TELKOMNIKA, Vol. 11, No. 10, October 2013, pp. 5627 ~ 5635
ISSN: 2302-4046

Research and Implementation of Congestion Control Scheme in ForCES Router

[1]Bangzhi Xiao, [1]Ming Gao*, [1]Weiming Wang and [2]Julong Lan
[1]College of Information & Electronic Engineering, Zhejiang Gong Shang University, Hangzhou, PR China
[2]Institute of Information Engineering, Information Engineering University, Zhengzhou, PR China
Corresponding author, e-mail: xiaobangzhi@pop.zjgsu.edu.cn, {gaoming, wmwang}@mail.zjgsu.edu.cn, ndsclj l@163.com

Abstract

With packet input rate rise ceaselessly, FoCES system will finally be congested. In order to solve this problem, aiming at the common local congestion and global congestion, this paper proposed a local congestion control strategy based on scheduling and a global congestion control mode based on the linkage between CE and FE. These two congestion control schemes can independently run inside the system, and also can operate simultaneously. Making that the forwarding rate, throughout and packet loss all have a great improvement in the FE end.

Keywords: *ForCES, congestion control scheme*

1. Introduction

Forwarding and Control Element Separation (ForCES) [1] is on the basis of the idea of programmable networks and has been paying more and more attention to. So far, the IETF ForCES working group [2] has successfully completed the Requirements (RFC3654) [3], Framework (RFC3746) [4] and Protocol Specification [5].

In the ForCES architecture, every router is considered as a Network Element (NE), as shown in Figure1. NE consists of Control Element (CE) and multiple Forwarding Elements (FEs). CE is responsible for instructing one or more FEs how to process packets via ForCES protocol. FE uses the underlying hardware to provide packet processing and handling as directed by a CE. The ForCES protocol runs on Fp reference point, which is single-hop or multi-hop network [6].

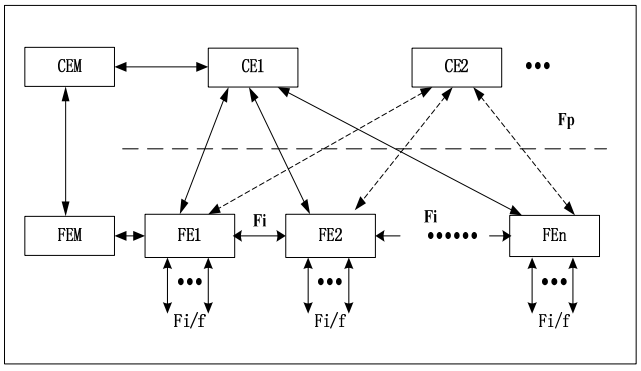

Figure 1. ForCES architecture

The main researches on ForCES are on the ForCES router's structure and the specific implementation. While not so much for the network congestion. The research of congestion control scheme on ForCES router is concentrated in effective and equitable distribution of resources and scheduling. Including the following aspects:

Received March 26, 2013; Revised June 26, 2013; Accepted July 12, 2013

In the research of ForCES channel transmission scheme and performance model [7], we can get a traffic model on the basis of the congestion avoidance control algorithm. A two-stage scheduling system was designed by solving the performance of the sending end, the network and the data receiving end to achieve the end-to-end performance in the whole system.

During the research of traffic model and flow control technology in ForCES router, the concept of flow control was introduced at the first time [8]. After the measurement and modeling towards the router inner network traffic, a predicted flow control allocation algorithm based on traffic matrix was proposed. Predicting the network throughput reasonably and fairly allocate the bandwidth to all links, so as to avoid network congestion [9].

Due to the limitations in the above-mentioned points, I proposed a local congestion control strategy based on scheduling and a global congestion control mode based on the linkage between CE and FE. These two congestion control strategies can be independently run inside the system, and also can operate simultaneously.

2. Related Work

According to RFC 5810 ForCES protocol specification requirements, transport mapping layer (TML) congestion control design requirements are as follows: preventing the control side or forwarding side from collapsing caused by transmission congestion, a congestion control scheme must be provided in this layer. TML has defined a congestion control scheme to prevent overload generated by the flow of CE or FE and eventually lead to a system crash. In addition, making the protocol layer (PL) informed the congestion happened, TML also need to define a congestion notification message. Once congestion occurs within the system, TML will send this message to PL.

The function of transport mapping layer is to transmit the ForCES message for PL. ForCES TML is internally divided into six sub-modules which can be seen in Figure 2, mainly includes the interface processing module, the message sending module, a message receiving module, message scheduling module, congestion monitoring module and log management module. The interface processing module is mainly used to manage the interfaces between the individual modules. The message sending module sends the encapsulated ForCES message to the external network. The message receiving module receives the incoming IP packets from the external network. The message scheduling module is used to schedule the IP packets from the external network pass through TML to PL. The congestion monitoring module is mainly responsible for monitoring congestion status within the system. Congestion control scheme is carried out by the congestion monitoring module and message scheduling module. When the congestion monitoring module notice the message sending module is abnormal, the TML congestion alarm events will be triggered. At the same time, TML will inform the PL about this.

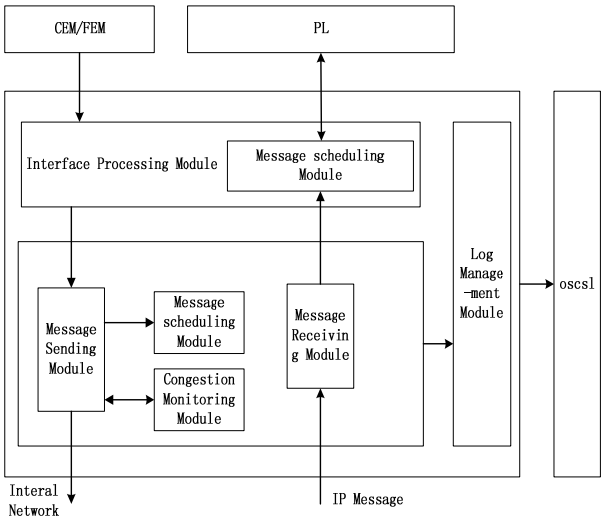

Figure 2. ForCES TML internal sub-modules

In the existing ForCES prototype system, we have developed a variety of strategies for congestion control. Mainly include dynamic probabilistic algorithm based on rate and queue control (RQ-DPA), flow control allocation algorithm based on flow matrix forecast, PGPS and SP two-stage scheduling algorithm based on network calculus. Next, more detailed description of these algorithms, and their advantages and disadvantages will be given.

2.1. Dynamic Probabilistic Algorithm Based on Rate and Queue Control

RQ-DPA algorithm is on the basis of rate and queue, its model is shown in Figure 3. This algorithm divides the message within the system into two types, control protocol message and redirect protocol message. For each message flow, build a FIFO message queue, then separately allocate an initial probability p_c and $(1-p_c)$ ($0 \le p_c \le 1$) [10]. According to the value p_c, the scheduler will execute the scheduling algorithm. During the scheduling process, with the change of the value of p_c, the bandwidth allocation of the different message queue will have the following changes:

- When p_c decreases, the bandwidth of control protocol message channel will be smaller, and the bandwidth of redirect protocol message channel becomes large.
- When p_c increases, the bandwidth of control protocol message channel will be larger, and the bandwidth of redirect protocol message channel becomes smaller.

Therefore, by adjusting probability parameters of p_c, we can reallocate the bandwidth of control message channel and redirect message channel, which can meet the bandwidth requirements between the different message channels.

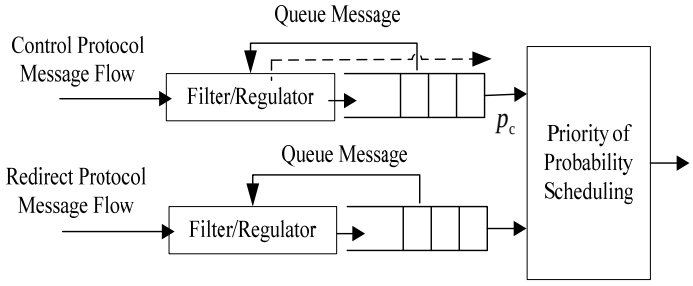

Figure 3. ForCES TML internal sub-modules

From Figure 3 we can know that in the RQ-DPA algorithm design model, there is a filter/regulator at the bottom each queue. Its role is to decide whether to adjust the bandwidth or discard redirect protocol packets according to the message arrival rate. If the arrival rate of control message is greater than the bandwidth provided by the system, then the system will increase the value of parameter p_c to increase the control protocol message channel bandwidth and reduce the redirect protocol message bandwidth. Conversely, if the arrival rate of the redirect protocol message flow is over its bandwidth, it will result in the queue in the buffer becomes too long, and then the system can determine whether to discard the message according to the arrival rate of the length of queue.

2.2. Flow Control Allocation Algorithm Based on Flow Matrix Forecast

This algorithm is one of algorithm that can be used in the ForCES system to prevent congestion. Here is how it works:

When CE and FE begin to communicate, the algorithm will allocate a initiate bandwidth B_k^{ini} and a dividual bandwidth B_k^{div} for each FE. B_k^{ini} indicates the initiate bandwidth of the K-th FE, while B_k^{div} presents the dividual bandwidth of the K-th FE. In the CE end, we can get the number N of FE that connected to it. The N is changeable, anytime may have new FE connect to the CE or connected FE may disconnect. Besides, use traffic matrix estimate the transmit rate r_k^{t+1} of each ForCES channel. The actual ForCES channel bandwidth value with

message flow arrival rate is proportional to the relationship [11]. This algorithm according to the change of N and r_k^{t+1} to adjust the dividual bandwidth B_k^{div}.

When the sum of message rate of all ForCES channel is larger than CE withstand bandwidth, and do not take any measures, the link will become congested, leading to ForCES message lost. If the link congestion is serious, a whole kind ForCES channel message may even completely be thrown away. Or control message may be discarded which can cause ForCES routing system cannot work normally.

The algorithm allocates the bandwidth for the individual ForCES channel reasonably to make sure that the total bandwidth will not be greater than CE withstand range, so as to effectively avoid the occurrence of congestion.

2.3. PGPS and SP Two-Stage Scheduling Algorithm Based on Network Calculus

PGPS scheduling policy is an implementation to make the ideal GPS model can be applied to a reality system. PGPS scheduling algorithm also can be called WFQ scheduling algorithm. GPS strategy is based on the flow model, while in the real network; what is always transmitted is data flow [12]. Therefore, the ideal model of the GPS algorithm for infinitely divisible data stream cannot be applied in a real network. It will bring some mistakes if we use the GPS algorithm in theory analysis. And PGPS scheme is the scheme that the length of packet is changeable in GPS. PGPS notices that the transmission is done until all the packets are received. In the ideal GPS scheduling algorithm, Fp is used to represent the time required to finish the scheduling. PGPS scheduling policy gives service to the data packets according to the order of Fp in the GPS. If the system is free in some time, and the next packets in need of service are yet not come, but the system itself is not able to determine when the packets to arrive. So the system cannot simultaneously meet to ensure that the service and in strict accordance with the order of Fp.

Strict priority scheduling algorithm (SP) is currently the most widely used in the Round Robin algorithm priority scheduling method. SP algorithm is very extensive in real-time scheduling system. Scheduling system will allocate a different priority for each queue in the buffer before scheduling, such as high priority, medium priority, and low priority. After the previous round of scheduling, the scheduling system serves for the packets from the highest priority to the lowest priority.

So far, the queue priority can be allocated dynamically or statically. During the process of scheduling using the dynamic priority scheduling algorithm, the priority's allocation is changeable with the change of attributes of queues, including the length, buffer occupancy etc [13]. The allocation scheme of static priority allocation system is to assign a static priority for each queue. This priority cannot be changed during the queue scheduling process. When the priority of each queue is assigned, the kernel of the operating system determines the high priority queue is able to use any processor resources though preemptive or non-preemptive way. Preemptive kernel is that when the process is in kernel, and there is a higher priority task. If the current kernel allows seizing, the current task will be hung and execute the higher priority process. However, in non-preemptive kernel, low-priority process cannot seize the CPU resource with a high-priority process. When a high-priority process is in the kernel mode, no process can seize the CPU resource with it, unless the process is to completed and f the CPU is free.

3. Design and Implementation

Before designing the congestion control scheme, we set a congestion threshold for each queue. Besides, we also set a total buffer congestion threshold. When a single queue buffer occupancy rate exceeds the threshold, it means that local congestion occurs, and we can solve it by scheduling. If the total buffer occupancy rate exceeds the threshold, then start using the congestion control mode of CE and FE linkage.

When the ForCES messages come to CE from the external network, the first step is to classify them based on the QoS requirements and send them to the appropriate queue. Assuming the total buffer capacity is C, there are N queues in the buffer, so the maximum capacity of each queue is the $\frac{C}{N}$ (assuming the buffer capacity for each queue is

equal). The current capacity of the queue is A byte, the speed of entering massage is V_i byte/s, and the speed of exiting message is V_o byte/s. The time of the burst traffic is set to S seconds, so the burst lasts for $S = \frac{C}{N(V_i - V_o)}$, and the packet transmission speed is $(V_i - V_o)$.

The congestion control scheme of ForCES system designed in this paper is divided into two parts: the local congestion control and the global congestion control. The message channel between CE and FE consists of control message channel, event message channel and redirect message channel. When the burst flow is encountered, a channel message stream may rise sharply, resulting in the stream in the buffer queue also sharply increased, eventually lead to local congestion [14]. At this time, the scheduling module will make some certain measures to ease the congestion. When all of the buffer queues reach to the saturated state, it means that the system is in the global congestion. TML congestion control module will monitor this to PL, and PL will inform all FE ends through news feedback, demand the FE end to reduce the transmission rate, so as to ease the congestion.

Message flows reached to CE are usually from different FEs, including various types of messages, and different message have different service requirements for ForCES system. The messages for higher service requirements and lower service requirements should be treated differently. We give different scheduling strategies to provide a better quality of service to the higher service requirements. For the different messages in the ForCES system, we divide different message channels. As to the reliable transmission messages (association message, configuration message, query message, etc.), we use a high-priority reliable channel (HP) for transmission. For the messages that can tolerate a certain timeout and lost (event message), using a semi-reliable medium priority channel (MP). With regard to the external messages (redirect message), use the low-priority unreliable channels (LP).

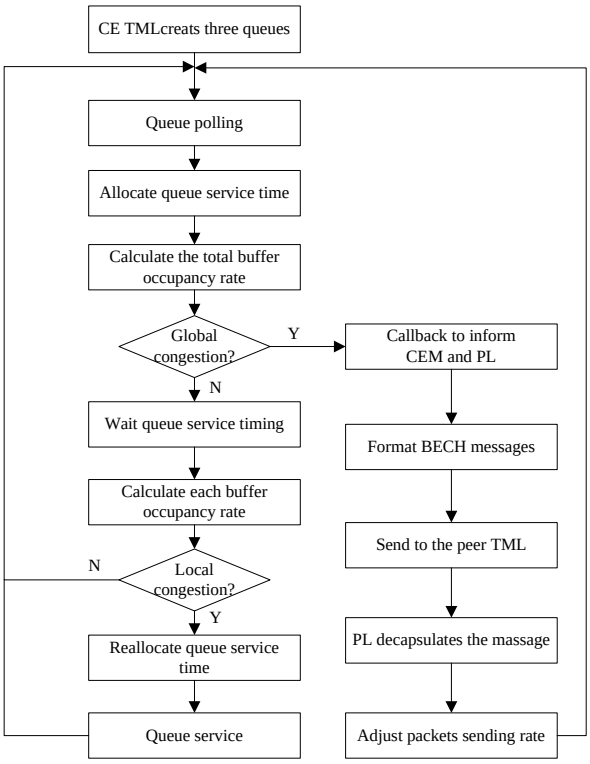

Figure 4. ForCES system congestion control process flow

ForCES system congestion control process is shown in Figure 4. The TML module in both CE and FE maintains three message queues for receiving messages: control message

queue, event message queue and redirection message queue. So the first step is to create these three message queues in TML and allocate a fixed-length buffer for each message queue. Each queue maintains a weighting factor, and this weighting factor is the queue buffer occupancy rate. A queue threshold of occupancy rate for each queue buffer is set in advance; meanwhile, a global threshold of occupancy rate for all of the three queues also should be set.

After each message queue created, scheduler in CE TML and FE TML will have polling services for this three message queues, and the polling time is equal. In each round of service, scheduler will allocate the queue service time according to the proportion of the weighting factors in the queue. When starting a new round of service, the scheduler will first calculate the overall three queues buffer occupancy rate and determine whether the rate is beyond the global threshold. If it is exceeded, the system enters the state of global congestion; otherwise, the message queue will wait for service as usual. When service time comes, the scheduler first calculates the occupancy of the queue buffer whether the rate exceeds the queue threshold [15]. If not, in accordance with the assigned service time for FIFO service; if exceeded, it indicates that the system is entering the local congestion mode.

In the local congestion mode, scheduler redistributes the remaining service time of the current round, to the maximum extent possible to meet the requirements of this queue. In the case of original allocated time is not enough, borrow it from the queue that is not serviced until the queue is empty or the current round of the remainder with graduated so far. Meanwhile, in the next round, the queue will be vacant for one round. The other queues receive service and after that waiting for the polling operation.

In global congestion mode, TML module notifies PL through CEM by the way of callback function. PL module constructs a congestion feedback message and sends it to the peer TML end. In this process, TML produces a congestion alarm to notify PL, making PL take appropriate measures to alleviate the congestion of the internal system, and PL gets the congestion message via CEM. When designing the congestion control scheme, in order to not break existing structure of TML and PL, we need do some extension towards functions and interfaces of the CEM. TML registers a callback function Congestion-Trigger () when CEM initialized. Meanwhile, PL also registers a callback function Congestion-Notification (). When the TML layer is in the state of global congestion, the PL layer will get this message by callback function. Normally, TML congestion will be a continuing event, after congestion alarmed; event status will be hold until the congestion is relieved. Given this, we need to define some state parameters for the function of Congestion-Trigger () and Congestion-Notification (), and these parameters should be associated with TML congestion events. This parameter uses “on” or “off” to indicate the congestion status in TML. When TML congestion occurs, congestion monitoring module will give an alarm to CEM module, and CEM management module will produce a congestion alarm command to inform PL that system congestion has occurred and take some appropriate measures.

After PL received the alarm command from CEM, it will encapsulate a backward explicit congestion notification (BECN) message in ForCES protocol type [16]. If we redefine a new type of ForCES message, it will increase the burden for the system. Therefore, this draft will generate a BECN message in the form of the existing internal message marked congestion alarm. Based on the existing ForCES message type and encapsulation, we have found that the heartbeat message used to check whether CE or FE still alive is the most suitable type for the reason that the message structure is relatively simple. It only has the message header but no message body, so the operation on it is very convenient, and the ForCES system would have sent the message every now and then. The header of heartbeat message consists of version number, reserved field, message type, length, source ID and destination ID, correlation factor and flag. As shown in Figure 5, the heartbeat message header flag field has the following contents: ACK, Pri, Resv, EM, AT, TP, Reserved. We set the alarm tag in the first two bits of Resv just behind the Pri position. The content of the mark occupies two bits, 00 indicates that no congestion occurs, 01 means that congestion is occurring, and 10 indicates congestion is relieved. When congestion occurs, TML in CE will send all FEs the BECN message. After receiving and decapsulating the message, FE end will take some measures to adjust the sending rate of packets.

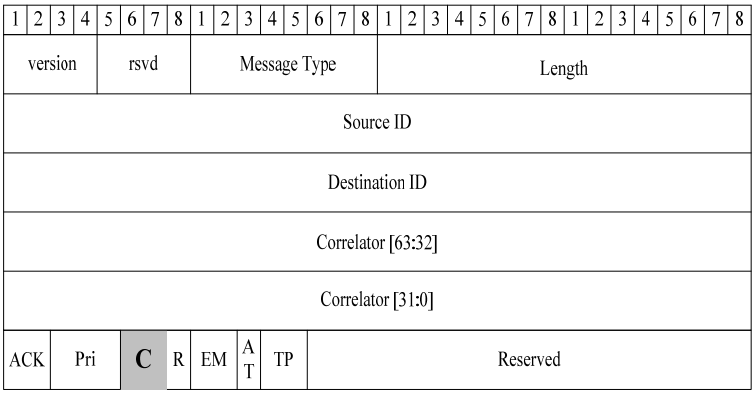

Figure 5. Extended ForCES message header with a congestion alarm tag

4. Performance Testing

4.1. Experimental Equipments

In order to test the superiority of our new designed congestion control scheme, we built a single CE and 3 FEs ForCES prototype system. A PC used as the CE, and FE by embedded network processor system.

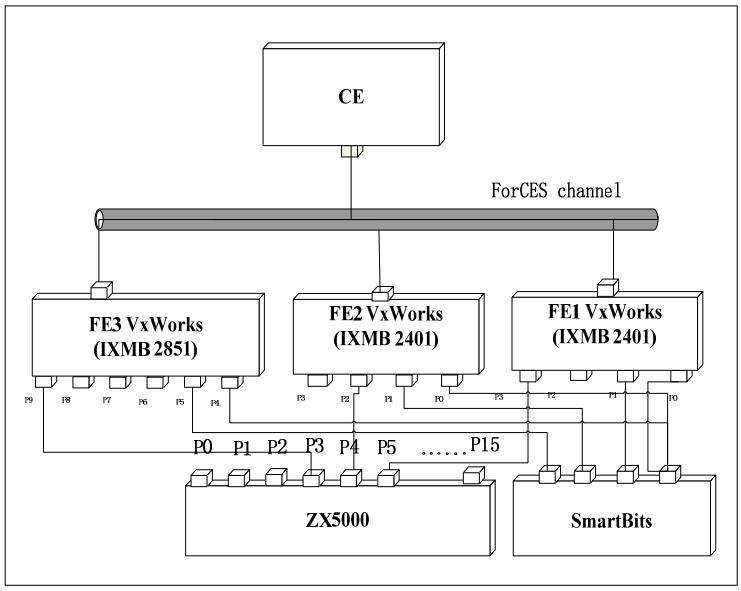

Figure 6. ForCES prototype system

As shown in the Figure 6, the entire system includes an IXMB2851 single network processor substrate, two IXMB2401 single network processor boards, a data communication tester Smart Bits, a ZX5000 high-speed switching board and a 5-slot advanced TCA standard chassis. CE communicates with FE through ForCES protocol on the Ethernet; all messages come to FE via the external interface and leave from another interface. So we can use the Smart bits to test the forwarding rate, throughout and packet loss of each FE.

4.2. Test Results and Analysis

Figure 7 shows the packet forwarding rate under the condition of with congestion control scheme and without it. Apparently, since the packet input rate soar to about 1500

packets per second, with our designed mechanism, the forwarding rate is much better than without using it.

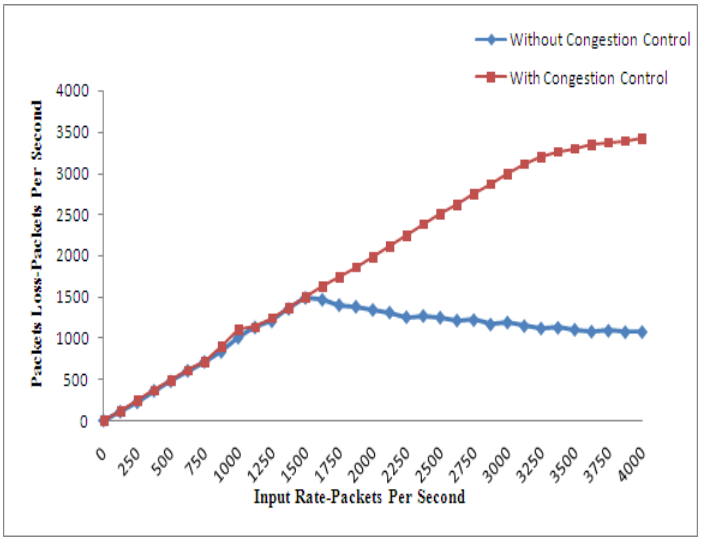

Figure 7. FE forwarding rate

Shown in the Figure 8, in normal conditions, with the packet input rate increasing, the packet loss will become very huge finally; sometimes even most of the packet will be lost. However, when we use the new designed congestion mechanism, no matter how the input rate is, the packet loss will become steady.

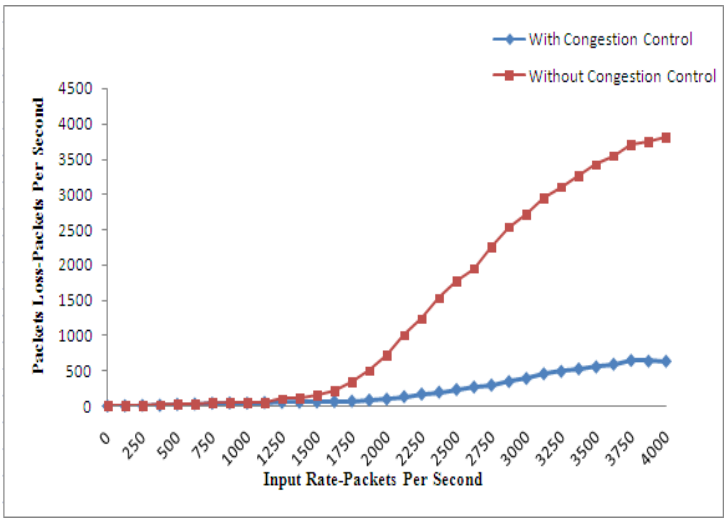

Figure 8. FE packet loss

5. Conclusion

First, we introduced some basic concepts about ForCES. Then two kinds of congestion control schemes used in ForCES system were given. In order to improve the two methods, aiming at the local congestion and global congestion, we separately put forward a new effective congestion control scheme. In the local congestion mode, scheduler redistributes the remaining service time of the current round, to the maximum extent possible to meet the requirements of

this queue. In global congestion mode, TML module notifies PL through CEM by the way of callback function. PL module constructs a congestion feedback message and sends it to the peer TML end. In this process, TML produces a congestion alarm to notify PL, making PL take appropriate measures to alleviate the congestion of the internal system.

Acknowledgements

This research was in part supported by the National Basic Research Program of China (973 Program) under Grant No. 2012CB315902, the Natural Science Foundation of China under Grant No. 61170215, 61102074, Zhejiang Provincial NSF of China under Grant No. Y1111117, Zhejiang Provincial College Students' Innovation and Entrepreneurship Incubation Projects under Grant No. 2012R408061, Zhejiang province key science and technology innovation team No. 2011R50010.

References

[1] Forwarding and Control Element Separation (ForCES). http://tools.ietf.org/id/draft-crouch-forces-applicability-01.txt,2002.
[2] IETF ForCES Working Group [OL]. http://www.ietf.org/wg/.
[3] H Khosravi, T Anderson. Requirements for separation of IP control and forwarding, http://www.ietf.org/rfc/rfc3654.txt,Accessed by Nov 2003.
[4] L Yang, R Dantu, T Anderson, R Gopal, Forwarding and control element separation (ForCES) framework, http://www.ietf.org/rfc/rfc3746.txt, Accessed by Apr 2004.
[5] A Doria, R Haas, W Wang, L Dong, Forwarding and control element separation (ForCES) protocol specification, http://tools.ietf.org/html/rfc5810, Accessed by Mar 2010.
[6] WM Wang. Forwarding and control element separation (ForCES) Technology and Application, Zhejiang University Press, China, ISBN 978-7-308-08296-9, Doc 2010.
[7] LY Ke. Research on channel transmission scheme and performance model in ForCES routers. Thesis of master's degree; 2011.
[8] Yu Cheng. Research on flow model and flow control in ForCES routers. Thesis of master's degree; 2011.
[9] Srisankar S Kunniyur, R Srikant. End-to-End Congestion Control Schemes: Utility Functions, Random Losses and ECN Marks. *IEEE/ACM Transactions on Networking*. 2003; 11: 689-702.
[10] Panos Gevros, Jon Crowcroft, Peter Kirstein. Congestion Control Mechanisms and the Best Effort Service Model. *IEEE Network Special Issue on the Control of Best Effort Traffic.* 2001: 16-26.
[11] D Kim, JT Chiang, A Perrig. A New Secure Congestion Control Architecture. Technical report UILU-ENG-2010-2511; 2010.
[12] JP Wagner, P Frossard. Distributed Congestion Control of Scalable Video Streams. *Journal of Communication*. 2012; 7(3).
[13] Sandip Chanda, Abhinandan. Congestion Relief of Contingent Power Network with Evolutionary Optimization Algorithm. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2012; 10: 1-8.
[14] GW Wu, F Xia. A Hop-by-hop Cross-layer Congestion Control Scheme for Wireless Sensor Networks. *Journal of Software*. 2011; 6(12).
[15] Zhenyu Na, Bao Peng, Liming Chen. AQM Algorithm with Adaptive Reference Queue Threshold for Communication Networks. *TELKOMNIKA Indonesian Journal of Electrical Engineering.* 2012; 10: 1062-1072.
[16] Nan Jiang, Becker DU, Michelogiannakis G, Dally WJ. *Network congestion avoidance through Speculative Reservation*. 2012 IEEE 18th International Symposium on High Performance Computer Architecture (HPCA). 2012; 1-12.