Research on Detectable and Indicative Forward Active Network Congestion Control Algorithm

Jingyang Wang
Hebei University of Science and Technology, Shijiazhuang, China
Email: ever211@163.com

Min Huang
Hebei University of Science and Technology, Shijiazhuang, China
Email: huangmin@hebust.edu.cn

Haiyao Wang
Fujian Jiangxia University, Fuzhou, China
Email: why-helen@163.com

Liwei Guo and Wanzhen Zhou
Hebei University of Science and Technology, Shijiazhuang, China
Email: {guoliwei, houwz}@hebust.edu.cn

***Abstract*—In order to solve the shortages of Forward Active network Congestion Control algorithm (FACC), this paper proposes a Detectable and Indicative Forward Active network Congestion Control algorithm (DIFACC). DIFACC uses RED buffer queue management algorithm instead of standard drop tail algorithm to increase the usage of bandwidth. It preserves the active detection and passive indication mechanism to realize load balance. It introduces sending speed adjustment policy, and designs different processing method for different kinds of service data according to their different characteristics and requests on network resources. DIFACC can not only relieve congestion in time when congestion happens, but also can avoid the congestion and increase the QoS of the different kinds of service data. A simulation experiment is given to analyze the performance of the algorithms. With the analysis of performance between FACC and DIFACC, it shows that DIFACC not only resolves some shortages in FACC, but also improves the OoS of different kinds of service data, reduces the package loss rate and decreases the processing delay of packets.**

***Index Terms*—DIFACC, congestion control, service data, active network**

I. INTRODUCTION

Active Network is an advanced research area on network in the world in recent years. It is a new network technology based on Internet. Active Network was put forward by Defense Advanced Research Projects Agency (DARPA) in the discussion on development direction of network system. With the rapid development of Active Network research, the research on active network congestion control is becoming increasingly important [1]. As a congestion control algorithm in Active Network, FACC [2],[3] uses active network technology to make feedback congestion control more responsive to network congestion. But FACC still has some problems to be resolved. The mainly shortages of the FACC are as follows.

- FACC is a kind of congestion control algorithm based on passive reaction. It merely relieves the network congestion passively. FACC doesn't take any actively preventive measure on network congestion according to the current network condition.
- The transmission priority of indication message is as the same as other packets. In this case, when the network node is at the congestion state, congestion control will be done immediately to control the congestion. This will lead to the loss of indication message. Thus FACC will be invalid.
- Network node queue management uses standard drop tail algorithm [2]. When network node queue is full, the node will discard all the data packets arriving at later. This will lead to a sharply loss on data packets coming from the same data stream.
- FACC can not ensure the Quality of Service (QoS) of different kinds of service data, because it processes different kinds of service data which has different characteristics and requests on network resources using the same method.

This paper puts forward DIFACC to solve the problems in FACC. DIFACC has improved upon FACC. It introduces the active detection and passive indication mechanism and combines with RED queue management [5] and the load balance technology. According to the transmission characteristics of different kinds of service

doi:10.4304/jsw.7.6.1195-1202

data, DIFACC introduces the concept of priority and uses different process methods to ensure the QoS [6].

II. DIFACC

A. RED Buffer Queue Management Algorithm

Compared with FACC, RED buffer queue management algorithm can be used to manage network nodes queue in DIFACC. The congestion condition of network nodes can be predicted by computing the buffer queue length with RED. The RED algorithm can control the buffer queue length of the router to a relative lower value, receive the data packets instantly and prevent data packets from the burst loss. So that it increases the usage of the bandwidth.

The RED algorithm uses a moving weighted average function to evaluate the buffer queue length and then predicts the congestion condition. The RED algorithm in DIFACC consists of two parts.

First part is to detect the initial network congestion. When a new data packet arrives, the buffer queue length can be calculated by a moving weighted average function [5] as follows:

$$avg = (1 - wq) * avg + wq * q \tag{1}$$

In this function, wq is the moving weighted average value; q is the current buffer queue length. RED compares the buffer queue length with two preplaced threshold parameters: min_th and max_th, and distinguishes whether the network is in congestion or not.

The second part is the congestion control rule. If avg $\geqslant$ min_th and $\leqslant$ max_th, the network is at the initial congestion state or the congestion avoidance state, and RED algorithm discards the data packets randomly with the probability Pa which is approximate in the buffer queue length function.

$$P_b = \frac{avg - \min\_th}{\max\_th - \min\_th} * \max_p \tag{2}$$

$$p_a = \frac{p_b}{1 - n * p_b} \tag{3}$$

In this function, $\max_p$ is the largest probability of dropping packets; n is the number of data packets from the previous time discarding the packets to current time. If avg is more than max_th, the network will be at congestion state and the router will discard each packet newly arriving with the probability.

With the description of all above, if avg is more than or equal to min_th and less than or equal to max_th, the network is at the initial congestion state or the congestion avoidance state, and RED algorithm discards the data packets randomly with the probability Pa which is approximate in the buffer queue length function. If Pa is more than or equal to 0 and less than 1, data packets are not to be discarded. Except this region, the data packets are to be discarded.

B. Realizing Load Balance

In FACC, when the network node is at the congestion state, it will send indication message immediately to the source node. After receiving the message, the source node will take some measures to relieve the network congestion. In fact, FACC is a kind of congestion control policy based on passive reaction. Indication message is sent at the congestion state. It can only indicate the congestion condition rather than prevent or predict the congestion. DIFACC preserves this passive indication message in FACC and then introduces the active detection and passive indication mechanism. In other words, DIFACC is a kind of preventive congestion control policy.

Active network consists of a group of network nodes which are called active nodes. Each active node can be a router or a switch. The active nodes comprise the execution environment of the active network. In DIFACC, the congestion information of each active node is given in Table Ⅰ.

TABLE I.
CONGESTION INFORMATION TABLE

Active node IP	Update Time	Queue Length
192.168.1.2	09:26:54	130
192.168.1.3	09:26:30	50

While starting, active nodes will send detection messages to all the neighboring nodes. After receiving the messages, the neighboring nodes calculate the buffer queue length using RED algorithm, put the results into the header of Active Network Encapsulation Protocol (ANEP) message and send it back to the source nodes, so that the active nodes can grasp the congestion information of the neighboring nodes and then create the table of congestion information.

In congestion information table, each record has an update time field to describe whether this record is latest or not. Compared the update time with a fixed threshold Tm, if it is less than Tm, we believe that this record is latest and effective. If not, we think that this record has been obsolete.

Before they transmit the data, active nodes inquire the congestion information table whether the records are latest firstly. If not, active nodes send detection message immediately to the neighboring nodes to update the records, and then find out the node whose buffer queue length is lowest. It indicates that the network resource of this node is most sufficient. And this node is the right one that data packets are sent to.

When the network is busy, the frequent transmissions of active detection messages can not avoid the congestion. On the contrary, it may add the burden of network and lead to more congestion. In order to avoid sending detection message so frequently, active node encapsulates its own buffer queue length value into the ANEP message with the transmission of data packets to all the nodes [6], [1] in order to update their tables. Furthermore, when the network node is at the congestion state, it will send the passive indication message to the source node. As same as the service data packets, active node also encapsulates the buffer queue length value into the header of the ANEP message. The network node which receives these two kinds of message (active detection message and

passive indication message) can update the congestion information table. In brief, to update the congestion information table has two methods: active detection and passive indication. However, when active node starts and the network is quite idle, the active detection message will be more. When the efficiency of use network resources is high, service data packets and the passive indication message (only at the congestion phase) are used to update the table. Due to the congestion information tables, active nodes can choose the optimum route [10] to transmit data, reduce the package loss rate and realize the load balance. The process of realizing load balance in active nodes is shown in Fig.1. DIFACC can prevent or predict the congestion in time. So compared with FACC, DIFACC is a kind of preventive congestion control policy.

The steps of realizing load balance are as follows.

(1) When active node starts, it sends indication packet to neighbor nodes to initialize congestion information table.

(2) Active node is at working states (listening states).

(3) If active node receives transmission data, it judges

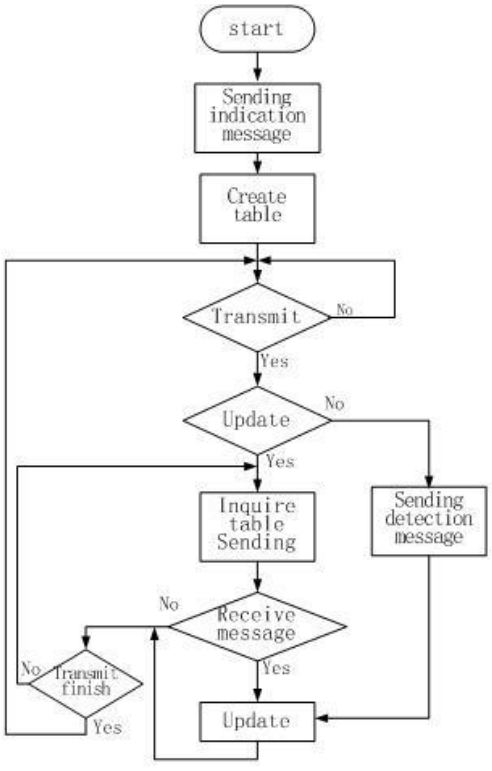

Figure 1. The process of realizing load balance

whether the congestion information is the latest, if not, goes to 7.

(4) Active node finds out the active node which buffer queue length is the shortest, then transfers the service data to that node.

(5) Active node judges whether the received data is the notification of decreasing speed, if yes, go to 8.

(6) Active node judges whether the transferring ends, if yes, go to 2, otherwise, go to 4.

(7) Sending indication packet to update congestion information table.

(8) Updating congestion information table.

C. Setting the Priority of Service Data

According to the different characteristics and demands for network resources of the various types of service data which is transmitted in the network, DIFACC algorithm designs different priorities. It implements that the active node uses different priority to process the congestion control based on different service data type. DIFACC can not only relieve congestion in time when congestion happens, but also can ensure that the special service data which has the higher priority can achieve the higer QoS. According to the different data types tranmitted in the network, the service data is abstracted by four types, which are video data, audio data, file data and message data. DIFACC implements that the active nodes discard the data packets according to different kind of service data and different priority. It also implements that the active node automatically chooses the congestion control algorithm which is suitable for corresponding service to process congestion control according to the different services.

In the FACC algorithm, when the congested node detects the congestion, it will process congestion control immediately; accordingly the load condition of the congested node will be relieved. The indication message packet actually plays a role in promptly responding to the congestion condition on the whole transmission process. However the indication message packet may be missed during transmission, the sending source node can not make necessary adjustments according to the current network usage, eventually FACC will be invalid. The priority of indication message packet is set the highest in DIFACC. The network nodes directly put the indication message packet at the head of the buffer queue to treat them preemptively, and ensure that the indication message packet is transmitted securely. So the problem of the algorithm invalidation which is caused by the loss of the indication message packet during transmission is solved. The priority of detection message packet is also set the highest. Indication message packet and detection message packet are collectively called notification packet.

D. Sending speed adjustment

When the active node is at the congestion state, it will randomly select and discard the data packet which has the lower priority, as well as will send the reduction speed notification, and it will be need to resend data for the non real-time data. The sending speed adjustment procedure includes the reduction sending speed process and the increase sending speed process. When the active node congestion condition is relieved, the terminal equipment which has reduced sending speed can increase the sending speed initiatively.

The mainly steps are shown in Fig.2.

(1) Define variables. The varialbes include the time of sending data Ts, the time of receiving the reduction speed notification Tr, the time of receiving the reduction speed notification last time Tr1, the count of consecutively

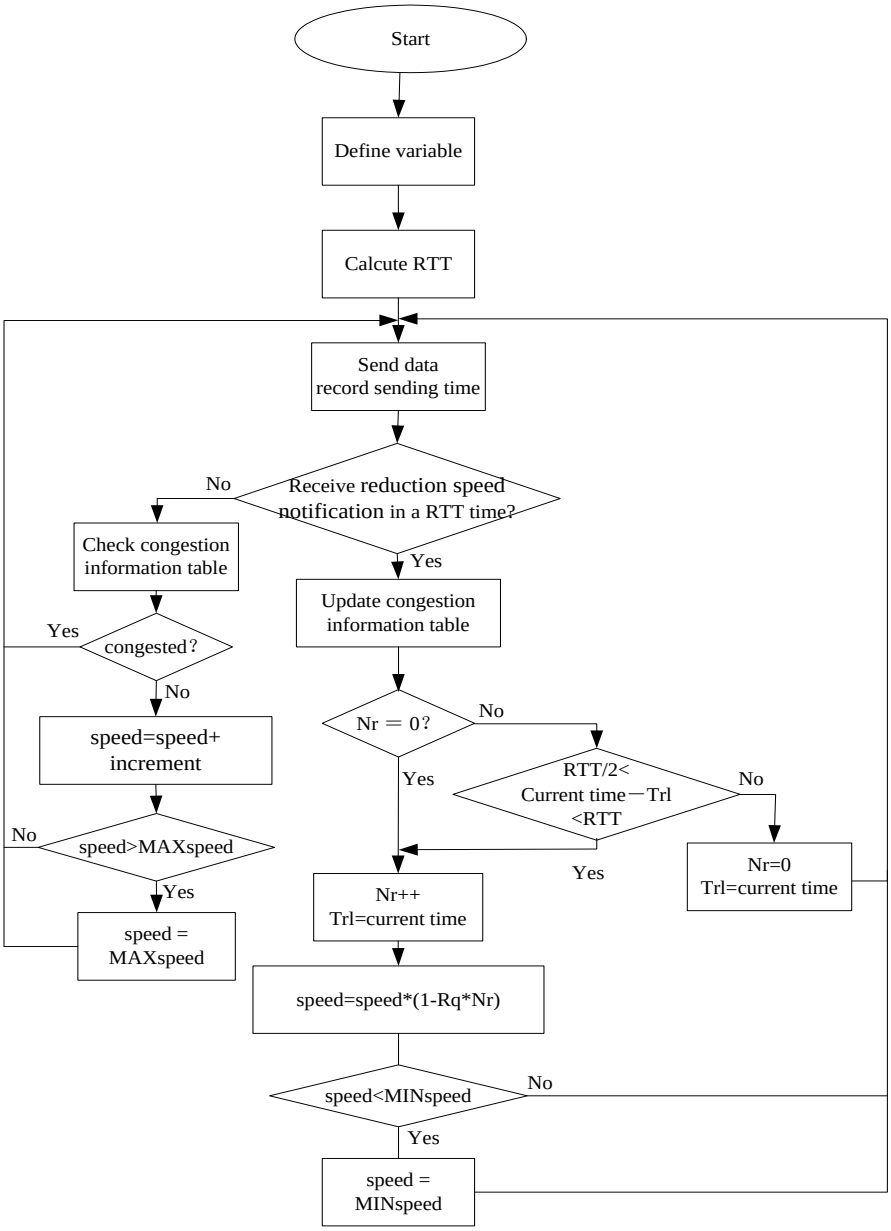

Figure 2. The process of the sending speed adjustment

receiving the reduction speed notification Nr, the average round-trip time RTT, the reduction ratio Rq.

(2) Calculate the average round-trip time RTT.

(3) Send data and record the sending time Ts.

(4) Check whether the terminal node receives the reduction speed notification packet during a RTT time. If yes, turn into the reduction speed process, go to step 8, otherwise, turn into increase speed process, and go on.

(5) Check the congestion information table, judge whether the active node is at the congestion state. If yes, the sending speed should not be changed, continued sending data, and go to step 3.

(6) If not at the congestion state, speed=speed+ increment.

(7) Judge whether the speed is greater than the speed upper limit MAXspeed, if not, continue to send data, go to step 3, otherwise, speed=MAXspeed, continue to send data, go to step 3.

(8) Update the congestion information table.

(9) Judge whether the count of consecutively receiving the reduction speed notification is 0. If not 0, go to step 13.

(10) Nr increases 1, record the current time as Tr1.

(11) Calcute speed=speed*(1-Rq*Nr).

(12) Judge whether speed is less than the speed lower limit, if yes, set the sending speed as the MINspeed, continue to send data, go to step 3, otherwise, the sending speed will not be changed, continue to send data, go to step 3.

(13) Judge whether RTT/2<The current Time-Tr1<RTT. If so, go to step 11, otherwise, set Nr as 0, Tr1 is the current time, go to step 3.

Through the above algorithm, the sending node can know that the transferring nodes get congested, can automatically adjust sending speed which can timely reduce the burden of active nodes, and can effectively relieve the congestion conditions. The sending node can increase speed automatically to ensure the data transmission timely after the active nodes' congestion is relieved.

E. Processing Service Data

According to the different characteristics and request of the different kinds of service data on network resources, different kinds of processing methods are designed for them. In this paper, active network transmits four kinds of service data, such as video data, audio data, file data and message data.

1) Video Data: Video data transmission in network is usually used MPEG format. MPEG format video data has three kinds of frames, the first one is I frame, the second is P frame, and the third is B frame. One graphics group consists of one I frame, several P frames and several B frames. The important characteristics of those frames are shown as follows.

- I frame is the most important in three kinds of frames, because it can be compressed or decompressed correctly not depending on other frames. If I frame is lost, other frames data in the same graphics group such as P frame and B frame can not be compressed or decompressed correctly.
- P frame is more important than B frame. In same graphics group, P frame can be compressed or decompressed after the I fame and P frames whose position are more anterior have been compressed or decompressed correctly. So the former P frame is more important than the one after in same graphics group.
- B frame is less important than any other frame; the importance of B frame data in different position is same.

According to the importance of different kinds of frames in MPEG format video data, we design the rule of discarding pocket as follows. When network node is at the congestion state, it judges the type of video frame in packet firstly, if the type of video frame is I frame, discards the current packet and all the packets after it until next I frame packet reaches. If the type of video frame is P frame or B frame, we only discard the current packet.

When network node discards video packets, it will send indication message immediately to the source node. Receiving the message, the source node will only decrease the speed of transmitting data, not need to resend the packet because video data need real-time processing.

2) Audio Data: Different packets of Audio data are irrelevant, but it needs to be transmitted on real-time. In order to ensure the sound can be played fluency and clarity, we allocate the higher priority to the audio data. When network node discards audio packets, it will only notify the source node to decrease the speed of transmitting data and not need to resend the packet.

3) File Data: File data does not need to be transmitted on real-time, but it need to be transmitted reliably because the file can not run correctly when any part of it is lost, so we allocate the lower priority to file data. When network node discards file data packets, it will notify the source node to decrease the speed of transmitting data and to resend the packet.

4) Message data: Message data does not need to be transmitted on real-time and has little relationship among different packets, but it need to be transmitted reliably, so we allocate the lower priority to message data. When network node is at the congestion state and discards message data packets, it will notify the source node to decrease the speed of transmitting data and to resend the packet.

F. Customization ANEP Protocol

According to the service demand, we customize several option fields of the ANEP protocol which is shown in Table II.

While the terminal equipment is transmitting data to the network, firstly it completes the active packet encapsulation according to the ANEP protocol. The active node receives the active packets and gets active packets' related information through analysing and calculating its related fields, and complements the service data retransmission. The several important fields' settings are shown as follows.

(1) The identification of service data and the congestion control algorithm are allocated automatically. The option field of the service type has four kinds of service code: 1 represents real-time voice; 2 represents ordinary message data; 3 represents real-time video; 4 represents file. After identifying the service type, the active node could automatically allocate the related active congestion algorithm according to the service type as soon as the network congestion happens.

(2) The setting of the service priority. According to the four kinds of service having the different transmission mechanism and the requirements for QoS, The different priority is set to them, which is used for the randomly selecting and discarding in RED algorithm, and is used for ensuring the transmission of special packets. Because the transmission of the video and audio service data is real-time and consecutive, once the data packet is lost, there is little significance for resending, higher priority is set to them. The file service priority is lower. Because the message service data transmits less data each time, and is uneasy to generate lots of data packets' sudden drop, its retransmission and restore is better, the lowest priority is set to it. To avoid the notification packet going missing during transmission and making the algorithm ineffective, the highest priority is allocated for various kinds of notification packets. Security transmission is ensured while the congestion happens.

(3) The mark of the notifying source node. Its total length is 32 bits. The first 16 bits are the length of the current weighted queue, which is used to record the length of the weighted queue of the current active node

TABLE II.
INFORMATION OF CUSTOMIZATION OPTIONAL FIELDS

Option Type	Defination	Option Values
1	Source identifier	The first 32 bits represent mechanism identifier. 1 indicates IPv4 address (32 bits); 2 indicates IPv6 address (128 bits); 3 indicates 802.3 address (48 bits); The last 32 bits represent IP address.
2	Destination dentifier (representing the relayed active node)	The first 32 bits represents mechanism identifier. 1 indicates IPv4 address (32 bits); 2 indicates IPv6 address (128 bits); 3 indicates 802.3 address (48 bits); Tthe last 32 bits represent IP ddress.
3	Integrated checksum	currently unavailable
4	Non-negotiable authentication	currently unavailable
5	Final destination identifier	Its total length is 64 bits.The first 32 bits represent mechanism identifier. 1 indicates IPv4 address (32 bits); 2 indicates IPv6 address (128 bits); 3 indicates 802.3 address (48 bits); The last 32 bits represent IP address.
6	Chose of the congestion algorithm	Its total length is 64 bits. The first 8 bits represent the number of the congestion algorithm. The other 56 bits represent the name of the class which realizes the algorithm. (The max length of the name is 7 letters.)
7	Service type	Its total length is 32 bits. 1 represents real-time voice; 2 represents ordinary message data; 3 represents real-time vedio; 4 represents files
8	Priority	Its total length is 32 bits; the values contain 1, 2, 3, 4 and 5.
9	Notifying source node	Its total length is 32 bits. The first 16 bits are the length of the current weighted queue. The third byte represents message type. 0 represents that it is not the reduction speed resending message; 1 represents the reduction speed resending message; 2 represents detection message. The forth byte represents whether need resending. 0 represents no resending; 1 represents resending.
10	Filename	Its total length is 32 bits. The first 16 bits are the automaticlly increasing file index value, which uniquely determines an object of Class Item, including ID, strDestIP, strFileName. The last 16 bits are the file sequence number seqID.

while the packet arriving. It prevents and avoids the congestion occurrence through the active detection and passive indication mechanism. The third byte represents message type. 0 represents that it is not the reduction speed and resending message; 1 represents the reduction speed and resending message; 2 represents detection message. The forth byte represents whether need resending. 0 represents no resending; 1 represents resending.

G. Design of code server

Active code is stored in the form of .class file in the code library of FTP server. We use the Internet Information Services (IIS) of windows to realize FTP server in which many active network congestion control algorithm is stored. When corresponding active network congestion control algorithm can't be found in active node local, active node sends request to FTP server to download the algorithm active code, then loads the algorithm dynamically using the loadClass method of ClassLoader class.

III. SIMULATION AND PERFORMANCE ANALYSIS OF DIFACC

A. DIFACC Simulation Topology Structure

This paper establishes a simulation experiment system and analyzes the performance of the two algorithms. The DIFACC simulation topology structure is shown in Fig. 3.

T1 and T2 are terminal devices, which can be used to send / receive data in the simulation experiment system. R1, R2, R3 and R4 are active nodes, which simulate route function and transmit data between T1 and T2. In order to

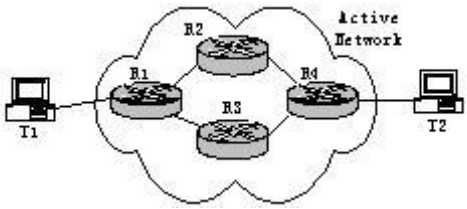

Figure 3. The DIFACC simulation topology structure

compare clearly, FACC and DIFACC algorithms are used in the same experimental environment and data.

Figure 4. Effect of transferring video in DIFACC

B. Analysis of Service Data Transmission Effect

When network transmits video data in a man-made congestion condition, DIFACC algorithm applies service priority and relativity to realize discard the selected packet. Video transmitting effect applied DIFACC algorithm in congestion condition is shown in Fig.4. It shows that the whole effect can be maintained and the image is clarity relatively, however, real-time of image becomes worse and produces dithering. In the same environment, Fig.5 illustrates the transmitting effect applied FACC algorithm, it shows that the image is illegibility and the quality is very poor.

In the same way, when network transmits audio data in a man-made congestion condition, it can be ensured to be transmitted reliably, because DIFACC algorithm allocates higher priority to audio data. When network discards audio packet in congestion condition, the sound is also clarity and continuous despite the real-time becomes poor. However, when network applies for FACC algorithm, the effect becomes worse clearly, the sound is not continuous as before and many audio data is discarded in transmission.

Figure 5. Effect of transferring video in FACC

The active congestion control algorithm applies for decreasing speed and resending mechanism to process file and message data. When network is at congestion states, the congestion can be relieved by decreasing speed of transmitting data. However FACC applies for drop tail algorithm which does not send message which is used for decreasing speed and resending, so it can not relieve congestion independently and lead to massive loss of service data, thereby it can not ensure service data transmit reliably.

B. Performance Analysis

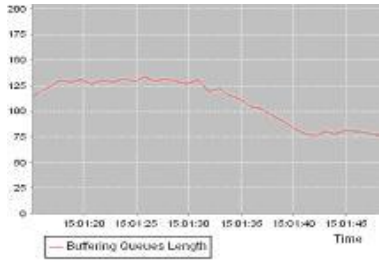

Figure 6. Buffer queue length in FACC

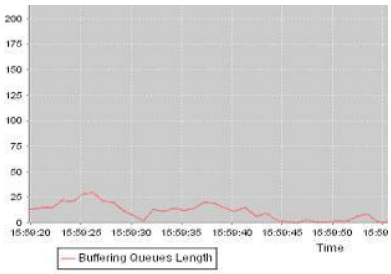

Figure 7. Buffer queue length in DIFACC

As shown in Fig.6 and Fig.7, axis of ordinate in Fig.6 is higher than Fig.7 obviously. It means FACC buffer queue is longer than that of DIFACC. Thus, the buffer queue is often longer in FACC. There is a longer waiting queue in the nodes transmission. In DIFACC, the buffer queue keeps short mostly. Therefore, there is a shorter waiting queue in the nodes transmission so that the processing delay of packets in network is decreased.

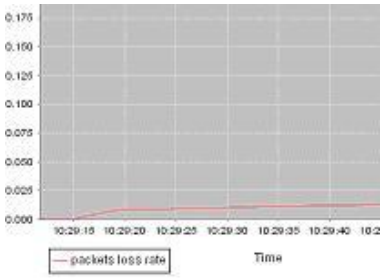

Figure 8. Packets loss rate in DIFACC

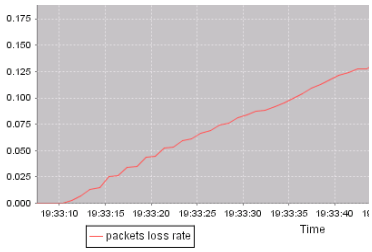

Figure 9. Packets loss rate in FACC

As shown in Fig.8 and Fig.9, packets loss rate curve is stable and raises slowly using FACC algorithm. In the same condition, packets loss rate curve raises faster using DIFACC algorithm.

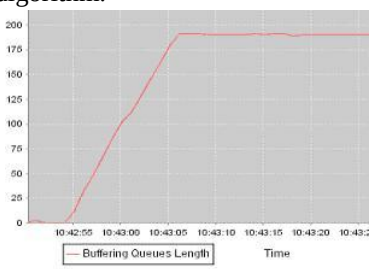

Figure 10. Congestion queue in FACC

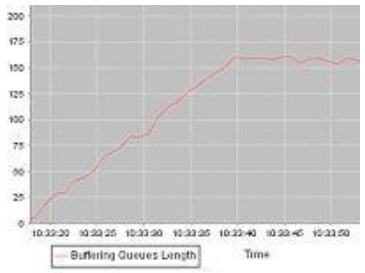

Figure 11. Congestion queue in DIFACC

As shown in Fig.10 and Fig.11, the buffer queue length rises quickly to its maximum using FACC algorithm; network is congested in short time; data packets are massively discarded. In the same condition, using DIFACC algorithm, data packets aren't massively discarded and the buffer queue curve rises slowly in respect to FACC. The curve of DIFACC doesn't reach the maximum of the buffer queue length but trends to stable at the minimum threshold value.

DIFACC introduces the active detection and passive indication preventive mechanism. Through this mechanism, active nodes can know the congestion information in time, actively choose the optimum route to transmit data, predict the congestion information

effectively, reduce the packets loss rate, realize load balance and increase the efficient use of network resources.

Ⅳ. CONCLUSIONS

DIFACC applies different processing methods to different kinds of service data (such as video data, audio data, file data and message data) because of the different characteristics and requests on network resources, adopts the cooperation of the load balanced management algorithm and RED queue management algorithm, and introduces the active detection and the passive indication mechanism. It succeeds in avoiding the algorithm invalidation caused by the indication message loss, promotes the efficient usage of network resources. With the analysis of performance between FACC and DIFACC in simulation experiment system, it is known that DIFACC not only resolves the problems in FACC, but also reduces the packets loss rate, decreases the processing delay of packets and promotes the efficient usage of network resources.

However, when active node starts just now or the network is idle, many active detection messages will be produced, the network resource will be wasted seriously. At the same time, there are also some limitations in processing video data, because it only aims at MPEG format.

ACKNOWLEDGMENT

The authors wish to thank the editor and referees for their careful review and valuable critical comments. This work is supported by the Science Fund of Hebei of China No. 11213522D.

REFERENCES

[1] Jingyang Wang, Xiaohong Wang et al, "The Research of Active Network Congestion Control Algorithm," *Proceedings of the WiCom2007*, September 2007.

[2] Wang Bin, Liu Zeng-Ji et al, "Forward active networks congestion control algorithm and its performance analysis," *Acta Electronica Sinica*, Vol 29, No. 4, April. 2001, pp. 483-486.

[3] Carlo Tarantola, "Dynamic Active Networks Services," *Proceedings of the 2004 IEEE International Conference on Mobile Data Management*, pp. 46-47, June 2004.

[4] Xu Jiali, and Liu Suqin, "Discussing and Implementing Method of Active Network Architecture," *Control & Automation*, Vol 11, No.2, pp. 232-245, Nov. 2004.

[5] LA Grieco and S. Mascolo, "TCP Westwood and Easy RED to Improve Fairness in High-Speed Networks," *Seventh International Workshop on Protocols For High-Speed Networks (PfHSN'2002)*, Berlin, Germany, pp. 130–146, April 2002.

[6] K.Psounis, "Active Networks; Applications, Security, Safety and Architectures," *IEEE Communications Surveys*, Vol 2, No. 1, 1999.

[7] Zhang Ke-ping, and Tian Liao, and Li Zeng-zhi, "A New Queue Management Algorithm with Priority and Self-Adaptation," *Acta Electronica Sinica*, Vol 6, No. 4, pp. 324-328, July. 2004.

[8] Xu Yongbo, and Wang Xingren, "The Research and Implementation of Time Management Services in Distributed Simulation System," *Proceedings of Asian Conference on System Simulation and ScientificSimulation Conference*, Shanghai, 2002.

[9] Appel A W, "Foundation Proof-Carrying Code," *IEEE Communication Magnize*, 2001.

[10] MAXEMCHUKNF, and LOWSH, "Active Routing," *IEEE Journal on Selected Areas in Communications*, 2001, pp. 552-565.

[11] S.Murphy, "Security Architecture for ActiveNets," *AN Security Working group*, July 15, 1998.

[12] A. B. Kulkarni and S. F. Bush, "Active network management and Kolmogorov complexity," *IEEE OPENARCH 2001*, Anchorage, AK, Apr. 2001.

Jingyang Wang, Associate Professor, born in 1971. He received the B.Eng. degree in computer software from Lanzhou University, China, in 1995. He received the M.Sc. degree in software engineering from Beijing University of Technology, China, in 2007. His main research areas include active network, transmition control, and distributed computing.

Min Huang, born in 1979. He received the B.Eng. degree in automatic control from Hebei University of Science and Technology, China, in 2000. He received the M.Sc. degree in computer science from Beijing Institute of Technology, China, in 2003. His main research interests include network and communication, system modeling and identification, image processing and distributed computing.

Haiyao Wang, born in 1976. She received the B.Eng. degree in machinery design and manufacture from HeFei University of Technology, China, in 1998. She received the M.Sc. degree in industrial engineering from HeFei University of Technology, China, in 2009. Her main research interests include algorithm design and industrial control.

Liwei Guo, Professor, born in 1956. He received the M.Sc. degree in automatic control from Harbin University of Science and Technology, China, in 1988. His main research interests include network and communication, system modeling and automatic control.

Wanzhen Zhou, Professor, born in 1966. He received the B.Eng. degree in applied mathematics from Harbin University of Technology, China, in 1988. He received the M.Sc. degree in computer science from Harbin University of Technology, China, in 1992. His main research interests include network and database, system modeling and image processing.