Journal of Computer Science

Original Research Paper

Enhancing Telecommunication Network Management with Autonomous Optimization Agents and Machine Learning

[1]Danny Adil Sibarani and [2]Evaristus Didik Madyatmadja

[1]Department of Information System Management, BINUS Graduate Program Master of Information System Management, Bina Nusantara University, Jakarta, Indonesia
[2]Department of Information Systems, School of Information Systems, Bina Nusantara University, Jakarta, Indonesia

Article history
Received: 12-08-2024
Revised: 23-09-2024
Accepted: 12-10-2024

Corresponding Author:
Danny Adil Sibarani,
Department of Information System Management, BINUS Graduate Program Master of Information System Management, Bina Nusantara University, Jakarta, Indonesia
Email: danny.sibarani@binus.ac.id

Abstract: This study explores the integration of Machine Learning (ML) and Autonomous Optimization Agents (AOAs) in the management and optimization of Radio Access Networks (RAN). The research addresses the growing challenges posed by the need for skilled network experts capable of managing and analyzing large-scale network data, including Key Performance Indicators (KPIs) and thousands of network configuration parameters. To overcome these challenges, the study proposes ML-based AOAs that autonomously monitor, manage, and optimize network performance, thereby reducing reliance on human expertise. Specifically, the study utilizes Deep Reinforcement Learning (DRL) to analyze network data and optimize key network parameters. Focusing on 4G LTE networks in a region of Indonesia, managed by a well-known operator, the study demonstrates the potential of AOAs in improving network efficiency, managing information overload, and optimizing critical KPIs. The findings highlight the significant impact of ML and AOAs on telecommunication network management, offering a more sustainable, efficient, and effective solution for RAN optimization.

Keywords: Machine Learning, Autonomous Optimization Agents, Telecommunication Networks, Radio Access Network Optimization, Deep Reinforcement Learning

Introduction

In the era of rapidly evolving digital communication, telecommunication networks have become the backbone of modern society, not only facilitating communication and data exchange but also advancing into an era of unprecedented connectivity speeds, supporting a wide range of applications and services (6G Use Cases and Analysis, 2022; D1.2 - Expanded 6G vision, use cases and societal values, 2021). This evolution has been pivotal in constructing interconnected digital networks, where every bite of data exchanged, whether in global communication or various applications and services, moves through increasingly complex telecommunication networks (Kunzmann *et al*., 2022; Qualcomm Technologies, 2015).

Particularly in the field of Radio Access Network (RAN) optimization, the demand for highly skilled experts has become increasingly crucial. These experts, often scarce, are required to manage and analyze vast amounts of network data, including Key Performance Indicators (KPIs), alarms, and thousands of network configuration parameters. Additionally, they must take actionable steps to maintain optimal network performance, ensuring smooth communication and data transmission across the network.

Despite the development of automated systems for network optimization, such as Self-Organizing Networks (SON), challenges persist. SON, codified since 3GPP Release 8 (3GPP TS 36.413, 2011; TSGS, 2022), has limitations in addressing every issue faced by networks, where each problem encountered necessitates the development of new algorithms, such as the Cell Outage Detection (COD) algorithm (De-La-Bandera *et al*., 2015). This approach, while beneficial in some contexts, may not be entirely efficient, especially considering the dynamic and complex nature of modern telecommunication networks.

The current network management approach, heavily reliant on manual intervention and continuous algorithm development, often fails to achieve optimal network performance. This inefficiency is exacerbated by the "information overflow" generated from the vast and ever-growing network data, posing a challenge to manage and analyze effectively and timely.

Given the challenges mentioned above, the transition from relying on "Network Experts" to implementing "Autonomous Optimization Agents (AOA)" based on Machine Learning (ML) is seen as a crucial forward step. These AOAs are designed to navigate "information overflow" by autonomously learning from past experiences (self-learning) and enhancing the efficiency of network performance optimization (RAN Automation | Reduce Effort on Radio Network Optimization, 2019). By employing ML-based agents, the system can analyze, learn from, and act upon various data, ensuring that the network is not only maintained in an optimal state but also adaptively evolves in response to changing demands and conditions. This approach aims to reduce the challenges faced by human experts and traditional systems, offering a more sustainable, efficient, and effective solution for RAN optimization.

The paper aims to enhance Radio Access Network (RAN) optimization by applying AI/ML techniques, including machine learning and deep reinforcement Learning:

- Implement Autonomous Optimization Agents (AOAs) using Deep Reinforcement Learning (DRL) with Deep Q-Network (DQN) to enhance RAN performance: AOAs utilize DRL, where an agent learns to make decisions based on rewards or penalties. Specifically, Deep Q-Network (DQN) is used, which employs a neural network to approximate the Q-function and optimize multiple Key Performance Indicators (KPIs) such as Session Setup Success Rate (SSSR), Session Abnormal Release Rate (SARR), Channel Quality Indicator (CQI), Handover Success Rate (HOSR) and Downlink Spectral Efficiency (DL SE). The approach involves tuning parameters like dlRsBoost to autonomously manage and improve RAN performance
- Evaluate the performance of the above approaches using a restricted telecommunication network simulated by a random forest model with historical datasets from 4G/LTE networks in Indonesia: This objective focuses on assessing the effectiveness of the AOAs and DRL techniques in a controlled environment. The simulation, using a random forest model and historical data, allows for thorough testing and refinement of the optimization methods, helping to avoid potential risks related to revenue loss, security issues, and user experience impacts that could occur if the methods were implemented directly in live networks

Literature Review

The continuous evolution of telecommunication networks, especially in the domain of Radio Access Networks (RAN), has necessitated advanced optimization techniques to manage the increasing complexity and performance demands. The integration of machine learning (ML) and data-driven approaches has become a focal point in enhancing the efficiency and adaptability of these networks. Machine Learning (ML) has emerged as a powerful tool in network management, offering the ability to analyze large datasets and make predictive decisions (RAN Automation | Reduce Effort on Radio Network Optimization, 2019). ML algorithms can autonomously learn from data, identify patterns, and make optimization decisions without human intervention. This capability is particularly beneficial in managing the "information overflow" in telecommunication networks, where traditional methods fall short.

Optimization Challenges in Machine Learning

The integration of machine learning in network optimization introduces several challenges, particularly concerning the scalability and efficiency of algorithms. A comprehensive review has systematically examined optimization methods used in machine learning, focusing on their application in increasingly complex and data-rich environments. This review explored common optimization challenges, such as the exponential growth of data and model complexity, and proposed various methods to address these issues. This study is essential for understanding the underlying challenges of applying machine learning to network optimization and lays the groundwork for further research in developing more robust and efficient optimization techniques (Sun *et al*., 2020).

While supervised learning has been extensively utilized in networking research, there is also a growing trend toward employing unsupervised learning techniques to improve network performance (Usama *et al*., 2019). Additionally, Reinforcement Learning (RL) is gaining momentum due to its ability to make decisions in dynamic and complex environments by learning optimal strategies through trial and error. RL is particularly valuable in scenarios where traditional supervised and unsupervised methods fall short, as it can continuously adapt to changing network conditions and optimize performance without requiring labeled data (Sun *et al*., 2020).

Recent advancements in both machine learning and reinforcement learning have led to several significant implementations in network optimization.

Recent studies on coverage and capacity optimization in wireless networks have developed and compared advanced optimization methods, including Deep Deterministic Policy Gradient (DDPG) and Multi-objective Bayesian Optimization (BO). These studies underscore the effectiveness of machine learning approaches in enhancing the self-optimization capabilities of wireless networks, which is crucial for advancing autonomous network management systems (Dreifuerst *et al*., 2021). While these studies highlight the significant evaluation requirements of DDPG, our paper introduces autonomous optimization agents using Deep Reinforcement Learning (DRL). This approach offers a comprehensive view of how autonomous systems can optimize network parameters with potentially different learning strategies and agent architectures, providing new insights into how these agents might outperform or complement existing methods.

Mobility Robustness Optimization (MRO) in Dynamic Small-Cell Networks, the MRO algorithm operates in two stages: Topology adaptation, where it gains prior knowledge to estimate optimal handover parameters, and mobility adaptation, where it refines these parameters using reinforcement learning based on real-time data. The study demonstrated that this approach significantly reduces adaptation time and enhances user satisfaction compared to traditional methods, highlighting the need for adaptive algorithms in managing the complexities of modern small-cell networks (Nguyen and Kwon, 2021). At the same time, the studies mentioned focus on optimizing a single KPI-handover performance approach can be designed to optimize a range of KPIs simultaneously. By defining a reward function that incorporates multiple KPIs (e.g., Session Setup Success Rate, Session Abnormal Release Rate, Handover Success Rate, Channel Quality Indicator, Downlink Spectral Efficiency), a DRL agent can learn policies that effectively balance these metrics.

Machine Learning in Self-Organizing Networks (SON), A comprehensive survey conducted and reviewed the application of machine learning algorithms in SON over the past fifteen years. The survey provided a detailed analysis of various machine learning techniques used in SON, categorizing them based on their learning solutions and self-organizing use cases. It also offered practical guidelines for selecting the appropriate machine learning algorithms depending on specific SON metrics. The survey emphasized the growing necessity of integrating more intelligent algorithms into SON to cope with the increasing complexity of modern cellular networks, especially as the industry moves towards 5G and beyond. The findings from this survey highlight the critical role of machine learning in enabling fully autonomous and flexible network management (Klaine *et al*., 2017). Building on these insights, this study will explore how Deep Reinforcement Learning (DRL) as an Autonomous Optimization Agent (AOA) supports such fully autonomous and adaptive network management.

Auto encoder-based frameworks for self-organizing networks, in the development of more intelligent and autonomous networks, auto-encoder-based machine learning frameworks have shown promise, particularly in Self-Organizing Networks (SON). A study introduced an Auto Encoder (AE)--based framework for cell outage detection, demonstrating superior accuracy over traditional machine learning approaches. This framework enhances the self-optimization, self-configuration, and self-healing capabilities of SON, making auto encoders a critical component in the evolution of intelligent network management systems (Asghar *et al*., 2019). Although auto-encoder-based approaches are promising for intelligent and autonomous networks, leveraging Deep Reinforcement Learning (DRL) can further enhance SON by providing more autonomous, adaptive, and optimized network management.

Resource management in Heterogeneous Networks (HetNets), and Heterogeneous Networks (HetNets) are crucial for boosting the capacity of 5G networks but introduce complexities due to interference between small cells and macrocells. A survey reviewed the application of deep reinforcement learning (DRL) in resource management for 5G HetNets, covering areas such as energy harvesting, network slicing, and coordinated multipoint transmission. The study provided a comparative analysis of various DRL-based resource management schemes and identified research gaps and future directions. This highlights the importance of DRL in managing resources in complex multi-tier network environments (Lee and Qin, 2019; Shi *et al*., 2020).

In Adaptive Resource Management in Radio Access Networks, the variability in traffic within Radio Access Networks (RAN) makes fixed network configurations inadequate for achieving optimal performance. A recent study explored the use of Deep Reinforcement Learning (DRL) to create an intelligent controller that does not require extensive domain knowledge of the RAN. This controller was tested in a lab environment with real eNodeB hardware and smartphones, demonstrating that DRL can significantly improve network performance. The study highlights the potential of DRL in enabling self-organizing, adaptive network management that operates autonomously around the clock, addressing the limitations of traditional rule-based and offline tuning approaches (Chen *et al*., 2021).

Toward Intelligent Network Optimization in Wireless Communication, the reviewed literature highlights the growing role of machine learning in addressing the challenges of modern telecommunication networks. The integration of machine learning techniques such as reinforcement learning and neural networks into network management processes demonstrates significant potential for enhancing network performance and efficiency. These advancements are particularly relevant for the development of Autonomous Optimization Agents (AOA), which aim to replace traditional network management approaches with more intelligent, machine-driven solutions (Zhang *et al*., 2019).

An overview of deep reinforcement learning for spectrum sensing in cognitive radio networks. This literature highlights the Recent advancements in Deep Reinforcement

Learning (DRL) that have shown promising results in optimizing resource management in telecommunication networks, particularly in Cognitive Radio Networks (CRNs). Techniques like Deep Q-Learning (DQN) have been used to improve spectrum sensing by enabling efficient detection of available spectrum, even in noisy environments. These methods leverage models like Markov Decision Processes (MDPs) to handle uncertainty and improve decision-making. Building on this foundation, our research integrates DQN with Autonomous Optimization Agents (AOAs) to optimize Radio Access Networks (RANs), providing a novel approach to enhance Key Performance Indicators (KPIs) in 4G LTE networks (Obite *et al*., 2021).

Deep Reinforcement Learning based Dynamic Reputation Policy in 5G based vehicular communication networks: The study on Deep Reinforcement Learning (DRL) based Dynamic Reputation Policy in 5G Vehicular Communication Networks demonstrates the potential of DRL in optimizing network operations and decision-making processes in real-time. While it primarily focuses on vehicular communication, the approach highlights how DRL, particularly the Deep Q-Network (DQN) algorithm, can effectively manage complex and dynamic environments. Traditional machine learning methods face challenges in such scenarios, but DRL's ability to learn optimal policies through trial and error has proven successful in optimizing network performance (Gyawali *et al*., 2021). Building on these findings, our study applies the DQN algorithm within Autonomous Optimization Agents (AOA) to optimize the Radio Access Network (RAN) performance, providing a novel approach to dynamic parameter adjustment in telecommunications networks.

Deep Reinforcement Learning based Wireless Network Optimization: A Comparative Study; The comparative study on Deep Reinforcement Learning (DRL) methods for wireless network optimization highlights the effectiveness of various DRL algorithms, such as Deep Deterministic Policy Gradient (DDPG), Neural Episodic Control (NEC) and Variance Based Control (VBC), in handling dynamic network environments. These algorithms are shown to optimize network operations like power control, resource management, and KPI improvements. DDPG, in particular, excels in continuous action spaces, making it a strong candidate for optimizing wireless network parameters. NEC, while offering fast convergence, struggles with large action spaces and VBC demonstrates promise in distributed multi-agent environments (Yang *et al*., 2020b). Our study leverages these insights by implementing Deep Q-Network (DQN) within Autonomous Optimization Agents (AOA) to optimize Radio Access Network (RAN) performance dynamically. Building on the success of DRL techniques in wireless networking, our approach addresses the unique challenges of real-time parameter tuning and performance optimization in RAN systems.

Deep Reinforcement Learning Based Energy-Efficient Resource Management for Social and Cognitive Internet of Things: Deep reinforcement learning (DRL) has been widely adopted to optimize resource management in dynamic wireless environments, including the Internet of Things (IoT). A notable study by Yang *et al*. (2020a) introduces a coordinated multi-agent DRL approach to manage resources in social and cognitive IoT networks. This study formulates resource management as a multi-agent reinforcement learning problem, where each device intelligently optimizes its Radio Block (RB) assignment and power control strategy. Key techniques such as Prioritized Experience Replay (PER) and coordinated learning are used to improve learning efficiency and network performance. The proposed DRL-based method demonstrates significant improvements in energy efficiency and Quality of Service (QoS) compared to traditional methods. These techniques align with our research, where DRL and Autonomous Optimization Agents (AOA) are leveraged for dynamic RAN optimization, providing insights into how DRL can be adapted to manage complex network parameters and improve system performance under varying conditions.

Deep reinforcement learning-based service-oriented resource allocation in smart grids. This study has applied Deep Reinforcement Learning (DRL) to optimize resource allocation in dynamic environments. A study by (Xi *et al*., 2021) proposed a service-oriented resource management framework using DRL to optimize communication, computing, and caching resources in smart grids, focusing on meeting diverse delay requirements. The DRL-based algorithm utilizes a polling mechanism to adapt resource scheduling based on service needs, achieving significant improvements in cache hit rates and reducing delays. This approach highlights the potential of DRL in managing complex resource allocation problems, providing valuable insights for its application in optimizing network performance in various domains, including telecommunications.

A study by Tang *et al*. (2020) explores deep reinforcement learning (DRL) for dynamic uplink/downlink resource allocation in high mobility 5G Heterogeneous Networks (HetNet), focusing on users like vehicles and UAVs. The proposed method leverages DRL and deep neural networks to predict traffic and channel conditions, allowing adaptive changes to Time Division Duplex (TDD) configurations. This approach significantly improves network throughput and reduces packet loss rates compared to traditional and shallow Q-learning methods. The research highlights the effectiveness of DRL in optimizing resource allocation in complex, high-mobility environments, which is closely aligned with our focus on intelligent resource management in dynamic networks.

Path planning of coastal ships based on optimized DQN reward function: This study applies Deep Reinforcement Learning (DRL) to optimize navigation in dynamic coastal environments. The research, proposed by (Guo *et al*., 2021), focuses on enhancing traditional path planning algorithms using an optimized Deep Q-Network (DQN), improving the convergence speed and safety of coastal ship navigation. Key improvements include the introduction of potential function rewards, reward areas near target points, and penalty areas near obstacles, which guide the ship toward its destination while avoiding obstacles efficiently. This approach significantly reduces path planning time, increases stability, and ensures compliance with maritime safety regulations. The study underscores the potential of DRL in handling real-time decision-making in dynamic, uncertain environments, aligning with our journal's focus on intelligent decision-making systems and optimization techniques in complex scenarios.

Autonomous Navigation of Robots: Optimization with DQN: This study by Escobar-Naranjo *et al*. (2023) explores the use of the Deep Q-Network (DQN) algorithm to improve the autonomous navigation of mobile robots. This research focuses on enabling robots to make optimal path-planning decisions in real time, utilizing reinforcement learning to avoid obstacles and reconfigure routes dynamically. By integrating neural networks with sensory inputs and the Robot Operating System (ROS), the system learns from its environment and optimizes navigation efficiency. The study demonstrates the DQN algorithm's potential in various applications, such as manufacturing, logistics, and search and rescue operations, emphasizing its effectiveness in complex environments. These findings align with our journal’s focus on intelligent decision-making systems and resource management, as the application of DQN in robotics highlights its ability to optimize processes and enhance system performance in real-time scenarios.

A study titled "Comparison of On-Policy Deep Reinforcement Learning A2C with Off-Policy DQN in Irrigation Optimization" (Alibabaei *et al*., 2022) applied reinforcement learning to optimize irrigation scheduling for a tomato crop in Portugal. The research compared two approaches: The off-policy Deep Q-Network (DQN) and the on-policy Advantage Actor-Critic (A2C). Both models showed a significant reduction in water consumption compared to traditional threshold-based irrigation methods, with slight variations in net yield. This study demonstrates how reinforcement learning can adapt irrigation practices based on environmental changes, such as rainfall variability, allowing for more efficient water management without compromising crop productivity. The insights from this study are valuable to our journal as they showcase how reinforcement learning can be utilized for intelligent decision-making in real-time, dynamic environments, which is applicable not only in agriculture but also in broader resource management scenarios.

The literature review above highlights the significant role of Machine Learning (ML) and Autonomous Optimization Agents (AOAs) in advancing Radio Access Networks (RAN). It underscores the challenges and opportunities associated with integrating ML, particularly Reinforcement Learning (RL) and Deep Reinforcement Learning (DRL), into network optimization. Recent studies demonstrate that DRL can effectively manage complex scenarios such as coverage, capacity, mobility robustness, and resource allocation in heterogeneous networks. While autoencoder-based frameworks have shown promise in self-optimization and network management, DRL offers enhanced adaptability and performance. The review identifies gaps and future research directions, emphasizing the need for intelligent, autonomous solutions to address the increasing complexity of modern telecommunication networks.

Problem Formulation

The growing complexity of Radio Access Network (RAN) optimization in modern telecommunication systems presents significant challenges due to the scarcity of skilled experts capable of managing and analyzing vast and intricate datasets. With the proliferation of Key Performance Indicators (KPIs), alarms, and a myriad of network parameters, the issue of "information overflow" has become increasingly prominent. To address these challenges, this study proposes the replacement of traditional "Network Experts" with Machine Learning (ML)-based Autonomous Optimization Agents (AOAs). These agents are designed to manage and optimize network parameters more efficiently by leveraging self-learning capabilities from past experiences with advanced AI/ML techniques.

In this study, the simulation environment is designed to model a telecommunication network without directly interacting with a live system. Instead of real-world implementation, a Random Forest model is employed to simulate the network’s behavior, allowing for safe and controlled testing of optimization strategies without compromising network integrity. By combining the Deep Q-Network (DQN) algorithm with the Random Forest model, the study establishes a robust and adaptable simulation environment. The DQN algorithm is chosen for its capability to manage complex decision-making and adapt over time, while the Random Forest model is selected for its accuracy and stability in predicting network performance. These choices ensure a reliable and effective simulation of telecommunication network management, enabling safe and insightful testing of optimization strategies.

About the Datasets

The dataset for this study is derived from a

DOI: 10.3844/jcssp.2025.322.335

telecommunication network in one region of Indonesia, managed by a well-known operator. This region was selected due to the wide variation in Key Performance Indicators (KPIs) it displays, providing a strategic advantage for analyzing different network behaviors under diverse conditions. The data, specifically related to 4G (LTE) network technology, was collected over a three-month period from July 1, 2023, to September 30, 2023.

Dataset Overview: The dataset includes:

- Key Performance Indicators (KPIs): These are metrics used to evaluate the performance aspects of the telecommunication network. The KPIs are Session Success Setup Rate (SSSR), Session Abnormal Release Rate (SARR), Handover Success Rate (HOSR), Channel Quality Indicator (CQI), and Downlink (DL) spectral efficiency
- Database Parameters: Specific to this study are dlRsBoost, which are frequently adjusted parameters for Radio Access Network (RAN) optimization

Data Analysis: The statistical summary of the dataset presented in Table (1), which includes dlRsBoost, SSSR, SARR, CQI, HOSR, and DLSE, provides insights into the performance metrics after normalization and outlier removal. The dlRsBoost values are normalized by dividing by the maximum value of 1600, resulting in a mean of 0.70, with a range from 0.44 to 1 and a standard deviation of 0.09. For SSSR and SARR, the metrics are highly stable with means of 0.990 and 0.998 respectively, indicating consistent performance with low standard deviations (0.02 and 0.01). The CQI, normalized by dividing by 15, has a mean of 0.65, indicating good channel quality, with values ranging from 0.09 to 0.99 and a standard deviation of 0.10. HOSR shows a mean of 0.98 and a maximum of 1, reflecting high handover success rates with minimal variation. DLSE, normalized by dividing by 10, has a mean of 0.19 and a standard deviation of 0.07, with values ranging from 0 to 1.37, illustrating variable downlink efficiency.

Table 1: Statistical summary of Key Performance Indicators (KPIs) and parameter

	dlRs Boost	SSSR	SAR R	CQI	HOS R	DLS E
count	4,277,353					
mean	0.70	0.990	0.990	0.650	0.980	0.190
std	0.09	0.020	0.010	0.100	0.050	0.070
Min	0.44	0.000	0.000	0.090	0.000	0.000
25%	0.63	0.990	0.998	0.591	0.984	0.139
50%	0.63	0.995	0.999	0.657	0.994	0.182
75%	0.81	0.997	0.999	0.720	0.996	0.228
max	1.00	1.000	1.000	0.990	1.000	1.370

These datasets will be utilized as follows:

- A Random Forest model will serve as a representation of the telecommunication network, instead of directly connecting to the network. This model will use 80% of the data for training and 20% for testing and will be employed to predict Key Performance Indicators (KPIs)
- The top 20 cells with the worst performance on the last day will be selected and fed into a Deep Reinforcement Learning (DRL) model for further testing

In this study, we chose to use daily data for regular optimization instead of focusing exclusively on peak-hour conditions for several reasons:

- Regular optimization focus: The primary goal of the study is to develop and test optimization strategies that can be applied consistently across various times and conditions throughout the day. Using daily data ensures that the optimization model can address a wide range of network scenarios, not just those experienced during peak hours
- Comprehensive Evaluation: Daily data provides a more comprehensive view of network performance across different times and conditions, rather than concentrating solely on peak hours. This approach allows for a broader evaluation of how optimization strategies perform under typical operational conditions, which is crucial for developing robust and adaptable solutions
- Realistic Network Management: Network management often involves dealing with a variety of conditions throughout the day, not just during peak traffic hours. By using daily data, the study reflects the real-world challenges faced by network operators who must optimize performance across all hours, including both peak and off-peak times
- Avoiding peak-specific bias: Focusing exclusively on peak-hour conditions could lead to optimization strategies that are overly tailored to high-stress scenarios, potentially neglecting the needs of the network during other times. Using daily data helps ensure that the optimization strategies are balanced and effective across all conditions

Conceptual Framework

Figure (1) Shown block diagram of an autonomous agent when the Autonomous Optimization Agent (AOA) uses a Deep Q-Network (DQN) algorithm as a decision-making process and epsilon-greedy policy as a component that defines the behavior of an agent by specifying the action to be taken in each state. For the Environment, instead of directly interfacing with the live telecommunication network, a Machine Learning (ML) model, specifically Random Forest, is utilized to predict critical KPIs such as Session Success Setup Rate (SSSR), Session Abnormal Release Rate (SARR), Handover Success Rate (HOSR), Channel Quality Indicator (CQI) and Downlink (DL) Spectral Efficiency based on input parameters like Band and dlRsBoost. This approach allows for a controlled and secure environment for testing and refining the optimization strategies without risking network integrity. The study by (Breiman, 2001) highlights the effectiveness of Random Forests as a tool for prediction, emphasizing that they do not overfit due to the Law of Large Numbers. The study notes that the randomness injected into the model contributes to its accuracy as a classifier and regressor, providing insight into the model's predictive capabilities through the strength of individual predictors and their correlations.

The primary objective of this study is to improve the overall Quality of Service by fine-tuning the dlRsBoost parameter, thereby enhancing key KPIs. The ultimate goal is to ensure that more sessions are successfully established (higher SSSR), session attempts are managed more efficiently (lower SARR), handovers occur seamlessly (higher HOSR), users experience better channel quality (higher CQI) and the network's spectral efficiency is maximized, leading to better utilization of available bandwidth and resources.

Markov Decision Processes (MDP)

In Fig. (1), the interaction between the agent and the environment, including actions, states, and rewards, is structured within the MDP framework, a key component of reinforcement learning. MDP is defined by a set of states (S), a set of actions (A), a transition function (T), and a reward function (R). The transition function $T(s, a, s')$ represents the probability of moving from state s to state s' after taking action a, and the reward function $R(s, a, s')$ provides the reward received after transitioning. The goal of an MDP is to find a policy $\pi(s)$, which tells the agent the best action to take in each state to maximize cumulative rewards over time (Gridin, 2021; Vanneschi and Castelli, 2018).

State: In this study, the observed states are key performance indicators (KPIs) of the telecommunication network, including SSSR, SARR, HOSR, CQI, and DL Spectral Efficiency, which are predicted using a Random Forest model.

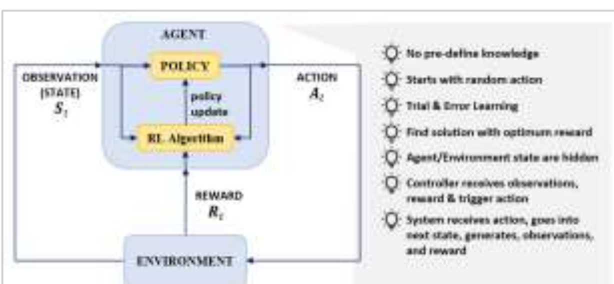

Fig. 1: Block diagram of autonomous optimization agent

Action: The action in this study involves setting or adjusting the dlRsBoost value. The Random Forest model, representing the telecommunication network, predicts the KPIs based on the selected dlRsBoost value. The dlRsBoost parameter can take one of six possible values: [700, 1000, 1177, 1300, 1477, 1600], providing the agent with six distinct actions to choose from.

Reward: For each action taken, if a KPI meets or exceeds its target optimization value, the agent receives a reward of +1; otherwise, the reward is -1.

Here is the MDP algorithm for agent-environment interaction, outlining actions, states, and rewards:

```
state_target = [SSSR_tgt, SARR_tgt, HOSR_tgt, CQI_tgt,
DLSE_tgt]

function step(action, state_target):
    reward = 0
    done = False
    done_reason = None

    dlRsBoost = [700, 1000, 1177, 1300, 1477, 1600]
    state =
get_kpi_prediction_from_random_forest_ml(dlRsBoost[acti
on])

  for i in range(len(['SSSR', 'SARR', 'HOSR', 'CQI',
'DLSE'])):
    if state[i] > state_target[i]:
      reward += 1
    Else:
      reward -= 1

if reward==5:
    done = True
    done_reason = "All KPIs >= previous"

if all_action_already_taken() and reward<5:
    done = True
    done_reason = "All steps complete!"
    state = the_best_solution()

  return (state, reward, done, info={done_reason,
cummulative_reward, steps})
```

Deep Q-Network

The reinforcement learning algorithm used in this study is Deep Q-Network (DQN) (Mnih *et al*., 2015), an extension of the classical Q-learning algorithm (Khenak, 2010), that utilizes neural networks to approximate the action-value function. Developed by Google's DeepMind in 2013, DQN effectively bridges the gap between reinforcement learning and deep learning, enabling the practical solution of complex problems in various environments.

State and Action Representation: In DQN, the environment is modeled as an MDP where the agent interacts with it by observing States (s), selecting Actions (a), receiving Rewards (r), and transitioning to new States (s′).

Q-Function: The Q-function $Q(s, a)$, which estimates the expected cumulative reward of taking action in state s and following the optimal policy thereafter, is central to DQN. The Q-function is learned through experiences and approximated using a neural network in DQN (Vanneschi and Castelli, 2018).

In Fig. (2), the neural network models the $Q(s, a)$ function by taking the input state s and outputting a vector a, where each value corresponds to the Q-value of an action a_i for that state. This approach is the foundation of the Deep Q-Network (DQN), which uses a neural network to determine the optimal actions for the agent.

Bellman Equation

The Bellman equation is used to iteratively update the Q-values based on the observed rewards and transitions. This is where the MDP comes into play: The Q-value is updated according to the expected future rewards, which are determined by the transition and reward functions of the MDP (Vanneschi and Castelli, 2018):

$$Q(s, a) = Q(s, a) + \alpha[r + \gamma max_{a'} Q(s', a') - Q(s, a)]$$

This equation updates the Q-value $Q(s, a)$ for a state-action pair by adjusting it based on the immediate reward r, the discounted estimate γ of the optimal future reward $max_{a'} Q(s', a')$, and the learning rate α to improve the agent's decision-making over time.

Replay Buffer

As the agent explores the environment, it stores experiences (s, a, r, s') in a replay buffer. These experiences represent the transitions defined by the MDP and are used to train the Q-network. The Replay Buffer serves an important role by allowing the agent to store and later revisit these past experiences during training. This process, called "experience replay," involves sampling random batches of these stored experiences to train the neural network.

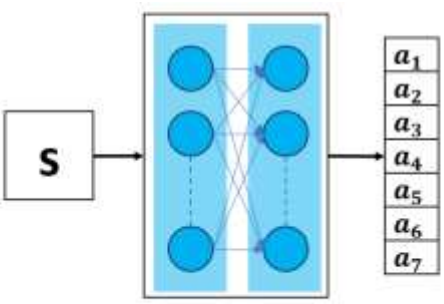

Fig. 2: Neural network as the *Q*(*s*, *a*) function

By doing so, the agent can learn from a diverse set of past experiences rather than relying solely on the most recent interactions. This helps break the temporal correlations between consecutive experiences, leading to more stable and efficient learning. The randomness in selecting batches from the Replay Buffer ensures that the agent generalizes better and prevents overfitting to recent experiences (Gridin, 2021.

In Fig. (3), the samples collected during exploration are stored in the Replay Buffer, which later provides random batches of these (s, a, r, s') samples for neural network training.

Epsilon-Greedy Policy

This study employs the Epsilon-Greedy Policy in reinforcement learning to balance exploration (trying new actions) and exploitation (using known information to maximize rewards). Initially, the agent explores the environment by trying different actions. After this exploration phase, the agent switches to exploitation mode, where it selects the action that yields the highest reward during exploration, assuming it remains the best choice. The "epsilon" in the Epsilon-Greedy Policy represents a small probability that the agent will continue to explore during exploitation, ensuring it doesn't overlook potentially better options (Gridin, 2021).

The Epsilon-Greedy Policy is defined with the following mathematical formula (Gridin, 2021; Vanneschi and Castelli, 2018):

- With probability ϵ
  Choose a random action $a \in A$
  where A is the set of all possible actions
- With probability $1 - \epsilon$
  $a = \arg max_{a'} Q(s, a)$
  where $Q(s, a)$ is the estimated value of taking action a in state s

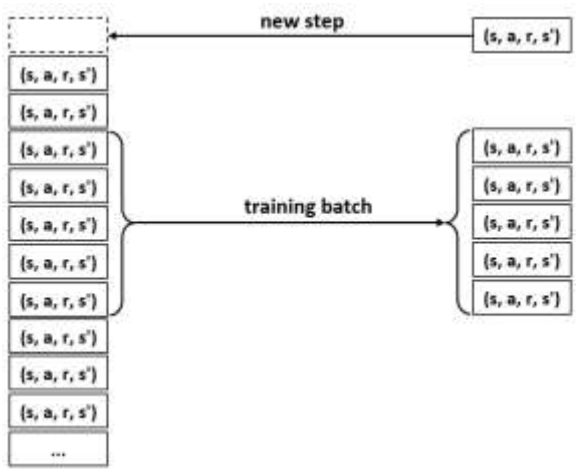

Fig. 3: Replay buffer

Here, ϵ is a parameter between 0-1 that controls the trade-off between exploration (choosing a random action) and exploitation (choosing the best-known action).

Random Forest Regressor as Network Telecommunication

In Fig. (1), the "environment" denotes a telecommunication network. Although direct connectivity to this network is technically feasible, the need to preserve network integrity compels the substitution of direct interactions with a Random Forest model. This model is utilized to simulate the network's behavior by predicting all relevant Key Performance Indicators (KPIs). The choice of the Random Forest approach is grounded in its proven effectiveness as a predictive tool, as extensively documented in studies by Breiman (2001). Employing this method ensures the maintenance of network stability and security, while also enabling the testing and refinement of optimization strategies.

The methodology of this study involves implementing a Random Forest model to predict KPIs and using a deep reinforcement learning approach to optimize network parameters. The approach aims to address the following objectives:

- Prediction: The Random Forest model predicts key performance indicators (KPIs) such as SSSR, SARR, HOSR, CQI, and DL SE based on input parameters like Band and dlRsBoost. This model will be trained and tested on historical network data to provide reliable KPI forecasts
- Optimization: Using the Deep Q-Network (DQN) algorithm, the study will optimize the dlRsBoost parameter to enhance KPI values. The DQN will be trained to maximize rewards based on the KPI predictions from the Random Forest model. The learning process involves adjusting dlRsBoost values and observing their impact on the KPIs to determine the optimal setting

Deep Reinforcement Learning Algorithm

Here is the algorithm to train the Deep Reinforcement Learning (DRL) model, bringing together all the key concepts discussed above:

```
def drl_train(episodes, epsilon):
  replay_buffer_init()
  evaluation_init() # exploratoin, exploitation, scores, steps

for episode in range(episodes):
  state = state_init()
  done = False
  While not done:
  if rand() < epsilon:
  action = random(action_space)
           exploration += 1
    Else:
      q_values = model.predict()
      action = q_values[0]
      exploitation += 1

      next_state, reward, done, info = step(action)
      replay_buffer_append(state, action, reward, next_state)
      state = next_state
      evaluation_update()

  train_dqn_model_with_label_from_bellman_equation()
```

Materials and Methods

Simulation Environment

The study focuses on modeling a telecommunication network in a controlled simulation environment to avoid potential risks associated with direct interactions with a live system. The simulation incorporates a Random Forest model to represent the telecommunication network's behavior. This model ensures a stable and accurate depiction of network dynamics by predicting Key Performance Indicators (KPIs) such as Session Success Setup Rate (SSSR), Session Abnormal Release Rate (SARR), and Downlink Spectral Efficiency (DLSE). The Random Forest model was selected for its robustness and ability to handle complex, high-dimensional data, ensuring accurate and reliable predictions.

To enhance decision-making and adaptability within the simulation, the Deep Q-Network (DQN) algorithm was integrated. DQN is particularly well-suited for managing complex, dynamic systems such as telecommunication networks due to its capacity for continuous learning and adaptation in real-time scenarios.

Deep Q-Network (DQN) Implementation

The Deep Q-Network (DQN) was implemented using TensorFlow and Keras libraries. The neural network architecture is designed to map the state space to the action space, enabling efficient decision-making. Key architectural features include:

- Input Layer: Configured to match the dimensions of the state space
- Hidden Layers: Two fully connected layers, each with 24 neurons, using the ReLU (Rectified Linear Unit) activation function to introduce non-linearity and improve learning
- Output Layer: The output layer corresponds to the action space, with a linear activation function to predict the Q-values for each action

The DQN was trained with the following hyperparameters:

- Optimizer: Adam optimizer with a learning rate of 0.001
- Loss Function: Mean Squared Error (MSE), used to minimize the difference between predicted and target Q-values

Performance Evaluation

The performance of the trained DQN agent was assessed using metrics such as episode rewards and the number of steps required to achieve the goal. Episode rewards measured the agent's cumulative performance, while the number of steps provided insight into the agent's efficiency in achieving its objective. A dataset of 100 simulation episodes was generated, with each episode capturing performance metrics to evaluate the agent's effectiveness in optimizing network parameters.

Statistical Analysis

Statistical analysis was conducted using Python's SciPy library to determine the significance of performance improvements achieved by the DQN-based optimization strategy. Metrics were compared across episodes to quantify improvements in decision-making and network optimization.

Results

In Fig. (4), the results of a 100-episode experiment are presented. The performance of the Deep Reinforcement Learning (DRL) model trained using the Deep Q-Network (DQN) algorithm is analyzed based on the following metrics: Episode scores and steps. In the initial episode, the Deep Reinforcement Learning (DRL) model trained using the Deep Q-Network (DQN) algorithm demonstrates its capability to meet the performance Key Performance Indicators (KPIs). Despite potentially starting with minimal prior knowledge, the DRL agent quickly learns and adapts to the environment, achieving a satisfactory performance level from the outset.

Furthermore, the subsequent episodes show consistent or improved performance, indicating that the initial success of the DRL DQN model is not merely a fluke. Instead, it demonstrates the model's ability to generalize learned knowledge and strategies across different scenarios. This rapid learning and consistent performance bode well for the model's future iterations and applications, suggesting that it has the potential to achieve even higher levels of performance with further training and optimization.

In Fig. (5), The provided data represents the utilization of the epsilon-greedy policy by the Deep Q-Network (DQN) algorithm, specifically in terms of exploration and exploitation actions taken during each episode of training. In the early episodes, there is a higher frequency of exploratory actions, this shows that the model initially prioritizes exploration to gather information about the environment. As training progresses, there is a gradual shift towards exploitation, with an increase in exploitative actions and a corresponding decrease in exploratory actions. This transition indicates that the model leverages learned knowledge to maximize rewards, demonstrating its ability to adapt its strategy based on experience. The balance between exploration and exploitation observed in the data reflects the model's dynamic learning process and its capacity to refine decision-making strategies over time.

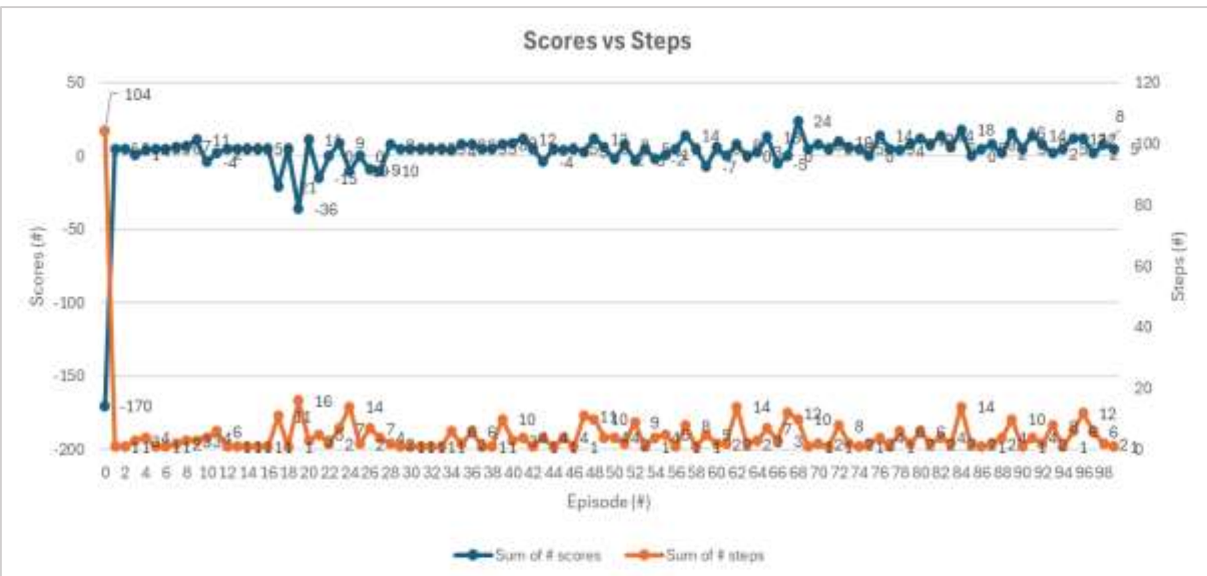

Fig. 4: DLR learning curve based on scores and steps

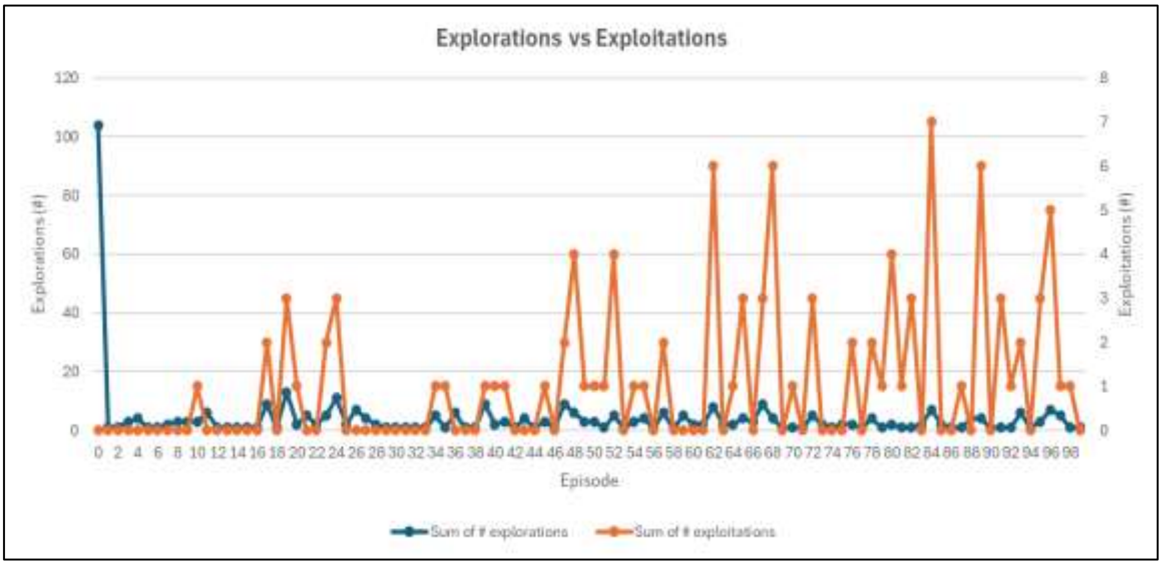

Fig. 5: DLR epsilon-greedy policy (exploration vs exploitation)

Model Performance: The data provided showcases the performance of a Deep Reinforcement Learning (DRL) model trained with the Deep Q-network (DQN) algorithm across multiple episodes. The model's performance is evaluated based on cumulative scores, steps taken, explorations, and exploitations. Analysis reveals fluctuations in scores and steps, indicating variability in the model's effectiveness in completing the task. This variability suggests a dynamic learning process influenced by the complexity of the environment and the model's decision-making efficiency. Furthermore, the balance between exploration and exploitation shifts over time, with early episodes characterized by higher exploration and later episodes emphasizing the exploitation of learned knowledge. This transition highlights the model's adaptability and ability to refine its strategies based on experience.

Overall, the results provide valuable insights into the learning dynamics and decision-making processes of the DRL model. Further exploration of these metrics can inform strategies for optimizing performance and enhancing the model's effectiveness in real-world applications.

Future Improvements: To address the observed limitations and enhance the model's performance, recommend the following strategies:

- Exploration-Exploitation Trade-off Adjustment: Fine-tuning the balance between exploration and exploitation is crucial for optimizing learning. Dynamic adjustment of the exploration-exploitation trade-off, such as incorporating adaptive epsilon values or exploring alternative exploration strategies like Upper Confidence Bound (UCB), may enhance the model's ability to efficiently explore the environment while maximizing rewards
- Model Architecture Optimization: Optimizing the neural network architecture and hyperparameters, such as network depth, width, and learning rates, can significantly impact the model's learning capacity and generalization performance. Experimenting with different network architectures, including deeper or wider networks, residual connections, or attention mechanisms, may uncover more effective representations and improve learning efficiency

Discussion

This study demonstrates the potential of Deep Reinforcement Learning (DRL) for adaptive and dynamic management of telecommunication networks, particularly through the integration of the Deep Q-Network (DQN) algorithm. The results reveal that the DRL model successfully transitions from exploration to exploitation within a short time frame, allowing for effective optimization of Key Performance Indicators (KPIs). This adaptability reflects the model's ability to autonomously refine its strategies and achieve optimal decision-making without predefined knowledge or human intervention.

The findings of this study align with existing literature on DRL applications in dynamic environments, where reinforcement learning models have shown efficacy in optimizing resource allocation and decision-making under uncertainty. However, this study specifically highlights the potential of combining simulation environments with DRL for telecommunication network optimization, bridging a significant gap in practical network management applications.

A notable innovation in this research is the adaptive exploration-exploitation strategy, which allows the DRL model to balance learning new strategies and exploiting known optimal actions effectively. This strategy not only accelerates the learning process but also ensures that the model adapts to changing network dynamics, ultimately improving performance consistency and robustness.

Despite these promising results, several limitations warrant discussion. Firstly, the use of simulation environments, while safe and controlled, may not fully capture the complexities of real-world telecommunication networks. Real-time deployments often involve stochastic variables, such as unpredictable user behavior and external environmental factors, which are not addressed in this study. Secondly, although the DQN algorithm worked well in this study, its current neural network design and fixed training settings might not be sufficient for more complex 5G networks. These networks involve more variables, larger decision spaces, and dynamic environments that may require more advanced models to handle effectively.

Lastly, while the training process requires significant computational resources due to the large state and action spaces, this challenge highlights opportunities for further optimization. Future work could focus on fine-tuning the exploration-exploitation balance, refining the neural network architecture, and improving the scalability of the DRL framework to handle more complex scenarios like 5G networks. These enhancements would ensure that the model remains efficient and adaptable as it is applied to increasingly dynamic and diverse network environments.

Conclusion

This study demonstrates that Deep Reinforcement Learning (DRL) has significant potential for managing telecommunication networks more effectively and adaptively. Using the Deep Q-Network (DQN) algorithm, the DRL model does not require predefined knowledge; it begins with random actions and learns through trial and error to find the optimal solution with the highest reward. In the first episode, the DRL model focuses on exploration to understand the environment. By the second episode, the model shifts towards optimizing Key Performance Indicators (KPIs) effectively. This transition from exploration to exploitation highlights the DRL model's capability to quickly adapt and improve its performance based on accumulated experience.

The results underscore that DRL can dynamically and adaptively optimize resources, enhancing overall service quality and network reliability. This adaptability is crucial for identifying and preventing network issues, ultimately improving customer experience and network management.

A key innovation in this study is the adaptive exploration-exploitation strategy employed by the DRL model. The model's ability to refine decision-making strategies by transitioning from exploration to exploitation based on accumulated experience highlights its effectiveness and adaptability. Future research should address any limitations and focus on further improving the DRL model, particularly by fine-tuning the exploration-exploitation balance and refining the model's architecture. These advancements will support more efficient and effective management of telecommunication networks.

Future Directions

While the study demonstrates the effectiveness of the DRL model, future research should address several limitations. Key areas of focus should include:

- Fine-tuning the exploration-exploitation balance: Refining this balance further through dynamic epsilon adjustments or advanced exploration strategies (e.g., Upper Confidence Bound) could improve the model's learning efficiency
- Optimizing the model's architecture: Investigating deeper or wider neural networks, as well as modern architectural enhancements like residual connections or attention mechanisms, can significantly improve learning and generalization performance across varied network conditions
- Scalability and application to 5G networks: Future research should explore the scalability of the DRL framework in more complex environments, such as 5G networks, where diverse traffic patterns, higher data rates, and new KPIs introduce additional layers of complexity

By addressing these areas, DRL-based solutions can continue to evolve, providing more efficient and autonomous network management systems capable of handling the increasing complexity of telecommunication networks, particularly as the industry moves toward 5G and beyond.

Acknowledgment

The authors wish to express their sincere gratitude to Bina Nusantara Indonesia for allowing the conduct of this research. We are also deeply thankful to the reviewers for their insightful comments that have significantly improved our work.

Funding Information

This research received no external funding.

DOI: 10.3844/jcssp.2025.322.335

Author's Contributions

Danny Adil Sibarani: Conceptualization, methodology, formal analysis, investigation and writing original draft preparation.

Evaristus Didik Madyatmadja: Written, reviewed, edited, and supervised.

Ethics

The authors consent to address any ethical issues that may arise after the publication of this manuscript.

References

Alibabaei, K., Gaspar, P. D., Assunção, E., Alirezazadeh, S., Lima, T. M., Soares, V. N. G. J., & Caldeira, J. M. L. P. (2022). Comparison of On-Policy Deep Reinforcement Learning A2C with Off-Policy DQN in Irrigation Optimization: A Case Study at a Site in Portugal. *Computers*, *11*(7), 104. https://doi.org/10.3390/computers11070104

Asghar, M. Z., Abbas, M., Zeeshan, K., Kotilainen, P., & Hämäläinen, T. (2019). Assessment of Deep Learning Methodology for Self-Organizing 5G Networks. *Applied Sciences*, *9*(15), 2975. https://doi.org/10.3390/app9152975

Breiman, L. E. O. (2001). *Random Forests*. *45*, 5–32.

Chen, Y., Chen, J., Krishnamurthi, G., Yang, H., Wang, H., & Zhao, W. (2021). Deep Reinforcement Learning for RAN Optimization and Control. *2021 IEEE Wireless Communications and Networking Conference (WCNC)*, 1–6. https://doi.org/10.1109/wcnc49053.2021.9417275

D1.2 - Expanded 6G vision, use cases and societal values. (2021). Hexa-X.

de-la-Bandera, I., Barco, R., Munoz, P., & Serrano, I. (2015). Cell Outage Detection Based on Handover Statistics. *IEEE Communications Letters*, *19*(7), 1189–1192. https://doi.org/10.1109/lcomm.2015.2426187

Dreifuerst, R. M., Daulton, S., Qian, Y., Varkey, P., Balandat, M., Kasturia, S., Tomar, A., Yazdan, A., Ponnampalam, V., & Heath, R. W. (2021). Optimizing Coverage and Capacity in Cellular Networks using Machine Learning. *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 8138–8142. https://doi.org/10.1109/icassp39728.2021.9414155

Escobar-Naranjo, J., Caiza, G., Ayala, P., Jordan, E., Garcia, C. A., & Garcia, M. V. (2023). Autonomous Navigation of Robots: Optimization with DQN. *Applied Sciences*, *13*(12), 7202. https://doi.org/10.3390/app13127202

Gridin, I. (2021). *Practical Deep Reinforcement Learning with Python*. Springer.

Guo, S., Zhang, X., Du, Y., Zheng, Y., & Cao, Z. (2021). Path Planning of Coastal Ships Based on Optimized DQN Reward Function. *Journal of Marine Science and Engineering*, *9*(2), 210. https://doi.org/10.3390/jmse9020210

Gyawali, S., Qian, Y., & Hu, R. (2021). Deep Reinforcement Learning Based Dynamic Reputation Policy in 5G Based Vehicular Communication Networks. *IEEE Transactions on Vehicular Technology*, *70*(6), 6136–6146. https://doi.org/10.1109/tvt.2021.3079379

6G Use Cases and Analysis. (2022). NGMN.

Khenak, F. (2010). V-Learning. *2010 International Conference on Computer Information Systems and Industrial Management Applications (CISIM)*, 228–232. https://doi.org/10.1109/cisim.2010.5643660

Klaine, P. V., Imran, M. A., Onireti, O., & Souza, R. D. (2017). A Survey of Machine Learning Techniques Applied to Self-Organizing Cellular Networks. *IEEE Communications Surveys & Tutorials*, *19*(4), 2392–2431. https://doi.org/10.1109/comst.2017.2727878

Kunzmann, G., Khorsandi, B. M., Ziegler, V., Raisanen, V., Ozturk, E., Kolding, T., & Mannweiler, C. (2022). *Technology Innovations for 6G System Architecture*.

Lee, Y. L., & Qin, D. (2019). *A Survey on Applications of Deep Reinforcement Learning in Resource Management for 5G Heterogeneous Networks*. 1856–1862.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., & Hassabis, D. (2015). Human-Level Control Through Deep Reinforcement Learning. *Nature*, *518*(7540), 529–533. https://doi.org/10.1038/nature14236

Nguyen, M.-T., & Kwon, S. (2021). Machine Learning–Based Mobility Robustness Optimization Under Dynamic Cellular Networks. *IEEE Access*, *9*, 77830–77844. https://doi.org/10.1109/access.2021.3083554

Obite, F., Usman, A. D., & Okafor, E. (2021). An overview of deep reinforcement learning for spectrum sensing in cognitive radio networks. *Digital Signal Processing*, *113*, 103014. https://doi.org/10.1016/j.dsp.2021.103014

Qualcomm Technologies. (2015). *5G - Vision for the next generation of connectivity*.

RAN Automation | Reduce Effort on Radio Network Optimization. (2019). Rantcell.Com.

Shi, Y., Sagduyu, Y. E., & Erpek, T. (2020). Reinforcement Learning for Dynamic Resource Optimization in 5G Radio Access Network Slicing. *2020 IEEE 25th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, 1–6. https://doi.org/10.1109/camad50429.2020.9209299

Sun, S., Cao, Z., Zhu, H., & Zhao, J. (2020). A Survey of Optimization Methods from a Machine Learning Perspective. *IEEE Transactions on Cybernetics*, *50*(8), 3668–3681. https://doi.org/10.1109/tcyb.2019.2950779

Tang, F., Zhou, Y., & Kato, N. (2020). Deep Reinforcement Learning for Dynamic Uplink/Downlink Resource Allocation in High Mobility 5G HetNet. *IEEE Journal on Selected Areas in Communications*, *38*(12), 2773–2782. https://doi.org/10.1109/jsac.2020.3005495

TSGS. (2022). *TS 128 313 - V17.4.0 - 5G; Management and Orchestration; Self-Organizing Networks (SON) for 5G Networks (3GPP TS 28.313 Version 17.4.0 Release 17)*.

Usama, M., Qadir, J., Raza, A., Arif, H., Yau, K. A., Elkhatib, Y., Hussain, A., & Al-Fuqaha, A. (2019). Unsupervised Machine Learning for Networking: Techniques, Applications and Research Challenges. *IEEE Access*, *7*, 65579–65615. https://doi.org/10.1109/access.2019.2916648

Vanneschi, L., & Castelli, M. (2019). *Multilayer Perceptrons* (S. Ranganathan, M. Gribskov, K. Nakai, & C. Schönbach, Eds.; Vol. 1, pp. 612–620). Elsevier. https://doi.org/10.1016/b978-0-12-809633-8.20339-7

Xi, L., Wang, Y., Wang, Y., Wang, Z., Wang, X., & Chen, Y. (2021). Deep Reinforcement Learning-Based Service-Oriented Resource Allocation in Smart Grids. *IEEE Access*, *9*, 77637–77648. https://doi.org/10.1109/access.2021.3082259

Yang, H., Zhong, W.-D., Chen, C., Alphones, A., & Xie, X. (2020a). Deep-Reinforcement-Learning-Based Energy-Efficient Resource Management for Social and Cognitive Internet of Things. *IEEE Internet of Things Journal*, *7*(6), 5677–5689. https://doi.org/10.1109/jiot.2020.2980586

Yang, K., Shen, C., & Liu, T. (2020b). Deep Reinforcement Learning Based Wireless Network Optimization: A Comparative Study. *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 1248–1253. https://doi.org/10.1109/infocomwkshps50562.2020.9162925

Zhang, W., Zhang, Z., Chao, H.-C., & Guizani, M. (2019). Toward Intelligent Network Optimization in Wireless Networking: An Auto-Learning Framework. *IEEE Wireless Communications*, *26*(3), 76–82. https://doi.org/10.1109/mwc.2019.1800350