. *IEEE Trans. on Mobile Computing*, 22(4):2295–2306, 2021.

[21] Mohammed Jaber Alam, Ritesh Chugh, Salahuddin Azad, and Md Rahat Hossain. Optimizing Cell Association in 5G and Beyond Networks: A Modified Load-Aware Biased Technique. *Telecommunication Systems*, 87(3):731–742, 2024.

[22] Pengcheng He, Yijia Tang, Fan Xu, and Qingjiang Shi. Optimization-Inspired Graph Neural Network for Cellular Network Optimization. *IEEE Trans. on Mobile Computing*, 2025.

[23] Navid Naderializadeh, Jaroslaw J Sydir, Meryem Simsek, and Hosein Nikopour. Resource Management in Wireless Networks via Multi-Agent Deep Reinforcement Learning. *IEEE Trans. on Wireless Communications*, 20(6):3507–3523, 2021.

[24] Zizhen Zhou, Jungang Ge, and Ying-Chang Liang. User Association and Coordinated Beamforming in Cognitive Aerial-Terrestrial Networks: A Safe Reinforcement Learning Approach. *arXiv Preprint arXiv:2502.13663*, 2025.

[25] Shaofeng Dong, Jinsong Zhan, Wei Hu, Amin Mohajer, Maryam Bavaghar, and Abbas Mirzaei. Energy-Efficient Hierarchical Resource Allocation in Uplink–Downlink Decoupled NOMA HetNets. *IEEE Trans. on Network and Service Management*, 20(3):3380–3395, 2023.

[26] Martín Randall, Pablo Belzarena, Federico Larroca, and Pedro Casas. GROWS: Improving Decentralized Resource Allocation in Wireless Networks Through Graph Neural Networks. In *Proceedings of the 1st Int. Workshop on Graph Neural Networking*, pages 24–29, 2022.

[27] Martín Randall, Pablo Belzarena, Federico Larroca, and Pedro Casas Deep Reinforcement Learning and Graph Neural Networks for Efficient Resource Allocation in 5G Networks. In *IEEE Latin-American Conference on Communications (LATINCOM)*, pages 1–6. IEEE, 2022.

[28] Arjun Anand, Gustavo De Veciana, and Sanjay Shakkottai. Joint Scheduling of URLLC and eMBB Traffic in 5G Wireless Networks. *IEEE/ACM Trans. on Net.*, 28(2):477–490, 2020.

[29] Thomas N. Kipf and Max Welling. Semi-Supervised Classification With Graph Convolutional Networks. *5th International Conference on Learning Representations (ICLR-17)*, 2017.

[30] Elvin Isufi, Fernando Gama, and Alejandro Ribeiro. EdgeNets: Edge Varying Graph Neural Networks. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, pages 1–1, 2021.