SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V12I9P106

Volume 12 Issue 9, 72-83, September 2025

Original Article

Enhancing Multi-Cell Dynamic TDD with Multi-Agent Deep Reinforcement Learning

Sarala Patchala[1], Sai Prasanth Kanuparthy[2], Vullam Nagagopiraju[3], Vijaya Babu Burra[4], Banda Snv Ramana Murthy[5], Rohini Rajesh Swami Devnikar[6]

[1]*Department of ECE, KKR & KSR Institute of Technology and Sciences, Guntur, Andhra Pradesh, India.*
[2]*RVR&JC College of Engineering, Guntur, Andhra Pradesh, India.*
[3]*Department of Computer Science and Engineering, Chalapathi Institute of Engineering and Technology, Guntur, Andhra Pradesh, India.*
[4]*Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur, Andhra Pradesh, India.*
[5]*Department of CSE-AIML, Aditya University, Surampalem, Andhra Pradesh, India.*
[6]*G H Raisoni College of Engineering and Management, Pune, India.*

[1]*Corresponding Author : saralajntuk@gmail.com*

Received: 06 July 2025 Revised: 08 August 2025 Accepted: 07 September 2025 Published: 29 September 2025

Abstract - *Dynamic Time Division Duplex (D-TDD) is an important feature in 5G and future 6G networks. It allows flexible allocation of Uplink (UL) and Downlink (DL) slots. This helps to manage traffic demands dynamically. However, two key challenges exist. First, the system determines the best TDD pattern to match user traffic. Second, cross-link interference occurs when different cells use different TDD configurations. This interference degrades network performance. The 3GPP standard does not provide an optimal method for TDD configuration. It does not solve cross-link interference issues. To address these gaps, we proposed a Multi-Agent Deep Reinforcement Learning (MADRL) approach. This approach models the TDD problem as a linear programming problem. Introduced the Multi-Agent Deep Reinforcement Learning-based 5G RAN TDD Pattern (MADRP) framework. This method is decentralized. Each cell has an independent agent that learns the best TDD configuration. The system reduces control latency and signaling overhead. The MADRP model monitors the buffer states of uplink and downlink data. It exchanges messages with neighboring cells to minimize cross-link interference. Each agent uses reinforcement learning to determine the best TDD allocation. The model adapts to traffic variations and prevents buffer overflows. It highlights the limitations of MADRP. Performance is degraded in high-interference environments. Future work will focus on implementing MADRP in real-world 5G systems. This aimed to integrate the model with OpenAirInterface (OAI) to demonstrate real-time adaptability. This will provide insights into practical deployment challenges. This research introduces a novel DRL-based TDD adaptation approach. It efficiently manages UL and DL allocation while minimizing cross-link interference. The method enhances performance in multi-cell 5G environments. It provides a scalable and effective alternative to static TDD configurations.*

Keywords - *Deep, Multi-agent, Multi-cell, TDD, Reinforcement, Resource allocation.*

1. Introduction

The rapid advancements in wireless communication have led to the emergence of 5G and future 6G networks [1]. These networks aim to provide Ultra-Reliable Low-Latency Communication (URLLC), massive Machine-Type Communications (mMTC) and enhanced Mobile Broadband (eMBB). The demand for high-speed data transfer is increasing [2]. Seamless connectivity is essential. Real-time applications like immersive holographic communication require Efficiency. The Internet of Skills and smart transportation need reliable networks [3]. These trends drive the need for adaptive resource allocation. One of the fundamental aspects of 5G New Radio (NR) is Time Division Duplex (TDD). It allows dynamic switching between UL and DL transmissions [4]. Traditional fixed TDD configurations are inefficient in handling dynamic and asymmetric traffic patterns observed in real-world networks. Video streaming generates more DL traffic [5]. Cloud-based Augmented Reality (AR) needs more UL bandwidth. Conventional TDD systems use static UL/DL slot allocation. This causes inefficient resource utilization. It leads to higher latency and increased packet loss [6]. Network performance becomes suboptimal.

D-TDD proposed to overcome these inefficiencies by dynamically adjusting the UL and DL slot allocation based on real-time traffic demands [7]. D-TDD allows base stations (gNBs) to allocate resources flexibly. It improves overall spectral Efficiency and Quality of Service (QoS) [8]. However, the implementation of D-TDD comes with significant challenges. One of the main issues is cross-link interference. It occurs when neighboring cells operate with different UL and DL patterns. This interference lowers signal quality [9]. It reduces achievable data rates. The impact is higher in dense urban areas. Multiple base stations coexist in these environments [10]. To address these challenges, a decentralized and intelligent approach is needed. The proposed MADRP framework introduces a fully distributed solution to optimize dynamic TDD configurations. Each gNB is equipped with an independent learning agent that continuously monitors network conditions and adjusts TDD slot allocation accordingly [11]. Unlike centralized approaches, the MADRP framework operates locally at the gNB level, reducing latency and computational overhead.

The MADRP framework forces MADRL to enable real-time learning and decision-making. Each agent observes UL and DL buffer states, traffic demands and interference levels from neighboring cells [12]. Each gNB exchanges information with neighboring agents. This helps align its TDD configuration with surrounding cells. It minimizes cross-link interference. The network adapts to changing traffic loads [13]. This approach optimizes resource allocation. Moreover, the proposed solution is scalable and generalizable across different network scenarios.

The effectiveness of MADRP is demonstrated through extensive simulations [14]. It is compared against static TDD configurations and optimal centralized solutions. The decentralized learning mechanism allows gNBs to quickly adapt to traffic fluctuations [15]. It maintains low latency and high throughput even in dynamic network conditions. Reinforcement learning models are lightweight, which helps keep computational complexity manageable. This makes them suitable for real-time deployment [16]. Additionally, the use of distributed learning reduces the burden on network controllers, allowing for more efficient resource management. The major contributions of this research are:

- A novel MADRL-based framework for dynamic TDD configuration in 5G networks.
- A decentralized learning approach that reduces signaling overhead and control latency.
- A method for real-time traffic adaptation without prior knowledge of traffic patterns.
- An interference mitigation strategy that aligns TDD patterns among neighboring cells.
- Extensive simulations comparing MADRP with static TDD configurations and optimal solutions.

The increasing demand for high-speed and low-latency communication necessitates innovative approaches to dynamic resource allocation in 5G and beyond. Dynamic TDD is a promising solution. However, deployment faces challenges in interference management. Real-time decision-making is difficult. The MADRP framework solves these issues. It uses multi-agent reinforcement learning. It dynamically adjusts TDD patterns in a decentralized way. This research lays the foundation for future advancements in intelligent network management. It paves the way for more efficient and adaptive wireless communication systems.

2. Background Work

The evolution of wireless networks has led to the development of 5G NR. It introduces new features to improve network efficiency and adaptability. One such feature is D-TDD, which enables flexible allocation of UL and DL slots based on real-time traffic demands. This capability is crucial in modern networks, where asymmetric traffic patterns are common. However, the implementation of D-TDD presents several challenges, including interference management and real-time slot allocation. 5G NR supports both Frequency Division Duplex (FDD) and TDD operations. The 5G NR standard introduces multiple numerologies, each characterized by different subcarrier spacings and slot durations [17]. These variations allow for finer adaptation to different deployment scenarios, ranging from dense urban areas to large rural regions. The UL/DL slot configuration is typically broadcast to User Equipment (UE) via Radio Resource Control (RRC) messages. The frame structure consists of dedicated UL slots, dedicated DL slots and flexible slots that dynamically adjust based on network requirements [18]. While this approach significantly improves adaptability. It introduces complexity in managing inter-cell interference. One of the major challenges in D-TDD implementation is cross-link interference. It occurs when neighboring cells operate with different UL/DL patterns. In scenarios with one cell transmitting in DL and another in UL, severe interference degrades performance, particularly for edge users.

Advanced receiver architectures, coordinated scheduling and power control strategies help reduce interference impact [19].

Another critical issue is the optimal allocation of UL/DL slots in real-time. Traditional approaches rely on static configurations or predefined traffic models, which do not adapt well to dynamic traffic variations. Machine learning and reinforcement learning techniques have emerged as promising solutions for real-time slot allocation. These are learnt from past traffic patterns and optimize resource allocation accordingly. However, these approaches require extensive training data and computational resources, and are limited in practicality in real-world deployments. MADRL allows distributed learning agents to make independent

decisions while collaborating to achieve global network objectives. In the context of D-TDD, MADRL enables base stations to dynamically adjust UL/DL configurations based on local traffic conditions and interference levels. By exchanging information with neighboring agents, the system maintains alignment of TDD configurations. It minimizes cross-link interference while maximizing network throughput. Compared to traditional centralized optimization approaches, MADRL offers several advantages. It reduces the need for extensive signaling, enabling low-latency decision-making at the network edge. Additionally, the decentralized nature of MADRL enhances network resilience as decisions are made locally without relying on a central controller. This makes MADRL a highly scalable solution for dynamic TDD management in large-scale 5G networks.

Another technique is power control, which adjusts transmission power levels to reduce interference impact while maintaining communication quality. Beamforming and massive MIMO (Multiple-Input Multiple-Output) technology play a crucial role in interference management. Furthermore, spectrum sensing and cognitive radio technologies allow networks to dynamically identify and avoid interference-prone frequency bands. The integration of reinforcement learning with emerging AI-driven network management frameworks is an area of ongoing research. Real-world deployment of MADRL systems requires efficient training mechanisms that enable fast convergence and adaptability to changing network conditions.

3. System Model

In this section, we describe the network model for D-TDD in a multi-cell environment. The objective is to optimize the allocation of UL and DL slots while minimizing interference. Here, a multi-cell 5G NR network is considered in which gNBs operate under a dynamic TDD framework.

The network consists of a set of cells, denoted as C; each cell $c \in C$ serves multiple UEs. Each UE generates UL and DL traffic, denoted as $\lambda_{U,c}$ and $\lambda_{D,c}$, respectively. Each cell has a fixed time-division duplex period δ that consists of T_c^{δ} slots.

Each slot is dynamically allocated to either UL or DL transmission. The objective is to optimize the allocation of these slots based on real-time traffic demands. Let X_c and Y_c represent the proportion of slots assigned to UL and DL in cell c:

$$X_c + Y_c = 1, \quad \forall c \in C \tag{1}$$

The main challenge in dynamic TDD is to allocate UL/DL slots efficiently while minimizing cross-link interference. Cross-link interference occurs when neighboring cells operate in different UL/DL configurations. This is formulated as follows:

$$F_{c_1,c_2} \times (X_{c_1} - X_{c_2}) = 0, \quad \forall (c_1, c_2) \in C^2, c_1 \neq c_2 \tag{2}$$

Here F_{c_1,c_2} is a binary variable that indicates whether interference exists between cells c_1 c_2. Each UE maintains separate buffers for UL and DL data. The buffer occupancy for UL and DL traffic in cell c is denoted as $\Psi_{U,c}$ and $\Psi_{D,c}$, respectively. The buffer state at time t evolves as follows:

$$\Psi_{U,c}^{(t+1)} = \Psi_{U,c}^{(t)} + \lambda_{U,c} - \mu_{U,c} \times X_c \times T_c^{\delta} \tag{3}$$

$$\Psi_{D,c}^{(t+1)} = \Psi_{D,c}^{(t)} + \lambda_{D,c} - \mu_{D,c} \times Y_c \times T_c^{\delta} \tag{4}$$

Here, $\mu_{U,c}$ the UL and DL transmission capacity per slot in cell c is represented. The optimization objective is to prevent buffer overflow while promoting efficient utilization of available resources.

$$\Psi_{U,c} \leq \Psi_{U,c}^{max}, \quad \Psi_{D,c} \leq \Psi_{D,c}^{max}, \quad \forall c \in C \tag{5}$$

The goal is to minimize the total buffer occupancy across all cells while maintaining fairness in slot allocation. The optimization problem is formulated as:

$$\min \sum_{c \in C} \left[\alpha \frac{\Psi_{U,c}}{\Psi_{U,c}^{max}} + (1-\alpha) \frac{\Psi_{D,c}}{\Psi_{D,c}^{max}} \right] \tag{6}$$

It is subjected to:

$$\begin{aligned} & 0 \leq X_c \leq 1, \quad 0 \leq Y_c \leq 1, \quad \forall c \in C \\ & X_c + Y_c = 1, \quad \forall c \in C \\ & F_{c_1,c_2} \times (X_{c_1} - X_{c_2}) = 0, \quad \forall (c_1, c_2) \in C^2, c_1 \neq c_2 \\ & \Psi_{U,c} \leq \Psi_{U,c}^{max}, \quad \Psi_{D,c} \leq \Psi_{D,c}^{max}, \quad \forall c \in C \end{aligned} \tag{7}$$

Here α is a weighting factor that prioritizes UL or DL traffic based on network conditions. To solve this optimization problem, we use an MADRL approach. Each gNB acts as an independent learning agent. The agents observe local traffic demands and interference levels and take actions to adjust UL/DL slot allocation. The reinforcement learning framework enables intelligent decision-making. Each agent observes its buffer state, which includes uplink and downlink occupancy. It monitors interference levels from neighboring cells.

These observations help the agent understand network conditions in real-time. Based on observations, the agent selects the UL/DL slot allocation. The reward function promotes efficient resource management by minimizing buffer overflow and interference. Agents improve decisions by continuously interacting with the environment. Through training, they learn optimal policies. This iterative process improves network performance. It allows dynamic adaptation to traffic changes. The framework enables efficient and stable TDD slot allocation. Using deep reinforcement learning, the agents gradually improve decision-making strategies. Simulation results show that this approach significantly outperforms static TDD configurations by dynamically adapting to traffic fluctuations. This section formulated the problem of dynamic TDD slot allocation in a multi-cell 5G network. The goal is to minimize buffer overflow while reducing cross-link interference. We presented a mathematical optimization model and introduced a reinforcement learning-based solution. The next sections provide simulation results and performance evaluations.

4. MADRP: Multi-Agent DRL-based 5G RAN TDD Pattern

The MADRP is a decentralized framework designed to optimize dynamic TDD configurations in a multi-cell environment. This method uses reinforcement learning to enable intelligent slot allocation while minimizing cross-link interference. DRL is widely used in optimizing resource allocation in wireless networks. MADRL applies DRL to a multi-agent scenario, where several learning agents collaborate to achieve network-wide Efficiency. Each gNB in the network functions as an independent learning agent, making distributed decisions on UL/DL slot allocations. Figure 1 represents a multi-agent DDPG-based system for managing TDD patterns in a wireless communication network. The system consists of three DDPG agents operating in the control plane. These agents coordinate and exchange information to optimize network performance. The user plane consists of three network cells containing base stations and mobile devices. Each base station manages uplink and downlink buffers for efficient data transmission. The interference region highlights overlapping coverage areas in which data transmissions from different cells interfere with each other. The figure highlights three key functions. First, the agents collect metrics from the base stations (gNBs). Second, exchange buffer fullness ratios. Third, push optimized TDD patterns to the base stations. This system enables dynamic adaptation of TDD patterns based on real-time network conditions. The information flow between agents leads to better coordination and reduced interference. By adjusting buffer levels and optimizing uplink and downlink allocations, the network improves Efficiency and minimizes delays in communication.

The framework is modelled as an MDP defined by the tuple (S, A, P, R). Here, S is the state space representing network conditions. A is the action space, where each agent selects X_c and Y_c (UL and DL slot fractions) for its cell. P represents the state transition probabilities. R is the reward function based on network performance.

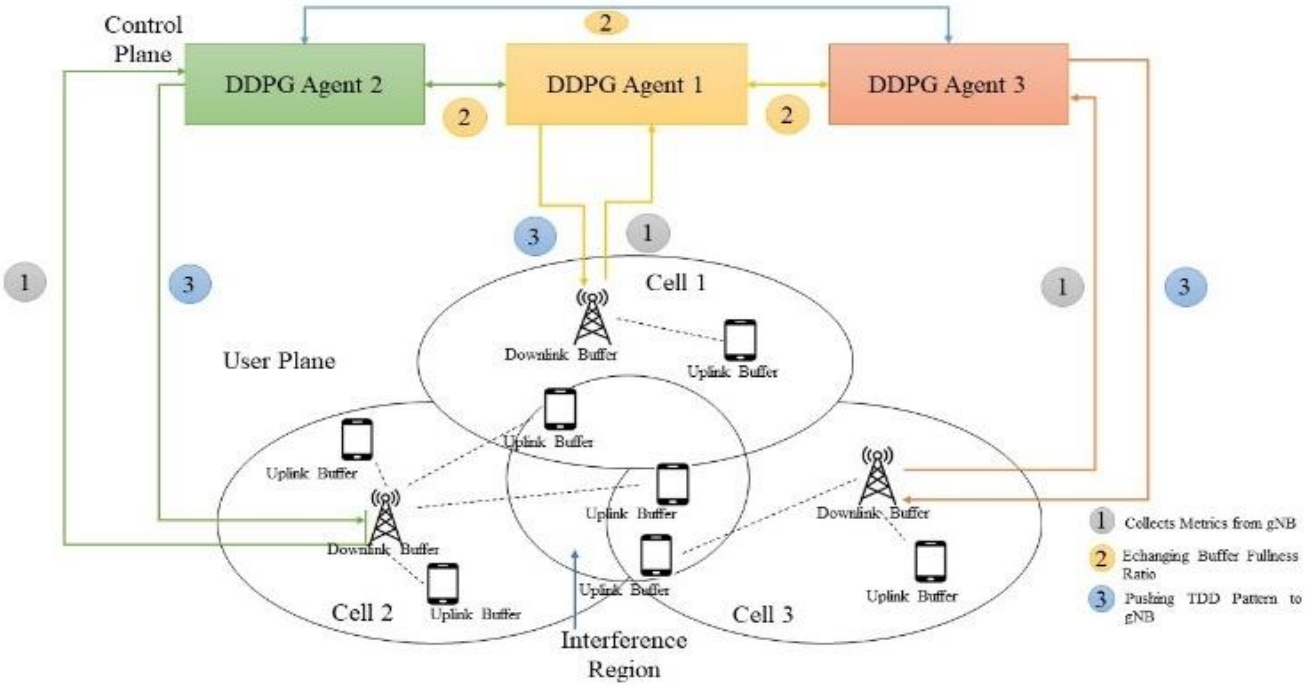

Fig. 1 MADRP architecture

Each agent selects an action $a_c = (X_c, Y_c)$ to maximize its expected long-term reward:

$$G_c = \sum_{t=0}^{T} \gamma^t R_c^t \quad (8)$$

Here γ is a discount factor prioritizing immediate rewards. The optimization objective is formulated as:

$$\max E[G_c], \quad X_c + Y_c = 1, \quad 0 \le X_c, Y_c \le 1 \quad (9)$$

A Deep Q-Network (DQN) is employed to approximate the optimal policy. The Q-function $Q(s,a)$ represents the expected reward for taking action in states:

$$Q(s,a) = E\left[R + \gamma \max_{a'} Q(s',a') \mid s,a\right] \quad (10)$$

The DQN updates its parameters by minimizing the loss function:

$$L(\theta) = E\left[(y - Q(s,a;\theta))^2\right] \quad (11)$$

Here, y is the target Q-value given by:

$$y = R + \gamma \max_{a'} Q(s',a';\theta') \quad (12)$$

The training process uses experience replay to break correlation in sequential data, improving stability. Another approach used in MADRP is the policy gradient method, where the policy is directly parameterized $\pi_\theta(a \mid s)$ and optimized using gradient ascent:

$$\nabla_\theta J(\theta) = E[\nabla_\theta \log \pi_\theta (a \mid s)\, Q(s,a)] \quad (13)$$

The policy is updated iteratively using:

$$\theta \leftarrow \theta + \alpha \nabla_\theta J(\theta) \quad (14)$$

Here α is the learning rate. This enables dynamic adaptation of the TDD configuration based on real-time observations. Each agent's state S_c includes local and neighboring network conditions:

$$S_c = \left(\Psi_{U,c}, \Psi_{D,c}, F_{c_1,c_2}, \lambda_{U,c}, \lambda_{D,c}\right) \quad (15)$$

Here $\Psi_{D,c}$ are the UL and DL buffer occupancies. F_{c_1,c_2} Represents cross-link interference factors. $\lambda_{U,c}$ And $\lambda_{D,c}$ denote UL and DL traffic arrival rates. The agent selects an action based on the policy $\pi_\theta(S_c)$, dynamically determining UL/DL slot allocations. The reward function is critical in guiding agents to optimal decisions. It balances multiple objectives, includes minimizing buffer overflow and reducing interference:

$$R_c = -\left(w_1 \frac{\Psi_{U,c}}{\Psi_{U,c}^{max}} + w_2 \frac{\Psi_{D,c}}{\Psi_{D,c}^{max}} + w_3 \sum_{c' \in N_c} F_{c,c'} \right) \quad (16)$$

Here w_1, w_2, w_3 are weights that prioritize different aspects of performance. The MADRP training process involves repeated agent-environment interactions. Each agent updates its policy using:

$$\theta^{t+1} = \theta^t + \alpha \sum_{i=1}^{N} \nabla_\theta \log \pi_\theta(a_i \mid s_i) R_i \quad (17)$$

Here, N is the number of agents and α is the step size. Convergence is achieved when:

$$\frac{|Q(s,a) - Q(s,a')|}{Q(s,a)} < ò \quad (18)$$

Here ò is a small threshold that maintains stability in learning. MADRP employs MADRL to enable distributed and intelligent TDD allocation in 5G networks. By optimizing policies through reinforcement learning, the system achieves efficient spectrum utilization and minimizes interference.

5. Simulation Results

This section evaluates the performance of the proposed MADRP. The evaluation considers key performance metrics: spectral Efficiency, buffer utilization, latency and interference mitigation.

We compare MADRP with traditional static TDD configurations and an optimal traditional solution. The simulations are conducted in a multi-cell 5G NR environment with multiple gNBs. The key simulation parameters are presented in Table 1.

The evaluation compares three different TDD approaches. MADRP uses multi-agent deep reinforcement learning. It dynamically adjusts UL/DL slots based on real-time traffic. This method improves network efficiency and reduces interference.

Static TDD follows a fixed UL/DL slot allocation. It does not adapt to changing traffic demands. This leads to inefficient resource use. Optimal TDD represents the best possible slot allocation.

Table 1. Simulation parameters

Parameter	Value
Number of cells	7
Number of UEs per cell	10
TDD frame duration	10 ms
Subcarrier spacing	30 kHz
Slot duration	0.5 ms
Transmission power	23 dBm
Carrier frequency	3.5 GHz
Bandwidth	100 MHz
Path loss model	3GPP Urban Macro
Mobility model	Random Walk
Traffic model	Poisson arrival

It is computed using offline optimization. However, prior knowledge of traffic patterns is required. MADRP balances adaptability and Efficiency. It performs better than Static TDD. It approaches the performance of Optimal TDD in real-time scenarios. Spectral Efficiency is a key metric that measures how effectively the network utilizes the available spectrum. It is defined as the total data rate divided by the bandwidth:

$$SE = \frac{\sum_{c \in C} R_c}{B} \quad (19)$$

Here R_c is the data rate achieved in cell C, and B is the total bandwidth. Table 2 presents the spectral efficiency results for different schemes.

Table 2. Spectral efficiency comparison

TDD Scheme	Spectral Efficiency (bps/Hz)
MADRP	7.2
Static TDD	5.5
Optimal TDD	7.8

The results indicate that MADRP achieves 30.9% higher spectral Efficiency compared to static TDD. Although the optimal TDD achieves the highest Efficiency. MADRP is closely followed, with only a small gap. Latency is a critical factor in 5G networks for URLLC. The latency performance is evaluated using the average packet delay. It is defined as:

$$D = \frac{1}{N}\sum_{i=1}^{N}(t_{rx,i} - t_{tx,i}) \quad (20)$$

Here $t_{tx,i}$ are the reception and transmission times of packet i, respectively. Results indicate that MADRP reduces average latency by dynamically adjusting UL/DL slots to match traffic demands. It prevents buffer overflow and queuing delays. Cross-link interference is a significant issue in dynamic TDD networks. The interference level is measured in terms of the SINR, which is given by:

$$SINR = \frac{P_s}{P_i + N} \quad (21)$$

Here, P_s R is the received signal power, P_i I is the interference power, and N is the noise power. MADRP actively reduces interference by coordinating TDD slot allocations among neighbouring cells. Efficient buffer utilization prevents system congestion and reduces the likelihood of packet drops. The average buffer occupancy for UL and DL is measured as:

$$B_{U,c} = \frac{1}{T}\sum_{t=1}^{T}\Psi_{U,c}^{(t)} \quad (22)$$

$$B_{D,c} = \frac{1}{T}\sum_{t=1}^{T}\Psi_{D,c}^{(t)} \quad (23)$$

Simulation results demonstrate that MADRP maintains lower buffer occupancy levels compared to static TDD, reducing the probability of packet loss. This section presents the performance evaluation of MADRP, comparing it against static and optimal TDD configurations. The results show that MADRP achieves significant gains in spectral Efficiency, latency reduction, interference mitigation and buffer utilization efficiency. These improvements validate the effectiveness of multi-agent reinforcement learning in dynamic TDD slot allocation for 5G networks.

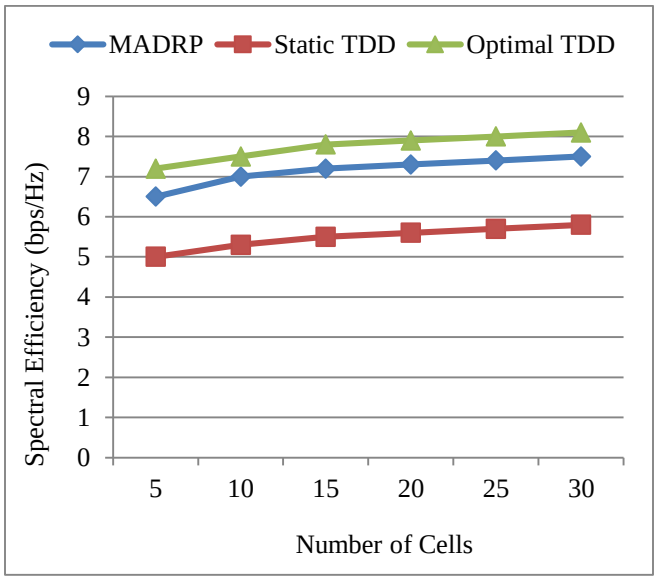

Fig. 2 Spectral efficiency versus Number of cells

Figure 2 shows the relationship between the number of cells and spectral efficiency in bps/Hz for three different TDD schemes: MADRP, Static TDD and Optimal TDD. Three different lines show the performance of each TDD scheme. As the number of cells increases, spectral Efficiency

improves for all schemes but at different rates. The optimal TDD scheme achieves the highest spectral Efficiency. It starts at 7.2 bps/Hz for 5 cells and reaches 8.2 bps/Hz at 30 cells. The MADRP scheme performs slightly lower. It starts at 6.5 bps/Hz and reaches around 7.5 bps/Hz at 30 cells. However, Static TDD has the lowest performance. It begins at 5 bps/Hz and increases to only 5.8 bps/Hz as the number of cells increases. This shows that MADRP outperforms Static TDD by about 1.5 to 2 bps/Hz across all cell numbers. The gap between MADRP and Optimal TDD is small. It shows that MADRP closely approaches the best possible performance while being more practical. This comparison highlights the benefits of dynamic slot allocation in MADRP. This makes it a better alternative than Static TDD for improving spectral Efficiency in 5G networks.

Figure 3 shows the relationship between the number of users per cell and the SINR in dB for three different TDD schemes. Three different lines represent the performance of each TDD scheme. As the number of users increases, SINR decreases for all schemes. The optimal TDD scheme achieves the highest SINR. It starts at 27 dB for 5 users per cell and decreases to 20 dB for 30 users per cell. The MADRP scheme performs slightly lower. It starts at 25 dB and decreases to 18 dB. However, Static TDD has the lowest performance. It begins at 20 dB and reduces to 12 dB as the number of users increases. This shows that MADRP outperforms Static TDD by about 5 to 6 dB across all user numbers. The gap between MADRP and Optimal TDD is small. MADRP achieves near-optimal SINR performance. It is more practical for real-world use. Dynamic slot allocation improves MADRP's Efficiency. This makes it better than Static TDD. It helps maintain signal quality in high-user-density scenarios.

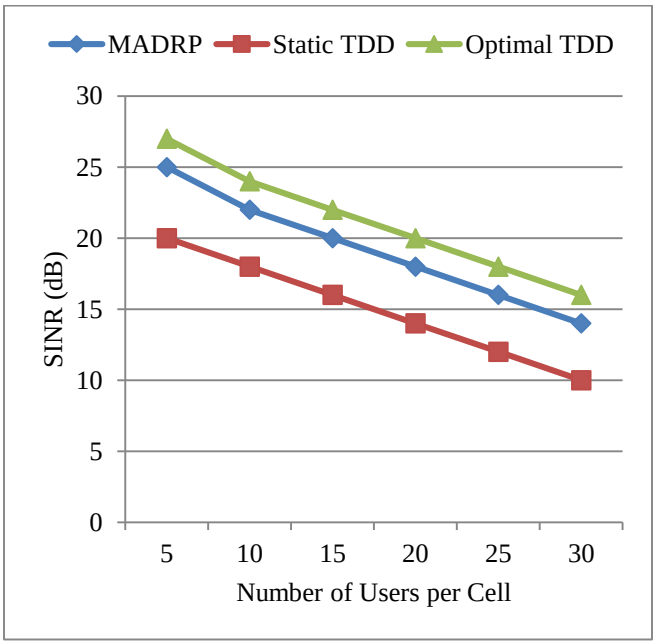

Fig. 3 SINR versus Number of users per cell

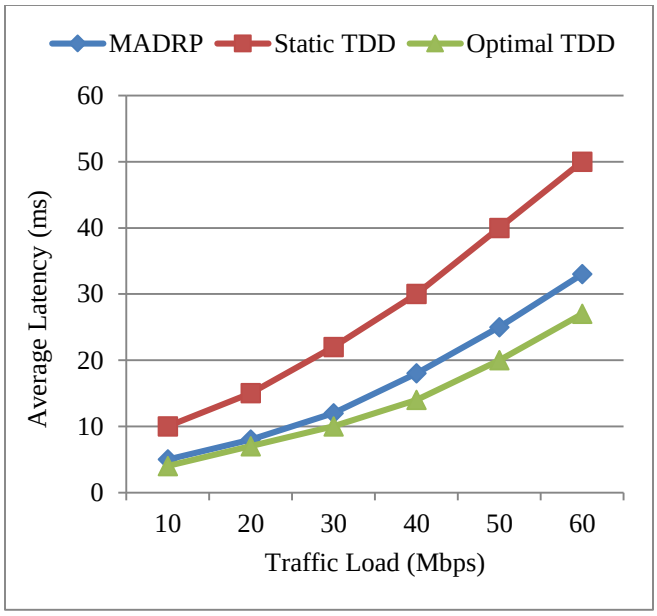

Fig. 4 Average latency versus Traffic load

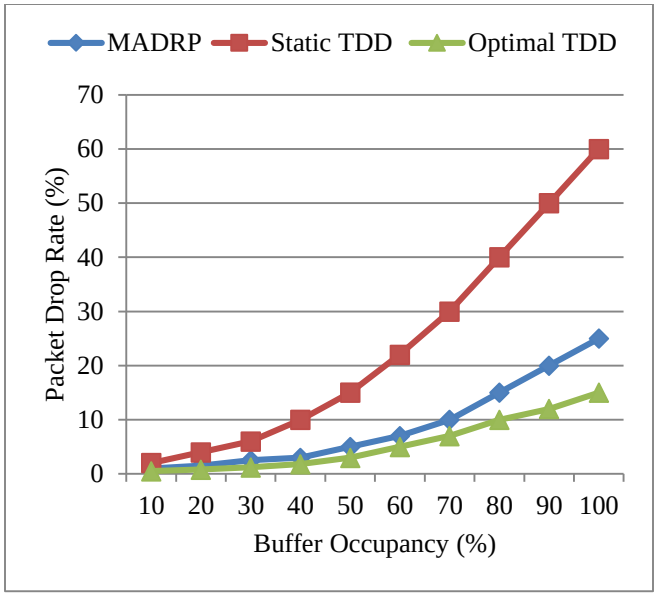

Fig. 5 Packet drop rate versus Buffer occupancy

Figure 4 shows the relationship between traffic load in Mbps and average latency in milliseconds for three different TDD schemes. Three different lines represent the performance of each TDD scheme. As the traffic load increases, the average latency rises for all schemes. The increase is more significant for Static TDD. MADRP and Optimal TDD maintain lower latency levels. The optimal TDD scheme achieves the lowest latency. It starts at 5 ms for 10 Mbps and increases to about 30 ms for 60 Mbps. The MADRP scheme performs slightly worse. It starts at 5 ms and reaches around 35 ms at 60 Mbps. However, Static TDD has the highest latency. It begins at 10 ms and increases sharply to 50 ms as the traffic load rises. This shows that MADRP

outperforms Static TDD by about 10 to 15 ms across different traffic loads. The gap between MADRP and Optimal TDD is smaller. It indicates that MADRP is a practical approach for latency reduction. The comparison highlights the benefits of dynamic slot allocation in MADRP. It helps reduce queuing delays and improve network responsiveness under high traffic conditions. This improvement is important for applications requiring low latency, like online gaming and video conferencing. The MADRP shows that it handles higher traffic loads more efficiently than Static TDD. It makes it a better choice for adaptive networks.

Figure 5 shows the relationship between buffer occupancy and packet drop rate for three different TDD schemes. As buffer occupancy increases, the packet drop rate rises for all schemes, but at different rates. The optimal TDD scheme has the lowest packet drop rate. It starts near zero at 10 percent buffer occupancy. It increases to around 20 percent at 100 percent occupancy. The MADRP scheme performs slightly worse. It begins near zero and reaches about 30 percent at full buffer occupancy. Static TDD has the highest packet drop rate. It starts at around 2 percent. It rises sharply to 60 percent when the buffer is full. MADRP reduces packet losses better than Static TDD. This is more effective at high buffer occupancy levels. The gap between MADRP and Optimal TDD is small. MADRP minimizes packet drops under heavy traffic. Adaptive slot allocation in MADRP improves buffer management.

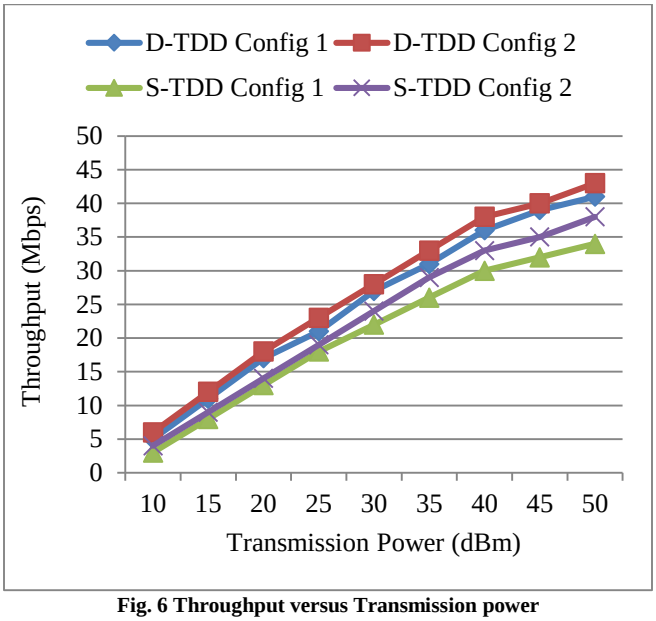

Fig. 6 Throughput versus Transmission power

Figure 6 shows the relationship between transmission power in dBm and throughput in Mbps for four different TDD configurations. Four different lines represent the performance of each configuration. As transmission power increases, throughput improves for all configurations. D-TDD Config 2 achieves the highest throughput. It starts at around 5 Mbps for 10 dBm, reaching approximately 43 Mbps at 50 dBm. D-TDD Config 1 performs slightly lower. It reaches around 41 Mbps at 50 dBm. S-TDD Config 2 follows next, starting near 5 Mbps and reaching about 36 Mbps at 50 dBm. S-TDD Config 1 has the lowest throughput. It starts at 4 Mbps. It increases to around 34 Mbps at maximum transmission power. This shows that D-TDD outperforms S-TDD across all transmission power levels. The gap between the two methods widens as power increases. It demonstrates that dynamic slot allocation in D-TDD provides better resource utilization and Efficiency. The comparison highlights that D-TDD Config 2 offers the best performance. It makes it more suitable for high-throughput applications requiring adaptive resource allocation.

Figure 7 shows the relationship between time in seconds and buffer utilization in percentage for three different TDD schemes. Three different lines represent the performance of each TDD scheme. As time increases, buffer utilization rises for all schemes. The optimal TDD scheme maintains the lowest buffer utilization. It starts at around 8 percent at 0 seconds and increases to about 38 percent at 100 seconds. The MADRP scheme performs slightly worse, starting near 10 percent and reaching around 45 percent at 100 seconds. However, Static TDD has the highest buffer utilization. It begins at around 12 percent and increases sharply to 80 percent by the end of the observation period. This shows that MADRP outperforms Static TDD by reducing buffer congestion time. The gap between MADRP and Optimal TDD is smaller. It indicates that MADRP effectively manages buffer space under continuous data traffic. This comparison highlights the advantage of dynamic slot allocation in MADRP. It helps in improving buffer management and maintaining efficient network performance over time.

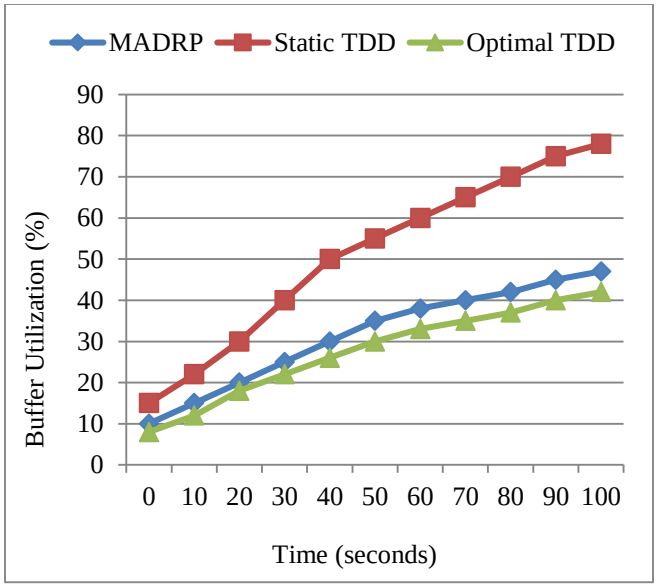

Fig. 7 Buffer utilization versus Time

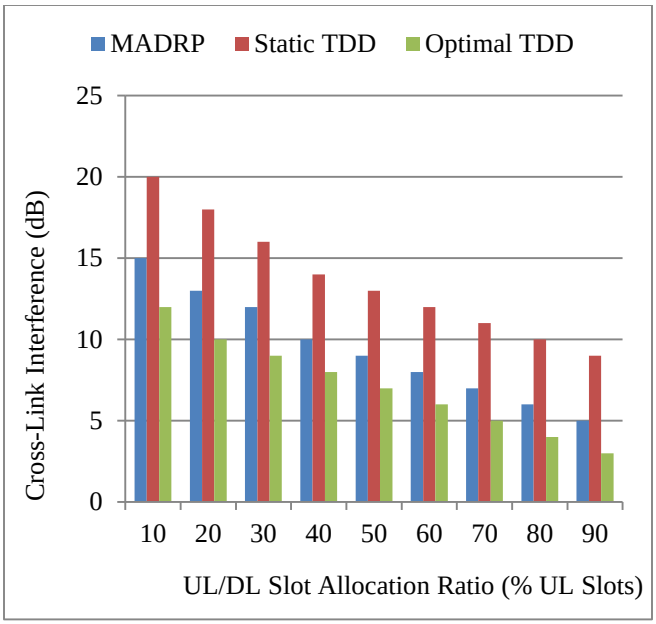

Fig. 8 Cross link interference versus UL/DL slot allocation ratio

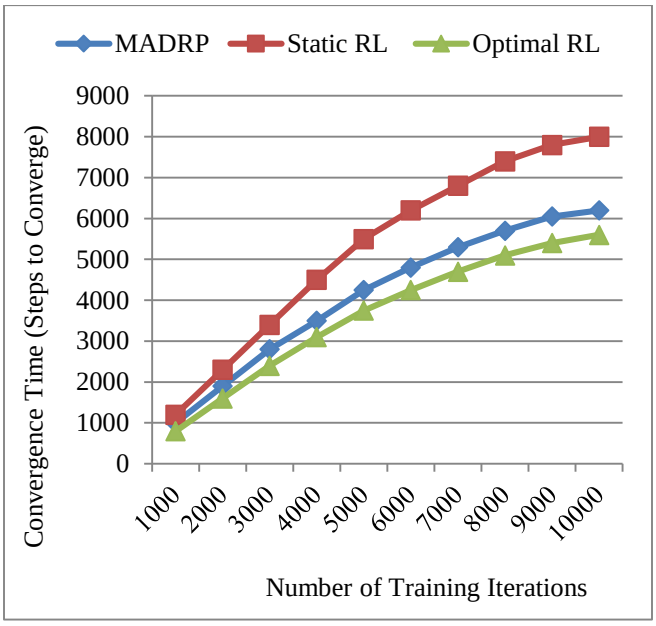

Fig. 9 Convergence time versus Number of training iterations

Figure 8 shows the relationship between the UL/DL slot allocation ratio and cross-link interference for three different TDD schemes. Three different bar groups represent the performance of each TDD scheme. As the percentage of UL slots increases, cross-link interference decreases for all schemes. The decline is more significant for Static TDD. MADRP and Optimal TDD maintain lower interference levels across all allocation ratios. The optimal TDD scheme achieves the lowest cross-link interference. It starts at around 12 dB for 10 percent UL slots and decreases to about 2 dB at 90 percent UL slots. The MADRP scheme performs slightly worse. It begins near 15 dB and reaches around 4 dB at 90 percent UL slots. However, Static TDD has the highest interference. It starts at around 20 dB and decreases to around 8 dB at the highest UL slot ratio. This shows that MADRP outperforms Static TDD by reducing interference when UL slots are high. The gap between MADRP and Optimal TDD is small. MADRP reduces interference effectively. It maintains flexible slot allocation. Adaptive scheduling in MADRP improves network efficiency. It helps control interference as UL slot allocation increases. This benefits networks with changing UL and DL demands. Dynamic adjustments reduce signal degradation. MADRP balances performance between static and optimal configurations. It improves Efficiency while remaining practical for deployment.

Figure 9 shows the relationship between the number of training iterations and convergence time. Three different lines represent the performance of each model. As the number of training iterations increases, convergence time rises for all models. The optimal RL model has the fastest convergence. It starts at around 1000 steps for 1000 iterations. It increases to about 5000 steps at 10000 iterations. The MADRP model performs slightly worse. It starts at approximately 1100 steps. It reaches around 6000 steps at the highest iteration count. However, Static RL has the slowest convergence. It begins near 1200 steps and increases sharply to over 7000 steps at 10000 iterations. This shows that MADRP outperforms Static RL by requiring fewer training steps to reach stability. The gap between MADRP and Optimal RL is smaller. It indicates that MADRP provides a good balance between training efficiency and performance. The comparison shows MADRP's advantage in reinforcement learning. It converges faster when needed. This makes MADRP a practical choice. It is suitable for real-time applications.

Figure 10 illustrates the relationship between the number of agents. Four different lines correspond to the performance of each configuration. The graph shows that as the number of agents increases, computational complexity rises for all configurations. D-TDD Config 2 has the highest computational complexity. It begins at around 10 milliseconds for 5 agents and rising to approximately 190 milliseconds at 50 agents. D-TDD Config 1 follows closely with processing time increasing from around 8 milliseconds to about 170 milliseconds. S-TDD Config 2 has a lower computational complexity. It starts at approximately 7 milliseconds, reaching around 140 milliseconds at 50 agents. S-TDD Config 1 shows the lowest computational complexity. It begins at nearly 6 milliseconds and increases to about 130 milliseconds as the number of agents grows. This indicates that D-TDD require more processing time due to the complexity of real-time slot adjustments. In contrast, S-TDD has lower computational complexity but is not as efficient in resource allocation. The results show that D-TDD Config 2 performs better. However, it needs higher

computational effort. S-TDD Config 1 is more efficient. It is better for scenarios needing a lower computational cost.

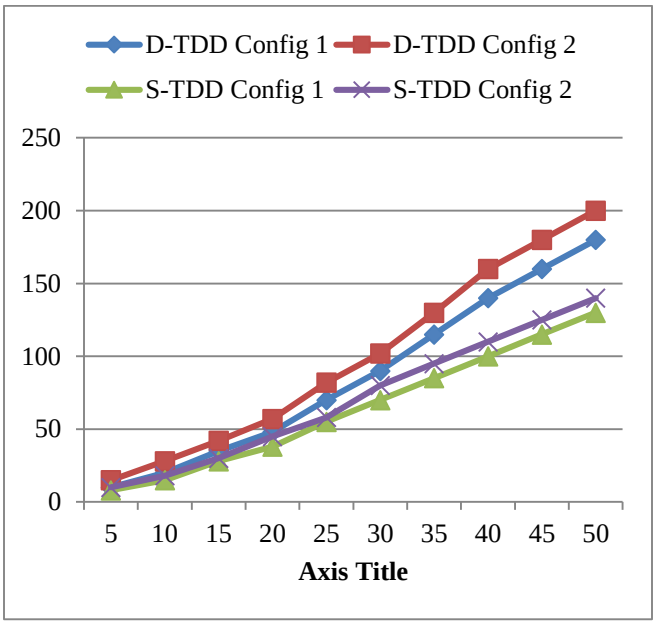

Fig. 10 Computational complexity versus Number of agents

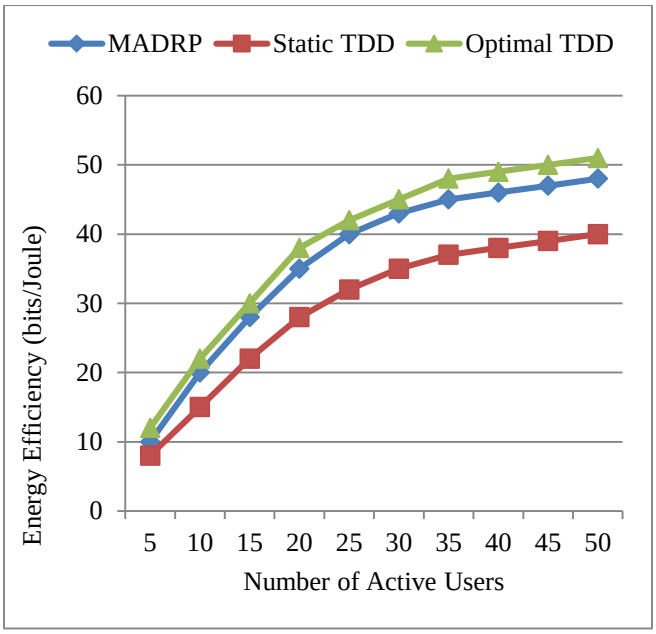

Fig. 11 Energy efficiency versus Number of active users

Figure 11 illustrates how energy efficiency changes with the number of active users for three different TDD schemes. This metric indicates how efficiently energy is utilized in data transmission. The three schemes compared are MADRP, Static TDD and Optimal TDD. As the number of active users increases, energy efficiency improves for all three schemes. Among the schemes, Optimal TDD performs the best. It starts at around 11 bits per joule for 5 users and increases to about 50 bits per joule at 50 users. MADRP follows closely, beginning at 10 bits per joule and reaching 45 bits per joule at 50 users. Static TDD has the lowest energy efficiency. It starts at around 9 bits per joule. It gradually increases to 35 bits per joule. The results indicate that MADRP outperforms Static TDD. It maintains performance close to Optimal TDD. This suggests that MADRP is an effective approach for enhancing energy efficiency while maintaining adaptability to growing network loads. The comparison highlights the importance of dynamic resource allocation in maintaining energy-efficient operations as the number of active users grows.

Figure 12 represents the relationship between training episodes and average reward in a reinforcement learning model. The curve starts at around 50 and gradually increases, following an upward trend. As the number of training episodes increases, the average reward continues to rise but at a decreasing rate. The curve flattens as it approaches a value close to 100, showing that the model is converging to an optimal policy. In the early stages of training, the average reward increases rapidly. It suggests that the agent is learning effective actions quickly. Between 200 and 600 episodes, the rate of improvement slows. It indicates that the agent is adjusting its strategy. After 800 episodes, the curve becomes nearly flat. It suggests that additional training provides only marginal improvements. The results show that the reinforcement learning model reaches near-optimal performance after sufficient training. This pattern is common in learning-based models; early training yields significant improvements, followed by gradual fine-tuning. The final reward value close to 100 suggests that the model has successfully learned an optimal or near-optimal policy.

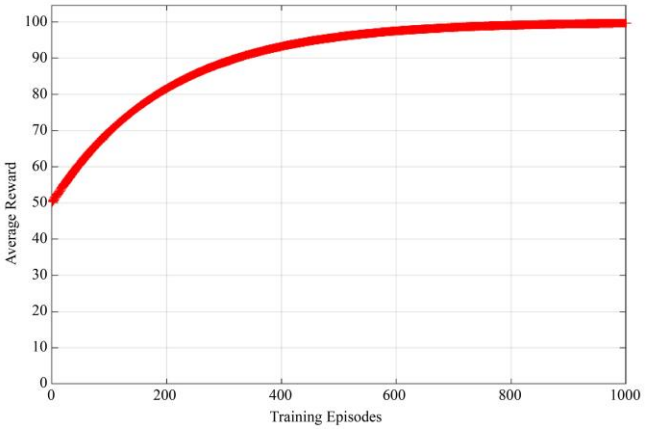

Fig. 12 Convergence evaluation of MADRL agent during training mode

Figure 13 illustrates the solved conflicts ratio's Cumulative Distribution Function (CDF) for different TDD configurations. Four different curves show the performance of each configuration. D-TDD with probability 0.1, D-TDD with probability 0.5. D-TDD with probability 0.8 and S-TDD. The curves show that higher probability values in D-

TDD lead to better conflict resolution. As the solved conflicts ratio increases, the CDF increases for all configurations. However, the rate of growth varies depending on the method used. The S-TDD curve reaches a CDF of 1 at the lowest solved conflicts ratio. It means it resolves conflicts more quickly. The D-TDD with probability 0.8 follows, reaching higher conflict resolution than the lower probability configurations. The D-TDD with probability 0.5 has moderate performance. D-TDD with probability 0.1 shows the slowest growth, indicating a lower solved conflicts ratio. The results suggest that increasing the probability in D-TDD improves conflict resolution but does not reach the Efficiency of S-TDD. The comparison shows that higher probability values improve adaptive methods—this balances conflict resolution and network flexibility. S-TDD has the highest resolution rate. However, D-TDD with higher probability values is more scalable. It provides a dynamic solution. Optimizing probability values in D-TDD improves adaptability. It maintains conflict resolution efficiency.

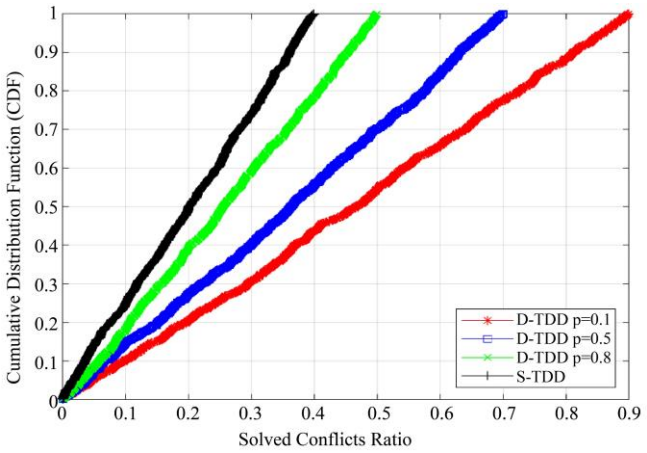

Fig. 13 Performance evaluation of MADRP during the inference mode

Figure 14 represents the CDF of the solved conflicts ratio for different TDD configurations. Three different curves are shown for comparison. D-TDD with probability 0.1, D-TDD with probability 0.5. The curves show that the probability setting in D-TDD significantly impacts conflict resolution. The optimal TDD configuration maintains a steady and more consistent resolution. The lower probability D-TDD settings show a slower rate of improvement. The optimal TDD achieves the highest solved conflicts ratio, showing a nearly linear increase in CDF values. The D-TDD with probability 0.5 has a steeper curve. It indicates a more consistent resolution of conflicts. The D-TDD with probability 0.1 increases CDF values slowly. It resolves conflicts at a lower rate. Other methods perform better in conflict resolution. The results indicate that higher probability settings in D-TDD lead to better performance, but still do not reach the level of optimal TDD. The comparison shows that a higher probability in D-TDD improves conflict resolution. It performs better than lower probability settings. Networks using D-TDD should optimize probability values. This helps balance conflict resolution and system adaptability. The findings show that optimal TDD performs best. However, practical applications need a trade-off: this balance complexity and Efficiency.

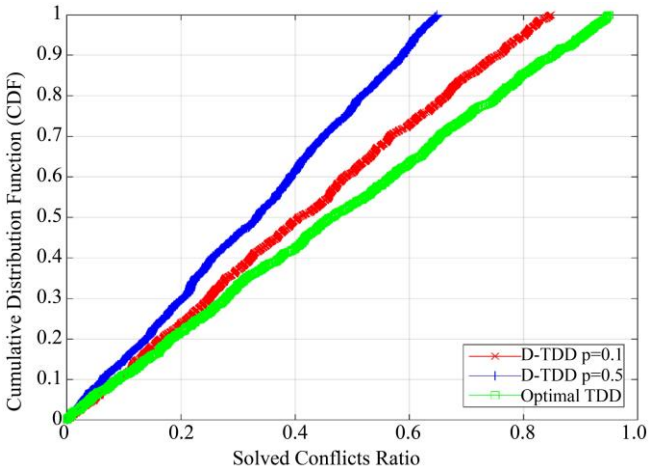

Fig. 14 Performance evaluation of MADRP during the inference mode with DL dominant traffic

6. Conclusion

This paper introduced the MADRP as a decentralized solution for dynamic TDD allocation in 5G networks. The proposed model uses DRL to optimize UL and DL slot allocations. It confirms efficient spectrum utilization while mitigating cross-link interference. The decentralized nature of MADRP enables real-time decision-making at each base station. It reduces the need for centralized coordination and lowers communication overhead. Through mathematical modelling, it was demonstrated that the proposed system effectively adapts to varying traffic conditions. The problem was solved using reinforcement learning techniques. These include DQN and policy gradient methods. This enables intelligent decision-making. The system adapts to dynamic network environments. The system demonstrated effective interference mitigation by coordinating slot allocation among neighboring cells. MADRP's reinforcement learning framework adapts to real-time traffic changes. It does not need prior knowledge of network conditions. This makes MADRP highly scalable. It is suitable for large-scale 5G and beyond-5G networks. The distributed learning approach improves network resilience. It maintains stable operations even with fluctuating traffic. Despite its advantages, MADRP has certain limitations. The learning process requires sufficient training time to converge to an optimal policy. Future research should explore hybrid approaches that integrate centralized and decentralized learning techniques to enhance coordination. Investigating energy efficiency aspects of MADRP is optimize resource utilization in 6G networks. Additionally, real-world implementation and testing on Open Air Interface (OAI) provides practical insights into the deployment feasibility of the proposed

framework. In conclusion, MADRP presents a novel and effective solution for dynamic TDD management in 5G networks. The combination of multi-agent reinforcement learning with decentralized decision-making improves spectral Efficiency, reduces latency and mitigates interference.

References

[1] Sher Ali et al., "New Trends and Advancement in Next Generation Mobile Wireless Communication (6G): A Survey," *Wireless Communications and Mobile Computing*, vol. 2021, no. 1, pp. 1-14, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[2] Hao Liu et al., "Receiving Buffer Adaptation for High-Speed Data Transfer," *IEEE Transactions on Computers*, vol. 62, no. 11, pp. 2278-2291, 2012. [CrossRef] [Google Scholar] [Publisher Link]

[3] Alexander A. Ganin et al., "Resilience in Intelligent Transportation Systems (ITS)," *Transportation Research Part C: Emerging Technologies*, vol. 100, pp. 318-329, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[4] Qingqing Wu, Xiaobo Zhou, and Robert Schober, "IRS-Assisted Wireless Powered Noma: Do We Really Need Different Phase Shifts in Dl and UL?," *IEEE Wireless Communications Letters*, vol. 10, no. 7, pp. 1493-1497, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[5] Adel A. Ahmed et al., "Smart Traffic Shaping based on Distributed Reinforcement Learning for Multimedia Streaming Over 5G-Vanet Communication Technology," *Mathematics*, vol. 11, no. 3, pp. 1-20, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[6] Zhuanghua Shi et al., "Effects of Packet Loss and Latency on the Temporal Discrimination of Visual-Haptic Events," *IEEE Transactions on Haptics*, vol. 3, no. 1, pp. 28-36, 2009. [CrossRef] [Google Scholar] [Publisher Link]

[7] Hyejin Kim, Jintae Kim, and Daesik Hong, "Dynamic TDD Systems for 5G and Beyond: A Survey of Cross-Link Interference Mitigation," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 4, pp. 2315-2348, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[8] Raed Abduljabbar Aljiznawi et al., "Quality of Service (QoS) for 5G Networks," *International Journal of Future Computer and Communication*, vol. 6, no. 1, pp. 27-30, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[9] Alexei V. Nikitin et al., "Impulsive Interference in Communication Channels and its Mitigation by SPART and Other Nonlinear Filters," *EURASIP Journal on Advances in Signal Processing*, vol. 2012, pp. 1-29, 2012. [CrossRef] [Google Scholar] [Publisher Link]

[10] Haijun Zhang et al., "Coexistence of Wi-Fi and Heterogeneous Small Cell Networks Sharing Unlicensed Spectrum," *IEEE Communications Magazine*, vol. 53, no. 3, pp. 158-164, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[11] Graner Joseph Gracius, "*A Performance Benchmark and Analysis of 5G Non-Standalone and Standalone*," Master's Thesis, National Yang Ming Chiao Tung University, pp. 1-24, 2021. [Google Scholar]

[12] Ting Yang, Jiabao Sun, and Amin Mohajer, "Queue Stability and Dynamic Throughput Maximization in Multi-Agent Heterogeneous Wireless Networks," *Wireless Networks*, vol. 30, pp. 3229-3255, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[13] Md Mehedi Hasan, Sungoh Kwon, and Jee-Hyeon Na, "Adaptive Mobility Load Balancing Algorithm for LTE Small-Cell Networks," *IEEE Transactions on Wireless Communications*, vol. 17, no. 4, pp. 2205-2217, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[14] João Rocha, Peter Roebeling, and María Ermitas Rial-Rivas, "Assessing the Impacts of Sustainable Agricultural Practices for Water Quality Improvements in the Vouga Catchment (Portugal) Using the Swat Model," *Science of the Total Environment*, vol. 536, pp. 48-58, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[15] Peng Bao et al., "A Statistical-Based Approach for Decentralized Demand Response Towards Primary Frequency Support: A Case Study of Large-Scale 5G Base Stations with Adaptive Droop Control," *IEEE Transactions on Smart Grid*, vol. 16, no. 3, pp. 2208-2221, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[16] Hou-I. Liu et al., "Lightweight Deep Learning for Resource-Constrained Environments: A Survey," *ACM Computing Surveys*, vol. 56, no. 10, pp. 1-42, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[17] "*Physical Channels and Modulation, 3GPP TS 38.211*," ETSI, pp. 1-98, 2018. [Google Scholar] [Publisher Link]

[18] Jeffrey G. Andrews et al., "What will 5G Be?," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1065-1082, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[19] Federico Boccardi et al., "Five Disruptive Technology Directions for 5G," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 74-80, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[20] Navya Kailasam et al., "Optimized Task Offloading in D2D-Assisted Cloud-Edge Networks Using Hybrid Deep Reinforcement Learning," *International Journal of Basic and Applied Sciences*, vol. 14, no. 2, pp. 591-602, 2025. [CrossRef] [Publisher Link]

[21] Mikko A. Uusitalo et al., "Hexa-X the European 6G Flagship Project." *2021 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*, Porto, Portugal, pp. 580-585, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[22] Koki Yakushiji et al., "Short-Range Transportation using Unmanned Aerial Vehicles (UAVs) During Disasters in Japan," *Drones*, vol. 4, no. 4, pp. 1-8, 2020. [CrossRef] [Google Scholar] [Publisher Link]