EAPO-NET: Adaptive Energy-Aware Profit Optimization for Next-Generation RANs

André Cédric Bessala, Guilherme Iecker Ricardo and Gentian Jakllari
Univ Toulouse, Toulouse INP, CNRS, IRIT, Toulouse, France
Email: {andrecedric.bessala, guilherme.ieckerricardo, gentian.jakllari}@toulouse-inp.fr

***Abstract*—The large-scale deployment of 5G networks has made energy-efficient operation of radio access networks a critical challenge, requiring the joint optimization of energy consumption, revenue, and quality-of-service compliance under dynamic traffic conditions. In this paper, we propose EAPO-NET, a Multi-Agent Reinforcement Learning (MARL) framework that optimizes these objectives across the Open Radio Access Network (O-RAN) control hierarchy. EAPO-NET employs a heterogeneous multi-agent architecture that aligns learning agents with radio unit (RU) level scheduling, admission control, and node-level energy management functions. Leveraging centralized training with decentralized execution (CTDE) and Value Decomposition Networks, the framework enables fully decentralized inference without inter-agent communication, ensuring scalability and practical deployment. Simulation results show that EAPO-NET learns near-optimal control policies that simultaneously reduce energy consumption, maximize network profit, and satisfy quality-of-service requirements, while achieving optimization performance comparable to integer linear program-based solutions with significantly greater scalability and adaptability.**

***Index Terms*—O-RAN, energy efficiency, MARL, RU-level scheduling, admission control.**

I. INTRODUCTION

The rapid growth of mobile data traffic and heterogeneous service demands has accelerated large-scale 5G deployment, with more than 3.5 million base stations (BSs) already operating worldwide [1]. While densification improves capacity and coverage, it significantly increases energy consumption. In particular, Radio Access Network (RAN) infrastructure accounts for most network power usage, with Radio Units (RUs) contributing up to 80% due to continuous physical-layer operations such as transmission, reception, and signal processing [1], [2]. As 5G evolves toward wider bandwidths and denser deployments, improving RU energy efficiency becomes critical. The disaggregated and virtualized Open Radio Access Network architecture enhances the flexibility and programmability of RAN resource management [3], [4]. By separating hardware and software and introducing entities such as the Near-Real-Time RAN Intelligent Controller (Near-RT RIC) and O-Cloud, O-RAN enables dynamic orchestration of radio and compute resources [4]. Although not originally designed for energy efficiency, its flexibility enables energy-aware RU operation and adaptive DU/CU placement, particularly for heterogeneous services with diverse quality-of-service (QoS) requirements [5].

In this paper, we propose a joint optimization framework for RU sleep mode, UE admission control, and DU/CU VNF placement in O-RAN. The objective maximizes net profit defined as UE revenue minus the energy cost of active RUs, nodes, and VNFs. This captures the trade-off between admitting more users and reducing energy consumption via sleep modes, while satisfying heterogeneous QoS constraints. We propose EAPO-NET, a MARL-based framework aligned with O-RAN, implementing Near-RT RIC xApps for radio control, an admission xApp, and O-Cloud dApps for VNF placement.

EAPO-NET employs a centralized training with decentralized execution (CTDE) architecture from Value Decomposition Networks (VDN) [6] built upon the model-free DQN algorithm. Taking as input a representation of current network conditions, EAPO-NET includes three types of cooperative agents — RU xApp agents, an admission xApp agent, and nodes dApp agents — each maintaining a local policy network that observes only its own domain-specific state and selects actions independently at execution time. During training, each agent type minimizes its own Q-learning loss function using targets computed under the CTDE framework, where joint observations and actions are exploited to coordinate learning without requiring inter-agent communication during execution. This decoupling allows EAPO-NET to scale to real O-RAN deployments while retaining the global optimization perspective needed to balance RU energy efficiency and O-Cloud consolidation against UE QoS demands.

This work advances the state-of-the-art as follows:

1) We formulate the joint RU sleep mode, UE admittance control, and DU/CU VNF placement in a 5G O-RAN system.
2) We model the problem as an MDP and design EAPO-NET, a MARL-based framework deployed as xApps within the Near-RT RIC and as dApps on O-Cloud nodes, enabling real-time optimization under non-stationary network conditions.
3) We evaluate EAPO-NET against a B&C solver, showing near-optimal performance.

II. PROBLEM DEFINITION

We first present the system model and its assumptions, and then formulate the problem as a stochastic optimization problem.

A. System Model

The network infrastructure consists of a set $\mathcal{R}$ of RUs, each composed of a set $\mathcal{B}$ of RBs and serving a set $\mathcal{U}$ of UEs over

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

the same geographical area. The computing infrastructure is modeled by a set $\mathcal{V}$ of cloud nodes hosting 5G VNFs, namely DU and CU functions. Each RU must be associated with exactly one DU and one CU, while cloud nodes may host multiple VNFs subject to resource availability. We assume a fully connected topology between RUs and cloud nodes.

Moreover, any network component (RU or cloud node) can be switched to sleep mode, as described in technical report [7]. We use variables

$$\mathbf{x} \in \{0,1\}^{\mathcal{R}\cup\mathcal{V}}, \tag{1}$$

to indicate that a component $i \in \mathcal{R} \cup \mathcal{V}$ is in active mode, i.e., $x_i = 1$, or in sleep mode, i.e., $x_i = 0$.

1) UE Admission and SLA: UEs may either be admitted into the system or rejected. Admitted UEs must be associated with exactly one RU, while rejected UEs receive no service. The admission and association decisions are controlled through variables

$$\mathbf{y} \in \{0,1\}^{R\times U}, \tag{2}$$

where $y_{r,u} = 1$ if UE u is associated with RU r and $y_{r,u} = 0$, otherwise. Hereafter, we refer to $\mathbf{y}$ as the *admission plan*. In 5G systems, SLAs define the QoS guarantees associated with each UE and directly impact the MNO's economic outcomes. In this work, SLA requirements are modeled solely through downlink throughput demands. Therefore, if UE $u \in \mathcal{U}$ is admitted, then

- its *individual profit* $p_u \in \mathbb{R}_{\geq 0}$ is generated for the MNO in line with the SLA, and
- the *number of RBs* $\boldsymbol{\xi} = \{\xi_{r,u} \in \mathbb{N}_{>0} : r \in \mathcal{R}, u \in \mathcal{U}\}$ is consumed from RU r to satisfy its QoS requirements. Parameter $\xi_{r,u}$ changes with the channel conditions[1].

2) RB States: In this paper, we assume that RUs are equipped with an RB blanking mechanism, as detailed in report [7]. When blanking is used, each RB may be (i) available, if it may be scheduled for transmission during the current frame, or (ii) blanked. The available RBs may be in the following states:

(i) An RB is in *transmitting* state if it is effectively scheduled for data transmission in the current frame. The number of transmitting RBs in RU r is given by

$$g_r^{\text{Tx}}(\mathbf{y}) = \sum_{u\in\mathcal{U}} \xi_{r,u} \cdot y_{r,u}, \tag{3}$$

which depends on the admission plan $\mathbf{y}$.

(ii) An RB is in an idle state if it is available but not transmitting. The number of idle RBs in RU r is

$$g_r^{\text{Id}}(\mathbf{y}) = g_r^{\text{Av}}(\mathbf{y}) - g_r^{\text{Tx}}(\mathbf{y}), \tag{4}$$

where

$$g_r^{\text{Av}}(\mathbf{y}) = \begin{cases} g_r^{\text{Tx}}(\mathbf{y}), & \text{if blanking is adopted,} \\ B, & \text{otherwise.} \end{cases} \tag{5}$$

In other words, if blanking is adopted, then $g_r^{\text{Id}}(\mathbf{y}) = 0$, i.e., there are no idle RBs.

3) VNF Placement: Each cloud node can implement multiple VNFs, and we introduce DU/CU *VNF placement* variables

$$\mathbf{z}^{\text{VNF}} \in \{0,1\}^{R\times V},\ \text{VNF} \in \{\text{DU}, \text{CU}\}, \tag{6}$$

to decide whether RU r's VNF is placed on node i. We denote the set of all VNF placements by $\mathbf{z} = (\mathbf{z}^{\text{DU}}, \mathbf{z}^{\text{CU}})$. The placement of DU and CU functions incurs a computational cost on the cloud nodes. Following a similar approach to prior work (e.g., [8]), we assume that the computing power required to run DU and CU tasks is proportional to the number of transmitting RBs in their associated RU. Therefore, the total computational load (in GOPS) performed by node $i \in \mathcal{V}$ is

$$w_i(\mathbf{y}, \mathbf{z}) = \sum_{r\in\mathcal{R}} g_r^{\text{Tx}}(\mathbf{y}) \cdot \left(\omega^{\text{DU}} \cdot z_{r,i}^{\text{DU}} + \omega^{\text{CU}} \cdot z_{r,i}^{\text{CU}}\right), \tag{7}$$

where $g_r^{\text{Tx}}(\cdot)$ is defined in (3) and parameters ω^{DU} and ω^{CU} are the volumes of computing workload per RB (in GOPS/RB) required for running DU and CU tasks, respectively.

4) Energy-Aware Profit Model: In this paper, we consider the energy expense to run network operations at the RU-level and at the cloud-level. The energy consumed by RU r is

$$e_r^{\text{RU}}(\mathbf{x}, \mathbf{y}) = \left[\epsilon^{\text{Av}} \cdot g_r^{\text{Av}}(\mathbf{y}) + \epsilon^{\text{Tx}} \cdot g_r^{\text{Tx}}(\mathbf{y})\right] \cdot x_r, \tag{8}$$

where ϵ^{Av} and ϵ^{Tx} (both in kWh/RB) are the power consumed to keep one RB in the available state and in the transmitting state [9]. The energy consumed by cloud node i is

$$e_i^{\text{Cloud}}(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \left(\epsilon_i^{\text{C}} \cdot w_i(\mathbf{y}, \mathbf{z}) + \epsilon_i^{\text{Base}}\right) \cdot x_i, \tag{9}$$

where ϵ_i^{C} denotes the energy consumed to perform a unit of computational tasks (in kWh/GOPS) and ϵ_i^{Base} represents the baseline power required to keep a cloud node active. The total *energy expense* to run network operations is

$$E(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \sum_{r\in\mathcal{R}} e_r^{\text{RU}}(\mathbf{x}, \mathbf{y}) + \sum_{i\in\mathcal{V}} e_i^{\text{Cloud}}(\mathbf{x}, \mathbf{y}, \mathbf{z}), \tag{10}$$

where $e_r^{\text{RU}}(\cdot)$ and $e_i^{\text{Cloud}}(\cdot)$ are introduced in (8) and (9), respectively.

The *revenue* obtained from admitting UEs into the network corresponds to the total income generated by the admitted UEs, i.e.,

$$H(\mathbf{y}) = \sum_{r\in\mathcal{R}} \sum_{u\in\mathcal{U}} p_u \cdot y_{r,u}, \tag{11}$$

where p_u denotes the revenue associated with admitting UE u, which may vary across UEs according to business policies or service priorities.

B. Problem Formulation

Our goal is to maximize the system profit, defined as the difference between the revenue and energy consumption, i.e.,

$$P(\mathbf{x}, \mathbf{y}, \mathbf{z}) = H(\mathbf{y}) - \sigma \cdot E(\mathbf{x}, \mathbf{y}, \mathbf{z}) \tag{12}$$

where $H(\cdot)$ is given by (11), $E(\cdot)$ is given by (10), and σ the energy-cost weight (in \$/kWh).

[1]We consider that $\xi_{r,u} = \left\lceil \frac{\lambda_u}{\eta_{r,u}} \right\rceil \in \mathbb{N}$ where $\lambda_u \in \mathbb{R}_+$ (in bps) is the average throughput and $\eta_{r,u} \in \mathbb{R}_{\geq 0}$ (in bps/RB) is the achievable wireless link capacity per RB between UE u and RU r.

Intuitively, admitting more UEs increases the MNO's revenue, as each admitted UE generates a profit depending on its service characteristics and associated priority. However, serving these UEs requires activating radio and computing resources, thereby increasing the network energy consumption. There is a trade-off: the revenue brought by a UE justifies the additional operational cost required to serve it.

The maximization of (12) is subject to several constraints, which are described next.

C1: At any time, each active RU must be associated with exactly one VNF of each type (i.e., one DU and one CU) deployed on cloud nodes, i.e.,

$$\sum_{i \in \mathcal{V}} z_{r,i}^{v} = x_r, \forall r \in \mathcal{R}, v \in \{\text{DU}, \text{CU}\}. \tag{13}$$

C2: Any admitted UE can only be associated with an active RU and to one RU only, i.e.,

$$y_{r,u} \leq x_r, \forall r \in \mathcal{R}, \forall u \in \mathcal{U} \tag{14}$$

and

$$\sum_{r \in \mathcal{R}} y_{r,u} \leq 1, \forall u \in \mathcal{U}. \tag{15}$$

C3: The set of admitted UEs in each RU is bounded by the existing number of RBs, i.e.,

$$g_r^{\text{Tx}}(\mathbf{y}) \leq B, \forall r \in \mathcal{R}, \tag{16}$$

where $g_r^{\text{Tx}}(\cdot)$ is introduced in (3).

C4: The number of VNFs hosted by cloud node i is limited by its processing capacity μ_i, i.e.,

$$w_i(\mathbf{y}, \mathbf{z}) \leq \mu_i, \quad \forall i \in \mathcal{V}, \tag{17}$$

where we introduce function $w_i(\cdot)$ in (7).

Problem 1. The problem is to maximize the total profit (12) while satisfying system constraints C1–C4.

$$\begin{aligned} \underset{\mathbf{x},\mathbf{y},\mathbf{z}}{\text{maximize}} \quad & P(\mathbf{x}, \mathbf{y}, \mathbf{z}) \\ \text{subject to} \quad & (1), (2), (6), (13) - (17) \end{aligned}$$

Proposition 1. Problem 1 is $\mathcal{NP}$-Hard.

A complete proof of Proposition 1 is available in the supplementary technical report [7].

In this problem, we have three coupled decisions: (i) network component activation, (ii) UE admission control, and (iii) placement of DU and CU VNFs across the cloud infrastructure. By coordinating these decisions, the system seeks to *consolidate* traffic on a limited subset of infrastructure resources whenever possible, allowing unused RUs or cloud nodes to remain in low-power states. However, Problem 1 faces two major challenges:

- The presence of $\boldsymbol{\xi}$ in the system dynamics makes it difficult to derive deterministic policies that remain effective across all possible realizations.
- In practice, the different decision types are handled by distinct system components that cannot necessarily communicate and coordinate their actions toward a globally optimal solution.

Multi-Agent Reinforcement Learning (MARL) provides a natural framework to address these challenges. First, by learning through repeated interactions with the environment, MARL agents can implicitly adapt to the stochastic dynamics induced by $\boldsymbol{\xi}$ and optimize the expected long-term objective without requiring explicit analytical models. Second, by assigning an autonomous agent to each decision-making component, MARL naturally supports decentralized execution, allowing agents to learn coordinated behaviors while operating with limited or no real-time communication.

III. EAPO-NET

In this section, we introduce Adaptive Energy-Aware Profit Optimization in Next-Generation RANs (EAPO-NET), a novel MARL framework for tackling Problem 1. The idea is to decompose the global optimization problem into a set of coordinated local decision processes, each handled by a different agent. Each agent performs localized decisions based on partial system observations, while collectively learning a policy that approximates a globally efficient solution. This distributed learning paradigm mitigates the computational complexity associated with centralized optimization and enables scalable policy learning with good practical performance. We first explain EAPO-NET's components, then introduce the underlying Markov Decision Processes (MDPs). Finally, we describe the multi-agent algorithm, its properties, and its distribution across the different components.

A. EAPO-NET *Components and MDP Formulation*

EAPO-NET is a framework where the multiple agents are distributed among O-RAN's (i) Near-RT RIC, in the form of xApps, and (ii) O-Cloud nodes, in the form of dApps. In the following, we define the different types of agents and describe how they relate to Problem 1.

1) Energy Control (EC): Each EC agent is deployed as an xApp that controls a single RU's "sleep mode" and "blanking." We introduce the state space of RU r's EC agent as $\mathcal{O}_r^{\text{EC}} = [x_r, g_r, \beta_r, \mathbf{y}_r]$, where:

- $x_r \in \{0, 1\}$ indicates if the RU is in active mode,
- $g_r \in \mathbb{Z}_{\geq 0}$ is the amount of transmitting RBs,
- $\beta_r \in [0, 1]$ is the fraction of blanked RBs, and
- $\mathbf{y}_r \subseteq \mathcal{U}$ is the list of UEs associated with RU r.

Moreover, we introduce the action space of RU r's EC agent as $\mathcal{A}_r^{\text{EC}} = [a_r, b_r]$, where:

- $a_r \in \{\text{ON}, \text{OFF}\}$ denotes the activation action, such that $a_r = \text{ON}$ sets $x_r = 1$ and vice-versa.
- $b_r \in \{0, 0.25, 0.5, 0.75, 1\}$ denotes the blanking action drawn from a discrete set of percentages.

The Lagrangian G^{EC} coming from the EC part is:

$$G^{\text{EC}}(\mathbf{x}, \mathbf{y}) = -\sigma \sum_{r \in \mathcal{R}} e_r^{\text{RU}}(\mathbf{x}, \mathbf{y}) + \alpha_r^{\text{C3}} \left[g_r^{\text{Tx}}(\mathbf{y}) - B \right] \tag{18}$$

2) VNF Control (VC): Each VC agent is deployed as a dApp in a given O-Cloud node that controls which VNFs are locally deployed and which ones should be delegated to other O-Cloud nodes. We introduce the state space of node i's VC agent as $\mathcal{O}_i^{\text{VC}} = [x_i, z_i, w_i]$, where:

- $x_i \in \{0, 1\}$ indicates if the node is in active or sleep mode,
- $z_i \in \{0, 1\}$ indicates which VNFs are being locally hosted, and
- $w_i \in \mathbb{R}_{\geq 0}$ is the amount of resources consumed, analogous to (7).

Moreover, we introduce the action space of node i's VC agent as $\mathcal{A}_i^{\text{VC}} = [a_i, \mathbf{m}_i^{\text{VNF}}]$, where:

- $a_i \in \{\text{ON, OFF}\}$ denotes the activation action, such that $a_i = \text{ON}$ sets $x_i = 1$ and vice-versa.
- $\mathbf{m}_i^{\text{VNF}} \in \{0, 1\}^{R \times V}$ denotes the migration action for each VNF. Here, such that $m_{i,r,j}^{\text{VNF}} = 1$ indicates that node i is moving RU r's VNF to node j.

The Lagrangian G^{VC} coming from the VC part is:

$$G^{\text{VC}}(\mathbf{x}, \mathbf{y}, \mathbf{z}) = - \sigma \sum_{i \in \mathcal{V}} e_i^{\text{Cloud}}(\mathbf{x}, \mathbf{y}, \mathbf{z}) + \sum_{i \in \mathcal{V}} \alpha_i^{\text{C4}} \left[w_i(\mathbf{y}, \mathbf{z}) - \mu_i \right] \tag{19}$$

3) Admission Control (AC): The AC agent is deployed as an xApp that controls which UEs are admitted to the system and which RU they are associated with. We introduce the state space of the AC agent as $\mathcal{O}^{\text{AC}} = [\{\mathcal{O}_r^{\text{EC}} : r \in \mathcal{R}\}, \{\mathcal{O}_i^{\text{VC}} : i \in \mathcal{V}\}, \boldsymbol{\xi}]$, which considers the states of the other agents and the vector of RBs $\boldsymbol{\xi} = \{\xi_{r,u} : r \in \mathcal{R}, u \in \mathcal{U}\}$ required by the UEs. Moreover, we introduce the action space of the AC agent as $\mathcal{A}^{\text{AC}} = \{0, 1\}^{\mathcal{U} \times \mathcal{R}}$, such that each action is a different UE-to-RU association mapping. The Lagrangian G^{AC} coming from the AC part is:

$$\begin{aligned} G^{\text{AC}}(\mathbf{x}, \mathbf{y}) = H(\mathbf{y}) &+ \sum_{r \in \mathcal{R}} \sum_{u \in \mathcal{U}} \alpha_{r,u}^{\text{C2}} \left[y_{r,u} - x_r \right] \\ &+ \sum_{u \in \mathcal{U}} \alpha_u^{\text{C2}'} \left[\sum_{r \in \mathcal{R}} y_{r,u} - 1 \right] \end{aligned} \tag{20}$$

Hence, the total [2] Lagrangian of the system is given by:

$$\begin{aligned} G(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \\ G^{\text{EC}}(\mathbf{x}, \mathbf{y}) + G^{\text{VC}}(\mathbf{x}, \mathbf{y}, \mathbf{z}) + G^{\text{AC}}(\mathbf{x}, \mathbf{y}), \end{aligned} \tag{21}$$

where $G^{\text{EC}}(\cdot)$, $G^{\text{VC}}(\cdot)$, and $G^{\text{AC}}(\cdot)$ are respectively defined in equations (18), (19), and (20).

The MDPs' transition dynamics are unknown, making model-based methods infeasible and requiring model-free learning techniques. They also feature large state-action spaces and high combinatorial complexity, as noted in our supplementary report [7]. These factors render conventional RL impractical, necessitating DRL methods.

[2]Constraint C1 is omitted from this formulation as it is structurally enforced by the causal action ordering in the environment; remaining constraints are integrated as penalty terms in the shared reward R_t.

B. EAPO-NET*'s Algorithm Description*

In a multi-agent setting, a key challenge is the non-stationarity of the environment: as agents update their policies, it disrupts the Markov assumption and destabilizes independent learning. Additionally, real-time synchronization for inter-agent coordination is impractical due to varying control loop latencies.

To tackle these issues, we draw from the VDN in the CTDE method [6], which addresses coordination through (i) *centralized training* using a global reward signal for agents' DQNs and (ii) *decentralized execution*, applying the learned policy without inter-agent communication.

Centralized Training Phase: A centralized trainer has access to the joint observation tuple $\langle o_1^t, \ldots, o_k^t \rangle$ and maintains one independent utility network $Q_k(o_k, a_k; \theta_k)$ per agent. All agents share a global replay buffer $\mathcal{D}$ that stores transitions $\big(\mathcal{O}_k(t),\, \mathcal{A}_k(t),\, R,\, \mathcal{O}_k(t+1)\big)$ collected across all k agents. We define the global reward as

$$R_t = G(\mathbf{a}_t(\mathbf{s}_t)) - G(\mathbf{s}_t), \tag{22}$$

where $\mathbf{s}_t \in \mathcal{O}$ is the state at time t, $\mathbf{s}_{t+1} = \mathbf{a}_t(\mathbf{s}_t)$ is the state at time $t+1$ after applying action $\mathbf{a}_t$, and $G(\cdot)$ is the Lagrangian function of Problem 1.

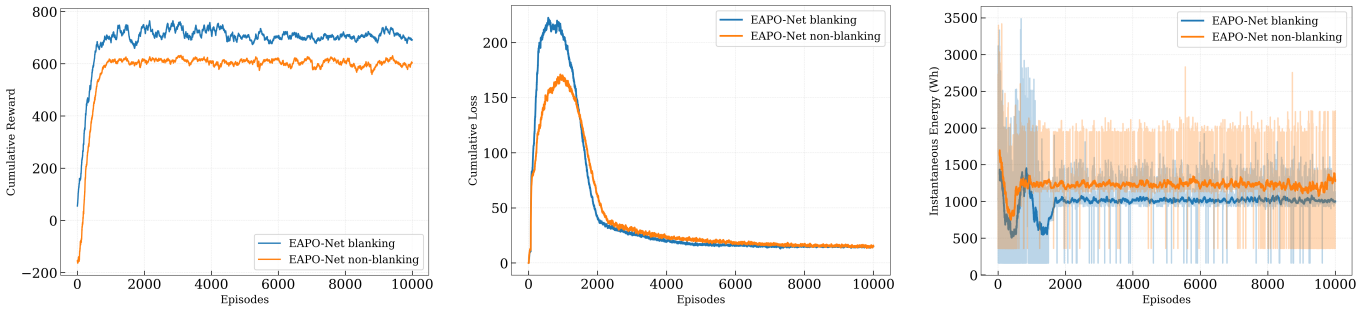
The global reward R_t serves as a common training signal that aligns all agents toward the system-wide objective. Despite sharing the same reward, each agent k learns an independent action-value function by minimizing its own temporal-difference (TD) loss. The loss function is defined as

$$\mathcal{L}(\theta) = \mathbb{E}_{\mathcal{D}} \left[\left(\Upsilon - \sum_{k=1}^{K} Q_k(o_k(t), a_k(t); \theta_k) \right)^2 \right], \tag{23}$$

where $\Upsilon = R_t + \gamma \sum_{k=1}^{K} \max_{a'_k} Q_k(o_k(t+1), a'_k; \theta_k^-)$ is the TD target whose parameters θ_k^- are periodically synchronized and γ is the discount factor.

Decentralized Execution Phase: After centralized training, each agent k receives only its trained network θ_k and operates fully decentralized. At each time slot t, agent k observes its local observation $o_k(t)$ solely and selects its action greedily without requiring any knowledge of other agents' observations or actions. No inter-agent communication takes place at decision time, and the replay buffer and the Q-learning loss functions are no longer involved. The IGM condition guarantees that this greedy per-agent maximization remains consistent with the joint optimum found during centralized training.

At execution time, although potential conflicts may arise between agents, they are structurally prevented rather than statistically avoided: the causal ordering enforced within each decision step constrains each agent to act only within the feasible space left by preceding agents, making inter-agent conflicts impossible by construction without requiring any explicit communication.



(a) Reward evolution over training. (b) Training loss progression. (c) Energy consumption over training.

Fig. 1: Analysis of reward, loss, and energy consumption.

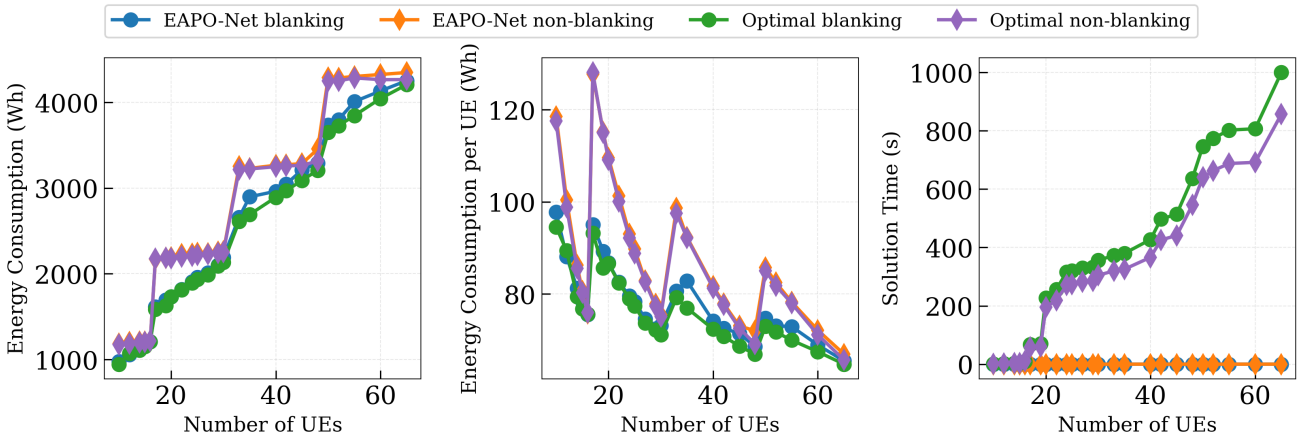


Fig. 2: Impact of UE count on energy consumption (left), energy per UE (center), and solution time (right).

System Parameters

Parameter	Value
Number of RUs $\lvert\mathcal{R}\rvert$	4
UE population $\lvert\mathcal{U}\rvert$ [min, max]	[10, 65]
UE data rate λ_u [min, max] Mbps	[1.5, 8.25]
Number of O-Cloud nodes $\lvert\mathcal{V}\rvert$	7
RU capacity B	100 (RBs)
Achievable wireless capacity $\eta_{u,r}$	0.25 (Mbps/RB)
RU energy $\epsilon^{\text{Av}}, \epsilon^{\text{Tx}}$	2, 6 (Wh/RB)
The energy-cost weight σ	0.1658 $/kWh
O-Cloud Energy ϵ_i^{C}, ϵ_i^{Base}	100, 30 (Wh)
O-Cloud node computing capacity μ_i^{Comp}	300 (GOPS)

Learning Parameters

Parameter	Value
Epoch	10000
Activation function	ReLU
Optimizer	Adam
Learning rate	1×10^{-4}
ϵ_0, ϵ_{min}, ϵ_{decay}	1.0, 0.01, 0.995
Discount factor	0.91
Replay buffer, Batch size	10^6, 128

TABLE I: Simulation Parameters

IV. Performance Evaluation

A. Experimental Setup

To evaluate the EAPO-Net framework, we developed a Python-based O-RAN simulation environment. We implemented the MA-DQN learning framework using PyTorch for efficient neural network training and practical policy learning. The benchmark optimal solution is obtained by implementing the ILP from Problem 1 in Pyomo and solving it with the GLPK solver, which utilizes a B&C algorithm for global optimality. System parameters are summarized in Table I.

B. Convergence Behavior

We evaluate the convergence of EAPO-Net using three metrics: average episode reward, training loss, and energy consumption. Results for both RB-blanking and non-blanking configurations are shown in Fig. 1.

Fig. 1 shows that both variants converge after approximately 2,000 training episodes. The reward stabilizes, the training loss decreases steadily, and the energy consumption converges, with the blanking variant achieving lower energy consumption than the non-blanking variant.

Takeaway 1: EAPO-Net converges reliably under both blanking and non-blanking configurations while learning energy-efficient policies.

C. Analysis of Energy Consumption

1) Energy consumption as a function of UE density: For a fixed average load of 1.5 Mbps per UE, Fig. 2 reports the energy consumption, energy consumption per UE, and per-slot decision time as the number of UEs increases.

Fig. 2(left) shows that energy consumption increases with the number of UEs. At RU saturation points (15, 30, 48, and 65 UEs), blanking opportunities disappear until additional RUs

are activated. The largest energy savings are achieved between 33 and 48 UEs, reaching 18.4% with EAPO-NET and 18.3% with the optimal model.

Fig. 2(center) shows that energy consumption per UE decreases with increasing UE density due to improved RB utilization and amortized static energy costs. Fig. 2(right) shows that EAPO-NET requires only 1.9–13 ms (blanking) and 1.7–12.3 ms (non-blanking), whereas B&C requires 0.35–999 s and 0.3–856 s, respectively.

Takeaway 2: EAPO-NET maintains near-optimal performance while reducing decision time from seconds to milliseconds, with its computational advantage increasing as the number of UEs grows.

V. Related Work

Mobile networks face a critical challenge: "energy consumption" is rapidly increasing while traffic becomes more dynamic and unpredictable. Current O-RAN control strategies are largely *static or rule-based*, keeping RAN and computing resources active during low traffic and wasting energy. Existing approaches mainly rely on cell sleeping (Cell Switching ON/OFF) and workload consolidation.

A large body of work exploits traffic fluctuations to deactivate base stations or sectors. For example, [10] jointly optimize cell clustering and BS sleeping to consolidate load during medium- and long-term low-traffic periods. However, these approaches require strong inter-cell coordination, which is difficult to achieve in dense and heterogeneous deployments.

At shorter timescales, energy savings can be achieved without shutting down entire cells. Cell Discontinuous Transmission (DTX) [11] enables sub-TTI micro-sleep during idle periods, while recent O-RAN xApps [12]–[15] exploit Advanced Sleep Modes (ASMs) and energy-aware scheduling. However, they generally assume ideal cell overlap or seamless UE mobility, limiting their applicability. Other works optimize UE–RU association, RB allocation, beamforming, or VNF placement [16]–[19]. While effective for resource management and orchestration, they rarely exploit "RB blanking", which reduces energy consumption within active RUs without requiring cell consolidation or ASM activation, and remains largely unexplored in O-RAN [20], [21].

VI. Conclusion

This paper presented EAPO-NET, a multi-agent RL framework for the joint optimization of RU sleep mode, UE admission control, and DU/CU VNF placement in 5G O-RAN systems. By decomposing the problem across heterogeneous agents and leveraging CTDE with VDN, EAPO-NET enables coordinated learning while supporting fully decentralized inference without inter-agent communication. Extensive simulations demonstrated that EAPO-NET learns near-optimal policies that effectively balance energy consumption, network profit, and SLA compliance. Future work will extend EAPO-NET to multi-cell scenarios and investigate online policy adaptation mechanisms to address long-term traffic dynamics.

Acknowledgements

This work was supported by ANR under the France 2030 program, grant NF-NAI: ANR-22-PEFT-0003.

References

[1] S. Han, S. Bian *et al.*, "Energy-efficient 5g for a greener future," *Nature Electronics*, vol. 3, no. 4, pp. 182–184, 2020.

[2] T. Li, L. Yu, Y. Ma, T. Duan, W. Huang, Y. Zhou, D. Jin, Y. Li, and T. Jiang, "Carbon emissions and sustainability of launching 5g mobile networks in china," *arXiv:2306.08337*, 2023.

[3] O.-R. Alliance, "O-ran whitepaper - building the next generation ran," October 2018, accessed: 2026-03-10. [Online]. Available: https://www.o-ran.org/resources

[4] M. Polese, L. Bonati, S. D'oro, S. Basagni, and T. Melodia, "Understanding o-ran: Architecture, interfaces, algorithms, security, and research challenges," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 2, pp. 1376–1411, 2023.

[5] O.-R. Alliance, "O-ran slicing architecture 14.01," June 2025, technical Specification, O-RAN.WG1.TS.Slicing-Architecture-R004-v14.01, Adopted by ATIS, R004.

[6] P. Sunehag, G. Lever, A. Gruslys, W. M. Czarnecki, V. Zambaldi, M. Jaderberg, M. Lanctot, N. Sonnerat, J. Z. Leibo, K. Tuyls *et al.*, "Value-decomposition networks for cooperative multi-agent learning," *arXiv:1706.05296*, 2017.

[7] A. C. B. et al, "Eapo-net: Adaptive energy-aware profit optimization in next-generation rans (extended technical report)," 2026, https://bessalacedric.github.io/uploads/EAPO_Net_TR.pdf.

[8] R. Singh, C. Hasan, X. Foukas, M. Fiore, M. K. Marina, and Y. Wang, "Energy-efficient orchestration of metro-scale 5g radio access networks," in *IEEE INFOCOM*, 2021, pp. 1–10.

[9] M. Lauridsen, A. R. Jensen, and P. Mogensen, "Reducing lte uplink transmission energy by allocating resources," in *IEEE VTC*, 2011.

[10] J. Kim, H.-W. Lee, and S. Chong, "Traffic-aware energy-saving base station sleeping and clustering in cooperative networks," *IEEE Transactions on Wireless Communications*, vol. 17, no. 2.

[11] P. Chang and G. Miao, "Optimal operation of base stations with deep sleep and discontinuous transmission," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, 2018.

[12] J. X. Salvat Lozano, J. A. Ayala-Romero, A. Garcia-Saavedra, and X. Costa-Perez, "Kairos: Energy-efficient radio unit control for o-ran via advanced sleep modes," in *IEEE INFOCOM*, 2025, pp. 1–10.

[13] A.-K. Dang, H. Khalifé, M. Sintorn, S. Rovedakis, and S. Secci, "Data-driven energy optimization in mobile networks with user experience guarantees," in *IEEE INFOCOM*, 2025, pp. 1–10.

[14] J. A. Ayala-Romero, L. Lo Schiavo, A. Garcia-Saavedra, and X. Costa-Perez, "Mean-field multi-agent contextual bandit for energy-efficient resource allocation in vrans," in *IEEE INFOCOM*, 2024, pp. 911–920.

[15] W. T. Pires-JR, G. M. Almeida, S. L. Corrêa, C. B. Both, L. L. Pinto, and K. V. Cardoso, "Optimizing energy consumption for vran placement in o-ran systems with flexible transport networks," *IEEE Open Journal of the Communications Society*, vol. 6, pp. 4279–4294, 2025.

[16] B. Han, V. Sciancalepore, D. Feng, X. Costa-Perez, and H. D. Schotten, "A utility-driven multi-queue admission control solution for network slicing," in *IEEE INFOCOM*, 2019, pp. 55–63.

[17] M. K. Motalleb, V. Shah-Mansouri, S. Parsaeefard, and O. L. A. López, "Resource allocation in an open ran system using network slicing," *IEEE Transactions on Network and Service Management*, vol. 20, no. 1, pp. 471–485, 2022.

[18] S. D'Oro, L. Bonati, M. Polese, and T. Melodia, "Orchestran: Network automation through orchestrated intelligence in the open ran," in *IEEE INFOCOM*, 2022, pp. 270–279.

[19] A. C. Bessala, G. Iecker Ricardo, and G. Jakllari, "E-MATRO: an adaptive and Energy-Efficient framework for slicing within the O-RAN architecture," in *IFIP Networking 2025*.

[20] F. Salahdine, J. Opadere, Q. Liu, T. Han, N. Zhang, and S. Wu, "A survey on sleep mode techniques for ultra-dense networks in 5g and beyond," *Computer Networks*, vol. 201, p. 108567, 2021.

[21] R. Smith, M. Farshchian, C. Freeberg, T. Machacek, and D. Li, "Enabling technologies for spectrum sharing with oran," in *2024 IEEE DySPAN*, pp. 102–107.