electronics

Article

Deep Reinforcement Learning-Based Joint Scheduling of 5G and TSN in Industrial Networks

Yuan Zhu [1], Lei Sun [1,2,*], Jianquan Wang [1,2], Rong Huang [3] and Xueqin Jia [3]

[1] School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China; m202120748@xs.ustb.edu.cn (Y.Z.); wangjianquan@ustb.edu.cn (J.W.)
[2] Key Laboratory of Knowledge Automation for Industrial Processes of Ministry of Education, University of Science and Technology Beijing, Beijing 100083, China
[3] China Unicom Research Institute, Beijing 100048, China; huangr27@chinaunicom.cn (R.H.); jiaxq21@chinaunicom.cn (X.J.)
* Correspondence: sun_lei@ustb.edu.cn; Tel.: +86-10-62332347

Abstract: 5th-Generation (5G) and Time-Sensitive Networking (TSN) are regarded as competitive new technologies for future industrial networks; 5G-TSN collaboration transmission has drawn more attention because it can provide a guarantee of low-latency, ultra-reliable and deterministic transmission for time-critical automation applications. However, the methodologies of resource scheduling mechanisms in 5G and Time-Sensitive Networking (TSN) are quite different, which may lead to an inefficient Quality of Service (QoS) guarantee for deterministic transmission across 5G and TSN. Therefore, an efficient 5G-TSN joint scheduling algorithm based on Deep Deterministic Policy Gradient (DDPG) is proposed and analyzed in this article. The proposed algorithm takes both 5G radio channel information and the Gate Control List (GCL) state in the TSN domain into consideration, aiming to provide a latency guarantee for time-triggered applications across 5G and TSN as well as a throughput guarantee for video applications in 5G systems. The simulation results compare the latency and throughput performance of the proposed joint scheduling algorithm with several traditional 5G scheduling algorithms; meanwhile, several GCL setting methods are given to verify the impacts on latency and throughput performance within the proposed algorithm. The simulation results demonstrate that the proposed DDPG-based joint scheduling algorithm can significantly enhance the multi-application-carrying capability of 5G-TSN collaboration architecture.

Keywords: 5G-TSN collaboration; joint resource scheduling; deep reinforcement learning; gate control list

Citation: Zhu, Y.; Sun, L.; Wang, J.; Huang, R.; Jia, X. Deep Reinforcement Learning-Based Joint Scheduling of 5G and TSN in Industrial Networks. *Electronics* **2023**, *12*, 2686. https://doi.org/10.3390/electronics12122686

Academic Editor: Stefano Scanzio

Received: 18 May 2023
Revised: 8 June 2023
Accepted: 13 June 2023
Published: 15 June 2023

1. Introduction

With the evolution of cellular mobile communication technology, 5G ultra-reliable low latency communication (uRLLC) is gradually being applied to factory automation fields because of its excellent QoS guarantee [1]. Factory automation information is usually time-sensitive and safety-critical, and is essential for the performance of automation systems, so it requires carrying networks qualified with capabilities of low latency, high reliability and deterministic guarantee. Therefore, the integration of 5G and factory automation applications is creating new technical solutions for Industrial 4.0 while also introducing more challenges to 5G.

Deterministic communication is usually fulfilled by assuring bounded latency and low jitter for time-critical applications [2]. TSN is an Ethernet-based networking technology and is standardized in the IEEE 802.1 task group. With time synchronization among bridges and stations, TSN proposes several traffic-shaping schemes based on a Gate Control List (GCL) to send time-critical applications at a precise point in time, and this time-triggered scheduling paradigm can provide a bounded latency guarantee for time-critical applications regardless of other ongoing flows competing for the same network

resources simultaneously [3]. In order to promote the deterministic transmission capability of 5G, 3GPP has introduced TSN to integrate with 5G systems (5GSs) in Release 16 [4]. According to 3GPP specifications, a set of new functionalities has been incorporated into 5GS architecture to connect TSN with 5G in both control and data planes. There are two main scenarios of integration between 5GS and TSN: one is 5GSs as a TSN bridge entity and the other is using TSN in 5G Xhaul networks (e.g., fronthaul, midhaul and backhaul).

The 3GPP Release 16 proposes an integration framework for 5G-TSN as well as new 5G function entities in both control and data planes to support TSN protocols. However, 5G and TSN are completely different networking methodologies in terms of physical layer techniques, MAC schemes and upper layer protocols. Therefore, how to guarantee time-critical communication services transmission across 5G and TSN has become a hot topic [5]. It requires making a resource allocation decision in 5G gNB dynamically based on GCL status. However, to the best of the authors' knowledge, limited literature addresses the resource scheduling of 5G radio access networks taking into consideration TSN scheduling.

The remainder of this article is organized as follows. In Section 2, we first overview the relevant TSN scheduling scheme, and give a detailed description of architecture and related research works about resource scheduling schemes in 5G-TSN integration networks. In Sections 3 and 4, we explain the system model and formulate and analyze the optimal problem. Then a joint resource scheduling algorithm based on DDPG and corresponding simulation results are described in Sections 5 and 6. Finally, this article is concluded in Section 7.

2. Related Work

2.1. Overview of TSN Scheduling Schemes

Standardization within the IEEE 802.1 TSN task group (TG) enables Ethernet-based networks to be reliable real-time communication networks, which can satisfy the demands of multiple time-critical and non-time-critical applications simultaneously. TSN standards define mechanisms in the aspects of time synchronization, bounded low latency, reliability and resource management to provide deterministic services in local areas. TSN scheduling schemes play an important role in bounded latency, such as Time-Aware Shaping (TAS) defined in IEEE 802.1Qbv, Cyclic Queueing and Forwarding (CQF) defined in IEEE 802.1Qch, Frame Preemption (FP) defined in IEEE 802.1Qbu, and Asynchronous Traffic Shaper (ATS) defined in IEEE 802.1Qcr. Among these TSN scheduling schemes, TAS is widely used for industrial scenarios due to its technological maturity and "no-waiting" scheduling character. Therefore, the TSN scheduling method considered in this article is based on TAS. More details on the standardization efforts of IEEE 802.1 TSN TG can be found in [6,7].

As shown in Figure 1, different applications are mapped into various priority queues in the egress of TSN switches; meanwhile, based on time synchronization achieved among devices, TAS adopts a time-triggered gate mechanism to enable (OPEN) or disable (CLOSE) transmission states of priority queues. Furthermore, GCL is used to set the gate state and accurate eligible transmission period of each priority queue, aiming to decide when and how many frames are selected for transmission.

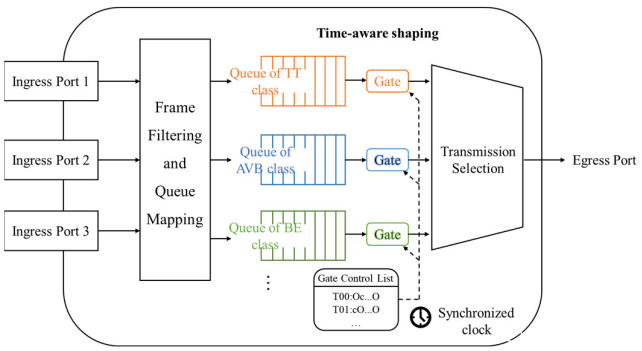

Figure 1. Time-aware shaping in TSN switch.

The GCL of each export in the switch is usually pre-configured and operated circularly, so the TAS can transmit frames based on a precise time. Furthermore, the eligible transmission period defined by GCL determines the amount of network resources that can be used for each queue, which regulates the transmission rate of each queue. With GCL settings, the higher priority applications, such as industrial control or real-time multimedia, are separated from other non-time-critical applications with lower priority and can achieve dedicated time slots for transmission. Therefore, TAS can provide a deterministic behavior that enables predicable and reliable communication over Ethernet for time-critical applications as well as meeting QoS requirements for other non-real-time applications.

2.2. 5G-TSN Integration Architecture

In order to support TSN protocols smoothly, several function entities are defined in 5G systems in both control and data planes, and the whole 5GS is exposed as a TSN bridge, which is indicated in Figure 2. In this article, we delve into a 5G-TSN joint scheduling algorithm under this 5G TSN bridge architecture.

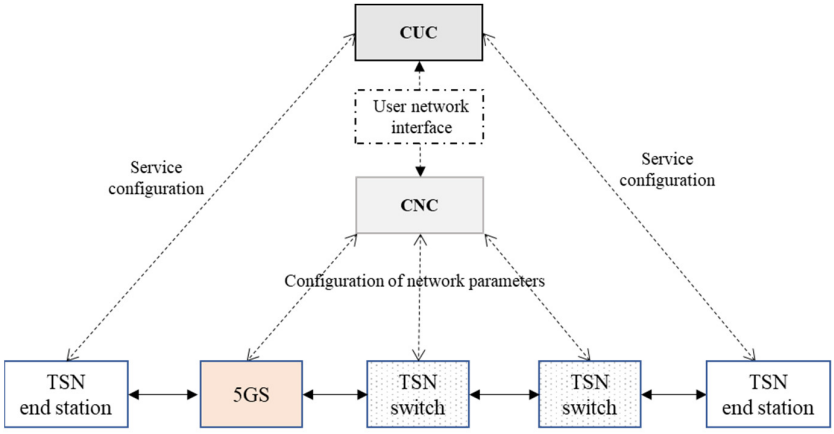

Figure 2. 5G system as a TSN bridge.

Centralized User Configuration (CUC) and Centralized Network Configuration (CNC) are two key elements in the TSN domain for service management and network controlling. CUC is responsible for service requirement collection and TSN capability discovery for end stations, while CNC acts as a network controller, which takes responsibility for network topology discovery, transmission path selection and GCL orchestration.

In order to keep signaling interworking with CNC, a new TSN Application Function (TSN-AF) is proposed in the 5G control plane. TSN-AF follows IEEE 802.1Qcc, reports information of the 5G TSN bridge to the CNC; meanwhile, it receives the service information (period, flow direction, packet, etc.) and QoS requirements of time-triggered applications from CNC. Then TSN-AF delivers QoS-related information received from the TSN domain to the Policy Control Function (PCF) in the 5G core network to make appropriate QoS profiles in 5GSs.

In the data plane, a Network-side TSN Translator (NW-TT) and a Device-side TSN Translator (DS-TT) act as 5G-TSN gateways to execute protocol translation and data forwarding between TSN and 5G domains. NW-TT and DS-TT monitor the time-triggered application status and report to the TSN-AF; meanwhile, they also receive configuration information from the CNC delivered through TSN-AF. In NW-TT and DS-TT, TAS is adopted for flow management and resource scheduling. As a result, data forwarding from NW-TT or DS-TT is based on a precise time, which can eliminate large delay jitter for time-triggered applications.

2.3. Related Research on 5G-TSN

Scheduling time-critical applications in a 5G or TSN network is challenging, and it becomes even more difficult in the complicated scenarios of 5G-TSN integrated networks.

Currently, a few works mainly pay more attention to how to meet latency demands of time-critical applications using uRLLC in 5G wireless networks.

Particularly to satisfy the strict latency requirements of time-critical applications, uRLLC has specified a mini-slot with durations of 0.125~0.25 ms to reduce scheduling delay for those applications. This is different from standard slot level access to radio resources for eMBB applications which has slot durations of 1 ms or more [8]. Reference [9] discusses QoS requirements for uRLLC applications and also proposes various methods to share radio resources among uRLLC and other application types. Reference [10] studies resource allocation schemes to maximize admissible uRLLC loads with associated latency constraints and the influence of retransmission schemes on uRLLC capacity.

Furthermore, there are a few other works addressing how to meet QoS constraints of uRLLC via preemptive puncturing of resources assigned for eMBB flows in multiple-application coexistence scenarios; preemption-based methods allow time-critical applications to interrupt ongoing non-time-critical transmissions, which is appropriate for transmitting urgent frames. Reference [11] studies a linear rate loss model for uRLLC preemptive overlap to maximize the utility of eMBB applications while immediately satisfying latency demands of uRLLC. Reference [12] proposes an RAN slicing method to meet the delay demand of deterministic applications through resource reservation and preemption.

Currently, reinforcement learning has been widely used for wireless resource scheduling problems [13]. In [14], a deep reinforcement learning algorithm is proposed to solve the channel allocation problem of a multi-beam satellite system, which effectively improved the spectrum resource efficiency and system capacity. Furthermore, some studies have paid attention to frequency resource allocation with reinforcement learning. In [15], an algorithm based on DDPG is proposed for 5G frequency allocation, which can guarantee a low bit-error rate and low latency for packet transmission, but this literature only considers a single application type, which makes the optimization problem easier and is not appropriate for actual network operation. In [16], a joint scheduling method based on DDPG is proposed to achieve a long-term QoS tradeoff between eMBB and URLLC services.

However, all the above works only focus on dynamic scheduling schemes in 5GSs to provide low latency and reliable transmission guarantees for time-critical applications in 5G, and never take the influences of TSN scheduling or TSN features into consideration.

Reference [17] introduces 5G-TSN developments in 3GPP standardization and identifies open issues in future research, such as time synchronization, session continuity and scheduling. In [18], a simulation model based on OMNet++ is addressed, including transmission procedures in 5G-TSN and analysis of several simulation results to prove that 5G-TSN can provide bounded latency for time-sensitive applications, but no joint scheduling scheme is proposed. Reference [19] takes both TSN shaping and 5G radio resource scheduling into consideration; the network constraints in the TSN and 5G domains are described, respectively, and the semi-persistent scheduling mechanism is adopted in 5GSs, which means that the 5G resources in each TTI for time-triggered applications are predefined. The problem in this work is formulated as multi-objects optimization to minimize the resource loss rate (the ratio of unused and total resources in 5G and TSN, respectively), subject to the deadline constraints of time-triggered applications and resources constraints in both 5G and TSN, and finally constraint programming is applied to determine feasible scheduling solutions. In [20], the authors present a system-level simulator integrated with 5G and TSN, and then investigate the impacts of GCL setting in TSN (consecutive transmission or uniformly distributed transmission) and radio channel quality of 5G on end-to-end latency of time-triggered applications and system throughput, but this work does not present any dynamic joint scheduling schemes. In [21], the authors propose a 5G-TSN joint scheduling mechanism based on radio channel information; the TSN scheduling considers the impacts of channel quality in 5G, and the flows bearing on radio channels with worse quality could be processed as priority. A transmission time budget in 5GSs is proposed, considering the maximum retransmission times in the correspondence channel quality, and this time budget in 5GSs is set as the constraints for scheduling. However, this work still

does not research dynamic 5G scheduling considering TSN shaping state. Reference [22] investigates how to achieve low-latency communication in a 5G-TSN integration network by considering non-real-time communication services with high-throughput requirements, and a two-step approach is proposed to solve the resource allocation problem and develop priority metrics for the TSN and eMBB streams according to their characteristics, respectively. However, this work also only considers 5G scheduling with latency constraints of time-triggered applications, and does not consider the impacts of TSN scheduling on 5G resource allocations. Reference [23] makes use of the network function virtualization (NFV) technique to implement QoS-aware mapping and scheduling in 5G-TSN networks; it first designs an incremental greedy algorithm to map VNFs to resources of 5G and TSN, and develops a preemption-based 5G resource scheduling scheme to offer no-wait transmission for higher priority applications. This work mainly focuses on the dynamic priority configuration, which aims to improve the resource preemption capability for time-critical applications in both 5G and TSN domains. However, this work does not pay attention to the collaboration of 5G and TSN scheduling schemes, and the 5G scheduling scheme is related in too simple a manner.

The above related research status on 5G dynamical scheduling and 5G-TSN resource allocation indicates that there is limited literature taking both 5G and TSN resource scheduling states into consideration. Therefore, a dynamic 5G-TSN joint scheduling algorithm is studied in this article; the proposed algorithm takes GCL setting status, features of multiple applications, available resources and radio channel quality in 5G into consideration, and the object of the optimization is to satisfy the latency requirements for time-critical application across 5G-TSN as well as to guarantee throughput requirements for other applications in 5GSs. Considering the optimization problem of 5G-TSN joint scheduling is more complex and cannot be solved with the convex optimization method directly; a reinforcement learning model is proposed to learn the optimal resource allocation policy from complex network environments.

3. System Model and Problem Formulation

3.1. System Model

Usually, there are several applications transmitted simultaneously in industrial networks, such as video monitoring, data collections from Programmable Logical Controller (PLC) and control information among PLCs or between PLCs to field devices. In our system model, the industrial network is composed of 5G and TSN, and applications are separated into two categories: time-triggered applications and video applications, which are denoted as $\mathbf{F}_{tt}$ and $\mathbf{F}_{vi}$, respectively. The active applications in the 5G system are denoted as $i = \{1, 2, 3, \ldots, N\}$ ($i \in \mathbf{F}$, $\mathbf{F} \equiv \mathbf{F}_{tt} \cup \mathbf{F}_{vi}$).

Furthermore, 5G-TSN industrial networks should satisfy various QoS requirements for different applications, such as end-to-end maximum latency requirements (EMLR) for both time-triggered and video, and minimum data rate requirements for video. The QoS requirements for various applications are described as follows:

$$d_i < d_i^{qos}, i \in \mathbf{F} \tag{1}$$

$$\overline{R}_j \geq R_j^{\min}, j \in \mathbf{F}_{vi} \tag{2}$$

where d_i is the queueing delay and transmission delay in a 5G system of the ith active flow, and d_i^{qos} is the latency requirement of the corresponding flow. $\overline{R}_j$ denotes the average data rate achieved by the jth video flow, and $R_j^{\min}$ denotes the minimal throughput requirement of the corresponding video flow.

Figure 3 illustrates the procedures of 5G and TSN joint scheduling for downlink data transmission based on an artificial intelligence agent, where the agent is a logical entity in gNB to make dynamic resource scheduling decisions in each Time Transmission

Interval (TTI); the agent can learn from the environment and make the appropriate "action". Detailed procedures are described as follows.

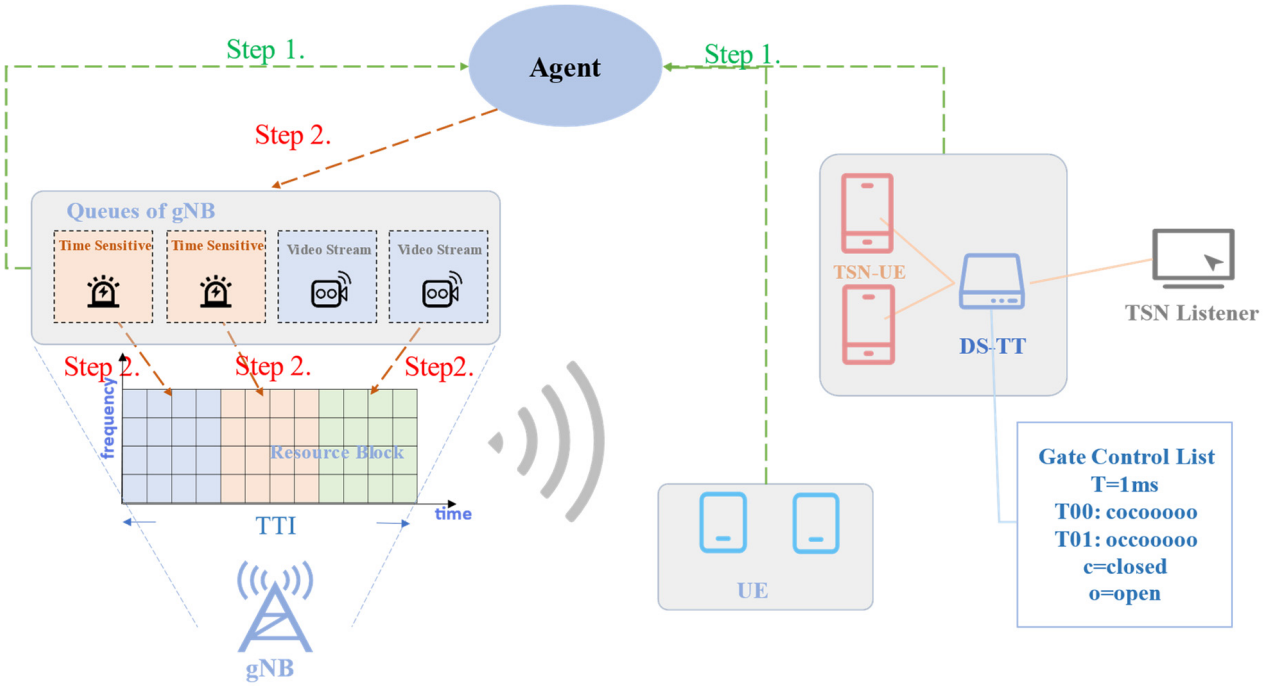

Figure 3. System model of joint scheduling in 5G-TSN architecture.

Step 1: Environment information collection. At the beginning of each TTI, environmental information is collected by an agent. The environment information includes the radio channel quality of each User Equipment (UE), the GCL state of the time-triggered queue in DS-TT, and the queue state of each activate flow in gNB, such as queueing packet number and queueing delay of each flow. The agent is physically located in the gNB, and is responsible for environment information collection, so the queue state in gNB is delivered to the agent with an internal interface, but the channel quality and GCL state should be reported from the UE to the agent using radio signaling.

Step 2: Decision-making and configuration. Packets of all applications are buffered for downlink scheduling in 5G gNB. Based on information collected from UEs, gNB and DS-TT, the agent makes decisions depending on the number of packets waiting for transmission, service priority, obtained average data rates, queueing time and deadline of the flow, and selects which flow can be scheduled and how many resources are allocated for packet delivery. After resource decision-making, the corresponding allocated resources are configured for packet delivery of each eligible flow. To avoid impacting the scheduling in the next TTI, resource decision-making, configuration and packet delivery with allocated resources should be implemented in a TTI.

After the agent allocates the radio resources for the corresponding flow, the environment, including GCL state in DS-TT, queue state in gNB and channel state of the user terminal, will be changed, and the agent will learn all these new states and make the appropriate action in the next TTI, which means that step 1 and step 2 will be executed cyclically in each TTI.

Furthermore, the GCL pre-defined in DS-TT is demonstrated in Figure 3; the packets of time-triggered flows are mapped into queue 7 in DS-TT, and the promised transmission duration for the time-triggered queue is 1 TTI, which means that the ON/OFF state is changed alternately in each TTI.

3.2. Problem Formulation

For a time-triggered application, its QoS requirement is to satisfy end-to-end latency; meanwhile, the QoS requirement of video is to guarantee transmission rate. Therefore, the agent, acting as a scheduler, aims to achieve a QoS tradeoff between time-triggered and video applications by jointly optimizing radio resource allocation. In this article, the purpose of scheduling is to provide a solid latency guarantee for time-triggered applications as well as to improve the throughput for video. In order to achieve this goal, the utility function of flow i at the tth TTI is defined as follows.

$$U_i(t) = \frac{p_i(t)}{d_i^{qos} - d_i^{gnb} + I_{ds-tt}(t) + \Delta} \tag{3}$$

where $p_i(t)$ is the resource allocation indicator of flow i at the tth TTI. $p_i(t) \in \{0, 1\}$, $p_i(t) = 1$ denotes the flow i has been scheduled at this TTI; otherwise, $p_i(t) = 0$. d_i^{qos} and d_i^{gnb} are the EMLR and queueing and processing delay in gNB of flow i, respectively. Moreover, $I_{ds-tt}(t)$ is a latency factor related to the GCL state in DS-TT. $\mathrm{GCL}(t) = 1$ means that the GCL is *OPEN* for time-triggered queues in DS-TT at the tth TTI; otherwise, $\mathrm{GCL}(t) = 0$. As shown in Equation (4), this indicates that $I_{ds-tt}(t)$ only affects the utility value of time-triggered applications. Finally, in order to keep the utility function stable, a small positive constant Δ is introduced.

$$I_{ds-tt}(t) = \begin{cases} 0, \forall i \in \mathbf{F}_{vi} \text{ or } \forall i \in \mathbf{F}_{tt} \text{ and } \mathrm{GCL}(t) = 1 \\ 1, \forall i \in \mathbf{F}_{tt} \text{ and } \mathrm{GCL}(t) = 0 \end{cases} \tag{4}$$

As discussed in Equation (2), the video flow is concerned more with the average throughput, and the average throughput of any video flow j is defined as follows.

$$\overline{R}_j = \frac{\sum_{k=1}^{t} n_j^k \cdot m_j^k}{t \cdot T_{TTI}} \tag{5}$$

This assumes that the scheduling is starting from the first TTI, and the period of each TTI is T_{TTI}. n_j^k represents the allocated Resource Blocks (RBs) for video flow j at the kth TTI, and m_j^k is the bit volume per RB, which depends on the radio channel quality of flow j at the kth TTI.

In each TTI, the available resources are finite, so the resources allocated to all active flows should not be larger than the available resources in gNB, which are described as

$$\sum_{i=1}^{N} p_i(t) \bullet n_i^t \leq \kappa_t, i \in \mathbf{F} \tag{6}$$

where κ_t represents the available RBs at the tth TTI.

As discussed above, the object of 5G-TSN joint scheduling is to ensure end-to-end latency requirements for time-triggered flows as well as to improve the overall throughput of video applications in each TTI. The optimization problem is formulated as follows:

$$\max \sum_{i=1}^{N} U_i(t) \tag{7}$$

subject to Equations (1), (2) and (6).

Based on Equation (3), it can be found that the problem described in Equation (7) is a non-convex function with integer and continuous variables, which is computationally complex. Therefore, a reinforcement learning model is proposed to achieve the optimal resource allocation policy following Equation (7).

4. Markov Decision Process Modelling

Reinforcement learning is a sequential decision process: it takes an action based on the state of the current environment, and then the environment generates an instantaneous reward and moves to a new state with a transition probability. It attempts to achieve an optimal policy, which can maximize the accumulated rewards from the environment [24].

A Markov Decision Process (MDP) provides a mathematical framework for reinforcement learning [25], and a MDP model is usually composed by state space **S**, action space **A**, instantaneous reward γ and transition probability **P**, which are represented mathematically by the tuple (**S**, **A**, **P**, γ). As a result, we should provide an appropriate MDP model for the problem formulated in Equation (7).

4.1. State Space

Usually, the state is an abstraction of the current environment, and the agent acts as an intelligent decision-maker, and makes decisions only according to the state observed from the environment. In this article, the agent makes scheduling and resource allocation decisions based on the state of each application queue in gNB, channel quality of each UE and GCL setting for a time-triggered queue in DS-TT. Therefore, the state space **S** is defined as:

$$\mathbf{S} = \left\{ \mathbf{G}, \mathbf{L}, \mathbf{D}, I_{GCL}^{TT} \right\} \tag{8}$$

Vector **G** denotes the radio channel quality of all activate flows in a 5G system. Vectors **L** and **D** indicate the queueing packet number and queueing delay of each flow in 5G gNB, respectively. Meanwhile, variable $I_{GCL}^{TT} \in \{0, 1\}$ denotes the GCL state in DS-TT for a time-triggered application, $I_{GCL}^{TT} = 1$ means the GCL state of a time-triggered queue in DS-TT is "OPEN" and $I_{GCL}^{TS} = 0$ means "CLOSED" correspondingly.

In order to facilitate the training process of neural networks, some simplification and normalization works are made for the elements in state space. Generally, the Signal-to-Interference-and-Noise Ratio (SINR) denotes the radio channel quality. However, the value of the SINR is consecutive and not suitable for training; it is preferable to choose the discrete variables with a finite value as the environment state. Therefore, we translate the SINR to the required RB numbers of flow i in each TTI. As shown in Equation (9), the maximum bits that can be transmitted in one RB is calculated [10].

$$m_i = B \ \log_2(1 + g_i) - Q^{-1}(\varepsilon)\sqrt{V(g_i)} \tag{9}$$

where ε is the tolerable bit error rate, $Q^{-1}(\varepsilon) = \frac{1}{\sqrt{2\pi}} \int_{\varepsilon}^{\infty} e^{-\frac{t^2}{2}} dt$, $V(g_i) = (\log_2(e))^2 \left(1 - \frac{1}{(1+g_i)^2}\right)$, and B is the frequency bandwidth of one RB. According to Equation (9), the required RB numbers of flow i can be obtained from Equation (10).

$$r_i^{expt} = \frac{s_i \times l_i}{m_i} \forall i \in \mathbf{F}, l_i \in \mathbf{L} \tag{10}$$

where r_i^{expt} indicates the required RB numbers of flow i; s_i and l_i are the packet size and packet number in the queue of flow i, respectively. Then, the radio channel state **G** can be replaced by the required RB numbers of all flows denoted as $\mathbf{R}^{expt}$. Furthermore, in order to reflect the relationship between queueing delay in gNB and EMLR, $d_i \in \mathbf{D}$ is normalized as follows,

$$d_i^{nor} = \frac{d_i^{gnb}}{d_i^{qos}} \tag{11}$$

Finally, the state space can be expressed as

$$\mathbf{S} = \left\{ \mathbf{R}^{expt}, \mathbf{L}, \mathbf{D}^{nor}, I_{GCL}^{TT} \right\} \tag{12}$$

4.2. Action Space

If the output action of the agent is the required RB numbers of each flow, it may lead to a high-dimension action space, which is computationally complex and not conducive to model training. Therefore, the action space is limited to binary, where "1" means the corresponding flow is scheduled in the current TTI, otherwise the output action is set to "0". Then the reinforcement learning model only needs to output whether to schedule the corresponding flow. As a result, the action space is defined as follows.

$$\mathbf{A} = \{p_i\}_{N*1} \quad p_i \in \{0,1\}, i \in \mathbf{F} \tag{13}$$

However, the output action does not consider the available RB amount in the tth TTI; the actual allocated resources for each flow are recalculated with the following formula.

$$r_i^{act} = \left\lceil \frac{p_i \cdot r_i^{expt}}{\sum_{i=1}^{N} p_i \cdot r_i^{expt}} * \kappa_t \right\rceil \tag{14}$$

4.3. Reward Function Design

For reinforcement learning, the agent observes the current environment and selects an action, and then the environment moves to the next state and the agent receives a reward value simultaneously. The reward is very important, because the reward is the quantitative description of the optimization goal for the reinforcement learning mode, which can guide the agent to learn a better policy.

The optimization goal of the proposed scheduling algorithm in 5G-TSN architecture is to provide a latency guarantee for time-triggered flows as well as to improve throughput of video flows. Hence, the reward function is designed as follows.

$$\psi_t(\gamma_i) = \begin{cases} (1+\lambda_1) * \gamma_i & i \in \mathbf{F}_{tt} \text{ and } \mathrm{GCL}(t) = 1 \\ (1-\lambda_2) * \gamma_i & i \in \mathbf{F}_{tt} \text{ and } \mathrm{GCL}(t) = 0 \\ \gamma_i & i \in \mathbf{F}_{vi} \end{cases} \tag{15}$$

$$\gamma_i = \begin{cases} \frac{1}{d_i^{qos} - d_i^{gnb} + \Delta} & \overline{R_i} \geq R_i^{\min},\ i \in \mathbf{F}_{vi} \ \text{ or } i \in \mathbf{F}_{tt} \\ 0 & \overline{R_i} < R_i^{\min}, i \in \mathbf{F}_{vi} \end{cases} \tag{16}$$

In Equation (15), $\psi_t(\gamma_i)$ is the reward function, which adopts the formulation of the potential function; it takes the GCL state at the tth TTI into consideration for time-triggered applications. As shown in Figure 4, in order to reflect the relationship between queueing delay of time-triggered flow in gNB and reward, λ is introduced as a self-adaption factor to adjust the reward. First of all, the rewards are different for the gated open and closed states. Therefore, λ_1 and λ_2 are introduced to adjust the reward value in different states. In addition, when the d_i^{gnb} of the time-triggered flow at the base station is close to d_i^{qos}, the reward value of scheduling the flow will be larger. Therefore, λ_1 will increase with an increase in d_i^{gnb}, and λ_2 is the opposite. The purpose of the reward function setting has two aspects: the first is encouraging time-triggered applications to be scheduled near the latency deadline, which leaves more opportunities for video flow transmission. The second is trying to schedule time-triggered flows when the GCL state of the time-triggered queue in DS-TT is "ON" as much as possible.

Equation (16) defines the basic reward expression for both time-triggered and video applications. We can see that when the queueing delay is closer to the EMLR, more reward value may be achieved. Furthermore, the throughput constraint of video flows is also considered in Equation (16); if the throughput of video flow cannot satisfy the minimum threshold, the reward value is set to zero, which aims to train the agent to

avoid making scheduling decisions leading to exceeding latency requirements during the training procedure.

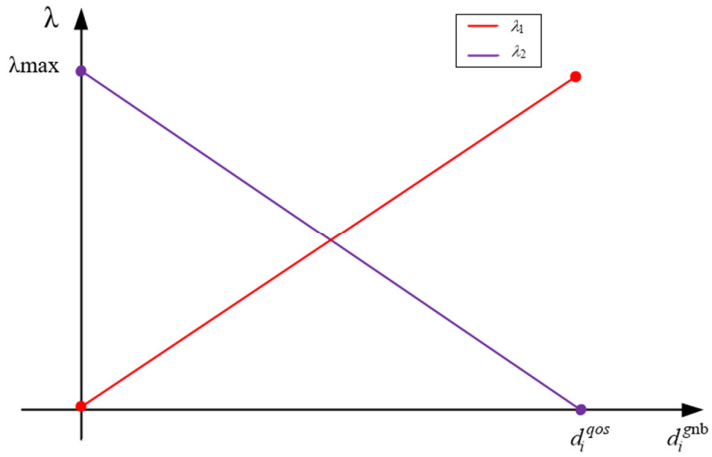

Figure 4. Relationship between λ and queueing delay in gNB.

5. DDPG-Based Resource Scheduling Algorithm

Based on the above elaboration, we have designed an MDP model with the appropriate state space, action space and reward function according to the optimization problem described in Equation (17) with constraints in Equations (1), (2) and (6). Utilizing the reinforcement learning model for resource scheduling, the best policy achieved from the training process can maximize the expected reward value, which is defined as follows.

$$\boldsymbol{\mathcal{A}}_t^* = \underset{\boldsymbol{\mathcal{A}}_t}{\operatorname{argmax}} \sum_{k=t+1}^{L} \sum_{i=1}^{N} \psi_k(\gamma_i) \tag{17}$$

In DQN, the general output consists of multiple dimensions of Q-values for each action. For example, when scheduling users, it outputs Q-values for both scheduling and not scheduling each user. However, DDPG combines the advantages of both policy-based and value-based methods. The actor network in DDPG learns a deterministic policy that directly maps states to actions, allowing for fine-grained control and explicit action outputs. In the case of scheduling users, it can directly output the actions for each individual user. Therefore, in this section, a DDPG-based resource scheduling algorithm is proposed to achieve the optimal action from the above formula, which is more suitable for multi-user tasks.

DDPG includes two neural networks: the actor network is denoted as $\pi(\boldsymbol{S}; \boldsymbol{\Theta})$, which takes the state $\boldsymbol{S}$ as input and outputs actions $\boldsymbol{A}$. The critic network is denoted as $Q(\boldsymbol{S}, \boldsymbol{\mathcal{A}}; \mathbf{W})$, and it takes the state $\boldsymbol{S}$ and actions $\boldsymbol{\mathcal{A}}$ as inputs and outputs the evaluation Q-value of actions $\boldsymbol{\mathcal{A}}$. Besides this, $\boldsymbol{\Theta}$ and $\mathbf{W}$ indicate the parameters of the actor and critic networks, respectively.

The training process of DDPG is to make the actor achieve an action maximizing the approximate Q-value given by the critic network, as well as to make the critic evaluate an approximate Q-value for the actions learnt from the actor closer to the true Q-value calculated by the Bellman Equation in Equation (18).

$$Q_t^{true} = \psi_t(\gamma_i) + \mu Q_{t+1} \tag{18}$$

where μ is the discount factor.

Once the number of transition sets $(\boldsymbol{S}_t, \boldsymbol{\mathcal{A}}_t, \psi_t(\gamma_i), \boldsymbol{S}_{t+1})$ meets the training requirements, the parameters of the actor and critic networks are updated according to Equations (19)–(23).

The critic calculates Q_t of action $\boldsymbol{\mathcal{A}}_t$ in state $\boldsymbol{S}_t$:

$$Q_t = Q(\boldsymbol{S}_t, \boldsymbol{\mathcal{A}}_t; \mathbf{W}) \tag{19}$$

The critic ranks the Q_{t+1} of the state $\boldsymbol{S}_{t+1}$ and action $\boldsymbol{\mathcal{A}}_{t+1}$:

$$Q_{t+1} = Q(\boldsymbol{S}_{t+1}, \boldsymbol{\mathcal{A}}_{t+1}; \mathbf{W}), where\ \boldsymbol{\mathcal{A}}_{t+1} = \pi(\boldsymbol{S}_{t+1}; \boldsymbol{\Theta}) \tag{20}$$

Calculation loss is defined as the difference between true Q-value and predicted Q-value by the critic with Equation (20):

$$\delta_t = Q_t - Q_t^{true} \tag{21}$$

Then the critic parameters are updated using gradient descent:

$$\mathbf{W} = \mathbf{W} - \alpha * \delta_t * \frac{\partial Q(\boldsymbol{S}_t, \boldsymbol{\mathcal{A}}_t; \mathbf{W})}{\partial \mathbf{W}} \tag{22}$$

The goal of the actor network is to achieve an action-maximizing approximate Q_t given by the critic network. Since $\boldsymbol{\mathcal{A}}_t = \pi(\boldsymbol{S}_t; \boldsymbol{\Theta})$, Equation (23) is obtained to calculate the gradient of parameter $\boldsymbol{\Theta}$ by the chain rule,

$$\nabla = \frac{\partial Q(\boldsymbol{S}_t, \boldsymbol{\mathcal{A}}_t; \mathbf{W})}{\partial \boldsymbol{\Theta}} = \frac{\partial \pi(\boldsymbol{S}_t; \boldsymbol{\Theta})}{\partial \boldsymbol{\Theta}} * \frac{\partial Q(\boldsymbol{S}_t, \boldsymbol{\mathcal{A}}_t; \mathbf{W})}{\partial \boldsymbol{\mathcal{A}}_t} \tag{23}$$

Then the actor parameters are updated with gradient ascent:

$$\boldsymbol{\Theta} = \boldsymbol{\Theta} + \beta * \nabla \tag{24}$$

The specific process of training the resource allocation model based on DDPG is shown in Algorithm 1.

Algorithm 1: Resource scheduling based on DDPG

Initialize the channel model parameters and DDPG model parameters
For episode = 1, 2, ... E do
  Initialize the packet model parameters
  For t = 1, 2, ... Step do
    Given the current state $\boldsymbol{S}_t$.
    Get action $\boldsymbol{\mathcal{A}}_t$ from $\pi(\boldsymbol{S}_t; \boldsymbol{\Theta})$.//*determine which flows are scheduled in this TTI*
    Calculate the numbers of allocated RB for each scheduled flow with Equation (14).
    Execute the $\boldsymbol{\mathcal{A}}_t$ and allocate RB numbers got in step 7 and get reward $\psi(\gamma_i)$ with Equation (15).
    Transit to the next state $\boldsymbol{S}_{t+1}$.//*After an action execution, the environment state will be updated*
    Store $(\boldsymbol{S}_t, \boldsymbol{\mathcal{A}}_t, \psi(\gamma_i), \boldsymbol{S}_{t+1})$.//this message is stored for DDPG model training
    Trainset num+ = 1.//calculate the size of train set
    if train set num > min trainset num do//When the size of the training set meets the training conditions, it begins to train the model.
      $Q_t = Q(\boldsymbol{S}_t, \boldsymbol{\mathcal{A}}_t; \mathbf{W})$//calculate Q_t of action $\boldsymbol{\mathcal{A}}_t$ in state $\boldsymbol{S}_t$
      $Q_{t+1} = Q(\boldsymbol{S}_{t+1}, \boldsymbol{\mathcal{A}}_{t+1}; \mathbf{W})$//Output the Q_{t+1} with the state $\boldsymbol{S}_{t+1}$ and action $\boldsymbol{\mathcal{A}}_{t+1}$.
      $\delta_t = Q_t - (r_t + \psi(\gamma_i) * Q_{t+1})$//Get the calculation loss
      $\mathbf{W} = \mathbf{W} - \alpha * \delta_t * \frac{\partial Q(\boldsymbol{S}_t, \boldsymbol{\mathcal{A}}_t; \mathbf{W})}{\partial \mathbf{W}}$//update the critic parameters using gradient descent
      $\nabla = \frac{\partial \pi(\boldsymbol{S}_t; \boldsymbol{\Theta})}{\partial \boldsymbol{\Theta}} * \frac{\partial Q(\boldsymbol{S}_t, \boldsymbol{\mathcal{A}}_t; \mathbf{W})}{\partial \boldsymbol{\mathcal{A}}_t}$
      $\boldsymbol{\Theta} = \boldsymbol{\Theta} + \beta * \nabla$//update the actor parameters with gradient ascent
    do end
  For end
For end

6. Results

6.1. Simulation Parameters

In this section, several parameters are list in Table 1 for simulation. As for the flow model, it is assumed that the packet burst obeys a 0–1 distribution, which represents whether the new packets are produced within each TTI. According to 3GPP TS 22.104, time-sensitive applications always have small packets, but video streams always generate large packets, so the packet size of time-triggered flows and video streams are set to 200 Bytes and 800 Bytes, respectively. The EMLR of time-triggered applications and the deadline of video flow are 6 ms and 20 ms, respectively. The 5G radio channel model follows a Rayleigh distribution with a standard deviation of 1.0, and the distance between gNB and mobile terminal should increase or decrease according to a 0–1 distribution [26]. In addition, the transmission power of the base station is 20 dbm, and the noise power of the channel is -90 dbm. κ_t indicates total resource bands. The subcarrier interval used in this paper is 15 kHz, indicating that the scheduling cycle interval is 1 ms. Δ is a small positive constant in Equation (3). $R_j^{\min}$ indicates the minimal throughput requirement of the corresponding video flow in Equation (4).

Table 1. Simulation parameters.

The probability of packets arriving within each TTI	0.5
The number of packets arriving within each TTI	The number is valued from 1–3, followed by a uniform distribution
Time-triggered application packet size	200 Bytes
Video flow packet size	800 Bytes
gNB transmission power	20 dBm
Channel noise power	-90 dBm
κ_t	100
Subcarrier interval	15 kHz
Δ	0.1
EMLR of time-triggered application (d_i^{qos})	6 ms
Deadline of video flow	20 ms
$\lambda_{\max}$	0.2
β	0.01
α	0.01
$R_j^{\min}, j \in \mathbf{F}_{vi}$	80,000 Bytes per second
Discount factor μ	0.9
min trainset num	800

In order to improve the learning rate and convergence rate, the structure of the neural network is designed as in Table 2. Both actor and critic have two hidden layers, and each hidden layer consists of a fully connected layer, a RELU activation function and a normalization layer (LN). LN is used to address the problem that the distribution of output data changes with the updating of neuron parameters [27]. Moreover, α and β indicate the learning rates for actor and critic, respectively. μ is the discount factor in Equation (18).

6.2. Results Analysis and Discussion

The convergence speed of the proposed model and the final stable reward value are very important for the reinforcement learning model. In order to accelerate the model convergence process and increase the reward value, several DDPG model related parameters are fine-tuned, including learning rate, model structure and other hyperparameters.

Table 2. Network parameters.

Layers	Actor Network	Critic Network
Input layer	Fully connected layer (40 × 400)	
	Rectified Linear Unit (Relu) nonlinear activation function	
	Normalization layer	
Hidden layer	Fully connected layer (400 × 400)	
	Relu nonlinear activation function	
	Normalization layer	
Output layer	Fully connected layer (400,10)	Fully connected layer (400,1)
	Tanh nonlinear activation function	——

As shown in Figure 5, we optimize model training process by adjusting the learning rate, using layer normalization and reshaping the reward function with Equation (15).

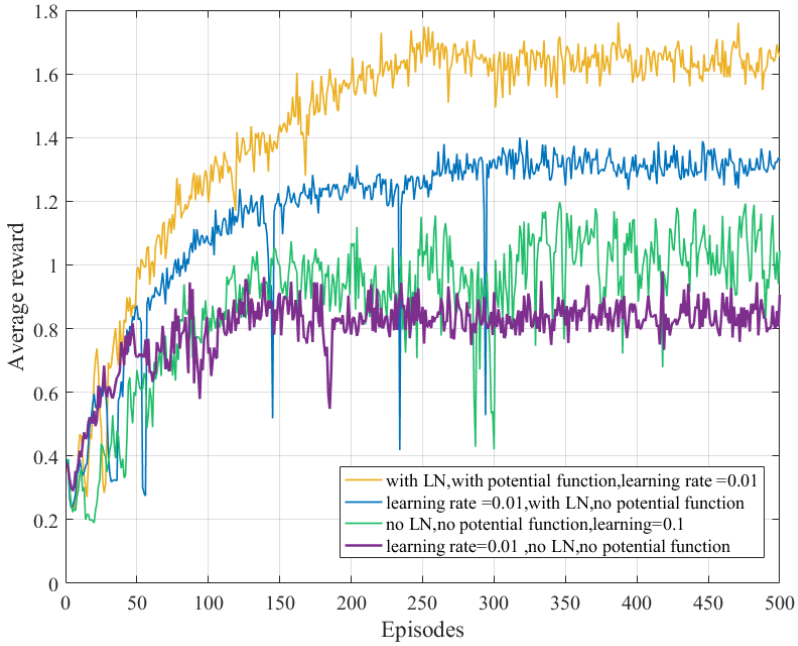

Figure 5. Average reward value vs. training process of DDPG.

As shown in Figure 5, the DDPG model without optimization has serious oscillation during training, because the learning rate and the step size are too large. When the learning rate is reduced from 0.1 to 0.01, it alleviates the problem of model oscillation. In addition, we can see that the reward value appears to decrease during the model training process when there is no layer normalization, because the distribution of input data may change continuously due to parameters updating, which leads to less optimal model training. Therefore, layer normalization is introduced for model training, which makes the reward value of the DDPT model increase markedly. Finally, the potential function is adopted for reward function, which makes the convergence process faster and the final obtained stable reward value larger.

In order to compare 5G system performance under various scheduling algorithms, Proportional Fairness (PF) and Earliest Deadline First (EDF) are used for resource allocation in 5G-TSN cooperation architecture. PF and EDF are commonly used scheduling algorithms in 4G and 5G. PF ensures that all applications have the opportunity to be scheduled. To some extent, PF can provide a throughput guarantee for the required flows. EDF is a classical scheduling for time-sensitive applications. For the algorithm

proposed in this article, the objective is to provide latency guarantee for time-triggered flows as well as to provide throughput guarantee for video flows, so we take these two schemes as a benchmark to compare the latency and throughput performance with the proposed algorithm.

As shown in Figures 6 and 7, DDPG with a potential function reward is compared with DDPG without a potential function reward, EDF and PF. For DDPG with a potential function reward and without a potential function reward, we use the bootstrap method to estimate the delay and throughput of 95% confidence interval. The specific practice is to generate multiple samples from the forecast result data with resamples, and then calculate the 95% confidence interval indicators based on these resamples.

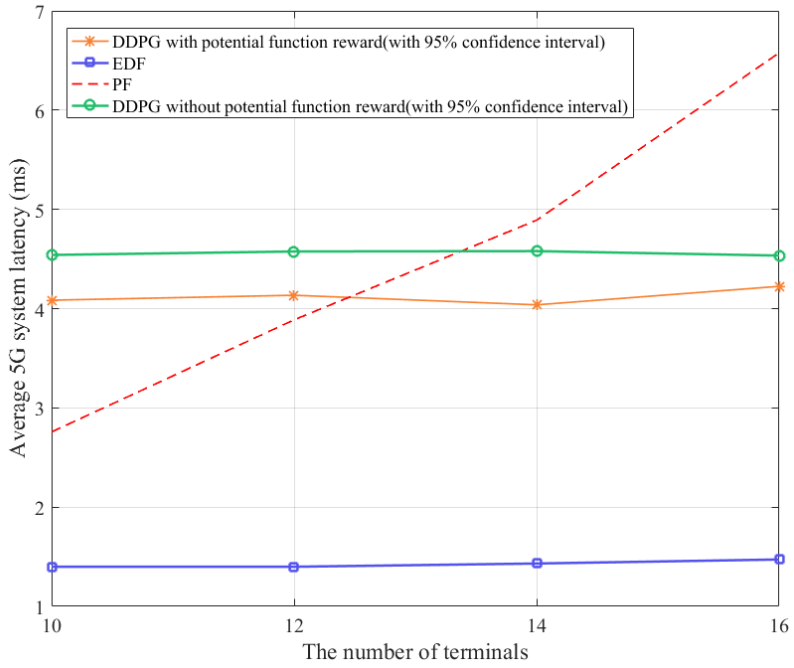

Figure 6. The 5G system latency comparison for time-triggered applications.

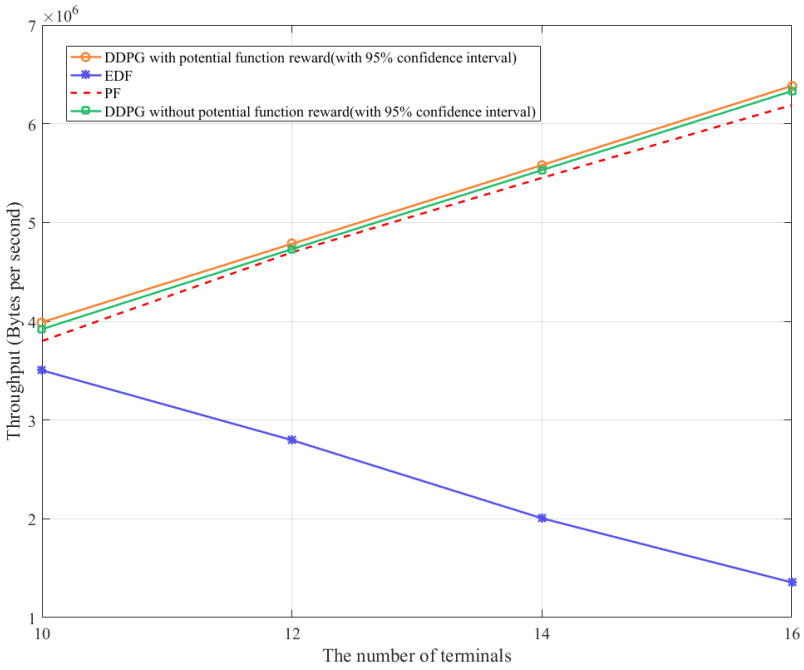

Figure 7. Throughput comparison for video applications.

As indicated in Figure 6, EDF achieves the best performance for time-triggered applications, because the EMLR of video is quite a lot more than that of time-triggered applications; then, the latency of time-triggered applications is closer to the deadline, so time-triggered applications can achieve the highest priority for scheduling. However, the PF algorithm is concerned more with the achieved data rate, but does not pay more attention to latency guarantee. Hence, the 5G system latency with PF increases with an increase in terminal number ($|\mathbf{F}_{tt}| : |\mathbf{F}_{vi}| = 1 : 1$), and the 5G system latency even exceeds the EMLR 6ms when the terminal number reaches 16. The time-triggered application latency of DDPG without potential function reward can remain stable at around 4.5 ms, because it can achieve a larger reward value when the queueing delay is 4.5 ms. However, the time-triggered application latency of DDPG with potential function reward almost remains stable at around 4 ms, because the potential function not only considers the queueing time as an impact factor on the reward value, but also takes the gating state of the time-triggered queue in DS-TT into account.

Unlike the time-triggered application, video is more concerned with the throughput. Figure 7 shows system throughput performance under the multi-application coexisting scenario. EDF prefers to schedule time-triggered applications, which means fewer available resources for video flow. Therefore, the throughput with EDF declines with an increase in terminal number. However, for DDPG without potential function reward, DDPG with potential function reward and PF-based scheduling algorithms, the throughput rises with an increase in terminal number. Because of the designed reward function, the DDPG-based scheduling algorithm provides more schedule opportunities for video flows with relaxed latency requirements for time-triggered applications, so the DDPG potential function reward achieves the best performance and improves by nearly 0.05% throughput compared with PF and by 0.01% throughput compared with DDPG without potential function reward. Generally speaking, the proposed algorithm is more versatile and can achieve better balance between various QoS guarantees in multi-application scenarios compared to EDF and PF.

In addition, in order to fully consider the impact of different GCL setting schemes in DS-TT on 5G scheduling, three different GCL setting schemes in DS-TT are evaluated. The first GCL setting scheme for time-triggered queues remains "ON" in one TTI and changes to "CLOSE" in next TTI; the time during the "ON" state for the time-triggered queue is one TTI, and the first GCL setting scheme is represented as "0101". As for the second GCL setting scheme, the GCL state of the time-triggered queue changes every two TTIs, which is represented as "0011". Furthermore, in the third GCL setting scheme, the GCL state of "ON" for the time-triggered queue is selected with probability 0.5, which is represented as "random".

Figures 8 and 9 show the latency and throughput performance of time-triggered and video flows within three GCL setting schemes in DS-TT, respectively. The bootstrap method is used to estimate the delay and throughput of the 95% confidence interval. As shown in Figure 8, it indicates that various GCL setting schemes can achieve nearly similar throughput performance, because the transmission opportunities of video in all three GCL setting schemes are the same; meanwhile, the proposed DDPG algorithm can guarantee throughput for video applications within various GCL setting schemes. However, the different GCL setting schemes have various impacts on 5G latency. As indicated in Figure 9, we can see that the first GCL setting scheme achieves the best latency performance of time-triggered applications in various scenarios, because the GCL state is changed every TTI interval, so the time-triggered applications can be scheduled in a timely manner, which leads to less queueing delay in gNB.

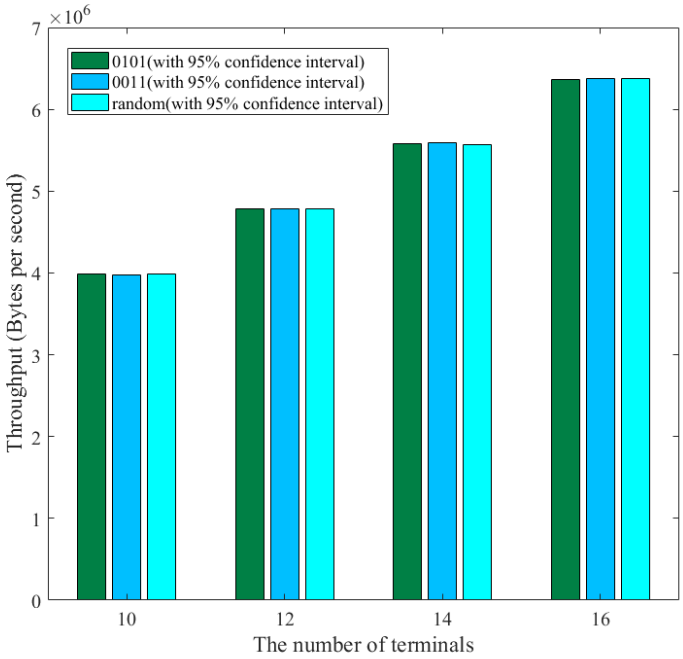

Figure 8. Throughput comparison in three GCL settings in DS-TT.

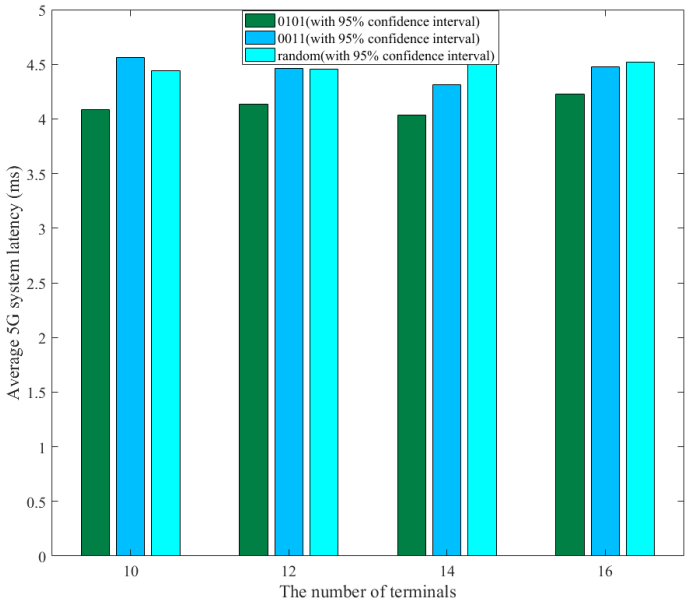

Figure 9. Latency comparison in three GCL settings in DS-TT.

7. Conclusions and Future Works

In this article, a joint resource scheduling problem within 5G-TSN collaboration architecture in a multi-application scenario is formulated and discussed. In order to solve the formulated problem efficiently and intelligently, a DDPG-based joint scheduling algorithm is proposed, which aims to guarantee both latency requirements for time-triggered applications and throughput requirements for video applications. The proposed reinforcement learning model combines latency requirement, throughput requirement and GCL state in DS-TT to design a reward function, which can encourage the agent to learn an optimized action during model training. Finally, the simulation results indicate that the proposed reinforcement learning model is convergent and efficient. Meanwhile, compared with EDF and PF scheduling algorithms at three different gate settings, the proposed DDPG-based algorithm can achieve a stable latency guarantee for time-triggered applications, as well

as improving the overall throughput for video. Furthermore, several different GCL setting methods are proposed to verify the latency and performance within the proposed DDPG-based joint scheduling algorithm; the more discrete GCL settings are more useful for time-triggered applications. Generally speaking, the proposed DDPG-based joint scheduling algorithm can provide solid QoS guarantee for multiple applications and achieve better multi-application carrying capability in 5G-TSN collaboration architecture.

However, several key aspects are still not considered in this article. Firstly, the scheduling problems of single base stations are mainly considered in this article, but in reality there are more scenarios of multiple base stations; secondly, we only consider that the radio channel quality affects the bits carried in one RB, but ignore the fact that a poor radio channel also leads to packet loss, so no retransmission procedure is considered in this algorithm. Finally, the proposed algorithm can only improve the throughput of video applications, but cannot maximize the throughput of the whole system.

Therefore, in order to solve those problems, there are several aspects to be investigated in the future. First, a 5G-TSN scheduling algorithm based on the multi-agent reinforcement learning model needs to be studied for multiple base station scenarios; meanwhile, the action space should be designed to be more complex. Second, the impact of a retransmission scheme on resource allocation should be considered. Third, for current 5G-TSN joint scheduling, we mainly investigate 5G scheduling considering TSN states, but in the future, how 5G scheduling affects GCL settings in the TSN domain should be studied.

Author Contributions: Conceptualization, Y.Z. and L.S.; methodology, L.S., Y.Z. and J.W.; software, Y.Z.; validation, Y.Z. and L.S.; formal analysis, L.S., Y.Z. and J.W.; investigation, Y.Z., L.S. and R.H.; resources, Y.Z. and L.S.; data curation, Y.Z.; writing—original draft preparation, Y.Z. and L.S.; writing—review and editing L.S., J.W. and X.J.; visualization, Y.Z.; supervision, J.W.; project administration, J.W. and L.S.; funding acquisition, J.W. and L.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Key Research and Development Program of China with Grant No. 2020YFB1708800 from Ministry of Science and Technology and in part by the Guangdong Provincial Key R&D Program with Grant No. 2020B0101130007 from the Administration of Science and Technology, Guangdong Province, China.

Data Availability Statement: This paper studies the scheduling algorithm based on reinforcement learning. There is no data set directly available for this kind of scheduling algorithm, and its model training is based on various formulas (such as channel state generated by channel model, etc.) and intermediate data generated by numerical simulation.

Acknowledgments: The authors would like to acknowledge the support from editors and comments from all the reviewers.

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

References

1. Bennis, M.; Debbah, M.; Poor, H.V. Ultrareliable and Low-Latency Wireless Communication: Tail, Risk, and Scale. *Proc. IEEE* **2018**, *106*, 1834–1853. [CrossRef]
2. Bello, L.L.; Steiner, W. A perspective on IEEE time-sensitive networking for industrial communication and automation systems. *Proc. IEEE* **2019**, *107*, 1094–1120. [CrossRef]
3. Lv, J.; Zhao, Y.; Wu, X.; Li, Y.; Wang, Q. Formal Analysis of TSN Scheduler for Real-Time Communications. *IEEE Trans. Reliab.* **2021**, *70*, 1286–1294. [CrossRef]
4. 3GPP Technical Specification TS 23.501 on System Architecture for 5G System. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3144 (accessed on 23 February 2023).
5. Atiq, M.K.; Muzaffar, R.; Seijo, O.; Val, I.N.; Bernhard, H.-P. When ieee 802.11 and 5g meet time-sensitive networking. *IEEE Open J. Ind. Electron. Soc.* **2022**, *3*, 14–36. [CrossRef]
6. Messenger, J.L. Time-sensitive networking: An introduction. *IEEE Commun. Stand. Mag.* **2018**, *2*, 29–33. [CrossRef]
7. Nasrallah, A.; Thyagaturu, A.S.; Alharbi, Z.; Wang, C.; Shao, X.; Reisslein, M.; ElBakoury, H. Ultra-low latency (ull) networks: The ieee tsn and ietf detnet standards and related 5g ull research. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 88–145. [CrossRef]
8. Marijanović, L.; Schwarz, S.; Rupp, M. Multiplexing Services in 5G and Beyond: Optimal Resource Allocation Based on Mixed Numerology and Mini-Slots. *IEEE Access* **2020**, *8*, 209537–209555. [CrossRef]

9. Ji, H.; Park, S.; Yeo, J.; Kim, Y.; Lee, J.; Shim, B. Introduction to Ultra Reliable and Low Latency Communications in 5G. *arXiv* **2017**, arXiv:1704.05565.
10. Anand, A.; de Veciana, G. Resource Allocation and HARQ Optimization for URLLC Traffic in 5G Wireless Networks. *IEEE J. Sel. Areas Commun.* **2018**, *36*, 2411–2421. [CrossRef]
11. Anand, A.; de Veciana, G.; Shakkottai, S. Joint Scheduling of URLLC and eMBB Traffic in 5G Wireless Networks. *IEEE/ACM Trans. Netw.* **2020**, *28*, 477–490. [CrossRef]
12. Ginthör, D.; Guillaume, R.; Schüngel, M.; Schotten, H.D. 5G RAN Slicing for Deterministic Traffic. In Proceedings of the 2021 IEEE Wireless Communications and Networking Conference (WCNC), Nanjing, China, 29 March–1 April 2021.
13. Alwarafy, A.; Abdallah, M.; Ciftler, B.S.; Al-Fuqaha, A.; Hamdi, M. Deep Reinforcement Learning for Radio Resource Allocation and Management in next Generation Heterogeneous Wireless Networks: A Survey. *arXiv* **2021**, arXiv:2106.00574.
14. Hu, X.; Liu, S.; Chen, R.; Wang, W.; Wang, C. A Deep Reinforcement Learning-Based Framework for Dynamic Resource Allocation in Multibeam Satellite Systems. *IEEE Commun. Lett.* **2018**, *22*, 1612–1615. [CrossRef]
15. Gu, Z.; She, C.; Hardjawana, W.; Lumb, S.; McKechnie, D.; Essery, T.; Vucetic, B. Knowledge-Assisted Deep Reinforcement Learning in 5G Scheduler Design: From Theoretical Framework to Implementation. *IEEE J. Sel. Areas Commun.* **2021**, *39*, 2014–2028. [CrossRef]
16. Li, J.; Zhang, X. Deep Reinforcement Learning-Based Joint Scheduling of eMBB and URLLC in 5G Networks. *IEEE Wirel. Commun. Lett.* **2020**, *9*, 1543–1546. [CrossRef]
17. Striffler, T.; Michailow, N.; Bahr, M. Time-Sensitive Networking in 5th Generation Cellular Networks—Current State and Open Topics. In Proceedings of the 2019 IEEE 2nd 5G World Forum (5GWF), Dresden, Germany, 30 September–2 October 2019.
18. Martenvormfelde, L.; Neumann, A.; Wisniewski, L.; Jasperneite, J. A Simulation Model for Integrating 5G into Time Sensitive Networking as a Transparent Bridge. In Proceedings of the 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Vienna, Austria, 8–11 September 2020.
19. Ginthör, D.; Guillaume, R.; von Hoyningen-Huene, J.; Schüngel, M.; Schotten, H.D. End-to-end Optimized Joint Scheduling of Converged Wireless and Wired Time-Sensitive Networks. In Proceedings of the 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Vienna, Austria, 8–11 September 2020.
20. Ginthör, D.; von Hoyningen-Huene, J.; Guillaume, R.; Schotten, H. Analysis of Multi-user Scheduling in a TSN-enabled 5G System for Industrial Applications. In Proceedings of the 2019 IEEE International Conference on Industrial Internet (ICII), Orlando, FL, USA, 11–12 November 2019.
21. Sun, L.; Wang, J.; Lin, S.; Ma, Z.; Li, W.; Qilian, L.; Huang, R. Research on 5G-TSN joint scheduling mechanism based on radio channel information. *J. Commun.* **2021**, *42*, 65–75.
22. Yang, J.; Yu, G. Traffic Scheduling for 5G-TSN Integrated Systems. In Proceedings of the 2022 International Symposium on Wireless Communication Systems (ISWCS), Hangzhou, China, 19–22 October 2022.
23. Zhang, Y.; Xu, Q.; Li, M.; Chen, C.; Guan, X. QoS-Aware Mapping and Scheduling for Virtual Network Functions in Industrial 5G-TSN Network. In Proceedings of the 2021 IEEE Global Communications Conference (GLOBECOM), Madrid, Spain, 7–11 December 2021.
24. Luong, N.C.; Hoang, D.T.; Gong, S.; Niyato, D.; Wang, P.; Liang, Y.C.; Kim, D.I. Applications of Deep Reinforcement Learning in Communications and Networking: A Survey. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 3133–3174. [CrossRef]
25. Alwarafy, A.; Abdallah, M.; Ciftler, B.S.; Al-Fuqaha, A.; Hamdi, M. The Frontiers of Deep Reinforcement Learning for Resource Management in Future Wireless HetNets: Techniques, Challenges, and Research Directions. *IEEE Open J. Commun. Soc.* **2022**, *3*, 322–365. [CrossRef]
26. Sklar, B. Rayleigh fading channels in mobile digital communication systems. I. Characterization. *IEEE Commun. Mag.* **1997**, *35*, 90–100. [CrossRef]
27. Ba, J.L.; Kiros, J.R.; Hinton, G.E. Layer Normalization. *arXiv* **2016**, arXiv:1607.06450.