

RESEARCH ARTICLE

Machine Learning-Based Multi-UAV Deployment for Uplink Traffic Sizing and Offloading in Cellular Networks

AHMED FAHIM MOSTAFA¹, (Graduate Student Member, IEEE),
MOHAMED ABDEL-KADER^{1,2}, YASSER GADALLAH¹, (Senior Member, IEEE),
AND OMAR ELAYAT¹

¹Department of Electronics and Communications Engineering, The American University in Cairo, New Cairo 11835, Egypt

²Department of Communications and Electronics Engineering, Alexandria University, Alexandria 21544, Egypt

Corresponding author: Ahmed Fahim Mostafa (ahmedfahim@ieee.org)

This work was supported by The American University in Cairo (AUC) under Grant SSE-ECNG-Y.G.-FY21-FY22-FY23-RG(1-21)-2020-Oct-01-14-33-33.

ABSTRACT Traffic offloading in cellular networks is considered an evolving application of unmanned aerial vehicles (UAVs). UAVs have attractive characteristics for this application, such as the ease of deployment, the relatively low cost and the line-of-sight signal propagation. This paper proposes a machine learning-based deployment of UAVs as temporary base stations (BSs) to complement cellular communication systems in times of excess traffic loads. In this role, the UAV is tasked with the proper sizing of the excess mixed traffic demands on the terrestrial BSs and the subsequent offloading of this traffic, given its different QoS requirements. We achieve this objective by optimizing the number of needed UAVs and their three-dimensional (3D) positions. A traffic estimation technique based on the autoregressive integrated moving average (ARIMA) model is utilized to estimate the mixed traffic demand. Our proposed machine-learning approach, based on the reinforcement learning (RL) methodology, aims to obtain real-time results close to the solution's optimal bound. Simulation results show that the proposed RL solution achieves its close-to-optimal real-time objectives. The proposed UAV deployment approach is also shown to clearly outperform a commonly used generic technique for UAVs deployment in such situations.

INDEX TERMS Mixed traffic estimation, network traffic offloading, reinforcement learning, UAV deployment.

I. INTRODUCTION

The unmanned aerial vehicles (UAVs) have attracted much attention in the past few years for use in the different areas of communication systems [1]. This is due to their high mobility, flexible deployment, low cost, and line-of-sight (LoS) propagation in air-to-ground communication links. In some public events, the communication traffic demands may become extremely high. This can also be the case when natural emergencies strike as the communication infrastructure may become unavailable or insufficient. In such cases, UAVs can be used to augment or replace parts of the communications

The associate editor coordinating the review of this manuscript and approving it for publication was Cong Pu¹.

infrastructure. That is, the UAV acts as the radio access station, or base station (BS), for the communication service users. The UAVs can also collect or offload the network data for the purpose of reducing the computational efforts of some network devices as in the machine-to-machine (M2M) communication applications [2], [3]. In these utilizations, the UAV must be equipped with the needed hardware and communication protocols to properly conduct its responsibilities. It should also be appropriately positioned, with sufficient bandwidth, in the area where the users need communication services to fulfill its mission efficiently.

Offloading high traffic demands using UAVs has recently been discussed extensively in the literature. The UAV-based solutions proposed in the previous research efforts have

focused mainly on a static approach by defining predetermined fixed trajectories. It is worth noting that most of the earlier studies addressed downlink communications, which are typically less challenging than the case of uplink communications. This is due to the fact that the case of uplink traffic involves considering the unpredictable traffic level and type of each user.

In this study, we consider the deployment of UAVs to offload excess uplink traffic in cellular networks due to significant events with a temporary high density of users, e.g., the case of large-scale exhibitions or sports/musical events. As illustrated in Figure 1, both malfunctioning and overloaded networks are modeled in our simulation scenarios while the UAVs are optimally deployed to support the communications in these situations. Also, this study evaluates the uplink performance of a mix of two traffic types: enhanced mobile broadband (eMBB) and the traffic of massive deployments of machine-type communication devices (MTCs). The traffic demand is estimated using a traffic sizing model based on the autoregressive integrated moving average (ARIMA) model. We focus our study on fulfilling user requirements for each traffic type while deploying a minimum number of UAVs. We use an optimization technique to obtain the optimal bound of the solution to this problem. Then, we propose a machine-learning-based (ML) technique to obtain a real-time solution that is close to the optimal bound but with significantly lower complexity.

The novelty of our work lies in the dynamic deployment of multiple UAVs in 3D space without having to rely on a fixed trajectory. The UAVs move within the deployment area according to the instantaneous excess uplink traffic demand of the users. The traffic demand of the users is forecasted using our developed traffic sizing model. Then, the proposed online ML approach learns the uplink data rate pattern of the users and determines the minimum required number of UAVs and their locations to fulfill users' demands and resource requirements.

A. RELATED WORK

Several approaches have been discussed in the literature for using UAVs in the downlink direction to provide communication services for excess traffic offloading. The UAV trajectory design approach is discussed in [4], [5], [6], [7], and [8] to achieve maximum downlink data rate for mobile users at overloaded cells or the cell edges of cellular networks. The study in [4] uses stochastic geometry to position multiple UAVs in a chain-like topology as a bridge between the overloaded and underloaded BSs. In [5], multi-UAVs coordination and offloading schemes are proposed to extend coverage to the cell-edge users. This scheduling problem is also considered in [6] and [7]. In [6], the fairness between the scheduled users is attained by maximizing the minimum rate according to the users' quality of service (QoS) requirements. This is done using a successive convex optimization technique in a configuration that positions a UAV at the edge of three

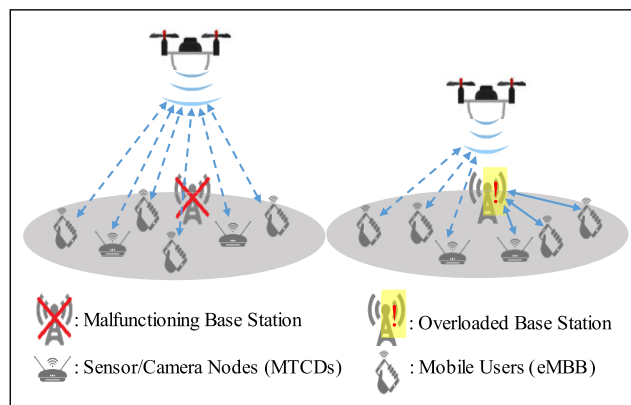


FIGURE 1. UAV-assisted coverage scenario.

adjacent cells. In [7], a spectrum sharing scheme is proposed to partition the total bandwidth orthogonally between the UAV and the BS in a UAV-assisted cellular offloading scheme. The energy efficiency of a single deployed UAV is maximized in [8] by jointly optimizing the resource allocation, user partitioning, and the UAV's trajectory selection.

The UAV dynamic positioning for traffic offloading basically in the downlink direction is addressed in [9], [10], [11], [12], and [13]. In [9], traffic offloading is done using UAVs based on a contract designed by the BSs. This work devises a two-stage contract optimization in multi-UAV cellular networks, considering both the current traffic demands as well as the required UAV energy consumption. Furthermore, the authors in [10] use an unsupervised learning approach combined with the concept of electrostatic forces of attraction and repulsion to obtain the minimum number of required UAVs and their 3D placements to fill the network downlink coverage gaps in the areas with some failed BSs. In [11], the minimization of the number of UAVs is done by using an optimization model with three constraints, namely, the ratio of covered users by each UAV, the downlink rate, and the limited UAV availability due to the charging time. The study in [12] proposes a deployment that limits the BSs to the minimum power levels that are sufficient to provide the users' minimum QoS demands with the help of UAVs. In [13], the sum of downlink data rates is maximized by the joint optimization of the UAVs' altitude, transmission power and the percentage of offloaded users.

Moreover, the traffic estimation-based UAV deployment is discussed in [14]. The data rate estimation is done by capturing the downlink traffic density using a Gaussian mixture function. The authors also utilize the weighted expectation maximization approach to estimate the areas of high traffic demands with respect to the users' distribution. The overloaded BS broadcasts a signal that contains information about the downlink demand and the service area to request the assistance of a UAV. The BS designs contracts for all UAVs and then chooses the UAV that mainly fulfills the transmission power requirements.

The studies discussed above investigate the use of UAVs to support traffic offloading in the downlink direction. However,

few other studies have investigated the uplink communications supported by deploying UAV-mounted BSs. The work in [15] tests the performance of augmenting a network of terrestrial BSs in both the uplink and downlink directions with a single UAV base station. The authors consider the problem of maximizing the average data rate while controlling the transmission power without considering traffic offloading scenarios. The M2M network deployments are studied in [16] and [17]. The network performance is regulated by establishing communications with the UAV-mounted BSs deployed at optimal locations while considering the physical resource allocation of the deployed UAVs in the absence of the terrestrial infrastructure.

In this study, we address the shortcomings in previous works by using a traffic estimation model to predict the different traffic demands of the users in the uplink direction. Then, we propose a low-complexity ML-based algorithm for the dynamic deployment of multi-UAVs in a 3D space to serve excess uplink traffic demands that the existing terrestrial BSs cannot normally handle. The proposed technique is designed to perform in real-time with performance close to the optimal bound that is also determined in this study.

B. PAPER CONTRIBUTIONS AND ORGANIZATION

The main objectives of this study are summarized as follows.

- We formulate a traffic estimation model to size dynamic uplink excess traffic demands using the ARIMA model. This traffic is a mix of eMBB and MTCD users.
- We propose an algorithm that provides the optimal bound for the 3D locations of the minimum required number of UAVs to satisfy the overload traffic demands.
- We propose an ML-based technique for the dynamic determination of the number and positions of the UAVs to satisfy the instantaneous excess uplink traffic demand.
- We provide a detailed analysis and evaluation of the computational complexity of the proposed ML-based solution against that of the optimal solution to show the relative merit from the real-time perspective.
- We perform a complete evaluation study for the proposed techniques as well as a selected benchmark technique to demonstrate the performance characteristics of the proposed solutions under different operating scenarios.

The rest of this paper is organized as follows. In Section II, the system model and the problem formulation are presented. Section III discusses the optimal solution to the problem and the proposed ML-based approach with their complexity analysis. Section IV introduces the evaluation results based on several operational scenarios. Finally, Section V concludes this paper.

II. SYSTEM MODEL AND PROBLEM FORMULATION

In this section, we present the system model that we use for our problem. It mainly considers the channel model and the network setup considered in this study. Then, we formulate

an optimization problem that depends on the excess traffic prediction model that we also present in this section.

A. CHANNEL MODELLING

We consider an urban area where terrestrial BSs (TBSs) are deployed to serve a population of cellular communication users. It has been reported that the coverage of these TBSs is not fulfilling the user communication needs under some events. Let T denote a set of TBSs covering a certain geographical area. Let U be a set of UAV-mounted base stations to be deployed to assist with handling the excess traffic demand on the TBSs. The set of all combined BSs is $G = T \cup U$. The serving BS is denoted by the superscript x where $x \in G$. The set of served users in the deployment area is denoted as E such that each user of this set is represented by a subscript i . The communication links between the BSs and users are modeled as block fading channels. Each channel is assumed to be constant within the fading block but generally changes from one block to another. The time duration of each fading block denoted as b is smaller than each time slot period, so the number of fading blocks in one time slot is denoted as L such that $b \in L$ and $L > 1$. The UAVs at high altitudes are likely to have LoS links with the users. The channel gain $h_i^x(t, b)$ between a user i and an aerial/terrestrial BS x in the fading block b of a time slot t is

$$h_i^x(t, b) = \sqrt{\rho_i^x(t)} g_i^x(t, b), \quad (1)$$

where $\rho_i^x(t)$ is the large-scale fading component of the average channel power gain that includes the channel attenuation caused by the path loss and shadowing between the user i and the serving BS x and $g_i^x(t, b)$ is the small-scale fading which is a function of the Rician factor K_i^x and is modeled as

$$g_i^x(t, b) = \sqrt{\frac{K_i^x(t, b)}{K_i^x(t, b) + 1}} g + \sqrt{\frac{1}{K_i^x(t, b) + 1}} \tilde{g}, \quad (2)$$

where g represents the deterministic LoS component of the channel as $|g| = 1$, \tilde{g} is a circularly symmetric complex Gaussian random variable that represents the random scattered components and $K_i^x(t, b)$ is the Rician factor of the user i in the fading block b of the time slot t . The Rician factor for each user differs from one time slot to another. However, it is found to be related to the elevation angle between the user i and the serving BS x [18]. When the elevation angle increases, the Rician factor increases because the communication link would have less scattering and larger portion of the LoS component. When the elevation angle in each time slot has a small change, the Rician factors in different fading blocks are assumed to be identical as $K_i^x(t, b) = K_i^x(t)$, $\forall b \in L$. The elevation angle-based Rician factor is calculated as

$$K_i^x(t) = \lambda_1 e^{\lambda_2 \theta_i^x(t)}, \quad (3)$$

where λ_1 and λ_2 are environmental coefficients and $\theta_i^x(t) = \frac{180}{\pi} \tan^{-1} \frac{H_x(t)}{B_i^x(t)}$ is the elevation angle between the diagonal distance $d_i^x(t)$ and the horizontal ground projection distance

$B_i^x(t)$ between the serving BS x and the user i at a time slot t . This direct communication link distance can be calculated as

$$d_i^x(t) = \sqrt{H_x^2(t) + B_i^{x2}(t)}, \quad (4)$$

where $H_x(t)$ is the altitude of the serving BS x . From the direct channel distance d_i^x , the average large-scale channel power gain $\rho_i^x(t)$ can be calculated as

$$\rho_i^x(t) = \rho_0 (d_i^x(t))^{-\sigma}, \quad (5)$$

where ρ_0 is the channel gain at a reference distance of 1 meter and can be calculated as $\rho_0 = \left(\frac{4\pi f_c}{c}\right)^{-2}$ as f_c is the carrier frequency, c is the speed of light, σ is the path loss exponent.

To formulate the channel capacity, the channel signal-to-interference-plus-noise ratio (SINR) is calculated as follows

$$\Gamma_i^x(t) = \frac{\sum_b^L |h_i^x(t, b)|^2 P_i}{N_0 + \sum_{z \in G \setminus \{x\}} \sum_b^L |h_i^z(t, b)|^2 P_i}, \quad (6)$$

where P_i is the transmission power and N_0 is the noise power.

B. PROBLEM FORMULATION

The problem that we try to solve is how to deploy a UAV-assisted heterogenous network to offload excess traffic of a cellular structure due to public events, for example, with a temporary high population of users. By this heterogenous aerial and terrestrial BSs deployment, we aim to size the users' traffic in the uplink direction to provide an acceptable SINR level per user and ensure that excess user data traffic demands are properly served. The instantaneous achievable data rate is targeted to satisfy the estimated traffic demands of the users. We consider a mix of two traffic types, namely, the eMBB traffic and the massive machine-type communications (mMTC) that consist of delay-tolerant camera devices with large packet sizes and wireless sensors with smaller packet sizes and lower arrival rate. The instantaneous achievable data rate $R_i^x(t)$ of the user i associated with a serving BS x can be calculated as

$$R_i^x(t) = BW \log_2(1 + \Gamma_i^x(t)), \quad (7)$$

where BW is the channel bandwidth.

The system can estimate the users' traffic demands using a periodic data traffic modeling approach. Depending on this prediction, the formulated problem is solved to determine the UAV deployment required to cover any excess traffic needs. We use the multiplicative ARIMA model to predict the mixed eMBB and mMTC traffic demands. The ARIMA model is the most widely used approach in time series forecasting [19]. It predicts the future traffic depending on the previous information known about the traffic using a linear combination of predictors. The term autoregression (AR) indicates that the changing traffic regresses on its own lagged, or prior, values. The forecasted traffic using the AR model of order p at a time instant t can be written as

$$y_i^{AR}(t) = c + \beta_1 y_i(t-1) + \dots + \beta_p y_i(t-p) + \varepsilon_i(t), \quad (8)$$

where $y_i^{AR}(t)$ is the estimated traffic of a user i at time t using the AR formulation, c is a constant, β_1, \dots, β_p are the regression weights which are obtained from the prior observations known about the concerned mixed traffic demands, $y_i(t)$ is the traffic demand associated with a user i at a time instant t and $\varepsilon_i(t)$ is a white noise which is sampled from a normal distribution at a time instant t . This AR model is similar to the typical multiple regression models, but it uses the lagged values of $y_i(t)$ as predictors.

If the time series data shows upward or downward trends, the moving average (MA) model is integrated to the AR regression model in (8) to enhance the estimation. The MA part of the overall ARIMA model uses the prior error terms of $\varepsilon_i(t)$ to cope with the trends of the traffic data in the regression model. The MA regression model of order q at instant t can be written as

$$y_i^{MA}(t) = c + \varepsilon_i(t) + \phi_1 \varepsilon_i(t-1) + \dots + \phi_q \varepsilon_i(t-q), \quad (9)$$

where ϕ_1, \dots, ϕ_q are the regression weights of the MA model. Since the traffic data might hold an upward or downward trend, the ARIMA model is obtained when the time series is differenced by a degree of d to develop a stationary time series with a constant mean for the AR regression equation. The differenced traffic demand time series of the user i is therefore written as

$$y_i'(t) = c + \beta_1 y_i'(t-1) + \dots + \beta_p y_i'(t-p) + \phi_1 \varepsilon_i(t-1) + \dots + \phi_q \varepsilon_i(t-q) + \varepsilon_i(t), \quad (10)$$

where $y_i'(t)$ is the differenced time series representing the change between the consecutive data points by a degree of d . This differenced time series can then be written as

$$y_i'(t) = y_i(t) - y_i(t-1) - \dots - y_i(t-d-1) - y_i(t-d), \quad (11)$$

If the time series holds a seasonality trend of the observation data, the ARIMA model degrees (p, d, q) are repeated considering the seasonality trend of degrees $(P, D, Q)_s$ where s is the degree at which the data trend is repeated. The overall ARIMA model that is used to predict each user's traffic demands $y_i(t)$ from the prior observations that are known about the mixed users' traffic demands can be represented as ARIMA $(p, d, q)(P, D, Q)_s$.

Based on the quantified traffic demands of the ARIMA model, the UAV deployment problem is formulated. The objective function of this problem is to minimize the total number of deployed UAVs to provide communication services to the anticipated excess traffic demands. The optimization problem can therefore be written as

$$\min_{X_x, Y_x, H_x, u} |U|, \forall x \in U, \quad (12)$$

$$\text{s.t. } B_i^x(t) \leq r_x(t), \forall i \in E, \quad (12a)$$

$$R_i^x(t) \geq y_i(t), \forall i \in E, \quad (12b)$$

$$\Gamma_i^x(t) \geq \Gamma^{\min}, \forall i \in E, \quad (12c)$$

$$u \leq u^{\max}, u \in \mathbb{N}, \quad (12d)$$

$$\mathbf{D}^{\min} \leq [X_x, Y_x, H_x] \leq \mathbf{D}^{\max} \in \mathbb{R}^3, \quad (12e)$$

where $|U|$ is the length of the set of UAVs such that $U = \{1, 2, \dots, u, u^{\max}$ is the maximum allowable number of UAVs that can be deployed, X_x and Y_x are the two-dimensional (2D) coordinates of the serving BS x in U , r_x is the ground coverage cell radius of the associated BS x in G , Γ^{\min} is the SINR minimum threshold and \mathbf{D}^{\min} and \mathbf{D}^{\max} are the boundaries of the deployment area. The constraint in (12a) defines the user associations to the BSs to ensure that all users are covered by at least one BS. Then, the constraint in (12b) controls the UAVs deployment using the instantaneous achievable data rate to cover the estimated overall traffic demand $y_i(t)$ at each time instant t for all users in E . Finally, the constraint in (12c) limits the air-to-ground channel attenuation and the possible interferences among the deployed BSs.

III. PROPOSED UAV DEPLOYMENT APPROACHES

In this section, we introduce a novel ML-based technique to address the uplink traffic sizing problem that properly deploys the UAV-mounted BSs to satisfy the excess traffic loads. The objective is to deploy the minimum possible number of UAVs to achieve this purpose at a cost close to the optimal minimum value. In order to calculate the optimal bound of this deployment, we also introduce an optimization-based algorithm that establishes this bound. This optimal algorithm cannot produce its results in real-time fashion in such dynamic environments due to its high complexity. Hence, our proposed ML technique, based on the reinforcement learning methodology, is devised to produce real-time results at a near-optimal deployment cost.

A. THE OPTIMAL BOUND OF THE SOLUTION

The optimization problem formulated in (12) is found to be a mixed-integer nonlinear programming (MINLP) problem that becomes intractable in high-dimensional spaces. Therefore, the optimal bound of the solution can be obtained using the branch and bound (BnB) algorithm [20] that branches along the integer variable of the problem. This integer variable represents the number of the deployed UAV BSs. The resulting subproblem is adopted to be a nonlinear programming (NLP) problem that can be solved using a non-differentiable optimization technique since the derivatives of the constraint functions cannot always be guaranteed along the search space dimensions. Hence, we utilize the particle swarm optimization (PSO) algorithm [21] to add some heuristics in the search for a suboptimal solution for the nonlinear subproblem. These added heuristics allow the problem to be solved within the class of NP-complete problems that use polynomial algorithms to find near-optimal solutions. Otherwise, the optimization problem becomes exponentially intractable with high dimensional spaces when we solve with highly dense network deployments.

To implement this optimization algorithm, the penalty method proposed in [22] is used to formulate an unconstrained objective function that represents the constraints of the problem in (12). Then, the NLP subproblem is reformulated as follows.

$$\begin{aligned} \min_{X_x, Y_x, H_x, u} f(X_x, Y_x, H_x) \\ = \min_{X_x, Y_x, H_x, u} \sum_{i \in E} \psi_i C_i(X_x, Y_x, H_x), \forall x \in U, \end{aligned} \quad (13)$$

where $f(X_x, Y_x, H_x)$ is the exact penalty function and $\psi_i > 0$ is the penalty coefficient that is chosen to give some priorities and tolerances to the infeasible constraints of the original problem in (12) and control the constraint penalties $C_i(X_x, Y_x, H_x)$. These priorities and tolerances are given with respect to each user's traffic demands such that the eMBB traffic is given the highest priority to access the deployed network since the eMBB is associated with the most urgent traffic transmissions in our network configuration. The constraint penalties are given as

$$\begin{aligned} C_i(X_x, Y_x, H_x) = & \max(0, B_i^x(t) - r_x(t))^2 \\ & + \max(0, y_i(t) - R_i^x(t))^2 \\ & + \max(0, \gamma_{th} - \gamma_i^x(t))^2, \forall i \in E, x \in G, \end{aligned} \quad (14)$$

These constraint penalties will be driven to zero if the locations of the serving UAV BSs satisfy the constraints of the problem in (12). Accordingly, each particle of the PSO algorithm represents a potential location of the UAVs. The heuristic learning exemplars of this technique utilize the information gathered by the global, local, and personal best positions obtained by the whole swarm particles. Algorithm 1 provides the details of the PSO procedures while the number of the UAV-BSs is obtained according to the branching rule of the BnB algorithm, as illustrated in the procedures of Algorithm 2.

B. THE MACHINE LEARNING-BASED SOLUTION

This algorithm is based on the Q-learning [23] technique which is an ML-based technique under the category of reinforcement learning. The algorithm is used to find the optimal policy that maximizes the total reward in successive steps. Q-learning is quite suitable for our dynamic UAV deployment problem because it mainly seeks to find the best set of UAV deployment actions by predicting the level of fulfillment of the excess traffic demands in successive algorithm iterations. Our adopted ML model aims to speed up reaching the best deployment scenario to meet the excess traffic demands (i.e., maximize the reward) of the UAV deployment calculator (i.e., agent) over the course of the progress of the algorithm. The adopted Q-learning model consists of four elements: the Q-value, the state space, the action space, and the reward [23], [24]. At each time slot t , the deployment agent chooses a UAV positioning action according to the Q-value to maximize the long-term reward.

1) Q-VALUE

As the UAVs' positions change to a given state s by a given deployment action a , the state-action value function $Q^\pi(s, a)$ represents the expected demand-fulfillment reward for selecting the action a in state s and then the following deployment policy π . The optimal Q-value function can be calculated for s_t and a_t by

$$Q(s_t, a_t) = Q(s_t, a_t) + \alpha \left[r_t + \gamma \max_a Q(s_{t+1}, a_t) - Q(s_t, a_t) \right], \quad (15)$$

where α is the learning rate and γ is the discount factor. The deployment agent observes a UAV positioning state s_t from a state space S . The agent carries out a deployment action a_t from the discrete action space A . The taken action a_t at the time instant t updates the current state s_t to a new deployment state s_{t+1} . The Q-value is updated $Q(s_{t+1}, a_t)$ and the agent receives a traffic demand fulfillment reward r_t .

The optimal policy π is the epsilon-greedy action selection policy [25] which discovers the next best action, according to the current state, to maximize the Q-function at each step. The selection policy has a decision parameter ϵ such that $\epsilon \in [0, 1]$. The agent sometimes picks random actions in order to visit new states and actions to explore the environment. The epsilon-greedy action is determined as

$$a_t = \begin{cases} rand, & \text{if } \delta > \epsilon \\ \arg \max_{a \in A} Q^\pi(s_t, a), & \text{otherwise} \end{cases}, \quad (16)$$

where δ is a uniform random variable updated at each step from the range $0 \leq \delta \leq 1$.

2) STATE REPRESENTATION

The state of the environment is represented by the positions of UAVs defined as $[X_x, Y_x, H_x]$, $\forall x \in U$. The boundaries of the state space of each UAV are defined in \mathbf{D}^{min} and \mathbf{D}^{max} .

3) ACTION SPACE

At each time step, the agent carries out an action a_t where $a_t \in A$, which involves picking a direction for each UAV. The action space has all the combinations for the possible directions for the UAVs to take which could be either an incremental or a decremental step in any of the 3D directions.

4) REWARD

The reward function is formulated in terms of the constraints in (12), representing the ratio of active users within the coverage of the associated cell and the ratio of users with satisfied data rates and the SINR. When the number of satisfied active users increases, the reward increases, indicating that the solution converges. The increase in the number of UAVs causes a negative reward because we aim at minimizing the number of deployed UAVs. The reward function is adapted from the exact penalties given in (14) such that

$$r_t = \sum_{i \in E} -C_i(X_x, Y_x, H_x), \forall x \in G. \quad (17)$$

Algorithm 1 Heuristic Solution of the NLP Subproblem

Input: $f(X_x, Y_x, H_x)$, \mathbf{D}^{min} , \mathbf{D}^{max}

Output: X_x^*, Y_x^*, H_x^* , $\forall x \in U$

1. set PSO parameters
//swarm size, unification factor, inertia and acceleration constants, neighbors ring size, the maximum iterations $iter^{max}$
2. initialize a swarm \mathcal{S} of size $|\mathcal{S}|$
//uniformly distributed swarm with upper and lower bounds \mathbf{D}^{min} , \mathbf{D}^{max} of \mathbf{P}_s : $\mathbf{P}_s = [X_x, Y_x, H_x]$, $\forall x \in U$, $s \in \mathcal{S}$
3. initialize stagnant counters $st = 0$ and $cnt = 0$
4. initialize refresh cycle $rc = 0.05 \times iter^{max}$
5. calculate $f(\mathbf{P}_s)$, $\forall s \in \mathcal{S}$
6. obtain the global, local and personal best locations of each particle in \mathcal{S}
// let the global best position be denoted as \mathbf{P}_g
7. set the maximum penalty bound of $f(X_x, Y_x, H_x)$
// such that $f^{max} = f(\mathbf{P}_g)$
8. initialize random particle velocities $\mathbf{V}_s = rand$, $\forall s \in \mathcal{S}$
9. **for** $iter \leftarrow iter^{max}$
10. **if** $cnt > rc$, **then**
11. **if** $st == 0$, **then**
12. set st and reset cnt
13. reinitialize $\mathbf{V}_s = rand$, $\forall s \in \mathcal{S}$
14. **else**
15. break
16. **end if**
17. **end if**
18. update \mathbf{P}_s : $\mathbf{P}_s \in [\mathbf{D}^{min}, \mathbf{D}^{max}]$ given \mathbf{V}_s , $\forall s \in \mathcal{S}$
19. compute $f(\mathbf{P}_s)$, $\forall s \in \mathcal{S}$
20. update the global, local and personal best positions of \mathcal{S}
21. update particles' velocities \mathbf{V}_s , $\forall s \in \mathcal{S}$
22. **if** $f(\mathbf{P}_g) == f^{max}$, **then**
// particles' stagnation detection
23. increment cnt
24. **else**
//update the penalty bound
25. reset st , cnt
26. $f^{max} = f(\mathbf{P}_g)$
27. **end if**
28. **end**
29. $[X_x^*, Y_x^*, H_x^*]$, $\forall x \in U = \mathbf{P}_g$

The max function in (14) allows the summation of the differences between the served traffic and the demand to be unbiased towards the users with satisfied requests. Algorithm 3 shows the steps of the proposed technique to obtain the minimum number of required UAVs and their respective positions. Based on the value of u^{max} , the agent's discrete action space is defined. Initially, only one UAV is deployed with its state (3D position) that is determined randomly. On each episode, the agent takes actions to place the UAV in the deployment area while observing the value of

Algorithm 2 BnB Technique for the Optimal Bound

Input: $f(X_x, Y_x, H_x)$, \mathbf{D}^{min} , \mathbf{D}^{max}
Output: X_x^* , Y_x^* , H_x^* , u^* , $\forall x \in U$

1. set an upper bound to the constraint penalty function f^{max}
2. branch to the first node $u = 1$
start at the first branch
3. **while** $u \leq u^{max}$
4. formulate the unconstrained objective function in (13)
5. solve the resulting NLP subproblem at u
6. get X_x^* , Y_x^* , H_x^* , $\forall x \in U$ by calling the function detailed in Algorithm 1
7. calculate $f(X_x, Y_x, H_x) : X_x, Y_x, H_x, \forall x \in G$
8. **if** $f(X_x, Y_x, H_x) < f^{max}$, **then** //optimal solution is found
9. break
10. **else**
11. bound the branch u
12. go to branch $u + 1$
13. **end if**
14. **end**
15. $[X_x^*, Y_x^*, H_x^*, u^*] = [X_x, Y_x, H_x, u], \forall x \in U$

the reward which indicates the fulfillment of the constraints in (12). The agent starts by exploring the environment. After several iterations, it would have improved its knowledge about the environment, allowing it to choose the next action that maximizes the reward. This process is iterated until the agent reaches the locations with the maximum reward in the deployment area. If the users are not satisfied after a defined number of episodes ep^{max} , the number of UAVs is increased by one, and the agent starts trying to satisfy the constraints with the new set of UAVs. Once the constraints are met, the agent returns the number of UAVs, which is the minimum possible number along with their 3D locations. The algorithm returns the solution with the highest reward if the constraints are not met.

C. COMPUTATIONAL COMPLEXITY ANALYSIS

In the following, we analyze the computational complexity of the optimal algorithm as well as that of the Q-learning based algorithm that we introduced earlier in this section.

1) COMPLEXITY OF THE OPTIMAL SOLUTION

The optimal solution is obtained by the procedures implemented in Algorithm 2. The worst-case complexity of the algorithmic steps can be analyzed using the big- O notation, and the overall complexity of Algorithm 2 can be expressed by summing up each step time complexity as:

- The parameter setting functions in the steps from 1 to 3 need a constant time complexity $O(1)$.

Algorithm 3 Q-Learning Method for the UAV Deployment

Input: $f(X_x, Y_x, H_x)$, \mathbf{D}^{min} , \mathbf{D}^{max}
Output: X_x^* , Y_x^* , H_x^* , u^* , $\forall x \in U$

1. set an upper bound to the constraint penalty function f^{max}
2. initialize state space S and action space A for $u = 1$
3. initialize $Q(s, a)$, $\forall s \in S, a \in A$
4. **while** $u \leq u^{max}$
5. **for** $ep \leftarrow ep^{max}$
6. initialize state s_t //place U randomly in the deployment
7. **for** $t \leftarrow t^{max}$ // t^{max} is the maximum number of episode time steps
8. find action a_t from s_t
9. apply the policy π
10. execute the action a_t
11. observe s_{t+1}
12. calculate r_t
13. update $Q(s, a)$
14. save $s_t \leftarrow s_{t+1}$
15. **end**
16. **end**
17. **if** $f(X_x, Y_x, H_x) < f^{max}$, **then**
18. break
19. **elseif** $u < u^{max}$, **then**
20. $u = u + 1$ //add one UAV to the set U
21. increase the state-space S and the action space A
22. Update $Q(s, a)$, $\forall s \in S, a \in A$ //add actions, states
23. **endif**
24. **end**
25. $[X_x^*, Y_x^*, H_x^*, u^*] = [X_x, Y_x, H_x, u], \forall x \in U$

- The steps from 3 to 14 are iterated $n := u^{max}$ times. Then, the linear time complexity $O(n)$ is required for steps 3 to 14 except for step 5.
- The complexity of step 5 that runs the PSO procedures of Algorithm 1 contributes to the following accumulative complexities in a loop of n iterations $O(n) + O(nm) + O(nm \log(m)) + O(n \log m) + O(nl) + O(nml) + O(nml \log(m)) + O(nl \log m)$ such that $m := |S|$, $l := iter^{max}$.

The typical BnB algorithm runs in exponential time complexity $O((n-1)!) = O(2^{n-1})$. Therefore, we adapted the solution of the optimal bound by utilizing the constraint that specifies a maximum number of UAVs to limit the BnB branches over the integer variable $u := |U|$. Since the branching rule runs over a single variable only, the complexity of the BnB step can be reduced to linear time complexity, as stated in each algorithmic step complexity. However, the multiple iterations of step 5 in Algorithm 2 derive the overall time complexity to be exponential due to the term of $O(nml)$.

Since the problem dimension n is bounded by the constraint concerning the maximum number of UAVs and the other PSO parameters, m and l , are also bounded, the overall complexity of this optimization algorithm is concluded to be exponentially bounded.

2) COMPLEXITY OF THE Q-LEARNING-BASED SOLUTION

The Q-learning time complexity analysis is also evaluated using the big- O notation for each step in the learning process given in Algorithm 3 in the worst-case scenario when all the learning episodes are executed. Therefore, the overall complexity of this solution can be stated as the augmentation of each step complexity as follows:

- The parameters initialization steps from 1 to 3 require a constant time complexity $O(1)$.
- The episodes loop from step 4 to step 16 require linear time complexity $O(nML)$ where $M := ep^{max}$ and $L := t^{max}$ that multiplies each step complexity within this loop. These steps within the learning iterations mainly perform in a constant time complexity $O(1)$ except for steps from 8 to 14.
- The search process that runs in step 9 requires $O(z \log z)$ as $z := |A|$ adding the looping complexity of $O(nML)$.
- The steps from step 17 to step 22 require a constant time complexity inside a loop producing a linear time $O(n)$.

Hence, the highest complexity term is noticed in the execution of the steps from 6 to 13. These step complexities are expressed as $O(nML) + O(z \log z)$. This time complexity can be observed as linear in the bounded Q-learning parameters, such as the episodes count ep , the learning time steps t and the action space size z . Therefore, the overall algorithmic complexity does not primarily depend on the formulated problem space size. This results in a much lower complexity than the optimization algorithm proposed in Section III-A, especially when solving in dense and large network deployments.

IV. EVALUATION RESULTS

In this section, we discuss the simulation results to evaluate the performance of the proposed optimal and ML solutions.

A. NETWORK SETUP

The deployed network consists of one TBS that is located at the center of the deployment area. Due to the excess traffic demand on the TBS, multiple UAV-mounted BSs are to be deployed within the area according to the functionality of the algorithm being evaluated. The network and algorithm parameters are given in Table 1. The simulated traffic is a mix of eMBB, MTCD camera and monitoring sensors traffic types. Due to the unavailability of real traffic data, we modeled the transmission requests of the different traffic profiles using the random Poisson distributions with the parameters in Table 2. For real network operations, the ARIMA model introduced in Section II can be used to anticipate the expected size of network users' data traffic.

TABLE 1. Network and algorithm parameters.

Parameter	Value
Channel bandwidth (BW)	3 MHz
Number of PRBs	15
Carrier frequency (f_c)	2 GHz
Transmission and noise power	20, -110 dBm
Environment parameters (λ_1, λ_2)	1, 0.7329
Path loss exponent (σ)	2
SINR threshold (Γ^{min})	2 dB
PSO Parameters $[\mathcal{S}], \alpha_{PSO}, w, c_1, c_2, rc$	50, 0.1, 0.729, 1.494, 1.494, 5 [26]
Learning rate (α), discount factor (γ), decision parameter (ϵ)	0.1, 0.8, 0.2

TABLE 2. Traffic configurations [17], [28].

Traffic Profile	eMBB	Camera	Sensor
Arrival rate (pkt/s)	586	30	2
Packet size (bytes)	1024	512	128
Contribution percent %	50	30	20
Delay bounds (msec)	10~20	125~250	800~900

The model parameters model can be determined following the analysis discussed in Section II-B. The available physical resource blocks are scheduled among the users according to a delay-based scheduler [27] in which the users with urgent deadline requirements have higher service priority. Because of the non-existence of any work in the literature concerning the deployment of multiple UAV-BSs serving in the uplink direction, we used a benchmark technique that has commonly been used in the literature [5], [6], [7] in which the UAVs are deployed along the cell edge of the TBS to serve the users with the worst channel quality conditions. In this setup, the UAVs fly in a circular trajectory with a constant speed equal to 30 m/sec at a fixed altitude of 100 m [7]. As the UAVs move, the users are associated with either the TBS or one of the UAVs according to the closest distance. This integrated deployment is named "the generic solution" in the rest of this discussion.

B. PARAMETERS SELECTION FOR THE ML SOLUTION

The learning rate, α , is normally selected to assume a small value between 0.1 and 0.3 while the discount factor, γ , is selected to assume a large value between 0.7 to 0.9 [24]. In order to determine the combination of parameter values that best suits our setup, we conduct parameter selection experiments such that we fix the learning rate at 0.1, which indicates slow learning from the previous actions, and the discount factor at 0.9, which allows the agent to look for high rewards in the long term. The epsilon-greedy decision parameter, ϵ , which indicates the exploration index, is then varied until we get the highest possible average reward.

We then repeat this procedure by fixing the decision parameter at this obtained value and the learning rate at 0.1 while trying different discount factor values. Finally, we repeat this

procedure for the selection of the learning rate. Based on these experiments, the best combination is given in Table 1.

C. PERFORMANCE EVALUATION OF THE PROPOSED SOLUTIONS

We now analyze the performance of the proposed optimal and ML solutions and compare their results against the generic solution. The evaluation metrics are mainly based on the cost of the solution in terms of the number of needed UAVs. Then, the network performance under this cost is represented in the deadline missing ratio and the network aggregate throughput. Therefore, we set the maximum allowed number of UAVs that can be used by the optimal and ML solutions to three, i.e., $u^{max} = 3$, to represent the limitation of the available resources. Since the generic solution has no control on the number of deployed UAVs, the maximum number of three UAVs is used for all different network configurations under the deployment that places the UAVs at the cell edge of the TSB. The generic solution is simulated under the same network configurations and setup of the other proposed solutions. The network configurations include deploying the UAVs to serve an overloaded terrestrial network with different numbers of users, starting from 50 to 250 users that are uniformly distributed in the deployment area with a distribution density of 50 users/km². The presented results are the average of several simulation runs. Hence, we indicate the 95% confidence interval limit bars with each of the result points.

1) THE NUMBER OF DEPLOYED UAVS

The average number of UAVs is an indication of the cost incurred by a solution to cover the excess traffic demand in the different user deployment configurations. The average number of UAVs is presented in Figure 2 for the different deployment solutions. The generic solution has a fixed number of UAVs hovering along the edge of the main TBS's cell such that the number is always equal to three, as indicated earlier. In both the optimal and ML solutions, the average number of deployed UAVs increases as the number of users increases, but it does not reach the limit of three UAVs in the simulated scenarios. This is due to the objective stated in (12) that tries to get the minimum number of UAVs required to satisfy the excess traffic demands. Both the optimal and ML solutions apparently use two UAV-mounted base stations along with the overloaded TSB to handle the traffic requests of the network users. The solutions of the optimization and ML algorithms guarantee a minimal cost of the deployment since these approaches control the locations of the UAVs with respect to the traffic demands of the users quantified by the formulated ARIMA estimation model.

2) THE DEADLINE MISSING RATIO

Figure 3 shows the deadline missing ratio as a percentage of the transmissions that missed their deadlines to the total number of communication requests for the different traffic profiles. The optimal and ML solutions guarantee a reduced

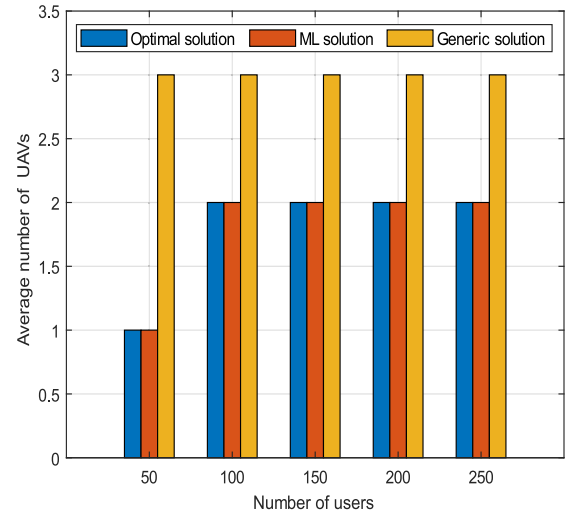


FIGURE 2. Average number of deployed UAVs.

deadline missing percentage in all user configurations when compared to the generic solution although the generic solution incurs a higher deployment cost than the optimal and ML solutions. In Figure 3(a), the network deadline missing ratio is presented for the three solutions. The proposed techniques provide an approximately zero deadline missing ratio in the network deployments of 50 and 100 users. The trend increases as the number of users increases since the proposed solutions maintain a minimum number of resources represented by the number of deployed UAV-BSs. Figure 3(b) shows the deadline missing trend for the eMBB traffic that exhibits a trend that is similar to the trend of Figure 3(a) since the eMBB traffic is 50 percent of the deployed network users.

In addition, in the simulated scenario, we utilize the delay-based scheduler that prioritizes the users with the lowest delay bound, such as the eMBB traffic, as given in the traffic characteristics of Table 2. The camera traffic in Figure 3(c) has a deadline missing ratio trend that is comparable to that of the eMBB traffic although the camera traffic is more delay-tolerant than the eMBB traffic. The reason behind this performance is that the proposed solutions tend to serve the traffic with respect to the anticipated demand according to the constraints in (12), and only the delay-based scheduler considers the delay bounds of the users. It is worth mentioning that the sensor traffic has zero deadline missing percentage at all the configurations because the modeled scenarios simulate the sensor MTCs with much smaller arrival rate traffic and tolerant delay bound, shown in Table 2, depending on the features of sensors sending data within fixed time intervals.

3) THE ACHIEVABLE THROUGHOUT

Figure 4 shows the system aggregate throughput demonstrating that the optimal and ML solutions outperform the generic solution although a greater number of UAVs is deployed all the time in this generic setup. The overall network

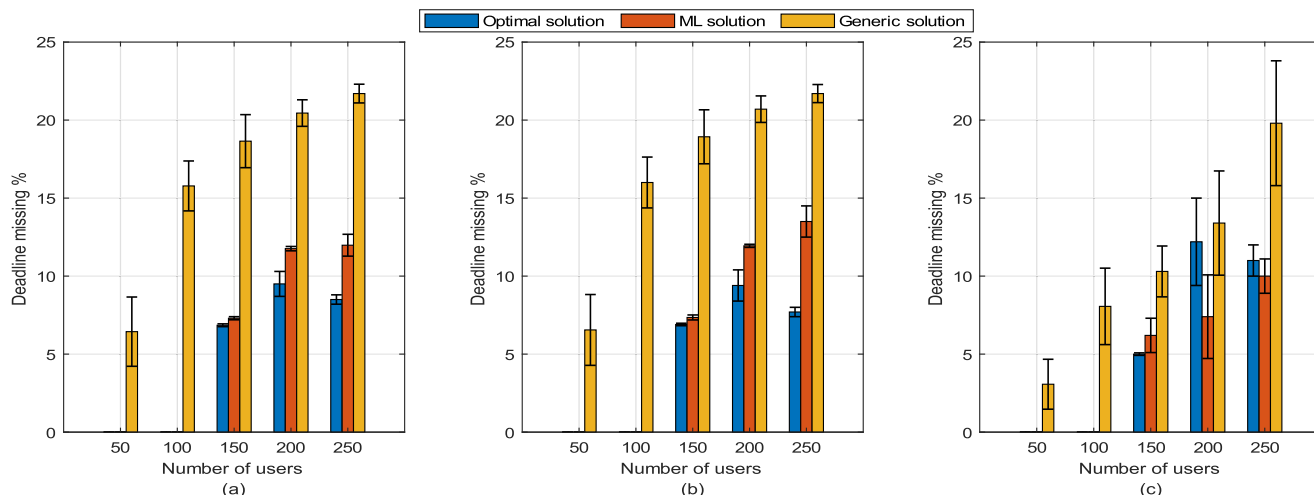


FIGURE 3. The average percentage of deadline misses for (a) the overall network (b) eMBB users (c) camera MTCDs.

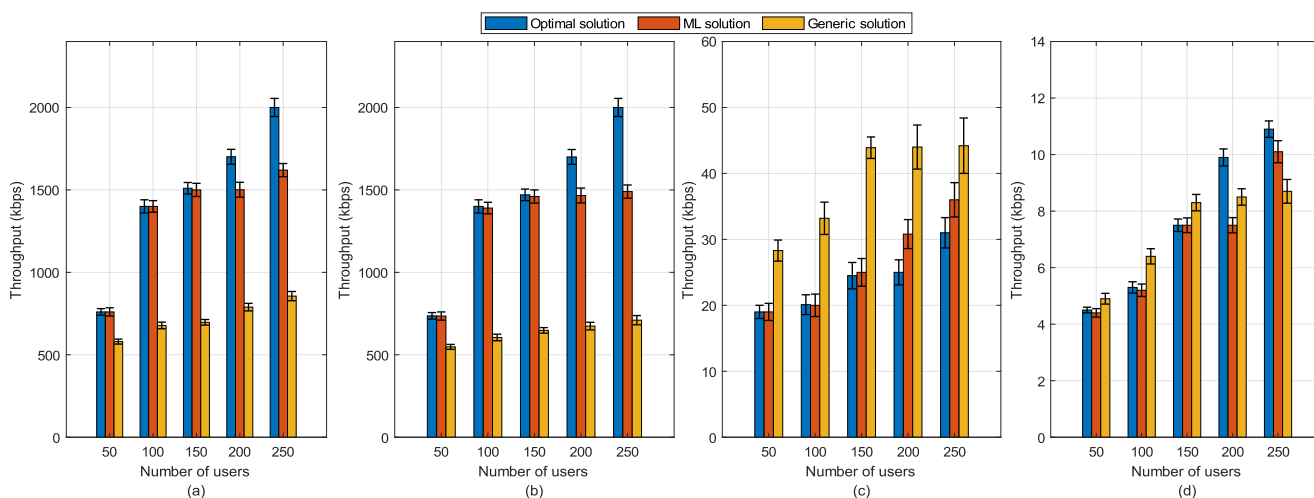


FIGURE 4. The aggregate throughput of (a) the overall network (b) eMBB users (c) camera MTCDs (d) monitoring sensor MTCDs.

aggregate throughput, in Figure 4(a), shows that the 3D UAV deployments of the optimal and ML solutions consider the measured traffic of the users. In addition, the proposed techniques prioritize the users with resources allocated by the delay-based scheduler. On the other hand, the generic solution might enhance the channel gains of the cell edge users without affecting the network throughput. The reason behind this behavior is that the generic algorithm only tries to maintain sufficient channel conditions regardless their traffic demands or even without considering whether there are physical resources assigned to the served users by the scheduler.

This generic deployment of the UAV-BSs does not guarantee effective utilization of the overall network resource since three UAVs are used in such deployment. The achievable throughput of the eMBB users is indicated in Figure 4(b). This traffic profile is characterized by larger packet sizes than the MTCDs’ traffic. Thus, the eMBB traffic contributes more to the network aggregate throughput. The throughput

of the camera traffic profile in Figure 4(c) demonstrates the insufficient utilization of the available resource of the generic solution. Although the camera throughput under the generic deployment is higher than that is obtained from the optimal and ML solutions, the deadline missing ratio increase has encountered for the same traffic profile under the generic solution, as shown in Figure 3(c). One can infer that the dynamic UAV deployment of the optimal and ML solutions accounts for the uncertain traffic demands and their associated data traffic measure using the minimum possible resources.

However, considering only the cell-edge users’ channel capacity is found to be inadequate, as in the case of the generic solution. This is because the other characteristics of the users’ traffic, such as delay bounds, data packet sizes and the available allocated resources, become congested in the access network requesting services. Similarly, the throughput of the monitoring sensor MTCDs, shown in Figure 4(d), has

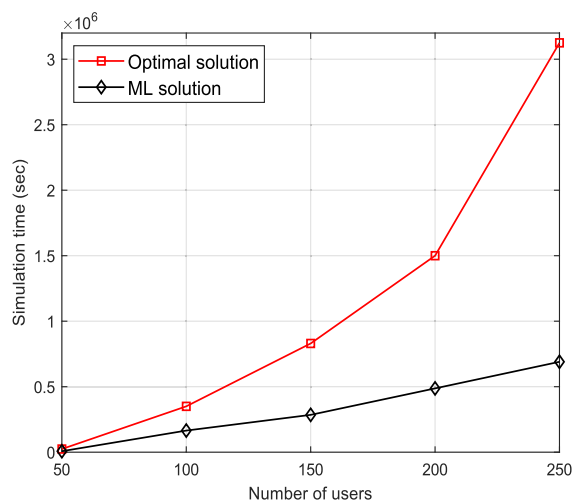


FIGURE 5. Simulation time of the proposed solutions.

comparable rates along the different solutions since this traffic profile is characterized by low arrival rates, small packet sizes and relaxed delay bounds. These characteristics do not contend much for network resources unless there are high demands by the intolerant traffic.

D. THE COMPUTATIONAL COMPLEXITIES OF THE PROPOSED SOLUTIONS

We now verify the complexity analysis of the optimal and ML solutions presented in Section III.C. This is done by examining the average simulation time of the experimental runs of each solution. Figure 5 shows the solutions' running simulation time. The running time of the ML algorithm increases linearly with a relatively small slope. This slope is mainly controlled by a tolerant termination criterion that can be selected when no significant improvements are observed between the successive iterations of the ML algorithm. Hence the complexity term $O(nML)$ can be easily bounded by the proper selection of the algorithm parameters and termination conditions.

In addition, the linear time complexity of the ML solution can be considerably influenced by the discrete step sizes chosen to define the state space S and the action space A . By regulating these discrete step sizes, the complexity term $O(z \log z)$ of the search algorithm can be maintained at low levels. These two control conditions directly impact the speed of convergence of the ML algorithm which can lead to a real-time performance depending on the size of the involved network.

On the other hand, the time complexity of the optimal solution is found to be increasing exponentially. The reason behind this performance is that the algorithm solves a subproblem with a linear complexity of $O(nml)$ at each branching node of the BnB algorithm. These observations of Figure 5 coincide with our analysis in Section III.C. To conclude this analysis, the advantage of the ML solution is that it allows for relatively fast convergence that can be used as a

practical solution in real-time arrangements. For the different evaluation metrics, the results of the ML solution are found to be close to those of the optimal solution in all simulated network configurations.

V. CONCLUSION

In this paper, we proposed optimal and machine learning-based UAV deployments as temporary BSs to offload the excess traffic demands that a terrestrial base station might encounter during certain events. For this purpose, uplink traffic sizing is carried out to determine the excess traffic that needs to be serviced by the UAV-mounted BSs. This excess traffic offloading goal is achieved by optimizing the number of deployed UAVs and their 3D positions in the area of interest. A traffic estimation technique was proposed based on the ARIMA model to estimate the excess traffic demands. We devised an optimal algorithm to determine the optimal bound of the solution and an ML algorithm to provide a practical implementation of the problem. Simulation experiments showed that the results obtained by the proposed ML solution are close to the optimal bounds while providing real-time performance. The resulting dynamic network outperforms that of the generic technique that deploys the UAV BSs at the cell edges when compared in terms of the achieved throughput and the traffic deadlines. A potential future direction for the field of traffic offloading using UAVs is to study the dynamic 3D UAV localization considering some other UAV deployment constraints that challenge flying UAVs in a given area such as the UAV transmission power constraints, and the effects of stormy weather/wind and some unreachable areas in the deployment space. In addition, there is an important future extension which is to consider the backhaul links of the deployed UAVs to the nearest sane infrastructure in the traffic offloading application. All these considerations should be included in the problem formulation as constraints or objectives of the UAV deployment problem.

REFERENCES

- [1] G. Alsuhli, A. Fahim, and Y. Gadallah, "A survey on the role of UAVs in the communication process: A technological perspective," *Comput. Commun.*, vol. 194, pp. 86–123, Oct. 2022, doi: [10.1016/j.comcom.2022.07.021](https://doi.org/10.1016/j.comcom.2022.07.021).
- [2] A. M. Seid, G. O. Boateng, B. Mareri, G. Sun, and W. Jiang, "Multi-agent DRL for task offloading and resource allocation in multi-UAV enabled IoT edge network," *IEEE Trans. Netw. Service Manage.*, vol. 18, no. 4, pp. 4531–4547, Dec. 2021, doi: [10.1109/TNSM.2021.3096673](https://doi.org/10.1109/TNSM.2021.3096673).
- [3] A. M. Seid, G. O. Boateng, S. Anokye, T. Kwantwi, G. Sun, and G. Liu, "Collaborative computation offloading and resource allocation in multi-UAV-assisted IoT networks: A deep reinforcement learning approach," *IEEE Internet Things J.*, vol. 8, no. 15, pp. 12203–12218, Aug. 2021, doi: [10.1109/JIOT.2021.3063188](https://doi.org/10.1109/JIOT.2021.3063188).
- [4] N. Tafintsev, D. Moltchanov, S. Andreev, S. Yeh, N. Himayat, Y. Koucheryavy, and M. Valkama, "Handling spontaneous traffic variations in 5G+ via offloading onto mmWave-capable UAV 'bridges,'" *IEEE Trans. Veh. Technol.*, vol. 69, no. 9, pp. 10070–10084, Sep. 2020.
- [5] H. Wu, Z. Wei, Y. Hou, N. Zhang, and X. Tao, "Cell-edge user offloading via flying UAV in non-uniform heterogeneous cellular networks," *IEEE Trans. Wireless Commun.*, vol. 19, no. 4, pp. 2411–2426, Apr. 2020.
- [6] F. Cheng, S. Zhang, Z. Li, Y. Chen, N. Zhao, F. R. Yu, and V. C. M. Leung, "UAV trajectory optimization for data offloading at the edge of multiple cells," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 6732–6736, Jul. 2018.

- [7] J. Lyu, Y. Zeng, and R. Zhang, "UAV-aided offloading for cellular hotspot," *IEEE Trans. Wireless Commun.*, vol. 17, no. 6, pp. 3988–4001, Jun. 2018.
- [8] M. Hua, Y. Wang, C. Li, Y. Huang, and L. Yang, "Energy-efficient optimization for UAV-aided cellular offloading," *IEEE Wireless Commun. Lett.*, vol. 8, no. 3, pp. 769–772, Jun. 2019.
- [9] N. Zhao, P. Fan, and Y. Cheng, "Dynamic contract incentives mechanism for traffic offloading in multi-UAV networks," *Wireless Commun. Mobile Comput.*, vol. 2020, pp. 1–11, May 2020.
- [10] S. Sharafeddine and R. Islambouli, "On-demand deployment of multiple aerial base stations for traffic offloading and network recovery," *Comput. Netw.*, vol. 156, pp. 52–61, Jun. 2019.
- [11] H. Huang, C. Huang, and D. Ma, "A method for deploying the minimal number of UAV base stations in cellular networks," *IEEE/CAA J. Autom. Sinica*, vol. 7, no. 2, pp. 559–567, Mar. 2020.
- [12] S. A. R. Naqvi, S. A. Hassan, H. Pervaiz, and Q. Ni, "Drone-aided communication as a key enabler for 5G and resilient public safety networks," *IEEE Commun. Mag.*, vol. 56, no. 1, pp. 36–42, Jan. 2018.
- [13] M. A. Ali and A. Jamalipour, "UAV-aided cellular operation by user offloading," *IEEE Internet Things J.*, vol. 8, no. 12, pp. 9855–9864, Jun. 2021.
- [14] Q. Zhang, W. Saad, M. Bennis, X. Lu, M. Debbah, and W. Zuo, "Predictive deployment of UAV base stations in wireless networks: Machine learning meets contract theory," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 637–652, Jan. 2021.
- [15] M. A. Ali and A. Jamalipour, "UAV placement and power allocation in uplink and downlink operations of cellular network," *IEEE Trans. Commun.*, vol. 68, no. 7, pp. 4383–4393, Jul. 2020.
- [16] R. Duan, J. Wang, C. Jiang, H. Yao, Y. Ren, and Y. Qian, "Resource allocation for multi-UAV aided IoT NOMA uplink transmission systems," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 7025–7037, Aug. 2019.
- [17] A. Fahim and Y. Gadallah, "An optimized LTE-based technique for drone base station dynamic 3D placement and resource allocation in delay-sensitive M2M networks," *IEEE Trans. Mobile Comput.*, vol. 22, no. 2, pp. 732–743, Feb. 2023, doi: 10.1109/TMC.2021.3089329.
- [18] C. You and R. Zhang, "3D trajectory optimization in Rician fading for UAV-enabled data harvesting," *IEEE Trans. Wireless Commun.*, vol. 18, no. 6, pp. 3192–3207, Jun. 2019.
- [19] A. Mahmood, M. L. M. Kiah, M. R. Z'Abu, A. N. Qureshi, M. S. S. Kassim, Z. H. A. Hasan, J. Kakarla, I. S. Amiri, and S. R. Azzuhri, "Capacity and frequency optimization of wireless backhaul network using traffic forecasting," *IEEE Access*, vol. 8, pp. 23264–23276, 2020.
- [20] D. S. Chen, R. G. Batson, and Y. Dang, "Branch-and-cut approach," in *Applied Integer Programming: Modeling and Solution*, 1st ed. Hoboken, NJ, USA: Wiley, 2010, ch. 11, pp. 272–280.
- [21] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. Int. Conf. Neural Netw. (ICNN)*, Perth, WA, Australia, Dec. 1995, pp. 1942–1948.
- [22] W. I. Zangwill, "Non-linear programming via penalty functions," *Manage. Sci.*, vol. 13, no. 5, pp. 344–358, Jan. 1967.
- [23] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, nos. 3–4, pp. 279–292, 1992.
- [24] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. Cambridge, MA, USA: MIT Press, 2018.
- [25] M. Tokic, "Adaptive ϵ -greedy exploration in reinforcement learning based on value differences," in *Proc. Annu. Conf. Artif. Intell.*, 2010, pp. 203–210.
- [26] F. Hafiz and A. Abdennour, "Particle swarm algorithm variants for the quadratic assignment problems—A probabilistic learning approach," *Expert Syst. Appl.*, vol. 44, pp. 413–431, Feb. 2016.
- [27] N. Afrin, J. Brown, and J. Y. Khan, "Performance analysis of an enhanced delay sensitive LTE uplink scheduler for M2M traffic," in *Proc. Australas. Telecommun. Netw. Appl. Conf. (ATNAC)*, Christchurch, New Zealand, Nov. 2013, pp. 154–159.
- [28] E. Dahlman, S. Parkvall, and J. Skold, "5G standardization," in *5G NR: The Next Generation Wireless Access Technology*, New York, NY, USA: Academic, 2018, ch. 2, pp. 7–25.



AHMED FAHIM MOSTAFA (Graduate Student Member, IEEE) is currently pursuing the Ph.D. degree in the area of automatic control theory and applications. His research focuses on networked control systems such as aerial-ground decentralized coordination. He aims to investigate stochastic models in dynamic system analysis and optimal control designs and integrate smart technologies such as computational vision, active learning, and optimization algorithms.



MOHAMED ABDEL-KADER received the B.Sc. degree (Hons.) in communication and electronics engineering from Alexandria University, Egypt, in 2019, where he is currently pursuing the M.Sc. degree. He is a Teaching and Research Assistant with Alexandria University and The American University in Cairo. His research interests include wireless communications and machine learning algorithms in communications.



YASSER GADALLAH (Senior Member, IEEE) is currently a Professor with the Department of Electronics and Communications Engineering, The American University in Cairo (AUC), New Cairo, Egypt. Before he joined AUC, he was an Associate Professor with the Electronics and Communications Department, Misr International University, Cairo. He was an Assistant Professor of networking systems with the College of Information Technology, UAE University. Prior to joining UAE

University, he was a Research Scientist in the area of wireless sensor networks with the Communications Research Centre, Ottawa, ON, Canada. He was with the networking and telecommunications industry. He was a Software Development Manager with Cisco Systems for several years, where he was involved in numerous research and development projects, which resulted in the introduction of several high-profile Cisco products. He was with Nortel Networks as the Software Project Leader and prior to that as a Software Designer. He has authored/coauthored several technical and scientific publications in the networking, communications, and IT fields. His current research interests include the Internet of Things issues, machine-to-machine communications, wireless sensor networks, mobile ad-hoc networks, and broadband wireless access networks. He was an organizer, a technical program committee member, and a reviewer of several international conferences and journals.



OMAR ELAYAT is currently pursuing the B.Sc. degree in electronics and communications engineering with a minor in computer science with The American University in Cairo. Early this year, he joined EPFL Processor Architecture Laboratory as an Intern where he worked on elastic and arithmetic circuits verification. His research interests include wireless communications, computer and processor architecture, and electronic design automation.

...