ISSN 2277-2685
IJESR/April-June. 2025/ Vol-15/Issue-2s/625-634
Kanaparthy Karuna Sahithi *et. al.,* / International Journal of Engineering & Science Research

Load Balancing In Mobile Networks Using Deep Reinforcement Learning And Traffic Prediction

[1]Dr. B Raveendranadh Singh, [2]Kanaparthy Karuna Sahithi, [3]Sampeta Navya
[1]Head, Department of CSE, Bhoj Reddy Engineering College for Women, India.
[2,3]B.Tech Students, Department of CSE, Bhoj Reddy Engineering College for Women, India.

ABSTRACT

Wireless communication networks are advancing at a rapid pace, driven by various challenges and ambitious goals. This rapid growth is driven by a range of applications, including technologies like the Internet of Things (IoT), as well as innovations in smart cities, autonomous vehicles, and more. Different applications demand specific performance criteria such as high data throughput, low latency, robust reliability, and efficient energy usage. In this thesis, we investigate two enhancements that can be adopted in wireless networks to tackle the challenges of resource optimization and network management. The motivation behind this is the fact that future networks will face challenges like severe congestion and varying traffic demands. The objective is to achieve higher network throughput and more data transmission by adjusting the network parameters. The first proposed approach introduces an enhanced self-optimization framework using deep reinforcement learning (RL) to dynamically adjust network parameters such as handover parameters, power levels, and MIMO technology. The proposed approach offers significant gains in network throughput by effectively balancing the load distribution. The proposed framework explores the trade-off between system complexity and performance improvement, demonstrating that adopting a scenario-aware optimized agent can outperform generalized agents under specific network conditions. The second approach we tackle is to adopt a proactive concept while controlling the network. The proposed approach is based on the ARIMA model used to predict the next states of the environment so that the RL agent considers them in the decision-making process. The simulation results demonstrate that the proposed approach leads to higher throughput and improved network performance, which underscores its potential as a robust alternative to the conventional agent existing in earlier works.

1-INTRODUCTION

Wireless communication is essential to our daily life, education, business, and leisure. There are communication systems all around us. For example, smart homes, surveillance, industry, and health care all employ wireless sensor networks, or WSNs. Cell phones, smartphones, smart TVs, outdoor internet, and other devices also use mobile networks. Additionally, WiFi is utilized to establish indoor internet connections and small networks. Moreover, the Internet of Things (IoT) has emerged as a transformative force across industries, enabling advancements in fields such as manufacturing, agriculture, transportation management, and home automation. IoT tech- nologies facilitate real-time data collection and analysis, leading to smarter decision-making and operational efficiencies. As communication technologies continue to evolve, new protocols and infrastructures are being developed to meet the increasing demand for seamless connectivity. 5G networks aim to enable massive connectivity and provide data rate speeds of 10 Gbps for low mobility and 1 Gbps for high mobility [1]. Global mobile data traffic is expected to reach around 160 exabytes (a thousand billion GB) per month in 2025 [2]. This increase in traffic re- quires significant optimization at the network level. With the huge number of devices, a lot of handover requests are initiated in order to keep the user's QoE unaffected. These handover pro- cesses should be reliable and seamless so as not to impact the user's connection or the quality served. The Network is also required to reduce delay in order to support real-time applications like real-time video streaming, online gaming, video conferencing, etc. Managing such highly dense networks requires proper allocation of network resources. As high traffic loads have be- come a primary concern in the design and operation of wireless communication systems, wire- less systems serve a huge number of devices with the aim of providing ubiquitous connectivity

and extremely high data rates not only for mobile phones but also for all newly emerged IoT de- vices distributed across a given area [3]. In order to serve a massive number of terminals, future networks will have serious congestion concerns. Network operators will have to provide more network resources in a timely manner to maintain good Quality of service (QoS) and Quality of experience (QoE) for users.

2-BACKGROUND AND CONCEPTS

In this chapter, we present a background of the main terminologies and concepts utilized in this thesis. We start by discussing Open Radio Access Networks (ORAN) as a key enabler for machine paradigm shift in RAN networks by promoting interoperability.

Our primary concern in this thesis is related to intelligence and automation. One major use case for O-RAN, according to the whitepaper [6] introduced learning algorithms that will be used afterwards. In section 2.1, we present the machine learning algorithms that will be used in our methodology including two main algorithms, Double Deep Q-Learning (DDQG) and Twin Delayed Deep Deterministic Policy Gradient (TD3). Then, in the last section, an explanation of the forecasting algorithm used in Chapter 4 will be introduced.

Open Radio Access Networks (O-RAN)

Traditional cellular networks have been around for decades. Much development has been done in the area of wireless communication that enhanced the quality and overall performance of the communication process. Newly developed technologies have been introduced like the en- hanced mobile broadband (eMBB), massive machine-type communications (mMTCs), and ultra- reliable low-latency communications (URLLCs) [5]. However, cellular network infrastructural devices were always proprietary hardware and software solutions, usually implemented by the same vendor to enable a good match for a proper communication process. This forced operators to rely on a single vendor for all their network needs. They, thus, guaranteed compatibility and interoperability based on the operators' needs at a price of limited innovation and creativity in designing the network.

Recently, O-RAN concepts were introduced to foster innovation and competition and facilitate flexible RAN deployments. Key components of O-RAN networks include cloudification, intel- ligence, automation, and open internal RAN interfaces. O-RAN is envisioned to introduce a

by the O-RAN alliance, is the conges- tion prediction and management. This resulted from the loads in modern 4G-5G networks, the fluctuating amount of moving cellular traffic and severe cell congestion that leads to a poor user experience due to radio link failures, handover failures, low data rates, etc.

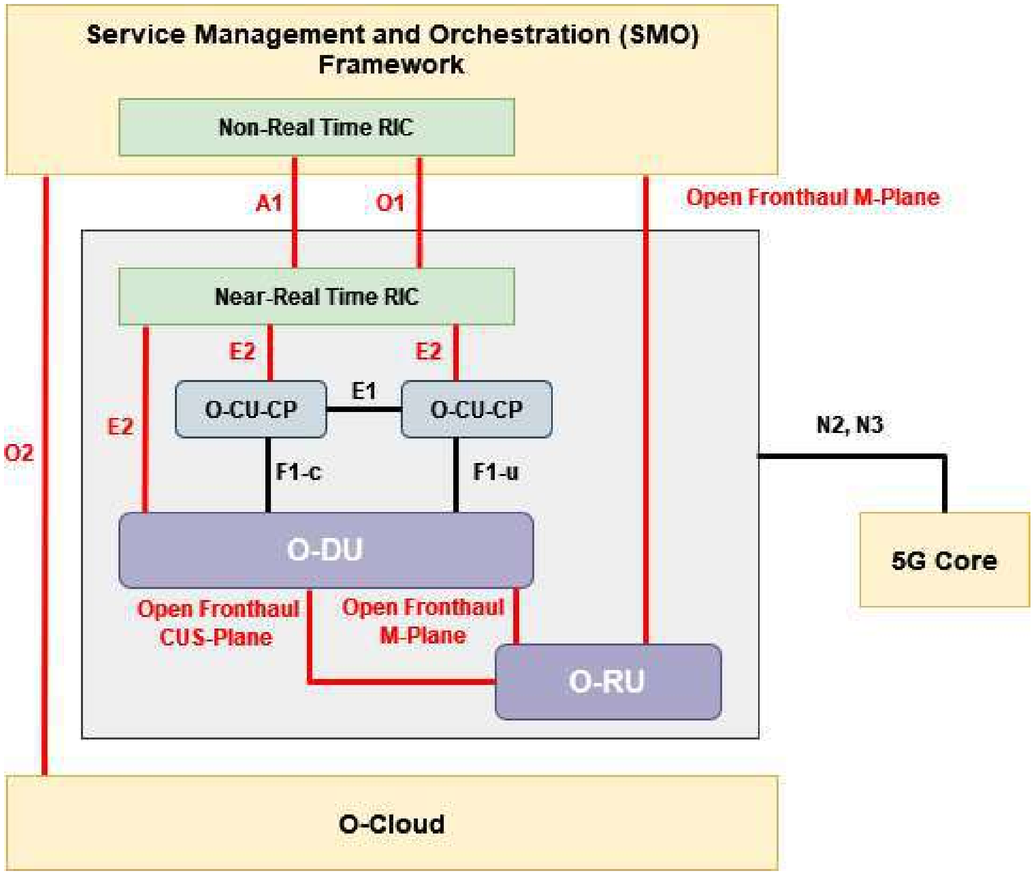


FIGURE 2.1: O-RAN Logical Architecture

3-SCENARIO-AWARE REINFORCEMENT LEARNING AGENT FOR MIMO USAGE OPTIMIZATION

In this chapter, we build upon the methodology presented in [4] to enhance the proposed approach and achieve performance improvements. RL is selected due to its effectiveness in optimizing long-term objectives without requiring training data. The work in [4] introduced a robust framework for self-optimizing cellular networks through deep reinforcement learning, aiming to improve network efficiency by balancing user load, enhancing coverage, improving user experience, and minimizing energy consumption. The proposed system incorporates a DDQN agent followed by a TD3 agent for fine-tuning handover settings, power allocations, and MIMO configurations. In this chapter, we refine the RL agents presented in [4] by incorporating an additional continuous-action TD3 agent specifically designed for a frequent, recurring scenario. During simulations, a particular DDQN decision state appeared more frequently than others. We accounted for this scenario during training and adjusted the relevant power and CIO values using the TD3 agent according to the network need. The inclusion of this specialized agent led to noticeable performance gains when the corresponding scenario occurred. However, this design results in three agents overall, increasing system complexity by having an additional agent, especially if multiple frequent scenarios exist. Despite this, the improvement in the overall network reward justifies

ISSN 2277-2685
IJESR/April-June. 2025/ Vol-15/Issue-2s/625-634
Kanaparthy Karuna Sahithi *et. al.*, / International Journal of Engineering & Science Research

the added complexity. Thus, a trade-off exists between the number of agents employed and the performance benefits achieved. The scenario-aware agent demonstrates enhanced performance compared to the general agent in [4] when optimizing network parameters, making it a compelling choice for self-optimization in networks where certain scenarios are more prevalent.

Proposed Algorithm

In this section, we provide a thorough explanation of the suggested algorithm.

The proposed scheme is outlined in the following Algorithm 1, where $s(t)$ is the observed state at time t, $a_M(t)$ is the MIMO enabling action vector, $a_C(t)$ is the CIO values action vector

$a_P(t)$ is the transmitted powers action vector.

Algorithm 1 Proposed RL framework

Determine Reward Function. Reset all values.

Repeat

procedure STAGE ONE Observe State ($s(t)$).

Select MIMO feature decision (DDQN) ($a_M(t)$).

Create a new augmented state ($s_{aug}(t)$ = [$s(t)$, $a_M(t)$]).

end procedure

procedure STAGE TWO

Observe state ($s_{aug}(t)$) and select the proper TD3 agent according to $a_M(t)$

Select relative CIO and power level actions (TD3) ([$a_C(t)$, $a_P(t)$])

Apply augmented action to the network a_{aug} = [$a_C(t)$, $a_P(t)$, $a_M(t)$].

end procedure

Calculate Reward.

Calculate the next state.

Decision-making occurs in two stages:

- **First Stage:** Based on the DDQN approach, the agent watches the state and decides when to turn MIMO ON or OFF [38]. The discrete set {0, 1} is used as an action space (for each eNB). Keep in mind that the selected action *is not applied* to the surroundings until the second stage is complete.
- **Second Stage:** Based on the output of the first stage, a TD3 agent is selected to take the second stage decision. We augment the first-stage action with the observed state. The second stage decides the CIO and the variation in power level actions based on the TD3

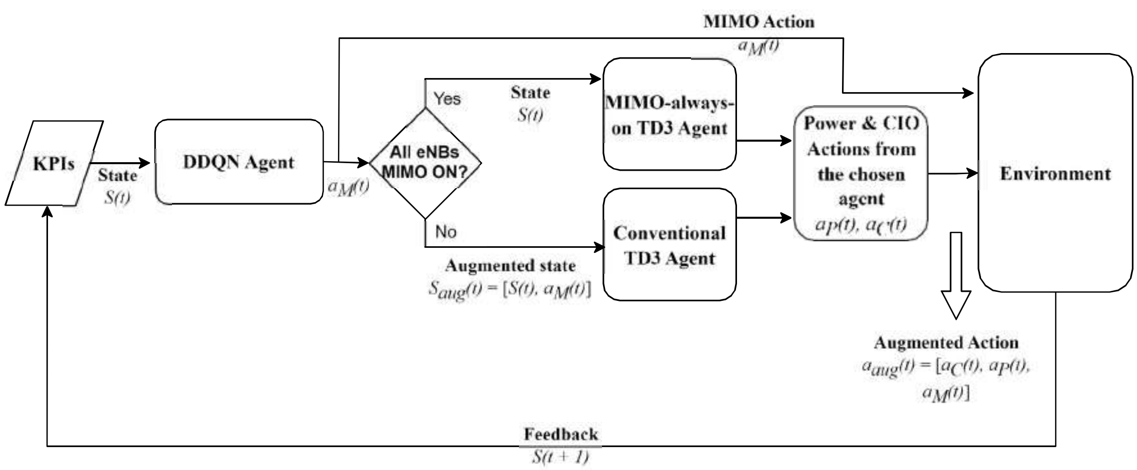


FIGURE 3.2: An Overview of the Decision-Making Process.

4-PROACTIVE CONGESTION PREDICTION AND OPTIMIZATION IN CELLULAR

NETWORKS

Recently, there have been several rapid changes in cellular networks [46]. For example, depending on current events, the number of users fluctuates significantly over time. Due to sudden fluctuations in metropolitan areas, the coverage requirements also vary. Within the last few years, providing high data rates that can support different applications has become a ubiquitous necessity. Cellular networks must be flexible enough to satisfy the new demands and handle the quick changes that are always occurring. Additionally, a criterion needs to be established to ensure that the network won't be disrupted by these quick adjustments. In other words, achieving network balance is essential. This will enable the network to self-heal from any certain issues and provide immunity against changes in the surrounding environment.

By automating the optimization and management of wireless mobile networks, it is possible to improve user experience and lower operating costs for network operators. Currently, a lot of network administration tasks require human intervention, from diagnosing customer problems to running drive testing to assess network performance and coverage [47]. We support the claim that machine learning methods can help with these duties by assisting in the diagnosis and prediction of network problems before they significantly impair network users' quality of service. Most of the interactions from operators with the cellular network are reactive actions. Network KPIs are being monitored, and once they are impacted, operators take the proper action to overcome this performance degradation. Controlled lab tests are the traditional method for evaluating network and service performance from the viewpoint of QoE end users [48]. Network operators aim to address network congestion through various strategies to ensure a positive user experience. However, these congestion mitigation approaches are typically reactive, leaving

the network vulnerable to performance degradation and negatively impacting user satisfaction. Overcoming sudden changes in the network should be in a timely manner to avoid out-of-service errors. Sufficient resources should be available to be allocated or correct handover decisions should be taken on the spot to provide an acceptable level of service availability.

SONs are being studied in literature to do automatic mitigation actions by continuous monitoring and updating network parameters without the need for human intervention. Our main focus here is congestion awareness and mitigation prior to occurrence using a part of the O-RAN architecture called Congestion Prediction Management (CPM) [6] that provides intelligent actions. This component of the network can be used to apply actions for the predicted congestion before it actually occurs. This is done through data collecting, data pre-processing, and then invoking an AI algorithm for future KPI prediction. After obtaining the prediction, it is then used to be part of the observation for the decision-making process. We propose that being able to predict the next state of the environment and applying updates to network parameters while considering this prediction will result in a better performance. In this chapter, we aim to establish an integrated algorithm for both prediction and optimization processes to produce an optimizing agent for congestion control. After this, we investigate the effect of this agent on the environment and network performance.

5-CONCLUSION

In this thesis, we explored two main enhancements for the cellular network. The first one we studied is the scenario-aware RL agent's performance, in which the environment is observed to turn the MIMO feature ON most of the time. We use a TD3 agent that is trained to only optimize the decision for this specific scenario. Using this scenario-aware agent when all the eNBs are enabling the MIMO feature, while using the original RL agent that is trained to optimize all scenarios of the observed environment. The case where all eNBs enable the MIMO scheme was selected due to its frequent occurrence during the environment simulation. The added agent works along with the conventional agent to provide better performance when each is used in the case for which it was trained. This suggestion yields a better performance and higher downlink throughput which will give the end users a better quality of experience. Whilst, it introduces some intricacy to the design. This trade-off should be assessed by system designers according to the environment conditions to adopt the new approach when a case occurs much more frequently than others. Specializing an agent for this case will result in better results for the agent's actions. While the thesis demonstrates the benefits of scenario-specific RL agents, future work should focus on exploring scalability across diverse and dynamic cellular network conditions.

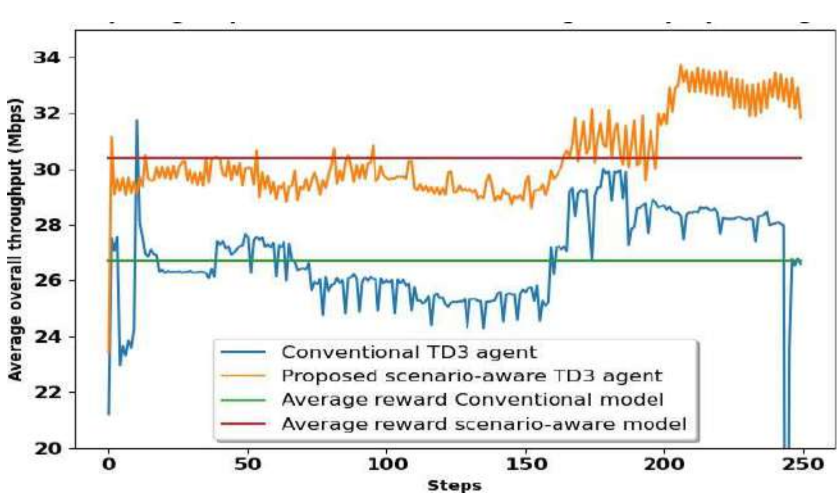
The second suggested enhancement is introducing a proactivity factor by optimizing the RL agent's decision while considering the future expected state of the environment. The agent's state contains the predicted value of the DL throughput in the next 5 15-minute spans. This allows the agent to be able to foresee the correct action to apply to the environment while acknowledging the predicted value by the ARIMA algorithm. The environment has dynamic user loads to present an increase in the network traffic. Taking this into consideration while training and

6-RESULTS

In this section, we assess the performance of our proposed approach by testing the effect of different hyperparameters on the sum throughput of the network. Testing scenarios are as follows:

1. *MIMO always on with penalty on the user coverage:* We employ a RL agent to optimize the CIOs and power levels only. We switch on the MIMO feature at all times for all eNBs and use a penalty on uncovered users with $\eta = 2$ in equation (3.5). This setup allows us

to compare the performance of both agents in the same environment, when all eNBs are forced to keep MIMO on, to observe the agents' performance



ISSN 2277-2685
IJESR/April-June. 2025/ Vol-15/Issue-2s/625-634
Sangannagare Chandana *et. al.*, / International Journal of Engineering & Science Research

in this specific case. We plot the network sum throughput (in Mbps) versus steps in Fig. 3.5 across a testing episode of 250 steps. The difference between both models is mostly around 4 $Mbps$.

FIGURE 1: Comparison of the sum throughput reward for the MIMO-on agent and original agent $\eta = 2$

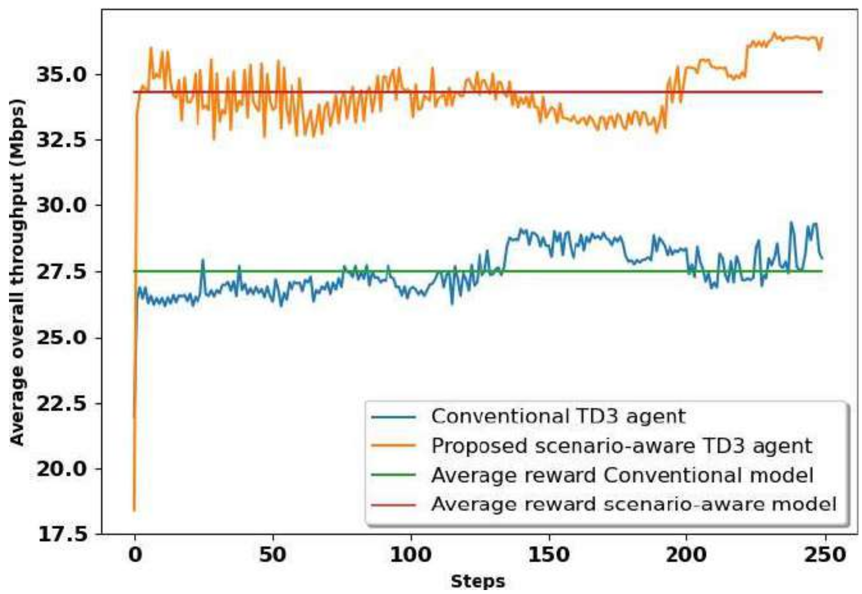


FIGURE 2: Comparison of the sum throughput reward for the all-MIMO-ON agent and original agent $\eta = 0$

2. *MIMO on/off with no penalty on the user coverage:* In this approach, the DDQN agent first decides whether to enable the MIMO feature on all eNBs. If MIMO is enabled for all eNBs, the scenario-aware TD3 agent (trained specifically for this all-MIMO-on setting) handles the continuous actions for CIO and transmitted power. Otherwise, the conventional TD3 agent from [4] is used to select the CIO and power values. The results of this case are shown in Fig. 3.7.

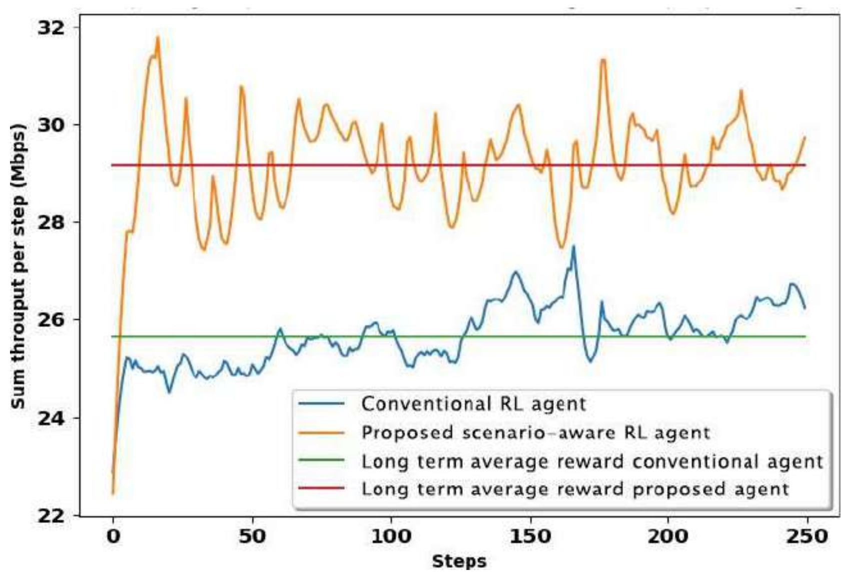


FIGURE 3: Average throughput per step over 20 episodes while alternating between conventional and all-MIMO-ON agents, $\eta = 0$

BIBLIOGRAPHY

[1] Ali Ö Ercan, M Oğuz Sunay, and Ian F Akyildiz. "RF energy harvesting and transfer for spectrum sharing cellular IoT communications in 5G systems". In: *IEEE Transactions on Mobile Computing* 17.7 (2017), pp. 1680–1694.

[2]Ghada Alsuhli et al. "Mobility Load Management in Cellular Networks: A Deep Rein- forcement Learning Approach". In: *IEEE Transactions on Mobile Computing* 22.3 (2023), pp. 1581–1598. DOI: 10.1109/TMC.2021.3107458.

[3]Pal Varga et al. "5G support for Industrial IoT Applications— Challenges, Solutions, and Research gaps". In: *Sensors* 20.3 (2020). Issn: 1424-8220. DOI: 10.3390/s20030828. URL: https://www.mdpi.com/1424-8220/20/3/828.

[4] Bishoy Salama Attia et al. "Self-Optimized Agent for Load Balancing and Energy Efficiency: A Reinforcement Learning Framework With Hybrid Action Space". In: *IEEE Open Journal of the Communications Society* 5 (2024), pp. 4902–4919.

[9]Mutasem Q. Hamdan et al. "Recent Advances in

[5] Petar Popovski et al. "5G Wireless Network Slicing for eMBB, URLLC, and mMTC: A Communication-Theoretic View". In: *IEEE Access* 6 (2018), pp. 55765–55779. DOI: 10.1109/ACCESS.2018.2872781.

[6] O-RAN Alliance. *O-RAN Minimum Viable Plan and Acceleration towards Commercializa- tion Architecture Description*. 2021. URL: https://www.o-ran.org/resources (visited on 06/04/2023).

[7] 5G Americas. *Transition Toward Open & Interoperable Networks*. https://www. 5gamericas.org/wp-content/uploads/2020/11/InDesign-Transition-Toward-Open-Interoperable-Networks-2020.pdf. Accessed: 07 08 2023.
2020.

[8]Michele Polese et al. "Understanding O-RAN: Architecture, Interfaces, Algorithms, Security, and Research Challenges". In: *CoRR* abs/2202.01032 (2022).

Machine Learning for Network Au- tomation in the

O-RAN”. In: *Sensors* 23.21 (2023). ISSN: 1424-8220. DOI: 10.3390/ s23218792. URL: https://www.mdpi.com/1424-8220/23/21/8792.

[10] Nguyen Cong Luong et al. “Applications of deep reinforcement learning in communica- tions and networking: A survey”. In: *IEEE Communications Surveys & Tutorials* 21.4 (2019), pp. 3133–3174.

[11] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[12] Ghada Alsuhli et al. “Optimized Power and Cell Individual Offset for Cellular Load Balancing via Reinforcement Learning”. In: *IEEE WCNC*. 2021, pp. 1–7.

[13] Richard S Sutton et al. “Policy gradient methods for reinforcement learning with function approximation.” In: *NIPs*. Vol. 99. Citeseer. 1999, pp. 1057–1063.

[14] Volodymyr Mnih et al. “Human-level control through deep reinforcement learning”. In: *nature* 518.7540 (2015), pp. 529–533.

[15] V. François-Lavet et al. “An introduction to deep reinforcement learning”. In: *Foundations and Trends® in Machine Learning* 11.3-4 (2018), pp. 219–354.

[16] S. Adam, L. Busoniu, and R. Babuska. “Experience Replay for Real-Time Reinforcement Learning Control”. In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 42.2 (2012), pp. 201–212.

[17] Thanh Thi Nguyen, Ngoc Duy Nguyen, and Saeid Nahavandi. “Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications”. In: *IEEE transactions on cybernetics* 50.9 (2020), pp. 3826–3839.

[18] Hado Van Hasselt, Arthur Guez, and David Silver. “Deep reinforcement learning with double q-learning”. In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 30. 1. 2016.

[19] S. Fujimoto, H. Van Hoof, and D. Meger. “Addressing function approximation error in actor-critic methods”. In: *arXiv preprint arXiv:1802.09477* (2018).

[20] Olivier Delalleau et al. “Discrete and continuous action representation for practical rl in video games”. In: *arXiv preprint arXiv:1912.11077* (2019).

[21] Rohan Dubey, Renuka Loka, and Alivelu Manga Parimi. “Maintaining the Frequency of AI-based Power System Model using Twin Delayed DDPG (TD3) Implementation”. In: *2022 2nd International Conference on Power Electronics & IoT Applications in Renewable Energy and its Control (PARC)*. IEEE. 2022, pp. 1–4.

[22] Nicholas Baard and Terence L van Zyl. “Twin-Delayed Deep Deterministic Policy Gradient Algorithm for Portfolio Selection”. In: *2022 IEEE Symposium on Computational Intelligence for Financial Engineering and Economics (CIFEr)*. IEEE. 2022, pp. 1–8.

[23] Mazen Shehab, Ahmed Zaghloul, and Ayman El-Badawy. “Low-Level Control of a Quadrotor using Twin Delayed Deep Deterministic Policy Gradient (TD3)”. In: *2021 18th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE)*. IEEE. 2021, pp. 1–6.

[24] Shuangxia Bai et al. “UAV maneuvering decision-making algorithm based on twin delayed deep deterministic policy gradient algorithm”. In: *Journal of Artificial Intelligence and Technology* 2.1 (2022), pp. 16–22.

[25] Zemin Eitan Liu et al. "An Intelligent Energy Management Strategy for Hybrid Vehicle with irrational actions using Twin Delayed Deep Deterministic Policy Gradient". In: *IFAC-PapersOnLine* 54.10 (2021), pp. 546–551.

[26] T. Lillicrap et al. "Continuous control with deep reinforcement learning". In: *arXiv preprint arXiv:1509.02971* (2015).

[27] S.L. Ho and M. Xie. "The use of ARIMA models for reliability forecasting and anal- ysis". In: *Computers Industrial Engineering* 35.1 (1998), pp. 213–216. ISSN: 0360-8352. DOI: https : / / doi . org / 10 . 1016 / S0360 - 8352(98) 00066 - 7. URL: https : / / www . sciencedirect . com / science / article / pii / S0360835298000667.