Data–Driven Image Resolution and Uplink Power Control for Mobile Augmented Reality Applications

Andrea WRONA[1,*], Danilo MENEGATTI[1], Emanuele DE SANTIS[1], and Andrea TORTORELLI[2]

***Abstract*— In the context of Mobile Augmented Reality, satisfying the challenging users' requirements about Quality of Service and Quality of Experience is not an easy task due to the limited computing capabilities of mobile devices, and the rapid, free movement of users within the environment. To deal with these issues, graphical computations are typically offloaded from mobile devices to edge servers. While traditional offloading strategies rely on static optimization or heuristics, this work proposes a multi–input data–driven dynamic control of uplink power and image compression rate, introducing a Policy Broadcasting Deep Reinforcement Learning approach, based on the Deep Deterministic Policy Gradient algorithm. The proposed solution is aimed at matching the challenging Quality of Service constraints, in terms of maximum round–trip latency and minimum resolution accuracy, while minimizing the energy consumption. Simulations show the effectiveness and scalability of the proposed approach for real–time applications.**

***Index Terms*— Augmented Reality, Power Control, Deep Reinforcement Learning.**

I. INTRODUCTION

The rapid evolution of wireless communication technologies has driven a paradigm shift in the design and deployment of wireless networks. One of the key developments has been the emergence of Heterogeneous Networks (HN) characterized by (i) the integration of wireless communication systems belonging to different Radio Access Technologies (RATs) and (ii) the coexistence of various network elements, such as macrocells, microcells, picocells, and femtocells [1]. The pivotal enabler of the HN deployment is 5G, the fifth generation of wireless communication technology, which is able to address the challenges posed by the increasing demand for higher data rates, low latency, massive device connectivity, and diverse user application requirements [2]. The evolution of wireless communication technologies is accompanied by the widespread diffusion of mobile devices for daily use, such as tablets and smartphones. Over the years, these digital devices made available to users have become increasingly sophisticated and are equipped with powerful processors (CPUs), very high resolution screens, video cameras and sensors of different types. This allows smartphones and tablets to run very complex applications and functions aimed at teleworking, virtualization and entertainment [3]. In the latter domain, one of the key emerging technology is Augmented Reality (AR), which has become an integral part of education and healthcare [4], as well as industry 4.0 [5].

In general, AR refers to the integration of computer-generated sensory information, such as visuals, sounds, and haptic feedback, into the real-world environment through user devices. This technology extends human perception by overlaying virtual objects, information, or experiences onto our physical surroundings, creating an immersive and interactive user experience [6]. Mobile AR (MAR) exploits this concept running the applications on a mobile device (MD). In this context, the inherent constraints posed by limited computational resources and battery capacity of MDs make the accomplishment of object analytics while adhering to stringent low-latency requirements a challenging endeavor [7].

Since MDs–like smartphones or headsets usually cannot mount built–in GPUs, they need to perform computation offloading towards network cloud servers, which arrange object detection or creation of tasks, sending back the augmented frames to MDs [8]. This operation usually is characterized by high latency, not allowing to enjoy real–time AR applications, especially in situations where the cellular network is overcrowded [9]. To address this challenge, some works integrated various different types of mobile networks to get the best performances of each [10], while in others the concept of Mobile Edge Computing (MEC), standardized by the European Telecommunications Standards Institute (ETSI) [11], emerges as a promising solution. The MEC offers computational resources to MDs at the network edge, positioned physically closer to MDs than conventional cloud servers. This allows to reduce the network communication latency, enabling real–time MAR application.

Recent research endeavors have focused on efficient computation offloading to MEC servers, taking into account latency, bandwidth, and computational resource limitations as Key Performance Indicators (KPIs) [12]–[14]. These works have been enriched considering also accuracy and resolution control [15], and MD's cache management [16] which constitute the distinct attributes of object analytics within the realm of MAR. While the above papers adopt a static nonlinear optimization, several other works consider the problem of uplink power management from a control systems

[1]Department of Computer, Control and Management Engineering "Antonio Ruberti" – Sapienza University of Rome. Email: {wrona, menegatti}@diag.uniroma1.it

[2]Department of Theoretical and Applied Sciences, eCampus University, Via Isimbardi 10, 22060 Novedrate, Italy. Email: andrea.tortorelli@uniecampus.it

*Corresponding author. Email: wrona@diag.uniroma1.it

Part of the reported research activities has been supported by the VADUS project - grant agreement No 4000132720 - (https://business.esa.int/projects/vadus), which received funding from the European Space Agency (ESA) in relation with the call ARTES 20 by ESA ITT AO/1-10065/19/NL/AF *Applications integrating space asset(s) and 5G networks in L'Aquila, the Abruzzo region, Roma Capitale and Municipality of Torino (L'ART)*. This work only reflects the authors' view.

perspective. As an example, in [17], the authors demonstrate the internal stability of standard power allocation dynamic procedures. More recent works (e.g., [18]–[22]), instead, have focused the attention on data–driven control methodologies, namely Reinforcement Learning (RL) [23]. A multi–agent Q-Learning formulation has been described in [24] where the authors consider a multitude of users requesting streaming services at the same time. In [25], the authors exploit Deep RL (DRL) to deal not only with power control, but also with user association in a HN environment.

However, none of the mentioned works have considered, jointly, the problem of dynamic uplink power and image resolution control for MAR applications in a MEC context.

This research advances the state of the art introducing and AI-based offloading policy for MAR applications capable of handling the rapid variability of the positioning data available to the MD of a user, which moves frequently and randomly in the environment. The main contributions of this work are the following:

- Simultaneous data-driven control of uplink power allocation and image compression by means of DRL, exploiting users' spatial information and implementing continuous control actions.
- Definition of a Policy Broadcasting approach, through which the training phase of the RL procedure is limited to one agent only, whereas the execution is broadcast to multiple agents.
- Novel formulation of reward function, leveraging on the trade-off between image accuracy and energy consumption, while guaranteeing operational constraints.

The subsequent sections of the paper unfolds as follows. Section II recalls the theoretical foundations of RL and details the proposed Policy Broadcasting learning/execution approach, whereas Section III presents an extensive modeling of the MAR wireless network system. Section IV presents the simulations and results, highlighting the effectiveness of the proposed control architecture. Eventually, in Section V, the key findings and practical implications of this study are synthesized, and potential avenues for future research are outlined.

II. Policy Broadcasting Reinforcement Learning

Decision-making problems within the RL paradigm can be formally modeled as Markov Decision Processes (MDPs) through the tuple $\langle \mathcal{S}, \mathcal{A}, P, R, \gamma \rangle$, [26] where:

- $\mathcal{S}$ denotes the finite state space, encompassing all configurations of the environment,
- $\mathcal{A}$ represents the finite collection of actions available to the agent in each state,
- $P(s, a, s')$ specifies the probability of transitioning from state s to state s' upon taking action a,
- $\gamma \in [0,\ 1)$ represents the discount factor, reflecting the preference of the agent for immediate rewards ($\gamma \approx 0$) over future ones ($\gamma \approx 1$).

The overall objective for a RL agent is to learn a policy $\pi : S \rightarrow A$ that maps states to actions in a way that maximizes the expected discounted total reward

$$J^{\pi,\chi} = \mathbb{E}_{\chi}\{V^{\pi}(s)\} = \sum_{s \in S} \chi(s) V^{\pi}(s), \tag{1}$$

where $\mathbb{E}_{\chi}\cdot$ denotes the expected value under the initial state distribution χ, and $V^{\pi}(s)$ represents the state-value function

$$V_{\pi}(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R_t | s_0 = s\right]. \tag{2}$$

The agent learns a policy through an interative procedure, balancing between exploration (experimenting with new, random actions) and exploitations (selecting action that are currently considered optimal). Upon completion of the training process, the agent implements the learned policy π^* within the environment during the execution phase.

In this work, a Policy Broadcasting Reinforcement Learning (PBRL) approach is introduced. Under the hypothesis that agents possess a homogeneous nature, characterized by identical capabilities and characteristics, a policy broadcasting mechanism is designed, involving the training of a single agent, as detailed in Algorithm 1, followed by the broadcasting of the learned policy to the entire group of agents during the execution phase, where agents simultaneously perform their action within the same environment, as illustrated in Algorithm 2. The core of the proposed approach lies in sharing the knowledge of the learned policy, thereby addressing one of the typical drawbacks of RL algorithms in multi-agent scenarios which is represented by the required training times. When the agents share an homogeneous nature, it is possible to save computational time by training only a single agent instead of dedicating a local training for each of the agents.

Algorithm 1 PBRL training phase

```
procedure SINGLE AGENT LEARNING
    Initialize policy π randomly
    for episode ← 1 to end do
        Initialize state s
        while s is not terminal do
            Take action a according to exploration
                strategy
            Observe reward r and new state s'
            Update internal state and policy π
        end while
    end for
end procedure
```

In this work, the PBRL training phase employs the Deep Deterministic Policy Gradient (DDPG) algorithm [27], which is particularly suitable at addressing problems characterized by continuous actions and state spaces. This algorithm utilizes an actor-critic neural network architecture, where the actor network approximate the policy, while the critic

Algorithm 2 PBRL deployment

procedure MULTI AGENT EXECUTION
  Initialize vector state $\boldsymbol{s} = [s_1, \ldots, s_N]$
  while $\boldsymbol{s}$ is not terminal **do**
    for agent $i \leftarrow 1$ to end **do**
      Observe state s_i
      Select action a_i using policy π
      Receive reward r_i and new state $s_{i,'}$
      Update internal state
    end for
  end while
end procedure

network the state-value function (2), estimating the expected cumulative rewards associated with taking a specific action in a given state. To enhance the stability of the learning process, DDPG introduces an experience replay buffer to store past experiences and a gaussian noise over the actions to explore the environment, thereby facilitating an efficient exploration of the environment [27].

The dynamic evolution of the MAR system, together with the underlying MDP formulation modeling choices, will be detailed in the next section.

III. MAR System Modeling

Let us consider a single–cell MEC system, consisting of an edge server wired (e.g., by means of optical fiber) to a 5G Base Station (BS), whose fixed 2D position is $\boldsymbol{x}^{\text{BS}} = \begin{bmatrix} x_1^{\text{BS}} & x_2^{\text{BS}} \end{bmatrix}^T$. The server hosts a virtual machine with guaranteed GPU capabilities for object detection and image augmentation purposes. The MDs are single–antenna devices, sparsely scattered within the coverage area of the BS and each one of them occupies a non–overlapping bandwidth W of the BS, running an active session of a MAR application. The latter aim is to capture images with the smartphone camera, send them to the edge server, thus receiving back the new images with modified and augmented information. The MDs have the same technological characteristics and cannot communicate one with each other.

The system scenario, depicted in Figure 1, is particularly suitable for applications involving a multitude of users moving with a defined area, such as an archaeological park. These users are connected to the same BS which offloads computational tasks to an edge server. The homogeneous nature of MDs and their inability to communicate with each other makes the system architecture ideal for scenarios requiring synchrorized and independent user interactions.

Consider now a single MD, and suppose its owner is allowed to move only within the coverage area of the BS with speed $\boldsymbol{v} = \begin{bmatrix} v_1 & v_2 \end{bmatrix}^T$ while running its session, thus moving according to the first order differential equation

$$\dot{\boldsymbol{x}} = \boldsymbol{v}, \tag{3}$$

where $\boldsymbol{x} = \begin{bmatrix} x_1 & x_2 \end{bmatrix}^T$. Moreover, suppose that the user can estimate, at any given time, its position $\boldsymbol{x}$ through a GPS–like navigation system.

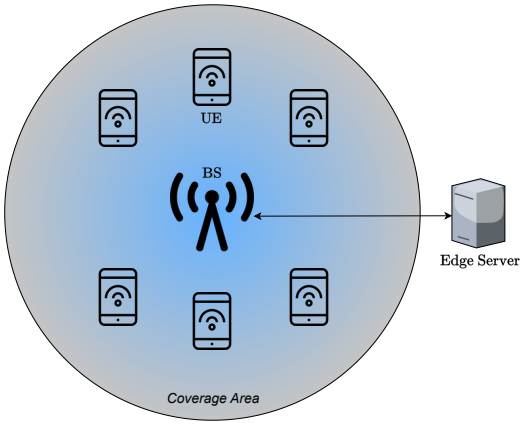

Fig. 1: System scenario.

Let p be the transmission power of the MD, then the Signal to Noise ratio (SNR) can be defined as:

$$\text{SNR} = \frac{p\, G_{\text{MD}}\, G_{\text{BS}}\, h}{N_0} \tag{4}$$

where G_{MD} and G_{BS} are respectively the transmitter and receiver gains, N_0 is the noise power spectral density, and h is the attenuation of radio energy, modeled following the free-space path loss propagation model as

$$h = \left(\frac{\lambda}{4\pi d}\right)^2, \tag{5}$$

where λ is the signal wavelength, and $d = ||\boldsymbol{x} - \boldsymbol{x}^{\text{BS}}||$ is the Euclidean distance between the MD and the BS.

As a result, remembering the Shannon–Hartley theorem, the uplink transmission rate cannot exceed the amount

$$R(p) = W \log_2\big(1 + \text{SNR}\big). \tag{6}$$

All the images captured by a MD undergo a compression process before being sent to the edge server which ultimately processes them.

We assume that the MD has compression capacity V, and captures K-pixel raw images which are compressed to s pixels, each one containing σ bits of information.

The accuracy of the image processing task can be computed as follows:

$$A(s) = 1 - 1.578 \exp(-6.5 \times 10^{-3} s^{1/2}). \tag{7}$$

Assuming that the server is capable of providing a minimal computation speed of f (TFLOPS), the processing workload is given by:

$$\psi(s) = 7 \times 10^{-10} s^{3/2} + 0.083. \tag{8}$$

The latency occurring for a round trip path can be divided into four components, defined as follows:

- $L_{\text{IC}} = \sigma K / V$ is the latency related to image compression at the i-th MD level;
- $L_{\text{T}} = \sigma s / R(p)$ is the latency due to the 5G connection between the MD and the BS;

- $L_{\text{ES}}^{\text{BS}}$ is the latency between the BS and the edge server. This quantity can be neglected compared to the other terms;
- $L_{\text{ES}} = \psi/f$ is the latency related to processing workload at the edge server level.

Hence, the total cumulative latency is

$$L(p, s) = L_{\text{IC}} + L_{\text{T}} + L_{\text{ES}}. \tag{9}$$

Eventually, energy consumption arguments at MD level are considered. Let $E(p, s)$ be the total energy consumed by the MD. It is made by two elements:

- $E_{\text{IC}} = \varepsilon\sigma(K - s)$ is the energy spent for image compression, with ε being the energy consumption for compressing 1 bit of data;
- $E_{\text{T}} = pL_{\text{T}}$ is the energy related to data transmission.

In what follows the mathematical formulation presented above is mapped on a MDP, in order to exploit the DRL control framework.

The state of the agent is given by

$$\mathcal{S} = \langle d \rangle, \tag{10}$$

i.e., it corresponds to the relative distance from the BS. This modeling choice, together with the assumption of the existence of non–overlapping bandwidths, allows to apply the PBRL approach, thus training only a single agent, and then deploying its policy to multiple MDs.

The action space

$$\mathcal{A} = \langle p, s \rangle \tag{11}$$

is composed of the transmitting power $p \in [p_{\text{min}}, p_{\text{max}}]$, and the image resolution after compression $s(t) \in [s_{\text{min}}, s_{\text{max}}]$.

The goal of the agent is to solve the power allocation and image processing problems in such a way that QoS and QoE KPIs are satisfied. To this end, let μ and δ be the maximum sustainable latency and the minimum feasible accuracy, respectively. Hence, it is possible to define the following two usage constraints:

$$\begin{aligned} L(p, s) &\leq \mu \\ A(s) &\geq \delta. \end{aligned} \tag{12}$$

In addition, for energy saving purposes, it is desirable to govern image compression and data transmission spending the least possible amount of energy. As a result, the reward is defined as follows:

$$r = -\big(E(p, s) + \alpha Q(s)\big) - \text{sgn}\big(L(p, s) - \mu\big) + \text{sgn}\big(A(s) - \delta\big), \tag{13}$$

where $Q(s) = 1 - A(s)$ is the accuracy loss, α is a weight factor balancing energy minimization and accuracy loss, and $\text{sgn}(\cdot)$ is the sign function.

This setup ensures efficient processing and low-latency communication, enabling real-time MAR services, as detailed in the next section.

TABLE I: System Parameters' Numerical Values

Parameter	Value	Unit
f	0.5	TFLOPS
G_{BS}	15	-
G_{MD}	3	-
λ	5×10^{-3}	m
K	4×10^{5}	pixel
N	3.98×10^{-16}	W
p_{max}	3.5	W
p_{min}	0	W
s_{max}	4×10^{5}	pixel
s_{min}	0	pixel
σ	24	bit/pixel
V	1×10^{8}	bit/s
W	1×10^{6}	Hz

IV. Simulations and Results

In order to validate the effectiveness of our proposed approach, we consider a scenario of 10 MDs moving inside a square having a side $l = 300$ [m], with the BS located at the center of the square. This scenario is typical of archaeological parks that offer AR services: each tourist is equipped with a MD and is free to move within a limited area.

The numerical constant parameters of the MAR system are summarized in Table I.

As for the users' QoS and QoE, the constraints parameters have been set such as $\delta = 0.85$ [%], $\mu = 0.85$ [s], while the cost parameter in (13) is fixed as $\alpha = 4$ in order to place more emphasis on accuracy over energy consumption.

A single agent is trained for $E_{\text{ep}} = 200$ episodes, each one lasting $T = 30$ [s]. The integration of the system dynamics, as described in equation (3), is performed using the Runge–Kutta 4th order method with a time step $dt = 0.1$ [s].

At the beginning of each episode the agent is located at a random distance d_0 from the BS and it moves with random 2D velocity $\boldsymbol{v} \in [0\ 1.11]$ [m/s], to mimic a typical human walking behavior. The episode ends if $A(s)$ or $L(p, s)$ fall below the constraints or if the time limit T is reached.

During the evaluation procedure, all agents are homogeneous as they correspond to MDs connected to the same BS. The proposed policy broadcasting mechanism enables the dissemination of the learned policy to all MDs which, despite differing in terms of initial positions and velocities $\boldsymbol{v}_i(t)$, share the same task, i.e., to develop an offloading strategy related to uplink power control and image compression rate. This strategy must simultaneously adhere to constraints on round-trip latency and minimum resolution accuracy, while minimizing energy consumption to extend MDs' battery life. The system dynamics evolve for the same amount of steps per episode as in the training phase.

The DDPG hyperparameters have been carefully selected as follows: the actor learning rate $\beta_{\text{a}} = 1 \times 10^{-3}$, the critic learning rate $\beta_{\text{c}} = 2 \times 10^{-3}$, the discount factor $\gamma = 0.9$, and the memory capacity is selected as 1000 transitions. Both the actor and critic share the same neural architecture composed of three layers, the first two of 512 and 128 neurons with `ReLU` activation function, and the last one of one neuron

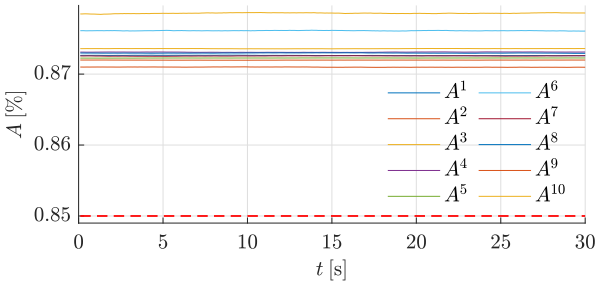

Fig. 2: Accuracy $A_i(s)$ evaluation for each MD $i = 1, \ldots, 10$.

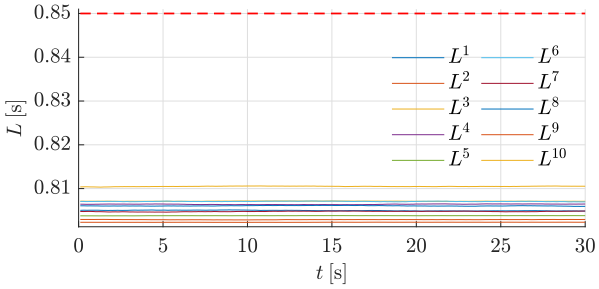

Fig. 3: Latency $L_i(s)$ evaluation for each MD $i = 1, \ldots, 10$.

only for the critic, and two neurons (one per each action) for the actor, with `tanh` as activation function.

The training process is carried out in almost 1 hour using Tensorflow and Keras on an Intel Xeon platform with 16 GB of RAM and Nvidia Tesla T4.

Figures 2-3 show the effectiveness of the proposed approach in terms of meeting the accuracy $A(s)$ and latency $L(p, s)$ requirements respectively. While the dotted red line in both figures indicates the KPI-correspondent value to be satisfied, it is evident that all MDs, while moving freely within the squared area, are capable of selecting the correct action pair $\langle p_i(t), s_i(t) \rangle$ at each instant, ensurig continuity of service at every time instant, achieving a resolution accuracy of approximately 87%, consistently exceeding the minimum requirement, and a service latency of approximately 0.80 [s], consistently being compliant with the corresponding constraint. In particular, the highest accuracy value is achieved by the tenth MD, $A_{10,\text{max}} = 0.8787$ [%], while the second agent achieves the lowest latency, $L_{2,\text{min}} = 0.8023$ [s].

As the initial positions and velocities of each MD are chosen at random, and their owners move freely within the squared area at dynamically changing radom speeds, their positions vary over time. Therefore, it is of interest to examine their corresponding mean values, computed over the evaluation steps, for distance $\bar{d}_i$ from the BS, transmission power $\bar{p}_i$, and image resolution $\bar{s}_i$, as illustrated in Table II. It is worth noting that MDs increase their uplink power and decrease image quality as users move farther from the BS, reflecting real-life expectations, thus demonstrating the effectiveness of the proposed approach.

Focusing on the most distant MD from the BS, Figure 4 illustrates the behaviour of the second MD, which we reckon is similar to that of the others. It starts from a distance

TABLE II: Distance, transmission power and image resolution mean values for all MDs.

Index	$\bar{d}_i$	$\bar{p}_i$	$\bar{s}_i$
1	129.05	205.40	1.50×10^5
2	152.78	277.10	1.48×10^5
3	118.04	174.69	1.51×10^5
4	124.03	190.35	1.50×10^5
5	133.35	219.02	1.49×10^5
6	96.94	131.79	1.53×10^5
7	129.74	207.56	1.50×10^5
8	125.37	194.27	1.50×10^5
9	141.66	246.69	1.49×10^5
10	87.08	114.09	1.56×10^5

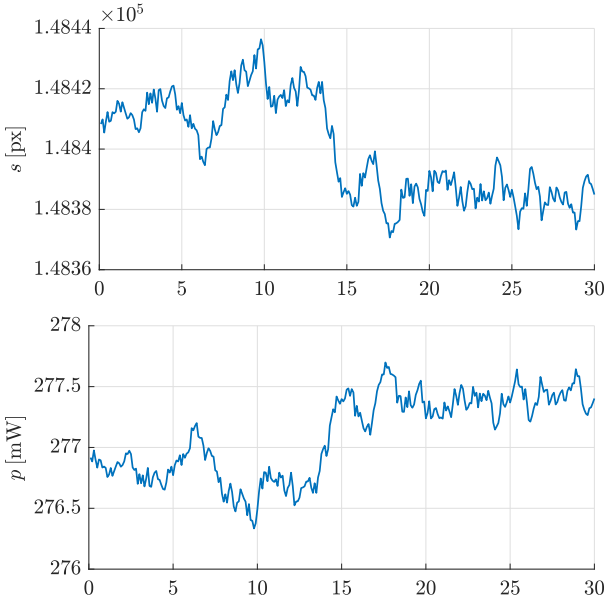

Fig. 4: Image resolution $s_2(t)$ (upper), and transmission power $p_2(t)$ (lower) related to the second MD.

of 152.69 [m] from the BS, and it dynamically adjust its transmission power $p_2(t)$ and image resolution $s_2(t)$ to ensure QoS to its user. As time goes by, small variations in distance are reflected in corresponding variations in transmission power and image resolution. It is worth mentioning that as the MD reduces its distance from the BS, less transmission power is required, resulting in an increase in image resolution to meet accuracy requirements. Conversely, starting from 13 [s] onward, transmission power increases due to the MD moving father from the BS, with image resolution decreasing to continue to satisfy latency requirements.

Similar considerations apply to energy consumption, as it is influenced by both energy expended for compression and energy required for data transmission. As observed from the mean energy consumption $\bar{E}_i(p, s)$ values reported in Table III, the greater the distance of the MD from the BS, the higher the energy consumption, with the data transmission component contributing more significantly than the compression component. Conversely, for MDs closer to the BS, the compression component outweighs the data transmis-

TABLE III: Average energy consumption $\bar{E}_i(p,s)$ for all the MDs (measurement unity Joule).

Index	$\bar{E}_i(p,s)$	Index	$\bar{E}_i(p,s)$
1	0.21	6	0.18
2	0.25	7	0.22
3	0.20	8	0.21
4	0.21	9	0.23
5	0.22	10	0.17

sion one, resulting in lower overall energy consumption. In particular, the second MD, which is the farthest from the BS, exhibits the highest mean energy consumption, whereas the tenth MD, being the closest to the BS, shows the lowest consumption value.

V. Conclusions

This paper has presented a novel DRL-based paradigm to cope with the computation offloading problem for a MAR application in the MEC scenario, in presence of QoS and QoE requirements and rapidly moving users. Leveraging on the PBRL control scheme and on the DDPG algorithm, the data knowledge of MDs' distance from the BS turns out to be sufficient to satisfy latency and accuracy requirements in scenarios where all MDs move within the BS's coverage area with non–controllable velocity. The introduction of an energy consumption term within the reward function extends MDs' battery life, thus guaranteeing continuity of service over a long time.

Future works shall involve the introduction of more complex connectivity protocols, together with a demonstration of the proposed method in a real network scenario using existing MAR devices.

ACKNOWLEDGMENT

The authors would like to thank the VADUS consortium for the fruitful discussions which provided significant inputs for this work.

References

[1] Y. Xu, G. Gui, H. Gacanin, and F. Adachi, "A survey on resource allocation for 5g heterogeneous networks: Current research, future trends, and challenges," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 2, pp. 668–695, 2021.

[2] M. Agiwal, A. Roy, and N. Saxena, "Next generation 5g wireless networks: A comprehensive survey," *IEEE communications surveys & tutorials*, vol. 18, no. 3, pp. 1617–1655, 2016.

[3] S. Melumad and M. T. Pham, "The smartphone as a pacifying technology," *Journal of Consumer Research*, vol. 47, no. 2, pp. 237–255, 2020.

[4] Y. Chen, Q. Wang, H. Chen, X. Song, H. Tang, and M. Tian, "An overview of augmented reality technology," *Journal of Physics: Conference Series*, vol. 1237, no. 2, p. 022082, 2019.

[5] J. Husár and L. Knapčíková, "Implementation of augmented reality in smart engineering manufacturing: literature review," *Mobile Networks and Applications*, vol. 29, no. 1, pp. 119–132, 2024.

[6] J. Carmigniani, B. Furht, M. Anisetti, P. Ceravolo, E. Damiani, and M. Ivkovic, "Augmented reality technologies, systems and applications," *Multimedia tools and applications*, vol. 51, pp. 341–377, 2011.

[7] L. Naveen, M. I. Khan, M. A. Saleh, and B. Campus, "The influence of mobile augmented reality on consumer behavior: Insights into affective, cognitive, and behavioral responses," *Computers in Human Behavior*, vol. 165, p. 108558, 2025.

[8] K. Kumar, J. Liu, Y.-H. Lu, and B. Bhargava, "A survey of computation offloading for mobile systems," *Mobile networks and Applications*, vol. 18, pp. 129–140, 2013.

[9] A. Pimpinella, F. D. Giusto, A. E. C. Redondi, L. Venturini, and A. Pavon, "Forecasting busy-hour downlink traffic in cellular networks," in *ICC 2022 - IEEE International Conference on Communications*, p. 4336–4341, IEEE, May 2022.

[10] N. Cassiau, I. Kim, E. C. Strinati, G. Noh, A. Pietrabissa, F. Arnal, G. Casati, T. Choi, Y.-J. Choi, H. Chung, S. Colombero, P. D. Zotto, E. D. Santis, J.-B. Dore, A. Giuseppi, J.-M. Houssin, J. Kim, M. Laugeois, F. Pigni, X. Popon, L. Raschkowski, M. Thary, and S. H. Won, "5g-allstar: Beyond 5g satellite-terrestrial multi-connectivity," in *2022 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*, p. 148–153, IEEE, June 2022.

[11] F. Giust, X. Costa-Perez, and A. Reznik, "Multi-access edge computing: An overview of etsi mec isg," *IEEE 5G Tech Focus*, vol. 1, no. 4, p. 4, 2017.

[12] Q. Liu, S. Huang, J. Opadere, and T. Han, "An edge network orchestrator for mobile augmented reality," in *IEEE INFOCOM 2018-IEEE conference on computer communications*, pp. 756–764, IEEE, 2018.

[13] D. Xu, Q. Li, and H. Zhu, "Energy-saving computation offloading by joint data compression and resource allocation for mobile-edge computing," *IEEE Communications Letters*, vol. 23, no. 4, pp. 704–707, 2019.

[14] P. Y. Zhou, S. Fu, B. Finley, X. Li, S. Tarkoma, J. Kangasharju, M. Ammar, and P. Hui, "5g mec computation handoff for mobile augmented reality," in *2024 IEEE International Conference on Metaverse Computing, Networking, and Applications (MetaCom)*, p. 129–136, IEEE, Aug. 2024.

[15] J. Ahn, J. Lee, S. Yoon, and J. K. Choi, "A novel resolution and power control scheme for energy-efficient mobile augmented reality applications in mobile edge computing," *IEEE Wireless Communications Letters*, vol. 9, no. 6, pp. 750–754, 2019.

[16] Y.-J. Seo, J. Lee, J. Hwang, D. Niyato, H.-S. Park, and J. K. Choi, "A novel joint mobile cache and power management scheme for energy-efficient mobile augmented reality service in mobile edge computing," *IEEE Wireless Communications Letters*, vol. 10, no. 5, pp. 1061–1065, 2021.

[17] F. Gunnarsson and F. Gustafsson, "Power control in wireless communications networks-from a control theory perspective," *IFAC Proceedings Volumes*, vol. 35, no. 1, pp. 183–194, 2002.

[18] A. Anzaldo and A. G. Andrade, "Deep reinforcement learning for power control in multi-tasks wireless cellular networks," in *2022 IEEE International Mediterranean Conference on Communications and Networking (MeditCom)*, pp. 61–65, IEEE, 2022.

[19] T. Zhao, L. He, X. Huang, and F. Li, "Drl-based secure video offloading in mec-enabled iot networks," *IEEE Internet of Things Journal*, vol. 9, no. 19, pp. 18710–18724, 2022.

[20] Y. S. Nasir and D. Guo, "Deep reinforcement learning for joint spectrum and power allocation in cellular networks," in *2021 IEEE Globecom Workshops (GC Wkshps)*, pp. 1–6, IEEE, 2021.

[21] Z. El Jamous, K. Davaslioglu, and Y. E. Sagduyu, "Deep reinforcement learning for power control in next-generation wifi network systems," in *MILCOM 2022-2022 IEEE Military Communications Conference (MILCOM)*, pp. 547–552, IEEE, 2022.

[22] F. H. C. Neto, D. C. Araújo, M. P. Mota, T. F. Maciel, and A. L. de Almeida, "Uplink power control framework based on reinforcement learning for 5g networks," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 6, pp. 5734–5748, 2021.

[23] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep reinforcement learning: A brief survey," *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 26–38, 2017.

[24] S. Dzulkifly, L. Giupponi, F. Said, and M. Dohler, "Decentralized q-learning for uplink power control," in *2015 IEEE 20th International Workshop on Computer Aided Modelling and Design of Communication Links and Networks (CAMAD)*, pp. 54–58, IEEE, 2015.

[25] H. Ding, F. Zhao, J. Tian, D. Li, and H. Zhang, "A deep reinforcement learning for user association and power control in heterogeneous networks," *Ad Hoc Networks*, vol. 102, p. 102069, 2020.

[26] R. S. Sutton, A. G. Barto, *et al.*, *Reinforcement learning: An introduction*, vol. 1. MIT press Cambridge, 1998.

[27] Y. Hou, L. Liu, Q. Wei, X. Xu, and C. Chen, "A novel ddpg method with prioritized experience replay," in *2017 IEEE international conference on systems, man, and cybernetics (SMC)*, pp. 316–321, IEEE, 2017.