Optimizing Throughput in Wi-Fi enabled UAV Network using Multi-agent Reinforcement Learning

Parshav Pagaria, Santosh Nagraj, Mahasweta Sarkar
Department of Electrical and Computer Engineering
San Diego State University
San Diego, USA
Emails:{ppagaria2278, snagaraj, msarkar2}@sdsu.edu

Dhruv Mishra, Souvik Deb, Shankar K. Ghosh*
Department of Computer Science and Engineering
Shiv Nadar Institution of Eminence
Delhi NCR, India
Emails:{dhruv27mishra, deb.souvik5, shankar.it46}@gmail.com

***Abstract*—Positioning of Wi-Fi equipped Unmanned Aerial Vehicle (UAV) and subsequent user association remains a complex problem due to unknown pattern of interference caused by UAV movements, user mobility and quality of service (QoS) constraints. Conventional deterministic approaches are known to under-perform in such non-stationary scenarios. In this manuscript, we proposed a decentralized multi-agent reinforcement learning (MARL) framework to *jointly* address the UAV positioning and user association problem for Wi-Fi enabled UAV network. The proposed MARL framework considers UAV collision, frequency of handover and number of frames successfully transmitted by UAVs. Based on the framework, three existing MARL approaches namely Value Decomposition Networks (VDN), QMIX and DeepNashQ have been evaluated through simulation. The MARL approaches have been compared with two existing deterministic algorithms as well. Results show the superiority of MARL algorithms over the deterministic counterparts, towards maximizing system throughput and fairness. Additionally, QMIX has been found to be the best MARL algorithm in this context.**

***Index Terms*—Multi-Agent Reinforcement Learning, UAV positioning, Wireless Fidelity, Throughput optimization.**

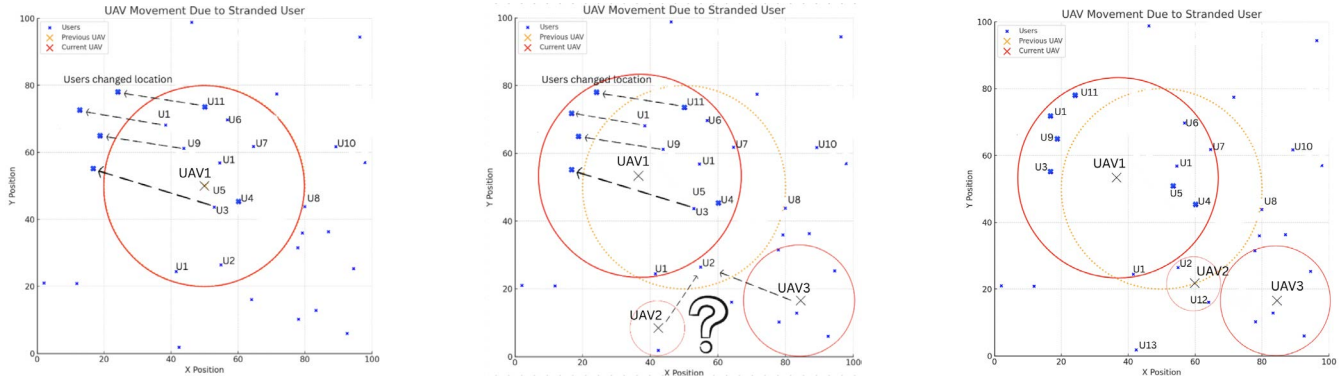
I. Introduction

Despite significant advances in wireless communication, large portions of the world's rural, remote, and sparsely populated regions continue to suffer from inadequate internet connectivity, because traditional cellular infrastructure deployment in those areas is often economically infeasible [1]. This motivates the use of Unmanned Aerial Vehicles (UAVs) as a mobile, cost-effective communication platform [2]. By equipping UAVs with IEEE 802.11ax Wireless Fidelity (Wi-Fi) modules, the aerial platforms can dynamically provide on-demand coverage without the logistical burdens of static infrastructure [3]. In such a Wi-Fi enabled UAV network, optimal positioning of UAVs and subsequent association with user equipments (UEs) are extremely important, in order to meet the high data rate requirements of enhanced mobile broadband (eMBB) applications. **Most of the existing studies [4]–[7] utilize UAV as relay to enhance coverage in cellular network; and the possibility of exploiting UAV as a mobile Wi-Fi platform is often overlooked.**

*All of the authors are equally contributing. Souvik Deb is currently associated with the School of Computer Science, University of Petroleum and Energy Studies, Dehradun, India.

Optimizing UAV positioning and subsequent UE association remain a complex problem due to UE mobility and stringent data rate requirements of eMBB services. Here, the goal is to maximize the system throughput. It may be noted that UAV positioning is a *recurrent* and *continuous* control problem. To illustrate, let us consider the scenario depicted in Fig. 1. Initially, UEs U1, U2, U3, U4, U5, U6, U7, U9 and U11 are residing within the coverage of UAV 1. At time t_1, four UEs (U1, U11, U9 and U3) move out of coverage of UAV 1. Subsequently, at time t_2, the UAV 1 repositions to regain coverage over those UEs. However, this shift leaves U2 uncovered. At this point, two UAVs (UAV 2 and UAV 3) are available to cover the stranded UEs: one with high coverage (UAV 3) and another with low coverage (UAV 2). It may be noted that the movement of the Wi-Fi equipped UAVs may cause interference to other UAV signals. In case UAV 3 moves in to cover U2, the coverage regions of UAV 1 and UAV 3 would overlap, resulting in interference to U1. Therefore, at time t_3, UAV 3 is moved to cover U2. In such a way, UAV positions should be updated using prior positional information of the UEs with the aim to mitigate inter-UAV interference and boost system throughput.

Existing geometric approaches for UAV positioning such as hexagonal tiling or grid-based layouts [6], [7] are static, and often assume full knowledge of user positions. Such approaches fail to adapt with mobility and interference. In [6], minimum number of stop points to completely cover the region of interest is computed assuming that full knowledge of the network scenario is available. In [7], neural network has been employed to find optimal UAV-UE association. Therein, the authors assume perfect user position and terrain knowledge. Key deficiencies of these approaches [6], [7] are the inabilities to handle UE mobility and inter-UAV interference. Such approach quickly become suboptimal in non-stationary scenarios. To overcome these limitations, data-driven and learning-based approaches such as [8] and [9] have gained traction. In particular, Reinforcement Learning (RL) has shown effectiveness in dynamically controlling UAV trajectories and UE association without prior knowledge of UE positions. However, most of the RL approaches adopt a single-agent perspective, which do not scale to UAV swarms where multiple UAVs must coordinate among themselves in a *decentralized* and



(a) Time t_1: Before user movement. (b) Time t_2: After user movement. (c) Time t_3: UAV positioned.

Fig. 1: UAV positioning at different time steps to maintain user coverage

partially observable environment [10]. This motivates the use of Multi-Agent Reinforcement Learning (MARL) algorithms, which enable decentralized agents to learn policies through interactions among themselves and with the environment [10]–[12]. However, these independent MARL methods suffer from nonstationary, where policy updates by one agent destabilize learning for others. To address this, value factorization methods such as Value Decomposition Networks (VDN) [13] and QMIX [14] were proposed. The VDN decompose the global Q-function into a sum of individual Q-values for each agent, thus supporting centralized training and decentralized execution (CTDE). However, its linearity constraint limits its expressiveness. On the other hand, QMIX extends the VDN with a non-linear, monotonic mixing network that preserves decentralization, while capturing more complex inter-agent dependencies. In wireless environments, UAVs often operate in a shared spectrum and may experience overlapping coverage or interference, leading to situations where the individual objectives are not fully fulfilled. To handle such partially competitive dynamics, game-theoretic MARL approaches such as Deep Nash Q-learning (DeepNashQ) are also particularly relevant [15]. This approach compute equilibrium strategies instead of greedy ones, enabling more principled coordination. Despite the promise of MARL, prior research rarely use MARL to address the non-stationarity in UAV positioning and subsequent UE association problem.

In this work, our *objective* is to leverage MARL framework to deal with UAV positioning and subsequent UE association in IEEE 802.11ax enabled UAV networks. The proposed framework accounts signal-to-interference plus noise ratio (SINR) feedback from UEs, eliminating the need for precise user location. Based on the framework, we evaluate two value-based approaches (i.e., VDN and QMIX) and one game-theoretic namely DeepNashQ. Our *contributions* are summarized below:

- We first formulate the UAV positioning and user association task as a constrained non-separable non-linear integer programming problem. Therein, the objective is to maximize the system throughput. Since, the aforesaid optimization problem is NP-complete [16], solving the problem at each time step is time consuming and impractical. Hence, to deal with the non-stationarity of the problem, we reformulate it using MARL framework, enabling decentralized and adaptive decision-making.
- We explore value-based MARL approaches, including VDN and QMIX, which allow us to decompose the joint action-value function across agents while maintaining cooperative behavior. In addition to value decomposition methods, we also evaluate DeepNashQ, a decentralized MARL algorithm designed for general-sum stochastic games.
- Through extensive simulations, we evaluate the performance of DeepNashQ, VDN, and QMIX in terms of *system throughput*, *lower bound on UE throughput* (LBT) and *goodness*. Here, system throughput is defined as the mean amount of data transmitted to all UEs per unit time. The LBT metric indicates fairness. The goodness metric is defined as the fraction of UEs achieving their requested data rate. Results show that learning-based strategies outperform the deterministic baselines [6], [7]. We further conclude that QMIX performs the best among the other considered MARL approaches.

In the next section, we describe the considered system model.

II. System Model

We consider N UEs, indexed by $j \in \mathcal{N} = \{1, 2, \ldots, N\}$, are distributed within a discrete spatial domain defined as a 100×100 square meters area. This region is partitioned into uniform two-dimensional grids, by discretizing the rows and columns into sides of 1 meter. A fleet of M UAVs provide wireless connectivity to the UEs via IEEE 802.11ax-based Wi-Fi modules mounted on the UAVs. These modules support high throughput through 256-QAM modulation, multiple input and multiple output (MIMO) antennas, and channel bonding up to 160 MHz in the 5 GHz band [17]. These Wi-Fi module use Carrier Sense Multiple Access with collision avoidance (CSMA/CA) for medium access control. We assume that these Wi-Fi modules are tethered to ground power sources,

thus eliminating energy constraints. The UAVs hover at fixed altitudes of either 10 or 15 meters, and are repositioned periodically to optimize network performance. Each UAV is positioned at the corner points of the 1 square meter grids. The UAV moves by 1 meter, in any one of the four directions, i.e., north, south, east and west. Time is divided into discrete slots of duration δ. At time slot t, UAV i has 3D position coordinates $u_i(t) = (x_i(t), y_i(t), z_i(t))$. Here $x(t), y(t) \in \{0, 1, \ldots, 99\}$, and $z(t) \in \{10,\ 15\}$.

A. Channel Model

Link quality is modeled using SINR, which accounts for both distance-based path loss and inter-UAV interference caused by CSMA/CA mechanism [18]. Now, the received power at UE j from UAV i is computed as:

$$P_{ij}(t) = \frac{P_t \cdot |h_{ij}|^2}{[d_{ij}(t)]^\alpha}, \tag{1}$$

where P_t is the fixed transmit power of each UAV, $|h_{ij}|$ is the Rayleigh fading gain, α is the path loss exponent and $d_{ij}(t)$ is the Euclidean distance between UAV i and UE j. Here $d_{ij}(t)$ is computed as: $d_{ij}(t) = \sqrt{(x_i(t) - x_j^u(t))^2 + (y_i(t) - y_j^u(t))^2 + z_i^2(t)}$. Accordingly, the SINR experienced by UE j from UAV i at time t is computed as:

$$\gamma_{ij}(t) = \frac{P_{ij}}{\sigma^2 + \sum\limits_{k \neq i}^{M} P_{kj}}. \tag{2}$$

To model $\tau_i(t)$, the downlink throughput (in frames per second) from UAV i in time slot t, we adopt the Bianchi's framework which is widely used in this context [19], [20]. Using this framework, $\mathbb{P}_{ij}(t)$, the probability of failure in frame transmission is computed as [20]:

$$\mathbb{P}_{ij}(t) = 1 - \prod_{k \neq i}^{M} \frac{1}{1 + \gamma_{req} e^{\mathbf{X}_i - \mathbf{X}_k} [\frac{d_{kj}(t)}{d_{ij}(t)}]^\alpha}. \tag{3}$$

where γ_{req} is the minimum required SINR threshold to successfully decode a frame at the UE, $\mathbf{X}_i$ is a Gaussian r.v. with zero mean and variance β^2 representing log-normal shadowing. Accordingly, $\mathbb{P}_i(t)$, the probability that transmission by all the UEs are unsuccessful under the coverage of UAV i is computed as:

$$\mathbb{P}_i(t) = \prod_{j \in \aleph_i} \mathbb{P}_{ij}(t). \tag{4}$$

Here, $\aleph_i = \{j \in \mathcal{N} |\ j$ is covered by UAV $i\}$. We use the M/M/1/K queue to model the buffer of the Wi-Fi access points mounted on the UAVs. The buffer can hold at most K frames. Subsequently, $\mathbb{P}_{Bi}(t)$, the probability that the buffer at UAV i is full in time slot t, is computed as follows [19]:

$$\mathbb{P}_{Bi}(t) = \frac{\left(\frac{\lambda_i(t)}{\mu_{\text{MAC},i}}\right)^K}{\sum_{j=0}^{K} \left(\frac{\lambda_i(t)}{\mu_{\text{MAC},i}}\right)^j}. \tag{5}$$

Here, $\lambda_i(t)$ is the arrival rate of frame at the buffer of UAV i at time slot t, and $\mu_{\text{MAC},i}$ is the expected time to process a frame in the MAC layer. Based on equations (4) and (5), $\tau_i(t)$ is computed as follows:

$$\tau_i(t) = \sum_{j \in \mathcal{I}_i} \tau_{ij} = \sum_{j \in \mathcal{I}_i} \lambda_{ij}(t) \left(1 - \mathbb{P}_{Bi}(t)\right) \left(1 - \mathbb{P}_i^{\nu+1}(t)\right). \tag{6}$$

Here, τ_{ij} is the downlink throughput from UAV i to UE j, $\lambda_{ij}(t)$ is the independent arrival rate of frames intended for UE j at the buffer of UAV i, $\mathcal{I}_i$ is the set of UEs being served by UAV i and ν is the maximum allowed number of retransmissions. Based on this system model, in the next section, we formulate the UAV positioning and UE association problem using MARL framework.

III. Problem formulation

In this section, we formulate the UAV positioning and subsequent UE association problem as a non-linear non-separable integer programming problem. Here the objective is to jointly optimize UAV positions and UE associations, aiming to maximize the system throughput while satisfying data rate requirements. In this formulation, $x_i(t)$, $y_i(t)$, $z_i(t)$, $c_{ij}(t)$ and $I_{ij}(t)$ are decision variables, where $i \in \{1, \ldots, M\}$ and $j \in \{1, \ldots, N\}$. Here, $(x_i(t), y_i(t))$ is the horizontal coordinate of UAV i, whereas $z_i(t) \in \{10, 15\}$ denotes the altitude of UAV i. The variable $c_{ij}(t) = 1$ if UE j is served by UAV i at time slot t, and 0 otherwise. The variable $I_{ij}(t)$ represents whether UE j switches from UAV i to any other UAV between consecutive time slots, and is defined in terms of $c_{ij}(t)$ as follows:

$$I_{ij}(t) = \begin{cases} 1, & \text{if } c_{ij}(t) \neq c_{ij}(t-1), \\ 0, & \text{otherwise.} \end{cases} \quad \forall i, j, t, \tag{7}$$

The objective is to maximize the system throughput, which can be expressed mathematically as follows:

$$\max \sum_{i=1}^{M} \sum_{j=1}^{N} c_{ij}(t) \lambda_{ij}(t) \times (1 - \mathbb{P}_{Bi}(t)) \times \left(1 - \mathbb{P}_i^{\nu+1}(t)\right), \tag{8}$$

It may be noted that $\mathbb{P}_{Bi}(t)$ and $\mathbb{P}_i^{\nu+1}(t)$ are functions of $d_{ij}(t)$, which in turn is a function of $x_i(t)$, $y_i(t)$ and $z_i(t)$. Now, the objective function (8) need to be maximized subjected to the following constraints:

$$\sum_{i=1}^{M} c_{ij}(t) = 1, \quad \forall j \tag{9}$$

$$\tau_{ij}(t) \geq r_{\min}, \quad \forall j \tag{10}$$

$$\sum_{t'=t-w}^{t} I_{ij}(t') \leq x, \quad \forall i, j \tag{11}$$

Here, constraint (9) ensures that each UE must be connected to exactly one UAV in every time slot. The constraint (10) ensures that each UE must receive the requested data rate. Here $\tau_{ij} = c_{ij}(t) \lambda_{ij}(t) \times (1 - \mathbb{P}_{Bi}(t)) \times \left(1 - \mathbb{P}_i^{\nu+1}(t)\right)$. Finally, constraint (11) ensure that the number of handovers for each

UE is restricted within a sliding window of length w. The parameter w controls the memory depth for recent handovers. A smaller w allows frequent handovers, whereas a larger w may restrict necessary handovers. The optimal value of w changes over time depending on varying channel conditions and UE positions. It is to be noted that, for a fixed w itself, the resulting optimization problem is a non-separable and a non linear integer programming problem, which is known to be NP-complete [16]. *Furthermore, the optimization problem has to be solved at each time step to compute the optimal UAV positions which is computationally intensive, and impractical from implementation perspective.* Given such a prevailing characteristics of the problem, it is worthy to develop a sequential decision making solution that can adaptively change the UAV positions based on adequate memory depth. In such a context, MARL can be employed which enables UAVs to cooperatively and adaptively adjust their positions over time, aiming to maximize the system throughput. The MARL framework is well-suited for decentralized decision-making with the objective to maximize the cumulative reward. In the subsequent section, we reformulate the UAV positioning problem using MARL framework.

A. MARL framework

Each UAV is equipped with an RL agent, operating in a 3D spatial grid. The movement of each UAV is driven by its current state and the action it takes. At each discrete time step, agent at UAV i (say agent i) observes the state of the environment and choose an action A_i. Based on the chosen action, agent i receives an individual reward r_i. The objective of all the agents in a cooperative MARL system is to learn the optimal policy such that the cumulative reward $(\sum_i r_i)$ for all agents is maximized. In this work, we translate the optimization problem presented in Section III to a MARL framework which aims to maximize the system throughput.

State space

The state space $U(t)$ at time t is the concatenation of all UAV positions at time $t-1$ i.e UAV positions at the previous time step:

$$U(t) = \langle u_1(t-1), u_2(t-1), \ldots, u_N(t-1) \rangle.$$

The system is partially observable (to each UAV), limited to itself and its neighboring UAVs.

Action Space

Each UAV is only allowed to observe and coordinate with its immediate neighbors in the 3D spatial grid. UAV j is considered to be a neighbor of UAV i, if it is located within two grid blocks in any direction.

The action space for agent i consists of 2-tuples of the form $A_i = (a_i(t), w)$. Here, $a_i(t)$ represents the movement OF UAV i, i.e., `North`, `South`, `East`, `West` and `Stay` (i.e., no change); whereas $w \in \{1, \cdots, W\}$ represents the sliding window length. Each action corresponds to a position alteration, which is determined by $a_i(t)$ as follows:

$$u_i(t) = u_i(t-1) + a_i(t), \tag{12}$$

where $a_i(t)$ is $(0,0,0)$ for action `Stay`, $(0,+1,0)$ for action `North`, $(0,-1,0)$ for action `South`, $(+1,0,0)$ for action `East` and $(-1,0,0)$ for action `West`. After computing the next state, each UE j is associated with the UAV i such that $i = \max_k \frac{\gamma_{kj}(t) - \gamma_{req}}{\zeta_{kj}(t)}$, where $\zeta_{kj}(t)$ is the duration between the current time slot t and the time slot when UAV k was assigned to UE j for the last time. Here, $\zeta_{ij}(t)$ is initialised to δ which allows the UEs to associate with UAVs based on SINR in the initial phase.

Reward structure

The reward function accounts UAV collision, frequency of handover and throughput. The individual reward of agent i at time t is defined as:

$$r_i(t) = \frac{\frac{1}{wM} \sum_{k=0}^{w-1} \sum_{j=1}^{N} c_{ij}(t-k)\Phi_{ij}(t-k)}{1 + \frac{1}{w} \sum_{k=0}^{w-1} \Psi_i(t-k)}, \tag{13}$$

when UAV i does not face any physical collision. Otherwise, $r_i(t) = 0$. Here, $\Phi_{ij}(t)$ is defined as the ratio of successfully transmitted frames to the total number of attempted transmissions and $\Psi_i(t)$ is the number of UE switches experienced by UAV i. We set $c_{ij}(t) = \Phi_{ij}(t) = \Psi_i(t) = 0$ when t is negative. Accordingly, the global reward $R(t)$ at time t is computed as the sum of all individual UAV rewards as follows:

$$R(t) = \sum_{i=1}^{M} r_i(t). \tag{14}$$

It may be noted that the reward function encourages the agents to maximize their throughput via favorable positioning while avoiding physical collision. Moreover, it drives the agent to learn the optimal value of w for which average number of frame transmissions is maximized while avoiding unnecessary handovers.

IV. Results and Discussions

In this section, we evaluate the performances of three MARL algorithms namely VDN, QMIX and DeepNashQ-based on the MARL framework proposed in Section III-A. In VDN, the global Q-function is additively decomposed into per-agent Q-values, allowing each agent to learn independently from local rewards. In QMIX, a monotonic mixing network is trained to combine individual Q-values into a global Q-value, thus supporting CTDE mechanism. Accordingly, all the UAVs have access to global state and shared rewards. Each UAV stores its experience in a replay buffer and updates its Q-network via backpropagation. The agents rely solely on their local observations and learned policies, ensuring scalability

and robustness. In DeepNashQ, each agent computes equilibrium strategies over joint actions using a local Q-network, enabling coordination via game-theoretic reasoning.

The aforementioned MARL approaches have been compared with two deterministic baselines [6], [21]. In [6] (say Deterministic I), fixed UAV placement has been considered, whereas UE association has been done based on SINR. In [21] (say Deterministic II), the UAV positioning problem has been considered as a geometric problem aiming to maximize the coverage using minimum number of circles. Both Deterministic I and Deterministic II approaches are snapshot-based which do not account UE mobility, varying channel condition due to fading and inter-UAV interference. We consider system throughput, LBT and goodness as performance evaluation metrics. In the next subsection, we describe the simulation set-up.

A. Simulation setup

TABLE I: MARL Parameters

Parameter	Value	Parameter	Value
Learning rate	0.001	Discount factor	0.99
Exploration rate	0.1	Replay buffer size	10,000
Batch size	64	Update frequency	100 steps
Collision penalty	-100.0	Boundary violation	-50.0
Altitude violation	-50.0	Handover penalty	2.0

We have developed a simulator using Python version 3.8[1]. We consider a 3D grid environment of size $10 \times 10 \times 5$ cubic meters, with UAVs constrained to maintain altitudes between 10 and 15 meters [6]. The UEs are moving according to a random way-point mobility model with UE velocity fixed to 1 m/s. UEs are distributed according to homogeneous Poisson Point Process (PPP), where the parameter is the number of UEs per square meter. The duration of a time slot (δ) has been set to 25.39ms, which is the channel coherence time [22]. We set $W = 5$. The UAV transmit power is set to 1.0 Watt, while the noise power is set to 10^{-6} Watts. The frame arrival rate is set at 4000 frames/second. The frame size is 1500 bytes [23]. We assume Raleigh fading channels with unit mean where the path loss exponent is set to 2.5. The total simulation time spans 10000 training episodes, with each episode running until termination conditions are met. The considered termination conditions are: collision between UAVs and boundary violations by the UAVs. The results have been generated by averaging over 10000 independent training runs, ensuring statistical significance of the performance metrics. The system employs a bandwidth of 1 Mbps, where the peak data rate requirement by a UE is 10 Mbps. The MARL specific parameters are depicted in Table I. The MOE measurements with 95% confidence interval show the accuracy of the reported results.

B. Performance evaluation

Fig. 2 presents the moving average of rewards and throughput (per episode) for each model, across 1000 training episodes. It is observed that QMIX converges faster and achieves the highest overall reward, showing more effective agent cooperation. DeepNashQ and VDN demonstrate performance with slight variations in stability, while DeepNashQ remains consistently lower due to its lack of learning adaptability. The reward curves further state that RL agents improve over time through exploration.

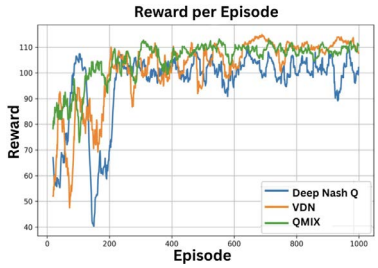
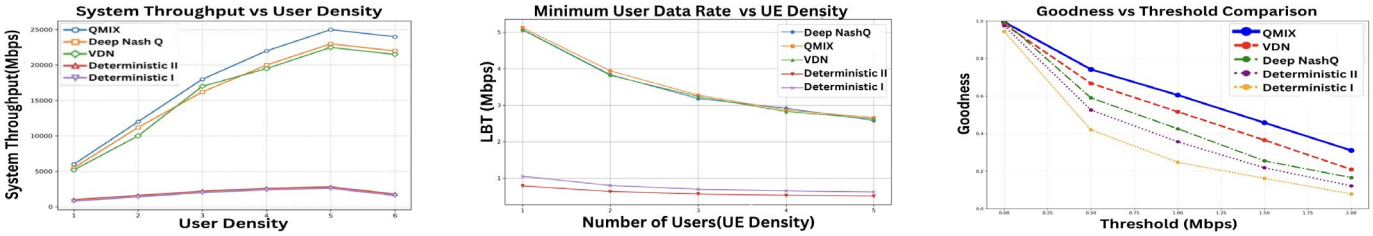


Fig. 2: Reward per episode across all models

Fig. 3a illustrates the variation in system throughput across different user densities for all models. The RL-based models (QMIX, DeepNashQ, and VDN) consistently outperform the deterministic models, especially as user density increases. Among them, QMIX achieves the highest system throughput, showing better resource coordination under high user density. In contrast, both deterministic models descend early, showing limited adaptability to growing user demand. This highlights the advantage of learning-based policies in handling complex association and repositioning situations.

Fig. 3b illustrates the LBT across varying UE densities for all models. As the number of UEs increases, all models experience a decrease in LBT due to increased resource usage. Among learning-based approaches, QMIX consistently achieves the highest LBT, which is closely followed by Deep NashQ and VDN, indicating the effectiveness of the coordinated multi-agent strategies in ensuring fairness. In contrast, the Deterministic models exhibit significantly lower LBT, with the Deterministic II model falling below 0.6 Mbps as UE density increases beyond 2. This highlights the advantage of learning-based models in optimizing worst-case user performance. Fig. 3c measures the impact of minimum throughput requirement on goodness. We observe that QMIX consistently outperforms other MARL approaches which is consistent with Fig. 3b. The VDN outperforms NashQ in overall goodness despite showing lower LBT. This stems from NashQ's higher computational complexity with increasing number of UAVs. Despite showing lower LBT, the VDN approach enables a higher fraction of UEs to achieve the required minimum throughput as compared to NashQ. In Fig. 4, we evaluate goodness as a function of minimum required throughput by each UE and the UE densities. The evaluation has been done for QMIX only, because QMIX performs best among other approaches. Here, goodness is defined as the fraction of UEs achieving throughput above the predefined threshold. The result shows that more than 90% of the UEs have throughput above 10 Mbps, underscoring the robustness of

[1]Code available at: github.com/Dhruv27Mishra/UAV-Repositioning



(a) System throughput vs. UE density (MOE $< 9 \times 10^{-3}$)

(b) LBT vs. UE density (MOE $< 9.8 \times 10^{-3}$)

(c) Goodness vs. threshold (MOE $< 9.3 \times 10^{-3}$)

Fig. 3: (a) Average throughput (b) LBT and (c) Goodness vs. threshold.

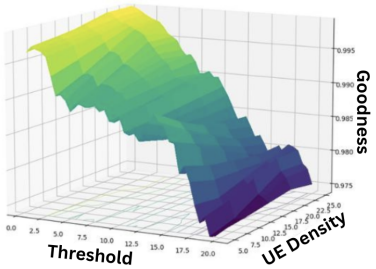


Fig. 4: Goodness as a function of UE density and threshold (MOE $< 9.3 \times 10^{-3}$)

QMIX. Moreover it is observed that the goodness value is consistent across all examined UE densities.

V. Conclusion

In this manuscript, we propose a MARL-based framework for UAV positioning and UE association in Wi-Fi enabled UAV network. Our approach considers UAV collision, frequency of handover and throughput; and demonstrates significant improvements as compared to deterministic baselines. Among the evaluated methods, QMIX showed the best performance in terms of throughput and fairness. In future work, we aim to extend our MARL framework to jointly optimize throughput and latency under the same network scenario.

References

[1] J. Valentín-Sívico, C. Canfield, S. A. Low, and C. Gollnick, "Evaluating the impact of broadband access and internet use in a small underserved rural community," *Telecommunications Policy*, vol. 47, no. 4, p. 102499, 2023.

[2] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "A tutorial on uavs for wireless networks: Applications, challenges, and open problems," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2334–2360, 2019.

[3] F. Pervez, J. Qadir, M. Khalil, T. Yaqoob, U. Ashraf, and S. Younis, "Wireless technologies for emergency response: A comprehensive review and some guidelines," *IEEE Access*, vol. PP, pp. 1–1, 11 2018.

[4] I. Sawad, R. Nilavalan, and H. Al-Raweshidy, "Backhaul in 5g systems for developing countries: A literature review," *IET Communications*, vol. 17, no. 6, pp. 659–669, 2023.

[5] X. Wang, H. Zhang, and Z. Hou, "Performance analysis of multi-satellite hybrid satellite-terrestrial relay networks," *Journal of Earth System Science*, vol. 131, no. 4, p. 217, 2022.

[6] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Unmanned aerial vehicle with underlaid device-to-device communications: Performance and tradeoffs," *IEEE Transactions on Wireless Communications*, vol. 15, no. 6, pp. 3949–3963, 2016.

[7] V. Sharma, S. Choudhury, and A. Banerjee, "Uav-assisted heterogeneous networks for capacity enhancement," in *IEEE International Conference on Communications (ICC)*, 2016, pp. 1–6.

[8] Y. Zeng and R. Zhang, "Energy-efficient uav communication with trajectory optimization," *IEEE Transactions on Wireless Communications*, vol. 18, no. 8, pp. 4570–4586, 2019.

[9] J. Zhang, L. Zhao, Y. Han *et al.*, "Multi-agent deep reinforcement learning for uav trajectory planning with qos guarantees," *IEEE Internet of Things Journal*, vol. 8, no. 17, pp. 13 492–13 504, 2021.

[10] Y. Chu, Y. Mao, and C. You, "Cooperative multi-uav trajectory optimization for wireless coverage via multi-agent reinforcement learning," *IEEE Transactions on Wireless Communications*, vol. 21, no. 12, pp. 10 638–10 653, 2022.

[11] T. D. Nguyen, N. D. Nguyen, and S. Nahavandi, "Deep reinforcement learning for multi-agent systems: A review of challenges, solutions and applications," *IEEE Transactions on Cybernetics*, vol. 50, no. 9, pp. 3826–3839, 2020.

[12] J. Chen, Y. Zou, H. Zhang, and C. Xu, "Cooperative multi-agent reinforcement learning: A survey of theories and applications," *IEEE Transactions on Neural Networks and Learning Systems*, 2022, early Access.

[13] P. Sunehag, G. Lever, A. Gruslys *et al.*, "Value-decomposition networks for cooperative multi-agent reinforcement learning," in *AAMAS*, 2018.

[14] T. Rashid, M. Samvelyan, C. de Witt *et al.*, "Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning," in *ICML*, 2018.

[15] J. Hu and M. P. Wellman, "Nash q-learning for general-sum stochastic games," *Journal of Machine Learning Research*, vol. 4, pp. 1039–1069, 2003.

[16] D. S. Hochbaum, "Complexity and algorithms for nonlinear optimization problems," *Ann. Oper. Res.*, vol. 153, no. 1, pp. 257–296, May 2007.

[17] *IEEE Standard for Information technology–Telecommunications and information exchange between systems–Local and metropolitan area networks–Specific requirements–Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications–Amendment 4: Enhancements for Very High Throughput for Operation in Bands below 6 GHz*, IEEE Std. IEEE Std 802.11ac-2013, 2013.

[18] G. Bianchi, "Performance analysis of the ieee 802.11 distributed coordination function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, 2000.

[19] D. Griffith, M. Souryal, C. Gentile, and N. Golmie, "An integrated phy and mac layer model for half-duplex ieee 802.11 networks," in *2010-MILCOM 2010 MILITARY COMMUNICATIONS CONFERENCE*. IEEE, 2010, pp. 1478–1483.

[20] M. Zorzi and R. R. Rao, "Capture and retransmission control in mobile radio," *IEEE journal on selected areas in communications*, vol. 12, no. 8, pp. 1289–1298, 2002.

[21] S. A. Hasan, L. Sau, and S. C. Ghosh, "Geometry based uav trajectory planning for mixed user traffic in mmwave communication," *Advanced Computing & Microelectronics Unit, Indian Statistical Institute*, 2025.

[22] H. Jung, K. Cho, Y. Choi *et al.*, "React: Rate adaptation using coherence time in 802.11 wlans," *Computer Communications*, vol. 34, no. 11, pp. 1316–1327, 2011.

[23] "Wi-fi 6 technology and evolution white paper," *ZTE Corporation*, 2020.