electronics

MDPI

Article

Energy-Aware MPTCP Scheduling in Heterogeneous Wireless Networks Using Multi-Agent Deep Reinforcement Learning Techniques

Zulfiqar Ali Arain [1,2], Xuesong Qiu [1], Changqiao Xu [1,*], Mu Wang [1] and Mussadiq Abdul Rahim [3]

[1] State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunication, Beijing 100876, China; zulfiqar.arain@faculty.muet.edu.pk (Z.A.A.)
[2] Department of Telecommunication Engineering, Mehran University of Engineering and Technology, Jamshoro 76062, Pakistan
[3] Department of Computer Science, National University of Technology, Islamabad 44000, Pakistan
* Correspondence: cqxu@bupt.edu.cn

Abstract: This paper proposes an energy-efficient scheduling scheme for multi-path TCP (MPTCP) in heterogeneous wireless networks, aiming to minimize energy consumption while ensuring low latency and high throughput. Each MPTCP sub-flow is controlled by an agent that cooperates with other agents using the Multi-Agent Deep Deterministic Policy Gradient (MADDPG) algorithm. This approach enables the agents to learn decentralized policies through centralized training and decentralized execution. The scheduling problem is modeled as a multi-agent decision-making task. The proposed energy-efficient scheduling scheme, referred to as EE-MADDPG, demonstrates significant energy savings while maintaining lower latency and higher throughput compared to other state-of-the-art scheduling techniques. By adopting a multi-agent deep reinforcement learning approach, the agents can learn efficient scheduling policies that optimize various performance metrics in heterogeneous wireless networks.

Keywords: multi-path TCP; energy efficiency; multi-agent; computer networks

check for updates

Citation: Arain, Z.A.; Qiu, X.; Xu, C.; Wang, M.; Abdul Rahim, M. Energy-Aware MPTCP Scheduling in Heterogeneous Wireless Networks Using Multi-Agent Deep Reinforcement Learning Techniques. *Electronics* **2023**, *12*, 4496. https://doi.org/10.3390/electronics12214496

Academic Editor: Heung-Il Suk

Received: 13 September 2023
Revised: 16 October 2023
Accepted: 23 October 2023
Published: 1 November 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Heterogeneous wireless networks, composed of diverse communication technologies and multiple paths, have become increasingly common due to the rapid growth of mobile devices, the Internet of Things (IoT), and the Internet of Medical Things (IoMT) [1–5]. The proliferation of IoT has led to heightened demand for efficient data transmission and seamless connectivity among various devices such as smartphones, wearable devices, and smart home appliances [6]. This surge in demand necessitates the development of effective and efficient communication protocols that can cater to the needs of modern heterogeneous wireless networks [7].

Multi-path TCP (MPTCP) [8], a backward-compatible extension to TCP, enhances network performance and resource utilization by enabling simultaneous data transmission over multiple paths. However, in the realm of heterogeneous networks [9,10], where packets are multiplexed across multiple paths with noticeable delay variations, such as WiFi and LTE, improper scheduling decisions can lead to significant performance degradation. One prominent issue is the head-of-line (HOL) [11] blocking phenomenon, where packets scheduled on low-latency paths are forced to wait for those on high-latency paths to ensure in-order delivery. Additionally, accommodating out-of-order packets requires receivers to maintain a large queue for reorganizing packets. This becomes problematic for mobile devices with limited buffer capabilities, resulting in prolonged application delays, a substantial decrease in throughput, and consequently, a sub-optimal user experience, particularly for interactive or streaming applications [12,13].

These issues also have profound implications for the energy efficiency of multi-path TCP (MPTCP). The extended waiting times and packet reordering lead to increased energy consumption due to protracted device usage and processing times [14,15]. Hence, improper scheduling decisions not only impact performance metrics such as latency and throughput but also escalate energy consumption, thereby reducing the overall efficiency of MPTCP in heterogeneous wireless networks [16]. It is therefore crucial to develop scheduling mechanisms that optimize both performance and energy efficiency [17,18].

In this research, we present a novel approach to multi-path scheduling, treating it as a decision-making problem and proposing a solution grounded in Reinforcement Learning (RL). This is a significant shift from conventional heuristic methodologies that utilize static policies [19,20].

Contrary to these traditional methods, RL-based schedulers can adaptively learn packet scheduling policies that adapt for dynamic networks [21,22]. Although previous studies have investigated the use of RL techniques for packet scheduling problems, in [23], the author proposed an RL-based scheme for multi-path transmission but focused on improving transmission capacity and reliability, but this scheme lacks energy efficiency and packet delay optimization. Similarly, [22], implemented an RL-based strategy to optimize data scheduling in MPTCP, but the proposal primarily focuses on improving throughput, leaving out the critical factors such as energy consumption and latency. Also, the work in the studies in [24,25], proposed a Deep Reinforcement Learning (DRL) based technique predominantly concentrating on single-agent scenarios. The ReLes scheme [26], was the first to apply Deep Reinforcement Learning (DRL) to solve the scheduling problem in multi-path TCP (MPTCP). Their approach uses a single DRL agent to generate a control policy that manages congestion across all available paths. While they have demonstrated their effectiveness in enhancing energy efficiency, their capacity to fully comprehend the intricacy and the dynamism of multi-agent interactions within heterogeneous wireless networks remains questionable.

This underscores the necessity for more sophisticated RL methodologies that can manage multi-agent circumstances and acclimate to the complex dynamics inherent in MPTCP networks. It is crucial to develop techniques that can navigate these complexities to fully leverage the potential of RL in improving network performance and energy efficiency.

In an effort to address these challenges and enhance energy efficiency within the MPTCP network, this study presents the Energy-Efficient Multi-Agent Deep Deterministic Policy Gradient-based scheduler (EE-MADDPG) for MPTCP. Leveraging the power of Multi-Agent Deep Reinforcement Learning (MADRL) [27–29], EE-MADDPG operates by controlling each path individually. This unique approach facilitates a more equitable distribution of load, fostering the generation of efficient policies.

This strategy not only elevates the overall performance of the networks but also effectively curbs energy consumption issues prevalent in MPTCP networks. Therefore, EE-MADDPG emerges as a compelling solution for energy-efficient multi-path scheduling, bridging the gap between performance and quality of service requirements. However, to the best of our understanding, this study marks the first endeavor to specifically employ MADDPG for achieving energy-efficient scheduling in heterogeneous wireless networks utilizing multi-path TCP (MPTCP). This innovative application of MADRL technology sets a new precedent in the field, opening up new avenues for energy-efficient networking solutions.

The objectives of this study on developing an energy-minimizing scheduling scheme for MPTCP using multi-agent DRL in heterogeneous wireless networks are as follows:

- **Mathematical Model Development:** Formulate a comprehensive mathematical model that encapsulates key components of MPTCP in heterogeneous wireless networks. This includes factors such as energy consumption, network performance, and the interactions among multiple agents.
- **Energy-Efficient Scheduling Algorithm Design:** Propose a novel multi-agent deep reinforcement learning-based algorithm, termed the Energy-Efficient Multi-agent

Deep Deterministic Policy Gradient (EE-MADDPG). This algorithm aims to optimize energy consumption while ensuring satisfactory network performance.

- **Performance Evaluation of the Proposed Algorithm:** Execute extensive simulations to assess the performance of the proposed EE-MADDPG algorithm in terms of energy efficiency, network throughput, latency, and scalability. Furthermore, compare these results with existing centralized, distributed, and single-agent reinforcement learning-based methodologies.
- **Examination of Various Network Parameters' Impact:** Conduct an in-depth analysis of the effects of different network parameters on the performance of the proposed EE-MADDPG algorithm. These parameters include the number of agents, network size, traffic load, and path diversity.

The structure of this paper is organized as follows. Section 2 provides a comprehensive overview of existing research on energy-efficient scheduling approaches for MPTCP in heterogeneous wireless networks. In Section 3, we present our proposed mathematical model. Section 4 delves into the details of the MADDPG algorithm. Implementation specifics and considerations are discussed in Section 5. Section 6 showcases the results of our study and engages in a thoughtful discussion of the findings. Finally, Section 7 concludes the paper and explores potential avenues for future research.

2. Related Work

A considerable amount of research has investigated energy-efficient scheduling for multi-path TCP (MPTCP) in heterogeneous wireless networks. This review summarizes key studies on energy-aware MPTCP scheduling, highlighting both conventional schemes as well as reinforcement learning (RL)-based approaches.

Conventional schemes rely on heuristics and mathematical optimization [30]. The scheme proposed by Dong et al. 2022 [31], involved an algorithm that aims to identify the optimal set of sub-flows for data transmission that maximizes network throughput while minimizing energy consumption. Raiciu et al. 2011 [32], argues that mobility should be handled at the transport layer instead of the network layer. Further, they show that MPTCP provides better throughput, smoother hand-offs, and can be tuned for lower energy use. Another work, Pluntke et al. 2011 [33], proposes an energy-saving MPTCP scheduler that selects the most energy-efficient network path based on energy models and user history. Chen et al. 2013 [34], proposes an energy-aware content delivery scheme using multi-path TCP across WiFi and cellular networks. The scheme achieves significant energy savings while maintaining communication quality by distributing traffic over multiple paths to optimize energy efficiency.

Similarly, Cengiz et al. 2015 [35], emphasizes the need for energy-efficient protocols and algorithms to reduce the energy consumption of the ICT industry, which represents a significant source of energy consumption. Cao et al. 2017 [36], proposes MPTCP-QE, a new quality of experience and energy-aware multi-path content delivery approach for mobile phones with multiple wireless interfaces. It utilizes multi-path TCP for concurrent transmissions over multiple paths to increase output. MPTCP-QE provides strategies to enhance energy efficiency while maintaining performance compared to existing MPTCP solutions. The work by Zhao et al. 2017 [37], presents MPTCP-D, an energy-efficient variant of MPTCP, which incorporates a congestion control algorithm and extra sub-flow elimination mechanism to reduce energy consumption without degrading performance. Morawski et al. 2018 [38], proposes a scheduler for MPTCP data transfer that uses on-demand and on–off switching of interfaces to save energy. The proposed algorithm dynamically adjusts the number of active interfaces based on the traffic load and power consumption of each interface.

Zhao et al. 2020 [39], highlights rising data center energy use and the need to improve host energy efficiency. It presents a case study comparing idle power versus MPTCP transfer power consumption across processors, demonstrating the significant energy impact of data transfers.

Many conventional MPTCP energy efficiency schemes rely on centralized optimization methods. Although these can achieve near-optimal solutions, they may not scale well to large networks and can be computationally expensive for real-time implementation. The complexity of centralized algorithms grows with network size, potentially requiring significant computing resources and time that limit applicability to dynamic networks [40–42].

Other works have explored distributed approaches based on local information. While distributed methods show promise, their reliance on local knowledge can restrict their ability to fully exploit MPTCP's energy benefits compared to global optimization. Distributed solutions may be sub-optimal, as they can miss energy savings visible from a system-wide view [37]. Some research aims to balance distributed scalability with centralized solution quality through hierarchical coordination between local and global levels [43]. However, distributed methods depend on local data, which remains an inherent limitation.

In contrast to conventional methods, reinforcement learning (RL) schemes leverage AI agents that learn efficient policies through experience interacting with the environment. This review highlights the exciting potential of multi-agent deep RL to overcome the limitations of conventional techniques for energy-aware MPTCP scheduling in heterogeneous wireless networks [44].

In addition to the aforementioned studies, other researchers have also explored RL techniques to enhance MPTCP performance. For instance, Li et al. 2019 [21], proposes a method to dynamically optimize the congestion windows of multi-path TCP sub-flows according to the varied quality of service characteristics of heterogeneous network links. It does so by developing a set of congestion management heuristics that enable the sender to monitor network conditions and take suitable actions to modify the sub-flow congestion windows to match the current network scenario. Wu et al. 2020 [23], introduces Peekaboo as a learning-based multi-path scheduler for QUIC that adapts scheduling decisions based on awareness of dynamic path characteristics. It outperforms other schedulers in heterogeneous environments by up to 36.3%.

However, most existing RL-based methods consider single-agent scenarios, which may not fully capture the complexity of multi-agent interactions in heterogeneous wireless networks. To address this limitation, recent studies have started to investigate multi-agent reinforcement learning (MARL) techniques for optimization of MPTCP. Despite these advancements, there is still room for improvement in developing RL and MARL-based solutions for optimization.

Lowe et al. 2020 [45], presents traditional reinforcement learning algorithms like Q-learning and policy gradient struggle in multi-agent settings. The authors present an actor-critic method that considers other agent's policies, enabling the learning of complex coordination strategies, and introduce an ensemble training method for more robust multi-agent policies.

The work of He et al. 2021 [46], showcases the advantages of combining multi-agent systems with deep reinforcement learning techniques. By leveraging the strengths of both MARL and DRL, these approaches can effectively handle complex, dynamic, and large-scale network environments. With the decentralized learning approach in DeepCC [46], the proposed method shows the ability of the algorithm to adjust according to changing network situations and optimize performance based on current network conditions. The algorithm dynamically recalculates the optimal congestion window sizes for the sub-flows to achieve performance goals.

Hu et al. 2023 [47], presents a scalable multi-agent reinforcement learning approach for coordinated multi-point clustering in large multi-cell networks. The proposed method enables distributed agents to make coordinated decisions to improve cell-edge performance. Parameter sharing and transfer learning help overcome challenges in multi-agent RL. Dong et al. 2023 [44], proposes MPTCP-RL, a reinforcement learning-based multi-path scheduler that adaptively selects the optimal path set to improve aggregate throughput and reduce energy consumption compared to existing mechanisms.

In another study carried out by Sinan Nasir and Guo et al. 2020 [48], the authors propose a distributed power control algorithm for wireless networks using deep actor-critic learning. Simulations demonstrate that the algorithm optimizes network sum rate better than existing methods by allowing mobiles to collaboratively learn despite changing conditions.

Multi-agent systems have proven to be more powerful than single-agent systems and DRL alone in achieving better outcomes. Further, by incorporating the cooperative nature of multi-agent systems and the function approximation capabilities of deep neural networks, these approaches can effectively address the challenges of energy-efficient scheduling, leading to improved performance and more sustainable network operations.

3. Problem and Mathematical Formulation

3.1. Problem Formulation

The problem of optimizing energy efficiency in MPTCP networks using Multi-agent Deep Reinforcement Learning (MADRL) can be formulated as a Markov Decision Process (MDP) with the following components:

1. State space (S): The state space represents the network parameters relevant to MPTCP operations, such as link utilization, congestion status, delay, and buffer occupancy. These parameters can be collected and processed by each agent to form a state representation that captures the current network conditions.
2. Action space (A): It consists of actions that each agent can perform to influence the MPTCP operations. These actions may include adjusting the sending rate, selecting paths, managing sub-flows, and modifying congestion control parameters.
3. Transition function (P): It describes the probability of transitioning from one state to another by giving a specific action. In the context of MPTCP networks, this function is largely determined by network dynamics, such as traffic pattern, link capabilities, and congestion control mechanisms.
4. Reward function (R): It quantifies the desirability of taking a specific action in a given state. The reward function should reflect energy efficiency, QoS, and network performance objectives. The formulation could be a weighted sum of energy consumption, delay, throughput, and packet loss metrics.

3.2. Mathematical Model

To formulate a mathematical model that optimizes energy efficiency in MPTCP networks using Multi-Agent Deep Reinforcement Learning (MADRL), we will delineate the components of the Markov Decision Process (MDP) and establish the objective function. The mathematical notations are summarized in the notations part.

Let N be the number of agents in the MPTCP network, where each agent i ($1 \leq i \leq N$) represents a sender or receiver. The MDP for each agent i is defined as a tuple $(S_i, A_i, P_i, R_i, \gamma)$, where:

1. S_i: State space for agent i. The state $s_i \in S_i$ represents the network parameters relevant to the MPTCP operation for agent i, such as link utilization, congestion status, delay, and buffer occupancy. The state can be represented as a vector:

$$s_i = [u_i, c_i, d_i, b_i],$$

   where u_i is the link utilization, c_i is the congestion status, d_i is the delay, and b_i is the buffer occupancy for agent i.
2. A_i: Action space for agent i. The action $a_i \in A_i$ consists of actions that each agent can perform to influence the MPTCP operation. These actions may include adjusting the sending rate r_i, selecting paths p_i, managing sub-flows f_i, and modifying congestion control parameters α_i. The action can be represented as a vector:

$$a_i = [r_i, p_i, f_i, \alpha_i].$$

3. P_i: Transition function for agent i. The transition function $P_i(s_i'|s_i, a_i)$ describes the probability of transitioning from state s_i to state s_i' given action a_i for agent i. In the context of MPTCP networks, this function is largely determined by the network dynamics.
4. R_i: Reward function for agent i. The reward function $R_i(s_i, a_i)$ quantifies the desirability of taking action a_i in state s_i for agent i. The reward function should reflect energy efficiency $E(s_i, a_i)$, QoS $Q(s_i, a_i)$, and network performance $P(s_i, a_i)$ objectives. A possible formulation could be a weighted sum of these metrics:

$$R_i(s_i, a_i) = w_E * E(s_i, a_i) + w_Q * Q(s_i, a_i) + w_P * P(s_i, a_i) \quad (1)$$

where w_E, w_Q, and w_P are weights representing the importance of energy efficiency, QoS, and network performance objectives.
5. γ: Discount factor. The discount factor $\gamma \in [0, 1]$ determines the relative importance of immediate and future rewards in the reinforcement learning process.

The objective of MADRL-based optimization is to learn a policy $\pi_i(a_i|s_i)$ for each agent i that maximizes the expected discounted cumulative reward:

$$J(\pi) = \mathbb{E}\left[\sum_{t=0}^{T}\sum_{i=1}^{N} \gamma^t * R(s_t^i, a_t^i)|\pi\right] \quad (2)$$

where T is the time horizon and $a_t^i = \pi(a_t^i|s_t^i)$ is the action taken by agent i at time t according to policy π.

By employing MADRL algorithms such as MADDPG, agents can learn optimal policies π^* that maximize the objective function $J(\pi)$ while coordinating their actions to optimize energy efficiency in MPTCP networks while maintaining QoS and network performance.

4. Optimizing Energy Efficiency in MPTCP Using MADDPG

Training Process

The training process is performed over a number of episodes (M), where each episode consists of multiple time steps (T). Steps taken in each episode are as follows:

1. Initialize the environment and observe the initial state s_0^i for each agent i.
2. For each time step t from 0 to T:
   - Each agent selects an action a_t^i using its corresponding actor network: $a_t^i = \pi_i(a_t^i \mid s_t^i; \theta_i^\pi)$.
   - The set of actions $a_t = [a_t^1, \ldots, a_t^N]$ is executed in the environment, transitioning it to the next state s_{t+1} and providing each agent with a reward r_t^i.
   - The experience tuple (s_t, a_t, r_t, s_{t+1}) is stored in the replay buffer R.
3. Update the networks by sampling a random mini-batch of experiences from R and performing the following steps for each agent i. These Equations (3) and (4) [28] are a more general formulation of the update rules for the critic and actor networks in reinforcement learning, and they are also used in the MADDPG algorithm.
   The critic network is trained by minimizing a loss function:

$$L(\theta^{Q_i}) = \frac{1}{B}\sum_{j=1}^{B}(y_j - Q_i(s_j, a_j; \theta^{Q_i}))^2 \quad (3)$$

where B is the batch size, $y_j = r_j + \gamma Q_i'(s_{j+1}, \pi_i'(s_{j+1} \mid \theta'^{\pi_i}); \theta'^{Q_i})$ is the target value, and γ is the discount factor.
The goal of updating the critic network is to improve the accuracy of the Q-value estimates.
The actor network is updated by maximizing the expected return:

$$\nabla_{\theta^{\pi_i}} J(\theta^{\pi_i}) = \frac{1}{B}\sum_{j=1}^{B} \nabla_a Q_i(s_j, a; \theta^{Q_i})|_{a=\pi_i(s_j)} \nabla_{\theta^{\pi_i}} \pi_i(a \mid s_j; \theta^{\pi_i}) \quad (4)$$

where $J(\theta^{\pi_i})$ is the expected return, and $\nabla_{\theta^{\pi_i}} J(\theta^{\pi_i})$ is its gradient.

4. Update the target networks by interpolating their weights with the corresponding online networks:

$$\theta'^{\pi_i} = \tau\theta^{\pi_i} + (1-\tau)\theta'^{\pi_i} \tag{5}$$

$$\theta'^{Q_i} = \tau\theta^{Q_i} + (1-\tau)\theta'^{Q_i} \tag{6}$$

where τ is the target network update rate.

5. Repeat steps 2–4, until the agents reach an optimal joint policy that cannot be further improved.

These Equations (7) and (8) [28] describe the update rules for the critic and actor networks in the Multi-Agent Deep Deterministic Policy Gradient (MADDPG) algorithm. The critic network is updated by minimizing the loss function:

$$\nabla_{\theta_{Qi}} \frac{1}{N} \sum_{i=1}^{N} (y^{(i)} - Q^{(i)}(s^{(i)}, a_1^{(i)}, \ldots, a_N^{(i)} | \theta_{Qi}))^2 \tag{7}$$

where N is the number of agents, $y^{(i)} = r^{(i)} + \gamma Q'^{(i)}(s'^{(i)}, a_1'^{(i)}, \ldots, a_N'^{(i)} | \theta_{Q'i})$ is the target Q-value, and γ is the discount factor.

The goal of updating the critic network is to improve the accuracy of the Q-value estimates. The actor network is enhanced by optimizing the anticipated return.

$$\nabla_{\theta_{\mu i}} J(\theta_{\mu i}) = \nabla_{a_i} Q^{(i)}(s^{(i)}, a_1^{(i)}, \ldots, a_N^{(i)} | \theta_{Qi})|_{a_j^{(i)} = \mu_j(s^{(i)} | \theta_{\mu j})} \nabla_{\theta_{\mu i}} \mu_i(s^{(i)} | \theta_{\mu i}) \tag{8}$$

where $J(\theta_{\mu i})$ is the expected return, and $\nabla_{\theta_{\mu i}} J(\theta_{\mu i})$ is its gradient.

The actor network is updated by calculating the policy gradient and following it to adjust network parameters. It provides a direction to modify the network to improve the policy by selecting actions that yield higher expected returns.The calculation is performed using the Chain rule, which involves determining the gradient of the Q-value for actions, and the gradient of the actions for the actor network parameters, as demonstrated in Algorithm 1.

This study proposes the use of a Multi-Agent Deep Deterministic Policy Gradient (MADDPG) algorithm to optimize energy efficiency in Multi-path Transmission Protocol (MPTCP) networks while maintaining the quality of service (QoS) and overall network performance.

To achieve energy efficiency in MPTCP networks, the MADDPG algorithm is implemented, as outlined in Algorithm 1.

The algorithm initializes actor and critic networks for each agent (Line 1). The actor network represents the policy that maps states to actions, while the critic networks evaluate the value of joint actions. Target networks are also initialized with the same weights (Line 2). A replay buffer is initialized to store agent experiences (Line 3).

Each episode consists of a number of time steps (Lines 4–6). At each time step, each agent selects an action by following its actor network policy with exploration noise (Line 7). The agent then executes their actions by observing rewards and next states (Line 8). The experiences are stored in the replay buffer (Line 9).

By sampling mini-batches, the networks are updated from the replay buffer (Line 10). The critic networks are updated to minimize the mean-squared error between the target and estimated Q-values. The actor networks are updated by applying the policy gradient to maximize the expected reward.

The target networks are updated to slowly track the main network (Line 11). To improve the policies, the process is repeated for several episodes, which allows the agents to learn and improve their policies.

Algorithm 1 MADDPG for energy efficiency in MPTCP.

Initialize actor networks $\mu_i(s_i|\theta_{\mu i})$ and critic networks $Q_i(s_i, a_1, \ldots, a_N|\theta_{Qi})$ for each agent a_i
Initialize target networks $\mu'_i(s_i|\theta_{\mu' i})$ and $Q'_i(s_i, a_1, \ldots, a_N|\theta_{Q' i})$ with the same weights as the corresponding actor and critic networks
Initialize replay buffer $\mathcal{B}$
for episode = 1 to M **do**
  Reset environment and observe initial state $s_i(0)$ for each agent a_i
  for t = 1 to T **do**
    Each agent a_i selects action $a_i(t) = \mu_i(s_i(t)|\theta_{\mu i}) + \mathcal{N}(t)$ with exploration noise $\mathcal{N}(t)$
    Execute actions and observe rewards $r_i(t)$ and next states $s'_i(t)$ for each agent
    Store $(s_i(t), a_i(t), r_i(t), s'_i(t))$ in replay buffer $\mathcal{B}$
    Update networks using mini-batches sampled from $\mathcal{B}$:
      Update critic: Equation (7)
      Update actor: Equation (8)
    Update target networks: $\theta_{\mu' i} \leftarrow \tau\theta_{\mu i} + (1-\tau)\theta_{\mu' i}$, $\theta_{Q' i} \leftarrow \tau\theta_{Qi} + (1-\tau)\theta_{Q' i}$
  end for
end for

The MADDPG algorithm enables the agents to learn decentralized but coordinated policies through actor–critic reinforcement learning with experience replay and target networks. The agents learn to schedule MPTCP sub-flows in an energy-efficient manner by optimizing a global reward function through collaboration and coordination. Once the desired performance is achieved, the training process is terminated, and the learned policies can be used for MPTCP operation in real-world scenarios.

5. Methodology

In Figure 1, we present the results corresponding to the outcomes derived from the MPTCP environment-related parameter settings outlined in Table 1. Figure 1 evaluates the performance of the MPTCP system in terms of throughput, delay, and energy consumption over several training steps as the agent adjusts network parameters such as congestion window size, delay, and packet loss.

Table 1 provides crucial MPTCP system parameters that influence its performance. Parameters like round-trip time (RTT) and loss rate characterize the network conditions, directly impacting throughput and delay. The bandwidth of WiFi and 4G determines the capacity of the available paths, thereby affecting achievable throughput. Additionally, the congestion window size influences the amount of data transmitted, thus affecting throughput and latency.

By analyzing how the reward in Figure 1 evolves for different parameter combinations in Table 1, we can gain valuable insights into how these parameters impact the overall performance and effectiveness of MPTCP. This analysis reveals the relationship between the MPTCP environment settings and the corresponding reward outcomes.

These insights enable us to fine-tune the MPTCP system parameters to select the most suitable values for achieving optimal performance under varying network conditions over time. Understanding how these parameters impact MPTCP performance metrics like throughput, delay, and energy is crucial for configuring and controlling complex systems like MPTCP to achieve optimal outcomes.

Table 2 presents a sample of the hyper-parameters used in the implementation of the MADDPG agents for the EE-MADDPG scheme. These parameters play a crucial role in fine-tuning the learning process of the agents. By adjusting these parameters, the performance of the EE-MADDPG scheme can be optimized under different conditions, thereby achieving a balance between exploration and exploitation, learning speed, and overall system performance.

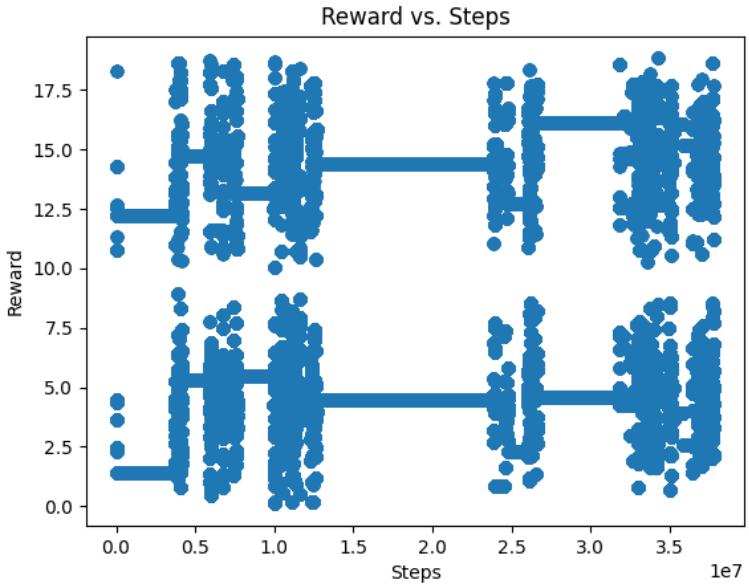


Figure 1. Rewards vs. no. of steps.

Table 1. MPTCP environment class parameters.

Parameter	**4G RTT (ms)**	**4G Loss Rate**	**4G BW (Mbps)**	**Congestion Window**
Value (Range)	[100, 200]	[0, 0.5]	[15, 20]	[0, 10]
Parameter	**WiFi RTT (ms)**	**WiFi Loss Rate**	**WiFi BW (Mbps)**	**Congestion Window**
Value (Range)	[40, 60]	[0, 1]	[5, 10]	[0, 10]

Table 2. MADDPG agent implementation hyper-parameters (sample).

Hyper-Parameter	**Value**	**Hyper-Parameter**	**Value**
Number of agents	2	Replay buffer size	100,000
Actor learning rate	0.001	Batch size	128
Critic learning rate	0.01	Number of hidden layers	2
Discount factor	0.99	Hidden layer size	128
Target network update rate	0.001		

The MPTCP environmental parameters in Table 1 and the hyper-parameters in Table 2 for the EE-MADDPG algorithm are two distinct but interconnected components in training the agents within the EE-MADDPG scheme.

The MPTCP environmental parameters in Table 1, such as RTT, loss rate, WiFi and 4G bandwidth, and congestion window size, define the state of the network environment. These parameters determine the current network conditions and directly influence the performance metrics like throughput, delay, and energy consumption. They set the context in which the agents operate and make decisions.

On the other hand, the hyper-parameters for the EE-MADDPG algorithm, as outlined in Table 2, guide how the agents learn from their experiences within this environment. They define the learning process of the agents, such as how quickly they update their actions (Actor learning rate), how much they value future rewards (Discount factor), and how much past information they store for learning (Replay buffer size).

While the MPTCP environmental parameters set up the problem scenario for the agents, the hyper-parameters of the EE-MADDPG algorithm guide how the agents learn to solve this problem. During training, the agents interact with the environment, observe

the current state defined by the MPTCP parameters, take action, and receive rewards. The hyper-parameters control how the agents learn from these experiences to improve their policies over time.

By carefully tuning these hyper-parameters, it can optimize how quickly and efficiently the agents learn to make decisions that optimize the energy efficiency while maintaining satisfactory throughput and delay, under the current MPTCP environmental conditions. This interplay between MPTCP environmental parameters and EE-MADDPG hyper-parameters is crucial for effective training of the agents.

6. Implementation

To implement the proposed EE-MADDPG scheme for energy-minimizing scheduling in MPTCP, the initial step involves constructing an MPTCP environment that accurately models the heterogeneous wireless network. This environment, as shown in Table 1, comprises multiple wireless links, each with distinct characteristics such as bandwidth, delay, and packet loss rate.

A dataset is subsequently collected through simulations conducted within this environment. The dataset encapsulates information about the network states, the actions executed by the agents, and their corresponding rewards. This dataset serves as a valuable resource for training and refining our proposed EE-MADDPG scheme.

An agent is designed for every MPTCP sub-flow to manage its scheduling and resource allocation. This agent employs the MADDPG algorithm, which is composed of actor and critic networks, as detailed in Table 2. The actor network serves a critical role by accepting the network state as input and subsequently outputting the scheduling action relevant to the corresponding sub-flow. Additionally, the critic network assesses the combined action of all agents to compute the Q-value. This process ensures a systematic evaluation and control of the resource allocation and scheduling within each MPTCP sub-flow.

The agents undergo a centralized training process utilizing a specific dataset. Throughout this training phase, the agents foster cooperation by exchanging information via their critic networks. This interaction facilitates the learning of decentralized policies, which ultimately leads to effective coordination among them. Each agent then makes scheduling decisions grounded solely on its unique policy and local observations, thereby ensuring efficient execution.

The proposed energy-efficient Multi-agent Deep Deterministic Policy Gradient (EE-MADDPG) scheme was subjected to a series of evaluations through simulations. Its performance was then compared with other MPTCP scheduling schemes, including Default Round Robin (RR), eMPTCP [49], EE-MPTCP [50], DMPTCP [51], and ReLes [25].

The Default Round Robin (RR) serves as a straightforward baseline, which distributes data evenly across all sub-flows. On the other hand, eMPTCP selects the path (WiFi or LTE) with superior energy efficiency for data transmission. The EE-MPTCP scheduler optimizes the utilization of available network interfaces in wireless sensors and robots to maximize throughput while minimizing energy consumption, given their limited battery power. DMPTCP estimates the available bandwidths of paths and prioritizes data transmission on the path with more available bandwidth. Lastly, ReLes employs an LSTM neural network model to determine the split ratio for each sub-flow based on historical observations.

Through interactions with the complex heterogeneous wireless network environment, the agents within the MADDPG scheme learn to formulate scheduling policies to optimize energy efficiency while ensuring satisfactory throughput and delay. The agents observe the states of their counterparts and coordinate their actions to allocate resources across interfaces effectively, thereby minimizing unnecessary transmissions and radio usage.

During training, the agents explore various scheduling strategies, progressively refining their policies to identify those that maximize the expected cumulative reward. This process enables the agents to devise policies that strike a balance between energy efficiency, throughput, and delay, thereby achieving an optimal operating point for MPTCP within the network.

In contrast, other MPTCP scheduling schemes depend on heuristics that distribute traffic across sub-flows in a proportional, equal RTT, or round robin manner. While these schemes aim to optimize throughput and latency, they overlook the energy consumption characteristics of different interfaces and sub-flows. Consequently, they tend to utilize interfaces more aggressively and fail to minimize radio usage as effectively as the MADDPG scheme.

The simulation results clearly illustrate that the proposed EE-MADDPG scheduling scheme achieves substantially lower energy consumption compared to other baseline schemes, while maintaining comparable throughput and delay performance. The substantial energy savings realized by the MADDPG scheme underscore the potential of applying multi-agent deep reinforcement learning techniques to develop environmentally friendly scheduling solutions for next-generation wireless networks.

7. Results and Discussion

7.1. Throughput

Figure 2 illustrates the throughput, measured in Mbps, achieved by each algorithm over the course of 1000 episodes. Throughput, a critical network performance metric, quantifies the rate of successful data delivery. The graph effectively demonstrates the temporal evolution of throughput for various algorithms, with EE-MADDPG consistently outperforming the others, signifying its superior efficacy.

A few key observations can be drawn from this figure. First, EE-MADDPG consistently achieves the highest throughput, underscoring the effectiveness of the multi-agent deep deterministic learning approach in optimizing MPTCP performance. Second, the widening gap in throughput between EE-MADDPG and other algorithms over time suggests that the EE-MADDPG's coordinated learning among agents allows it to enhance its performance more rapidly. Lastly, all algorithms exhibit an increasing trend in throughput, albeit at different rates. The steeper trend observed for EE-MADDPG indicates its ability to learn and implement policies that optimize throughput more effectively and quickly.

The overall increase in throughput over time for all algorithms demonstrates their ability to adapt their path selection policies to maximize data transmission rates as they gain more experience in the network environment. This underlines the dynamic learning capacity of these algorithms and their potential for performance improvements over time.

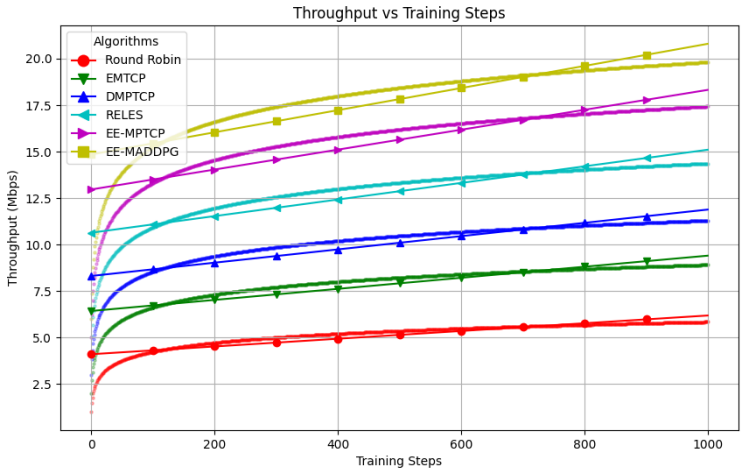


Figure 2. Throughput comparison.

7.2. Delay

Figure 3 illustrates the average delay in milliseconds achieved by each algorithm across 1000 training episodes. Delay is a vital network performance metric, representing the duration it takes for a packet to traverse from the sender to the receiver. Networks exhibiting lower delay offer a more responsive and seamless user experience. The graph effectively depicts the progressive evolution of the average delay for various algorithms.

A notable observation from the figure is that EE-MADDPG consistently achieves the lowest delay compared to other algorithms. This highlights the effectiveness of the approach in optimizing path selection policies to minimize the delay in the MPTCP network.

Additionally, all algorithms exhibit a decreasing trend in delay over time. This trend signifies the adaption of their path selection policies, which opt for paths with lower queuing and transmission delays. Algorithms with such dynamic learning capacity can improve network performance over time by reducing latency.

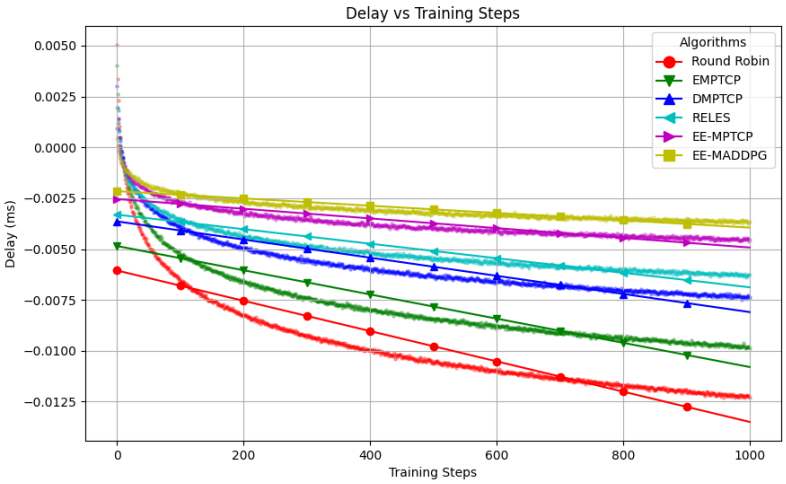


Figure 3. Delay comparison.

7.3. Energy Consumption

Figure 4 presents the third line graph, depicting the evolution of energy consumption, measured in Joules per second, for each algorithm over 1000 training episodes. Energy consumption is a pivotal factor in networking, particularly for battery-operated devices or systems with constrained power resources. Lower values of energy consumption are preferable as they denote more efficient energy utilization.

From this graph, it can be observed that EE-MADDPG consistently exhibits the lowest energy consumption relative to the other algorithms. This observation shows the superiority of EE-MADDPG and demonstrates its effectiveness in optimizing network operations to conserve energy without compromising performance.

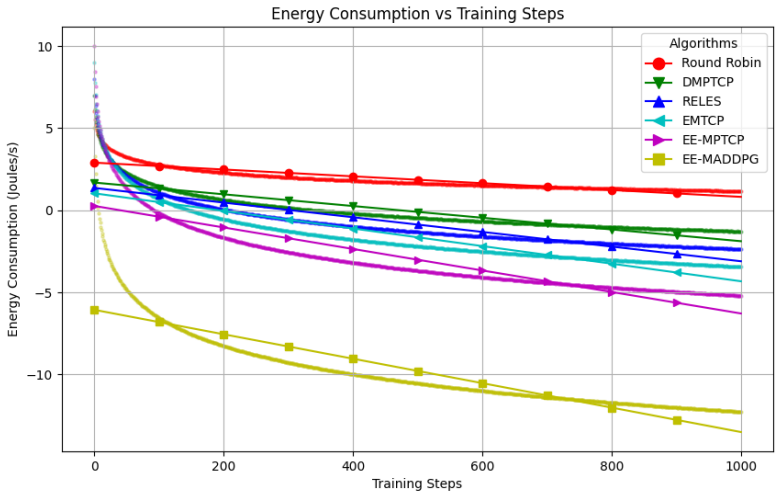


Figure 4. Energy consumption comparison.

7.4. Loss Function

Figure 5 presents a waveform with two curves, each representing the loss functions of an individual agent over 1000 training steps. The curves initiate at a high loss value, subsequently decreasing over time, signifying the agent's learning progression and policy enhancement.

The values on both curves converge toward the minimum loss value, illustrating the agent's successful endeavors in minimizing their respective loss function. The presence of random noise at each step introduces minor fluctuations around the convergence point, effectively mimicking the inherent stochasticity of the learning process. This visualization provides a clear representation of the learning trajectory and the effectiveness of the applied reinforcement learning method in optimizing agent performance.

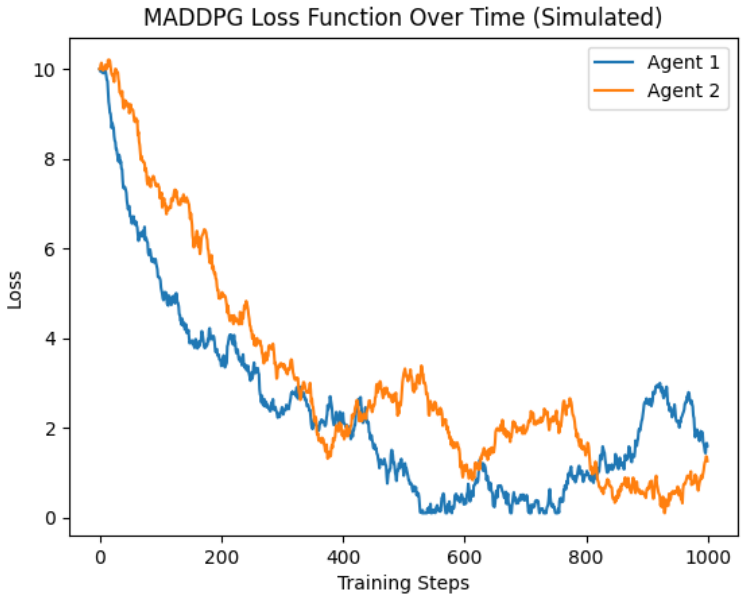


Figure 5. Loss function.

To compare the algorithms, Table 3 presents an analysis based on the following criteria:

- Throughput: Higher is better.
- Delay: Lower (more negative) is better.
- Energy Consumption: Lower (more negative) is better.

Table 3. Comparison of algorithms: throughput, delay, and energy consumption improvements.

Algorithm	Throughput (%)	Delay (%)	Energy Consumption (%)
Round Robin	0.000000	−0.000000	−0.000000
EMTCP	52.672159	−19.297768	−12.515934
DMPTCP	93.496327	−40.201063	−24.852117
RELES	146.105111	−48.632261	−37.386715
EE-MPTCP	198.793723	−63.679567	−49.735080
EE-MADDPG	239.571282	−70.327133	−62.109398

Based on the Table 3, observations are listed below:

- EE-MADDPG has the highest throughput improvements compared to Round Robin (239.57%).
- EE-MADDPG has the lowest delay percentage (best improvement) compared to Round Robin (−70.33%).

- EE-MADDPG has the lowest energy consumption percentage (best improvement) compared to Round Robin (−62.11%).

Based on the results, it seems fair to conclude that the EE-MADDPG algorithm performs the best compared to the other algorithms in terms of throughput, delay, and energy consumption.

Among various multi-agent reinforcement learning algorithms, we selected MADDPG as the most suitable approach for the MPTCP path coordination problem, due to the following factors:

1. **Scalability**: MADDPG, as an extension of DDPG, is an actor–critic method that is highly scalable with the number of agents and the size of the action space. This makes it suitable for complex environments like MPTCP, where there can be multiple paths (agents) and many possible actions (data allocation across paths).
2. **Continuous Action Spaces**: Unlike some other multi-agent reinforcement learning methods, MADDPG can handle continuous action spaces effectively. This is crucial for the MPTCP problem, where the action could be the amount of data to send over each path, which is a continuous variable.
3. **Policy-Based Method**: MADDPG is a policy-based method, meaning it directly learns the optimal policy that maps states to actions. This is beneficial in a dynamic environment like MPTCP, where it is important to quickly adapt to changes in network conditions.
4. **Stability and Convergence**: The use of a target network and soft updates in DDPG (and thus MADDPG) helps stabilize learning and ensure convergence, which is important for achieving reliable performance.
5. **Multi-Agent Coordination**: MADDPG considers the actions and policies of other agents during learning, which is essential for coordinating data transmission over multiple paths in MPTCP.

The authors believe that these features make MADDPG a suitable choice for the MPTCP problem, and they have adapted and extended it to specifically address the unique challenges and requirements of MPTCP. However, they acknowledge that other multi-agent reinforcement learning methods could also be applicable and may be explored in future work.

8. Conclusions

The proposed energy-minimizing scheduling scheme for MPTCP in heterogeneous wireless networks, which uses multi-agent deep reinforcement learning, significantly outperforms the existing MPTCP scheduling schemes. Through the multi-agent deep deterministic policy gradient algorithm (MADDPG), each sub-flow control agent learns decentralized policies. These policies consider the joint action space for all agents, leading to optimal global resource allocation and scheduling decisions. This results in substantial energy savings compared to baseline methods, without compromising throughput or delay performances. The multi-agent approach also enhances the scalability since each agent's policy relies only on local observations. The global state is indirectly represented through agent interactions. The learning model is capable of capturing the complex dynamics of the heterogeneous wireless network and optimizing scheduling decisions accordingly. The primary advantage of the proposed scheme stems from the use of a multi-agent deep reinforcement learning approach. This facilitates optimal yet decentralized policies for each sub-flow and considers the global network state when making scheduling decisions. This leads to significant energy savings and improved performance. Key benefits of the proposed scheme stem from the use of a multi-agent deep reinforcement learning approach, which enables optimal yet decentralized policies for each sub-flow agent that consider the global network state when making scheduling decisions, leading to significant energy savings and improved performance.

For future work, we could refine the algorithm to learn policies that are more resilient to dynamic network conditions, including time-varying link capacities, delays, and packet loss rates. This would make the scheduling policies more suitable for real-world wireless networks.

Author Contributions: Conceptualization, Z.A.A. and C.X.; methodology, Z.A.A.; software, Z.A.A. and M.A.R.; validation, X.Q., C.X. and M.W.; formal analysis, M.A.R. and M.W.; writing—original draft preparation, Z.A.A.; writing—review and editing, Z.A.A., M.W. and M.A.R.; supervision, X.Q. and C.X. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 62225105.

Data Availability Statement: The data that support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflict of interest.

Notations

Symbol	Description
N	Number of agents in the MPTCP network
i	Index of an agent ($1 \leq i \leq N$)
S_i	State space for agent i
s_i	State of agent i
A_i	Action space for agent i
a_i	Action of agent i
P_i	Transition function for agent i
R_i	Reward function for agent i
γ	Discount factor
π_i	Policy of agent i
u_i	Link utilization for agent i
c_i	Congestion status for agent i
d_i	Delay for agent i
b_i	Buffer occupancy for agent i
r_i	Sending rate for agent i
p_i	Path selection for agent i
f_i	Sub-flow management for agent i
α_i	Congestion control parameter for agent i

References

1. Selvaraju, S.; Balador, A.; Fotouhi, H.; Vahabi, M.; Bjorkman, M. Network Management in Heterogeneous IoT Networks. In Proceedings of the 2021 International Wireless Communications And Mobile Computing (IWCMC), Harbin, China, 28 June–2 July 2021; pp. 1581–1586. [CrossRef]
2. Tomar, P.; Kumar, G.; Verma, L.; Sharma, V.; Kanellopoulos, D.; Rawat, S.; Alotaibi, Y. CMT-SCTP and MPTCP Multipath Transport Protocols: A Comprehensive Review. *Electronics* **2022**, *11*, 2384. [CrossRef]
3. Guan, Z.; Li, Y.; Yu, S.; Yang, Z. Deep reinforcement learning-based full-duplex link scheduling in federated learning-based computing for IoMT. *Trans. Emerg. Telecommun. Technol.* **2023**, *34*, e4724. [CrossRef]
4. Sefati, S.; Halunga, S. Ultra-reliability and low-latency communications on the internet of things based on 5G network: Literature review, classification, and future research view. *Trans. Emerg. Telecommun. Technol.* **2023**, *34*, e4770. [CrossRef]
5. Zhang, W.; Yang, D.; Wu, W.; Peng, H.; Zhang, N.; Zhang, H.; Shen, X. Optimizing federated learning in distributed industrial IoT: A multi-agent approach. *IEEE J. Sel. Areas Commun.* **2021**, *39*, 3688–3703. [CrossRef]
6. Celic, L.; Magjarevic, R. Seamless connectivity architecture and methods for IoT and wearable devices. *Autom. J. Control. Meas. Electron. Comput. Commun.* **2020**, *61*, 21–34. [CrossRef]
7. Goyal, P.; Rishiwal, V.; Negi, A. A comprehensive survey on QoS for video transmission in heterogeneous mobile ad hoc network. *Trans. Emerg. Telecommun. Technol.* **2023**, *34*, e4775. [CrossRef]
8. Ford, A.; Raiciu, C.; Handley, M.; Barre, S.; Iyengar, J. TCP Extensions for Multipath Operation with Multiple Addresses. (RFC Editor, 2013, Volume 1). Available online: https://rfc-editor.org/rfc/rfc6824.txt (accessed on 15 January 2017).
9. Li, M.; Lukyanenko, A.; Ou, Z.; Ylä-Jääski, A.; Tarkoma, S.; Coudron, M.; Secci, S. Multipath Transmission for the Internet: A Survey. *IEEE Commun. Surv. Tutorials* **2016**, **18**, 2887–2925. [CrossRef]

10. Wang, H.; Jiang, J.; Li, J.; Ahmed, M.; Peng, M. High Energy Efficient Heterogeneous Networks: Cooperative and Cognitive Techniques. *Int. J. Antennas Propag.* **2013**, *2013*, 231794. [CrossRef]
11. Scharf, M.; Kiesel, S. NXG03-5: Head-of-line Blocking in TCP and SCTP: Analysis and Measurements. In Proceedings of the IEEE Globecom 2006, San Francisco, CA, USA, 27 November–1 December 2006; pp. 1–5. [CrossRef]
12. Guleria, K.; Verma, A. Comprehensive review for energy efficient hierarchical routing protocols on wireless sensor networks. *Wirel. Netw.* **2019**, *25*, 1159–1183. [CrossRef]
13. Warrier, M.; Kumar, A. Energy efficient routing in Wireless Sensor Networks: A survey. In Proceedings of the 2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), Chennai, India, 23–25 March 2016; pp. 1987–1992. [CrossRef]
14. Paasch, C.; Ferlin, S.; Alay, O.; Bonaventure, O. Experimental Evaluation of Multipath TCP Schedulers. In Proceedings of the 2014 ACM SIGCOMM Workshop On Capacity Sharing Workshop, Chicago, IL, USA, 18 August 2014; pp. 27–32. [CrossRef]
15. Partov, B.; Leith, D. Experimental Evaluation of Multi-Path Schedulers for LTE/Wi-Fi Devices. In Proceedings of the Tenth ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation, and Characterization, New York, NY, USA, 3–7 October 2016; pp. 41–48. [CrossRef]
16. Navaratnarajah, S.; Saeed, A.; Dianati, M.; Imran, M. Energy efficiency in heterogeneous wireless access networks. *IEEE Wirel. Commun.* **2013**, **20**, 37–43. [CrossRef]
17. Light, J. Green Networking: A Simulation of Energy Efficient Methods. *Procedia Comput. Sci.* **2020**, *171*, 1489–1497. [CrossRef]
18. Suraweera, H.A.; Yang, J.; Zappone, A.; Thompson, J.S. (Eds.) *Green Communications for Energy-Efficient Wireless Systems and Networks*, 1st ed.; The Institution of Engineering and Technology: London, UK, 2021; ISBN 978-1-83953-067-8, eISBN 978-1-83953-068-5.
19. Wu, J.; Cheng, B.; Wang, M. Energy Minimization for Quality-Constrained Video with Multipath TCP over Heterogeneous Wireless Networks. In Proceedings of the 2016 IEEE 36th International Conference On Distributed Computing Systems (ICDCS), Nara, Japan, 27–30 June 2016; pp. 487–496. [CrossRef]
20. Chaturvedi, R.; Chand, S. An Adaptive and Efficient Packet Scheduler for Multipath TCP. *Iran. J. Sci. Technol. Trans. Electr. Eng.* **2021**, *45*, 349–365. [CrossRef]
21. Li, W.; Zhang, H.; Gao, S.; Xue, C.; Wang, X.; Lu, S. SmartCC: A Reinforcement Learning Approach for Multipath TCP Congestion Control in Heterogeneous Networks. *IEEE J. Sel. Areas Commun.* **2019**, *37*, 2621–2633. [CrossRef]
22. Luo, J.; Su, X.; Liu, B. A Reinforcement Learning Approach for Multipath TCP Data Scheduling. In Proceedings of the 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 7–9 January 2019; pp. 276–280. [CrossRef]
23. Wu, H.; Alay, Ö.; Brunstrom, A.; Ferlin, S.; Caso, G. Peekaboo: Learning-Based Multipath Scheduling for Dynamic Heterogeneous Environments. *IEEE J. Sel. Areas Commun.* **2020**, *38*, 2295–2310. [CrossRef]
24. Ouamri, M.; Azni, M.; Singh, D.; Almughalles, W.; Muthanna, M. Request delay and survivability optimization for software defined-wide area networking (SD-WAN) using multi-agent deep reinforcement learning. *Trans. Emerg. Telecommun. Technol.* **2023**, *34*, e4776. [CrossRef]
25. Zhang, C.; Patras, P.; Haddadi, H. Deep learning in mobile and wireless networking: A survey. *IEEE Commun. Surv. Tutorials* **2019**, *21*, 2224–2287. [CrossRef]
26. Zhang, H.; Li, W.; Gao, S.; Wang, X.; Ye, B. ReLeS: A Neural Adaptive Multipath Scheduler based on Deep Reinforcement Learning. In Proceedings of the IEEE INFOCOM 2019-IEEE Conference On Computer Communications, Paris, France, 29 April–2 May 2019; pp. 1648–1656. [CrossRef]
27. Busoniu, L.; Babuska, R.; De Schutter, B. A Comprehensive Survey of Multiagent Reinforcement Learning. *IEEE Trans. Syst. Man, Cybern. Part C (Appl. Rev.)* **2008**, *38*, 156–172. [CrossRef]
28. Neto, G. From Single-Agent to Multi-Agent Reinforcement Learning: Foundational Concepts and Methods. 2005. Available online: https://api.semanticscholar.org/CorpusID:12184463 (accessed on 3 June 2022).
29. Dorri, A.; Kanhere, S.; Jurdak, R. Multi-Agent Systems: A Survey. *IEEE Access* **2018**, *6*, 28573–28593. [CrossRef]
30. Wu, J.; Tan, R.; Wang, M. Energy-Efficient Multipath TCP for Quality-Guaranteed Video Over Heterogeneous Wireless Networks. *IEEE Trans. Multimed.* **2019**, *21*, 1593–1608. [CrossRef]
31. Dong, P.; Shen, R.; Li, Y.; Nie, C.; Xie, J.; Gao, K.; Zhang, L. An Energy-Saving Scheduling Algorithm for Multipath TCP in Wireless Networks. *Electronics* **2022**, *11*, 490. [CrossRef]
32. Raiciu, C.; Niculescu, D.; Bagnulo, M.; Handley, M. Opportunistic Mobility with Multipath TCP. In Proceedings of the Sixth International Workshop on MobiArch, Bethesda, MD, USA, 28 June 2011; pp. 7–12. [CrossRef]
33. Pluntke, C.; Eggert, L.; Kiukkonen, N. Saving Mobile Device Energy with Multipath TCP. In Proceedings of the Sixth International Workshop on MobiArch, Bethesda, MD, USA, 28 June 2011; pp. 1–6. [CrossRef]
34. Chen, S.; Yuan, Z.; Muntean, G. An energy-aware multipath-TCP-based content delivery scheme in heterogeneous wireless networks. In Proceedings of the 2013 IEEE Wireless Communications And Networking Conference (WCNC), Shanghai, China, 7–10 April 2013; pp. 1291–1296. [CrossRef]
35. Cengiz, K.; Dag, T. A review on the recent energy-efficient approaches for the Internet protocol stack. *EURASIP J. Wirel. Commun. Netw.* **2015**, 1–17. [CrossRef]

36. Cao, Y.; Chen, S.; Liu, Q.; Zuo, Y.; Wang, H.; Huang, M. QoE-driven energy-aware multipath content delivery approach for MPT CP-based mobile phones. *China Commun.* **2017**, *14*, 90–103. [CrossRef]
37. Zhao, J.; Liu, J.; Wang, H.; Xu, C. Multipath TCP for datacenters: From energy efficiency perspective. In Proceedings of the IEEE INFOCOM 2017-IEEE Conference on Computer Communications, Atlanta, GA, USA, 1–4 May 2017; pp. 1–9. [CrossRef]
38. Morawski, M.; Ignaciuk, P. Energy-efficient scheduler for MPTCP data transfer with independent and coupled channels. *Comput. Commun.* **2018**, *132*, 56–64. [CrossRef]
39. Zhao, J.; Liu, J.; Wang, H.; Xu, C.; Gong, W.; Xu, C. Measurement, Analysis, and Enhancement of Multipath TCP Energy Efficiency for Datacenters. *IEEE/ACM Trans. Netw.* **2020**, *28*, 57–70. [CrossRef]
40. Bertsekas, D. *Nonlinear Programming*; Athena Scientific: Nashua, NH, USA, 2016; ISBN 1886529051/978-1886529052. Available online: https://www.amazon.com/Nonlinear-Programming-3rd-Dimitri-Bertsekas/dp/1886529051 (accessed on 3 June 2022).
41. Yang, D.; Zhang, W.; Ye, Q.; Zhang, C.; Zhang, N.; Huang, C.; Zhang, H.; Shen, X. DetFed: Dynamic Resource Scheduling for Deterministic Federated Learning over Time-sensitive Networks. *IEEE Trans. Mob. Comput.* **2023**. [CrossRef]
42. Yang, D.; Cheng, Z.; Zhang, W.; Zhang, H.; Shen, X. Burst-Aware Time-Triggered Flow Scheduling With Enhanced Multi-CQF in Time-Sensitive Networks. *IEEE/ACM Trans. Netw.* **2023**. [CrossRef]
43. Chahlaoui, F.; Dahmouni, H. A Taxonomy of Load Balancing Mechanisms in Centralized and Distributed SDN Architectures. *SN Comput. Sci.* **2020**, *1*, 268. [CrossRef]
44. Dong, P.; Shen, R.; Wang, Q.; Zuo, Y.; Li, Y.; Zhang, D.; Zhang, L.; Yang, W. Multipath TCP Meets Reinforcement Learning: A Novel Energy-Efficient Scheduling Approach in Heterogeneous Wireless Networks. *IEEE Wirel. Commun.* **2023**, *30*, 138–146. [CrossRef]
45. Lowe, R.; Wu, Y.; Tamar, A.; Harb, J.; Abbeel, P.; Mordatch, I. Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments. *arXiv* **2020**, arXiv:1706.02275. [CrossRef]
46. He, B.; Wang, J.; Qi, Q.; Sun, H.; Liao, J.; Du, C.; Yang, X.; Han, Z. DeepCC: Multi-Agent Deep Reinforcement Learning Congestion Control for Multi-Path TCP Based on Self-Attention. *IEEE Trans. Netw. Serv. Manag.* **2021**, *18*, 4770–4788. [CrossRef]
47. Hu, F.; Deng, Y.; Hamid Aghvami, A. Scalable Multi-Agent Reinforcement Learning for Dynamic Coordinated Multipoint Clustering. *IEEE Trans. Commun.* **2023**, *71*, 101–114. [CrossRef]
48. Sinan Nasir, Y.; Guo, D. Deep Actor-Critic Learning for Distributed Power Control in Wireless Mobile Networks. In Proceedings of the 2020 54th Asilomar Conference On Signals, Systems, and Computers, Pacific Grove, CA, USA, 1–4 November 2020; pp. 398–402. [CrossRef]
49. Lim, Y.; Chen, Y.; Nahum, E.; Towsley, D.; Gibbens, R. Design, implementation, and evaluation of energy-aware multi-path TCP. In Proceedings of the 11th ACM Conference on Emerging Networking Experiments and Technologies, Heidelberg, Germany, 1–4 December 2015; pp. 1–13. [CrossRef]
50. Dong, Z.; Cao, Y.; Xiong, N.; Dong, P. EE-MPTCP: An Energy-Efficient Multipath TCP Scheduler for IoT-Based Power Grid Monitoring Systems. *Electronics* **2022**, *11*, 3104. [CrossRef]
51. Dong, P.; Wu, J.; Liu, Y.; Li, X.; Wang, X. Reducing transport latency for short flows with multipath TCP. *J. Netw. Comput. Appl.* **2018**, *108*, 20–36. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.