Computer Life (CPL) ISSN: 1819-4818
DELIVERING QUALITY SCIENCE TO THE WORLD

Research on Scheduling Algorithm Based on Guaranteeing Service Time

Wenfang Hou [1, a], Hui Wang [1, b, *] and Bo Zeng [1, c]

[1]School of Information Engineering, Henan University of Science and Technology, Henan, China

[a]408559635@qq.com, [b]wh@haust.edu.cn, [c]wbzeng_hn@163.com

---

Abstract: In order to solve the problem of traffic transmission timeout and transmission failure caused by TCP Incast problem in data center network (DCN), This paper presents a scheduling algorithm BST-TCP (Business Service Time TCP) based on guaranteed service time, In order to ensure the service time of the business, Firstly, the queue model of the sink node in the data center network is modeled, After that, a scheduling algorithm based on ensuring the service time of the service is proposed. By introducing an access mechanism, the data packets that cannot be transmitted within the time limit are discarded, and a hierarchical queue of buffers is set up at the convergence switch to ensure that the services with higher urgency can be transmitted first. Compared with the existing protocols, BST-TCP protocol can better ensure that the flow is completed within the service time and improve the network performance.
Keywords: Data Center Network, Service Time, Service Quality Assurance, Hierarchical Queue.

---

1. INTRODUCTION

In recent years, with the wide application of cloud computing technology and the development of emerging application modes such as virtualization technology, the data center, as an information infrastructure, is gradually in the core position of data transmission, computing and storage. This has brought about profound changes in the application mode and scale of the data center network. In the data center, online interactive application services [1]are gradually emerging, including Web services, online retail, search engines, advertising systems and social networks, etc., making data center applications face some soft real-time restrictions . If the response time of the user request exceeds the deadline, it will directly affect the performance of various services and further affect the user experience and return on investment. In the current data center network, the design pattern of split aggregation and fair sharing transmission protocol are generally adopted. However, the data transmission characteristics of the data center network, such as high bandwidth, low delay and high throughput, make the traditional TCP congestion control mechanism unable to adapt well to the data center network environment, resulting in congestion, timeout, packet loss and network performance degradation problems [2-4].

In order to solve this problem, it is necessary to design a new transmission control mechanism for the special network environment of the data center network. Researchers have proposed a fair and shared congestion control algorithm DCTCP [5], which adjusts the congestion window according to the

degree of link congestion instead of the traditional TCP method. Although DCTCP can ensure the fairness of the flow, it cannot distinguish the deadline of shunt, resulting in some flows missing the deadline. Some studies have focused on the deadline of streams, such as D3 [6]and D2TCP [7], but they do not distinguish the priority between delay-sensitive streams and delay-insensitive streams, which makes the delay-insensitive streams likely to be transmitted first during a short period of network congestion and increases the possibility of streams missing the deadline. In this paper, based on the guarantee of service time, firstly, by adding access control mechanism in sink node, the problem of more service transmission failure caused by simultaneous service for too many services is avoided, and the success rate of network transmission is improved. Secondly, by dividing priority queues for services with different urgency levels, priority is given to ensuring the service quality of services that are about to timeout. Finally, through the scheduling algorithm, in order to ensure that the service will not have timeout transmission, the overall service time of the service is reduced as much as possible.

2. MODELING AND ANALYSIS

2.1 Data Center Network Service Time Analysis

There is a problem in the data center network, that is, when there are a large number of data packets to be sent in the sink node, the data packets of some services often cannot be sent in time, resulting in these service services exceeding the service deadline, service transmission failure, and affecting the user experience.

As shown in FIG. 1, it is assumed that there are two kinds of services in the sink node, namely, the data packets of service 1 and service 2, and the data packets of service 2 (white) are in front of the data packets of service 1 (gray). If the sink node sends data packets in turn according to the current data packet sending sequence, then the data packets of service 1 cannot completely send the data packets of service 1 within the deadline, thus causing the overall transmission failure of service 1 data. This situation will lead to two results: one is that the transmission of traffic 1 packets in the current queue is invalid, and the other is that the packets sent through the network before traffic 1 are meaningless. In other words, the data center network resources occupied by service 1 are wasted, and the service time of other services is also reduced.

Although some research results have considered the guarantee of the deadline of a single data packet in the data center network, they have not considered the service time problem mainly at the service level, and there is also a lack of research on the service time from the scheduling algorithm level of the sink node. In view of the existing problems, This paper proposes a scheduling algorithm BST-TCP (Business Service Time TCP) based on guaranteed service time. In order to guarantee service time, the queue model of sink node in data center network is modeled at first, then a scheduling algorithm based on guaranteed service time is proposed, and finally the performance of the proposed scheduling algorithm is verified.

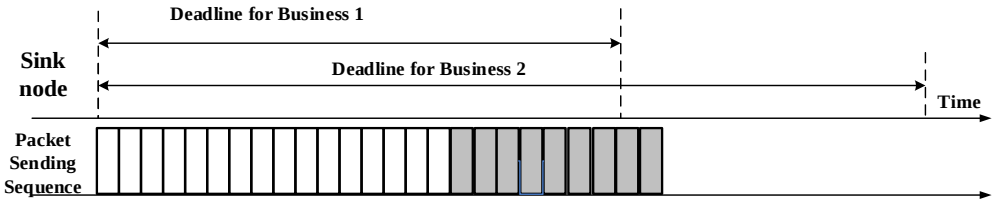

Fig. 1 Schematic of Business service time-out

2.2 Modeling and Analysis of Service Time

2.2.1 Service transmission process

In order to propose a scheduling algorithm to ensure the service time, it is necessary to understand the basic process of service transmission, which links in the process will affect the service time and to what extent, as shown in Figure 2. When the workstation receives the transmission request from the server, it starts to transmit the data packet of the corresponding service. The sink node will receive the service data packets from the workstation, and these data packets will first enter the queue to wait for transmission. Generally, the queue model of first-in first-out or random packet loss is adopted. When the sink node schedules these packets, it will send the corresponding packets to the server. In the process of one service transmission, the above process often exists many times until the server obtains all the service data and completes the data transmission

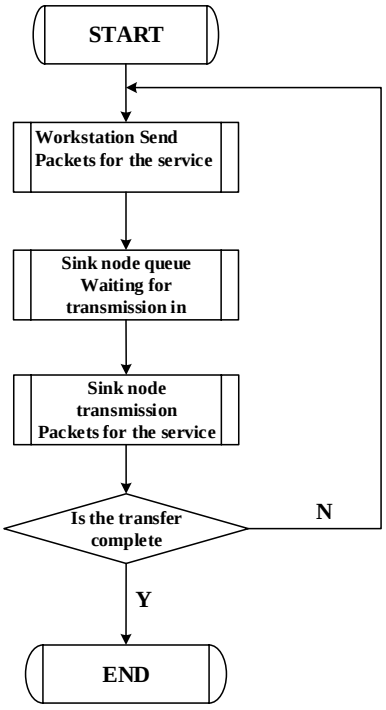

Fig. 2 Process of data center service transmission

2.2.2 Definition and Analysis of Business Service Time

As shown in FIG. 3, the schedule diagram of service transmission in the data center network shows the process of service transmission. Where $T_{\max}$ represents the maximum service time corresponding to the service. If the maximum service time is exceeded, the service transmission will fail, and the transmission of all relevant service data is meaningless. The service time of the service mainly includes two aspects: one is the request time, that is, the time when the server sends the request

information to the workstation receives the request information; The other is the service transmission time, that is, the time when all data packets of the service are sent from the workstation to the server. For the request time, because the amount of data transmitted is small and the link from the server to the workstation is not congested, it takes less time and the main time is consumed in the service transmission time. In addition, from the perspective of the sink node, the request information sent by the server can be sensed at the first time, and it can be considered that the sink node can obtain the time point when the server sends the request information.

In the transmission time of the service, it is mainly composed of three parts: T_w represents the time when the service data is sent from the workstation to the sink node, T_q represents the time when the data packet waits in the sink node queue, and T_s represents the time when the data packet is transmitted from the sink node to the server. It can be seen that in the process of service transmission, this transmission process lasts many times (assumed to last n times in FIG.

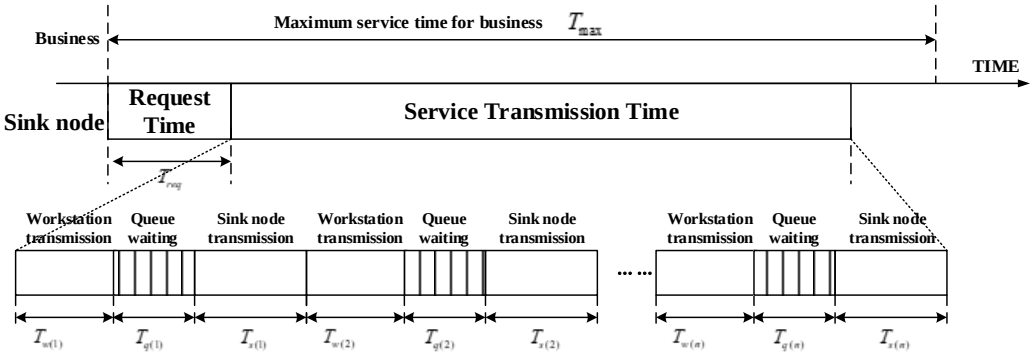

Fig. 3 Data Center Network Data Transmission Timing Diagram

2.2.3 Methods to Guarantee Business Service Time

Through the above description, it can be found that if the service time is guaranteed, two conditions need to be met: one is that the service data should be sent to the sink node as soon as possible, and the other is that the data packet should be sent to the server as soon as possible when it arrives at the sink node. This paper focuses on how to prioritize and ensure more services to complete transmission within the time limit when data packets are piled up in sink nodes. The main solutions are as follows:

(1) For newly initiated services, it is decided to serve the service according to the current service situation of the sink node, and it is not necessary to serve multiple services at the same time so that most services cannot complete data transmission within the maximum service time.

(2) For services that have already started service, the remaining service time of the corresponding services should be obtained in real time (the maximum service time minus the currently consumed service time), and services with less remaining service time should be preferentially scheduled.

3. SCHEDULING ALGORITHM

3.1 Access Control Method of Sink Node

Through the previous analysis, we can know that when the sink node provides services for all service packets, then a likely result is that some services cannot complete data transmission within the maximum service time, thus affecting the performance of the overall network. In order to avoid the above situation, the sink node needs to introduce access control for.

The bandwidth between the sink node and the server is BW, and the sink node maintains the variable service load ratio η. Assuming that the sink node provides services for a total of M services at this time, it is defined as η:

$$\eta = \frac{\sum_{i=1}^{M} L(i)}{BW} \tag{3-1}$$

Where $L(i)$ is defined as the bandwidth resource occupied by the first i service, and the sink node will respectively count or update the bandwidth resource occupied by the current services service every other period K, and $L(i)$ is defined as:

$$L(i) = \alpha L(i)' + (1-\alpha)E(i)/K \tag{3-2}$$

Where $L(i)'$ represents the situation that the service occupies bandwidth resources counted in the previous period, $E(i)$ represents the amount of data that the service enters the sink node in the current period, and α is a fixed parameter used to describe the influence of the previous service occupies bandwidth on the present.

When a new service enters the sink node, the sink node needs to make judgment. If η is less than 1, it means that the sink node has additional bandwidth resources for transmitting the new service, and provides services for the new service. If η is greater than or equal to 1, the data packet of the corresponding service is discarded.

3.2 Multi-queue Model of Sink Node

There are data packets of multiple services in the internal queue of the sink node. In order to fully guarantee the service time of different services, it is necessary to classify the priority levels of different data packets first, and to schedule services with more urgent service time first. Define the remaining service time for business i as:

$$T_{remain}(i) = T_{\max}(i) - T_{used}(i) \tag{3-3}$$

Where $T_{used}(i)$ represents the total time elapsed after the current service sends the request from the server, and this parameter can be obtained because the sink node records the time when the server sends the request. The sink node can obtain the total data amount $S_{total}(i)$ of each service, record the data amount $S_{trans}(i)$ that has been sent, and obtain the total amount of remaining services:

$$S_{remain}(i) = S_{total}(i) - S_{trans}(i) \tag{3-4}$$

Define the service transmission priority as:

$$P(i) = \frac{S_{remain}(i)/L(i)}{T_{remain}(i)} \tag{3-5}$$

Where $L(i)$ is defined as the bandwidth resource occupied by the service, and the time required to complete the transmission of all services according to the current transmission state can be obtained by dividing the total amount of remaining services $S_{remain}(i)$ by the bandwidth. $P(i)$ indicates the priority level of service i. Generally, $P(i)$ is a value greater than 0 and less than 1. When $P(i)$ is greater than 1, it indicates that the current network state cannot provide services for the service and all data packets of the service need to be discarded. When $P(i)$ is larger, it means that the service needs relatively more time to transmit and has a higher degree of urgency, so it has a higher transmission

priority. When $P(i)$ is smaller, it means that the service needs relatively less time to transmit and has lower urgency, so it has lower transmission priority. Specifically, the service is divided into four priority queues according to different urgency levels in the paper:

$$Priority = \begin{cases} \text{First Priority Queue}, 0.9 \le P(i) < 1.0 \\ \text{Second Priority Queue}, 0.7 \le P(i) < 0.9 \\ \text{Third Priority Queue}, 0.5 \le P(i) < 0.7 \\ \text{Fourth Priority Queue}, 0 \le P(i) < 0.5 \end{cases} \tag{3-6}$$

As shown in FIG. 4, the sink node will periodically update the priority information of different services. According to the above division criteria, there are four queues. When the sink node receives the data packet, it will enter different queues according to the priority of the service. When the sink node sends data, the data packets of the high priority queue are preferentially sent, and the specific data transmission service is determined according to the scheduler (the scheduler is introduced in the next section). It should be noted that after the sink node periodically calculates the priority levels of different services, it will adjust the packet queues of different services according to the calculated priority level results.

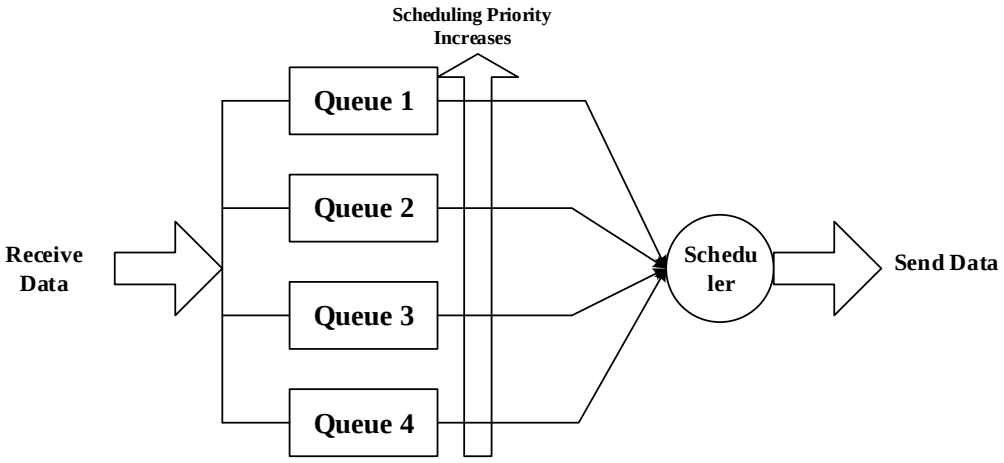

Fig. 4 Multi-Queue Model of Aggregation Nodes

3.3 Scheduling Algorithm for Guaranteeing Service Time

When the sink node completes the priority queue division of different service data, in order to further ensure the service time of the service and minimize the service time of different services, a scheduling algorithm needs to be introduced after determining the priority queue for transmission. The goal of the scheduling algorithm in the sink node is to minimize the service time of the service. In order to achieve this goal, the sink node needs to collect the status information of different services, including: the total data amount of each service, the data amount that each service has sent, the remaining transmission time of each service, and the bandwidth resources that the current service can occupy. Through the scheduling algorithm, the sink node will calculate the scheduling priority of each service in the current transmission opportunity. In order to reduce the computational complexity of the sink node, the sink node will determine the service number of obtaining channel resources in a short

period of time in the future every once in a while. The calculation method of scheduling weights for each service will be described separately below.

Defines the scheduling weights $\gamma(i)$ for different services as:

$$\gamma(i) = \frac{P(i)}{S_{remain}(i)} \tag{3-7}$$

Where $P(i)$ represents the ratio of the transmission time required for the remaining data to the remaining service time, and $S_{remain}(i)$ represents the amount of data remaining for the service. When the proportion of transmission time required by the service to the remaining service time is higher, the urgency of the service is higher, so it has higher scheduling weight. In addition, when the amount of remaining data of the service is small, it means that the service can complete the transmission soon, so the sink node should allow the service to complete the transmission as soon as possible, thus reducing the service time of the service.

4. ANALYSIS OF EXPERIMENTAL RESULTS

In order to verify the performance of the scheduling algorithm based on guaranteed service time, the DCTCP protocol and D2TCP protocol are compared, and the service transmission success rate and the average packet delay of the three protocols are mainly compared. The service transmission success rate is defined as the number of all services arriving in the maximum service time divided by the total number of services.

4.1 Experimental environment

The experimental parameter settings are shown in Table 1, where the service rate is 100Mbps to simulate the performance of the three protocols under the condition of increasing network load. Each simulation is often 10 seconds, and each scene is simulated 100 times, and the simulation result takes the average value of 100 simulations.

Table 1 Configuration of experimental parameters

Parameters	Parameter Settings
Average rate of service	100 Mbps
Packet Size	1460 byte
Workstation Link Delay	0.025ms
Data Center Link Delay	0.025ms
Workstation Bandwidth	1 Gbps
Data Center Bandwidth	2 Gbps
Workstation Queue Model	FIFO model
Data Center Queue Model	RED
Data Center Queue Length	250 packet
Emulation Time	10 seconds

4.2 Experimental results

FIG. 5 shows the simulation results of the service success rate of the three protocols with the increasing number of services (the maximum service time is 40ms). It can be seen that with the increasing number of services, the success rate of the three services shows a downward trend. When

the number of services is 25, the total bandwidth required by the services is higher than the bandwidth between the sink node and the server. The success rate of transmission control protocol is the lowest. Only when the congestion state is already very serious can transmission control protocol perceive congestion and carry out congestion control. DTCP protocol adjusts in time through congestion estimation, but does not consider the service time of the service. On the one hand, the algorithm proposed in this paper limits the number of services accessing the network through access control; On the other hand, it gives priority to dispatching more urgent services by putting services into different priority queues, so it can still better ensure the transmission success rate of services.

FIG. 6 shows the simulation results of the average packet delay of the three protocols with the increasing number of services. With the increasing number of services, the average delay of the three services shows an upward trend. For the traditional TCP protocol, because the congestion state of the link cannot be sensed in time and the congestion control is the latest, the average delay is the highest. DCTCP protocol adjusts the congestion window size in time according to ECN feedback to avoid a large number of packet losses, so its performance is higher than that of TCP protocol. However, the average delay of the proposed algorithm is always at a low level, This is because the algorithm can avoid serving services exceeding the network bandwidth at the same time through access control, and can further reduce the delay of transmission packets through priority-based multi-queue and scheduling algorithm guaranteeing service time, so its performance is the best.

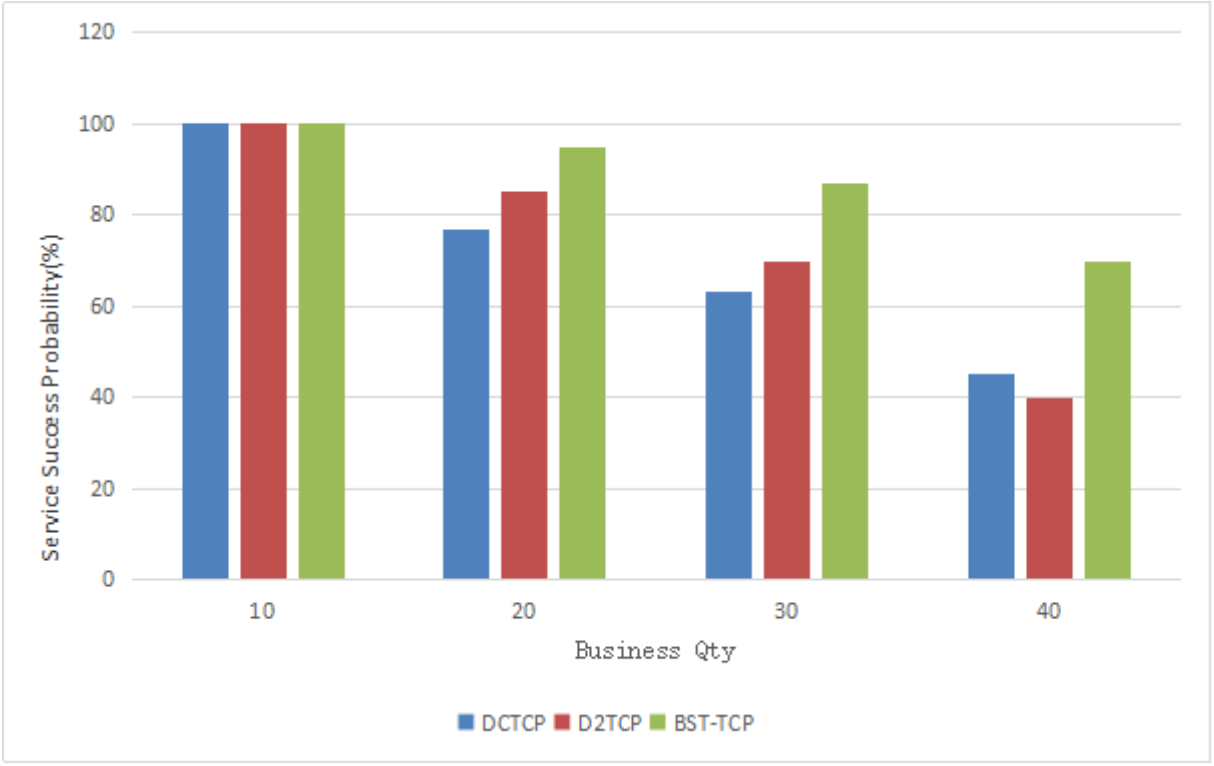

Fig.5 The simulation results of the business success rate with the change of the number of services

When the number of fixed services is 25, the simulation results of the service success rate of the three protocols that continuously decrease with the maximum service time are shown in FIG. 7. With the continuous decrease of the maximum service time, the service success rate of the three services has decreased, and the success rate of the algorithm proposed in this paper is the closest to the theoretical upper limit. When the traffic is fixed, The probability of node success is only affected by the sending

order of the sink node, However, the algorithm proposed in this paper can directly carry out data prioritization queue and scheduling based on guaranteed service time in the sink node, so its adjustment mode is effective and direct, and the sink node itself has mastered the current information of all services when scheduling data packet transmission, which is more conducive to the sink node's data transmission scheduling. Under the control of the traditional TCP protocol, the congestion window cannot be adjusted in time, but the window is halved after the congestion is sensed, resulting in severe window oscillation. Therefore, with the increase of the number of services, the network load increases, and the probability of service success also decreases. Although DCTCP protocol adjusts the congestion window in time, due to the allocation principle of fair sharing, it does not give priority to the transmission of high-priority services, resulting in some data packets being cancelled only half of the deadline. However, BST-TCP protocol can arrange a more reasonable queue sequence according to the current network situation and the number of services and other information to ensure the priority completion of services with high urgency.

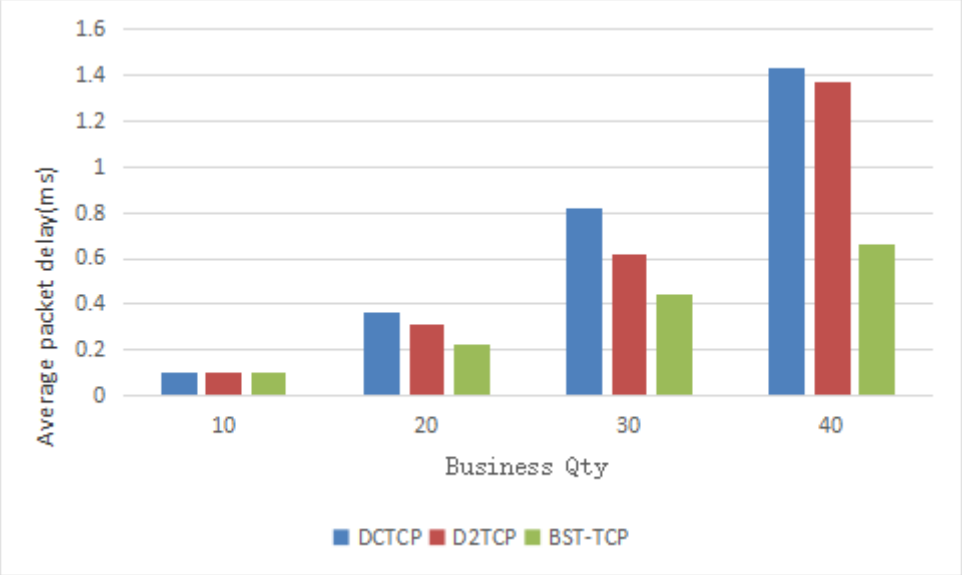

Fig.6 The simulation results of average packet delay with the change of the number of services

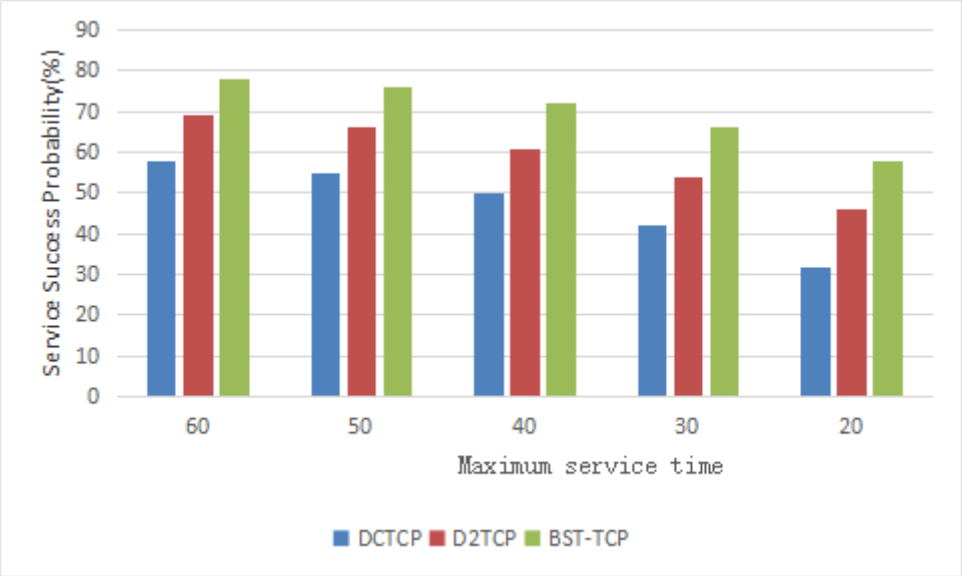

Fig.7 Simulation results of business success rate with the change of maximum service time

Figure 8 shows the proportion of flows that miss the time limit as the number of services increases. It can be seen that with the increase of the number of services and the increase of network load, the proportion of flows that miss the deadline under DCTCP protocol increases rapidly. D2TCP can better ensure the completion of flows within the deadline than DCTCP. However, with the increasing network load, the overall throughput of D2TCP decreases and its performance also deteriorates rapidly. However, the BST-TCP proposed in this paper can better ensure the flow to complete transmission within the deadline and improve the success probability of the service.

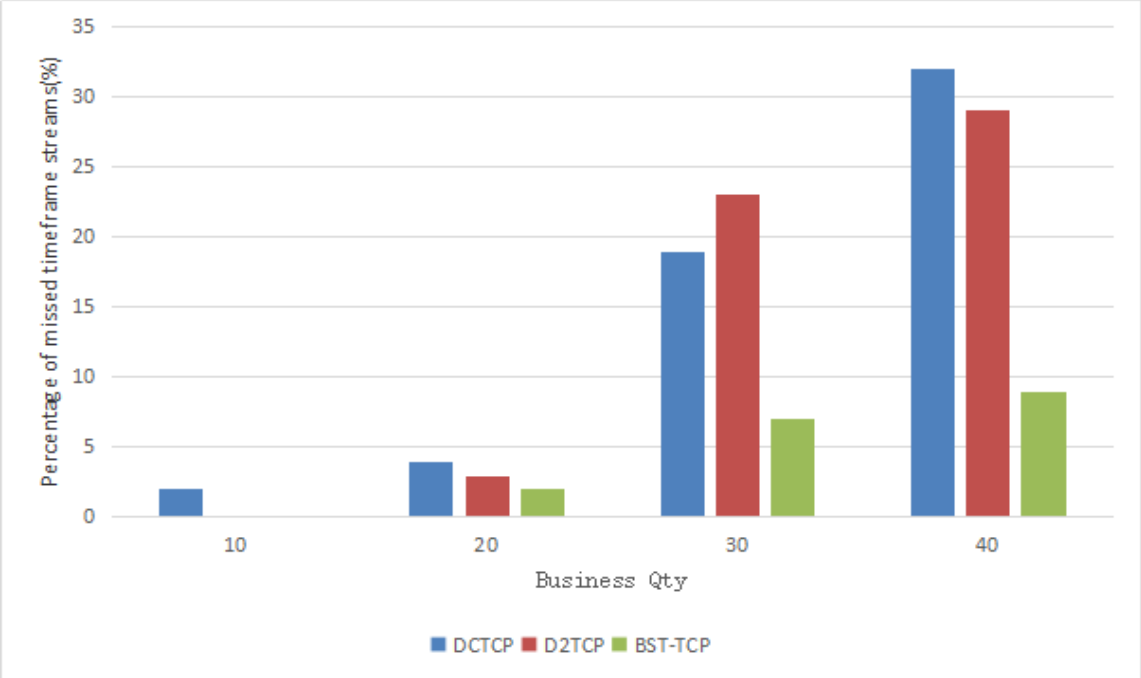

Fig.8 Simulation results of the missed deadlines flows

5. CONCLUSION

In the data center network division-aggregation transmission mode, the aggregation switch receives a large number of data packets at the same time, resulting in cache overflow and packet loss, which affects the online interactive application to feedback the results to users immediately. Therefore, according to the congestion degree of the network and the proportion of the transmission time required for the remaining data and the remaining service time, a 4-level queue is set up at the convergence switch to flow different services into the corresponding queue level to ensure that more urgent tasks are completed and queued first. The scheduling algorithm ensures that services with relatively high priority can be fed back to users as soon as possible, thus improving the service performance of the network.

REFERENCES

[1] Meisner D , Sadler C M , Luiz André Barroso, et al. Power management of online data-intensive services[C]// 38th International Symposium on Computer Architecture (ISCA 2011), June 4-8, 2011, San Jose, CA, USA. ACM, 2011.

[2] Zhao B Y . Tapestry: a Resilient Global-Scale Overlay for Service Deployment[J]. IEEE, 2004, 22.

[3] Chandrasekaran S , Cooper O , Deshpande A , et al. TelegraphCQ: Continuous Dataflow Processing[C]// Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data, San Diego, California, USA, June 9-12, 2003. ACM, 2003.

[4] Laptev N , Mozafari B , Mousavi H , et al. Extending Relational Query Languages for Data Streams[M]//

Data Stream Management. Springer Berlin Heidelberg, 2016.

[5] Alizadeh M, Greenberg A G, Maltz D A, et al. Data center TCP (DCTCP)[C]// 2010.

[6] Wilson C , Ballani H , Karagiannis T , et al. Better never than late:meeting deadlines in datacenter networks[J]. ACM SIGCOMM Computer Communication Review, 2011, 41(4):50-61.

[7] 7. Vamanan B, Hasan J, Vijaykumar T N. Deadline-aware datacenter tcp (D2TCP)[C]. Acm Sigcomm Conference on Applications. ACM, 2012:115-126.