The Effects of Different Congestion Management Algorithms over Voip Performance

Szabolcs Szilágyi
Faculty of Informatics
University of Debrecen
Debrecen, Hungary

***Abstract*—This paper presents one of the features of DS (Differentiated Services) architecture, namely the queuing or congestion management. Packets can be placed into separate buffer queues, on the basis of the DS value. Several forwarding policies can be used to favor high priority packets in different ways. The major reason for queuing is that the router must hold the packet in its memory while the outgoing interface is busy with sending another packet. The main goal is to compare the performance of the following queuing mechanisms using a laboratory environment: FIFO (First-In First-Out), CQ (Custom Queuing), PQ (Priority Queuing), WFQ (Weighted Fair Queuing), CBWFQ (Class Based Weighted Fair Queuing) and LLQ (Low Latency Queuing). The research is empirical and qualitative, the results are useful both in infocommunication and in education.**

Keywords*—CBWFQ; congestion; CQ; FIFO; LLQ; Pagent; PQ; queuing; WFQ

I. Introduction

At the beginning computer networks were designed mainly for data transfer such as FTP and email, where delay was considered to be unimportant. In most cases the delivery service was effective, and the TCP protocol dealt with data losses. As the multimedia applications became popular (voice transfer, video conferences), separate telephone and video communication networks were set up (see Fig. 1). Nowadays, office and company networks are transformed into one converged network (see Fig. 2), in which the same network infrastructure is used to ensure all the requested services [1].

Although converged networks have many advantages, there are some disadvantages too, namely the competition for network resources (buffers of routers), which leads to congestion. Delay in delivering the packets, jitter, loss of packets are consequences of congestion. Different applications show different sensitivity to these issues. For example, FTP is not impacted by delay and jitter, whereas the multimedia applications (e.g. interactive voice) are very sensitive to them and the loss of packets too [2]. Quality of Service (QoS) was introduced to handle this problem, and it is able to provide different priority to different applications, users, or data flows, or to guarantee a certain level of performance to a data flow [3], [4].

This research was realized in the frames of TÁMOP 4.2.4. A/2-11-1-2012-0001 „National Excellence Program – Elaborating and operating an inland student and researcher personal support system”. The project was subsidized by the European Union and co-financed by the European Social Fund.

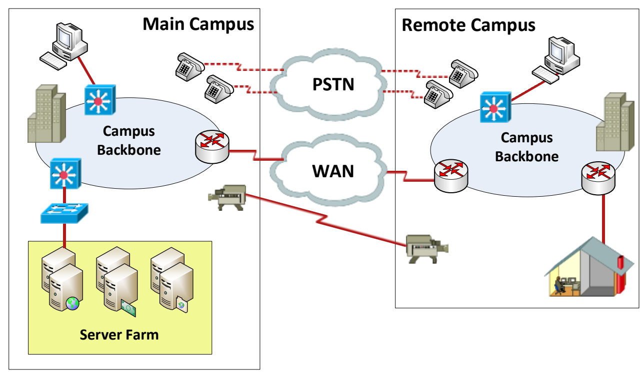

Fig. 1. A classical non-converged network

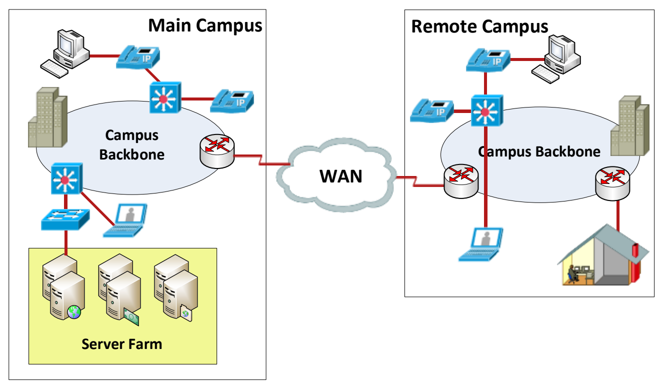

Fig. 2. A converged infocommunication network

In accordance with the QoS requirements and recommendations for the interactive voice traffic packet loss should be no more than 1%, one-way latency should be not exceed 150ms and the average one-way jitter should be targeted at less than 30ms [5].

In the IP header there are some fields which can be used to make distinction between the packets of different applications, for example the Type of Service (ToS) field [6]. Different technics are used for congestion management (PQ, CQ, WFQ, CBWFQ and LLQ). Congestion avoidance (WRED), traffic shaping and traffic policing are also used by the QoS technology in order to control data traffic [7]. This article focuses on the most important component, namely the congestion management.

Our purpose is to analyze and evaluate the efficiency of these congestion management algorithms using a laboratory environment. This paper examines the following methods: FIFO, PQ, CQ, CBWFQ, WFQ and LLQ. It is important to note that these algorithms have real effect only in the case of congestion.

The network topology for the performance evaluation is identical to the one used in former articles (see e.g. [8]-[10]). This paper continues to study the queuing technologies for congestion management. In [8] and [11] the authors considered three algorithms: FIFO, PQ and WFQ. The conclusion was that WFQ is the most efficient for multimedia applications. In addition to these three new algorithms were investigated: CQ, CBWFQ and LLQ. The main result of this paper is that for multimedia applications (mainly for voice transfer) LLQ is more efficient then WFQ.

The detailed description of the algorithms has been discussed in several papers already (see e.g. [12]-[14]).

II. The Measurement Environment

The measurement environment network topology is shown on Fig. 3, which was built at the network laboratory of the Faculty of Informatics, University of Debrecen.

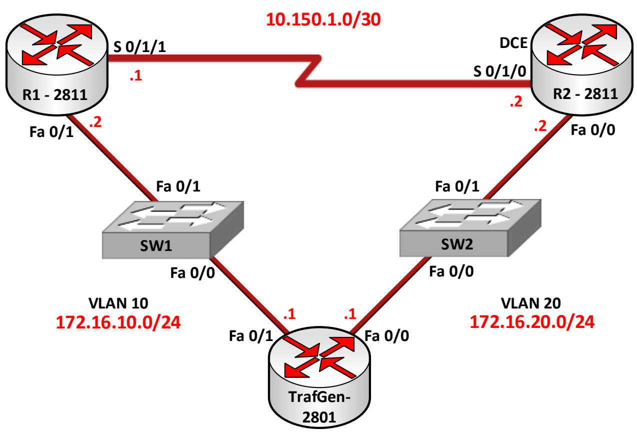

Fig. 3. The measurement environment

The measurement environment consists of three routers (two Cisco 2811 type routers and one Cisco 2801 router) and two switches. The routers are connected with a point-to-point link, having the speed of 128000 cycles per second. The rest of the network devices are connected with 10 Mbps Ethernet links. The part between the two routers is actually a narrow cross-section where congestion can happen. For this reason the congestion management algorithms are activated in this area (see [15]-[16]).

The *Cisco IOS 12.4* operating system was running on the *R1* and *R2* routers, represented on Fig. 3. The *TrafGen* router was responsible for the functioning of the communication endpoints. This was used to generate the traffic, and the operating system run on *TrafGen* was *c2801-tpgen+ipbase-mz.PAGENT.4.3.0* [17], which enabled the traffic generation, attached timestamps to the outgoing packages, and performed the statistical analysis based on the rate of incoming packets.

In order to distinguish between the generated and incoming traffic, two Cisco 2960 switches were used (*SW1*, *SW2*). These created two Virtual LANs, namely *VLAN 10* and *VLAN 20* [18]. A serial connection was created between *R1* and *R2*, which simulated a slow WAN. Three types of traffic were generated, similarly to the previous papers: an FTP, Video and VoIP traffics. In the next section the traffic generation code is shown used by *TrafGen* router.

A. *The traffic generation*

The following code was used for traffic generation [19]:

```
wait-after-stop 1 ! Waiting time (sec)
!FTP traffic generation
fastethernet0/1 ! The TrafGen router output interface name
add tcp ! Adding a traffic stream (TCP)
datalink ios-dependent fastethernet0/1 ! The output interface name
l2-arp-for 172.16.10.2 ! Layer 2 ARP message to default gateway
l3-src 172.16.10.1 ! Layer 3 source IP address
l3-dest 172.16.20.1 ! Layer 3 destination IP address
l3-tos 0x00 ! Layer 3 packet header ToS byte value
l4-src 21 ! Transport layer source port number
l4-dest 21 ! Transport layer destination port number
name FTP ! Name of the generated traffic
rate 20 ! Setting the packet send rate
length 1434 ! Setting packet length (Bytes)
delayed-start 0 ! Delaying start of packet generation (sec)
send 206 ! Sending packets
fastethernet0/0 ios-dependent capture
! The TrafGen router input interface name
!VIDEO traffic generation
fastethernet0/1
add tcp
datalink ios-dependent fastethernet0/1
l2-arp-for 172.16.10.2
l3-src 172.16.10.1
l3-dest 172.16.20.1
l3-tos 0x22
l4-src 4249
l4-dest 1720
name VIDEO
rate 50
length 1500
burst on ! Sending traffic stream in bursts
burst duration off 1000 to 2000
burst duration on 1000 to 3000
delayed-start 0
send 333
fastethernet0/0 ios-dependent capture
!VOICE traffic generation
fastethernet0/1
add udp
datalink ios-dependent fastethernet0/1
l2-arp-for 172.16.10.2
l3-src 172.16.10.1
l3-dest 172.16.20.1
```

```
l3-tos 0x2E
l4-src 44899
l4-dest 5060
name VOICE
rate 50
length 150
delayed-start 0
send 513
fastethernet0/0 ios-dependent capture
```

B. The implementation of congestion management algorithms

The part between the R1 and R2 routers is actually a narrow cross-section where congestion can happen. For this reason the congestion management algorithms were activated in this area (between the *R1*' S 0/1/1 and *R2*' S 0/1/0 interfaces). These codes (for FIFO, PQ, CQ, WFQ, CBWFQ and LLQ) can be found in *APPENDIX*.

III. MEASUREMENT RESULTS

As in the previous works [9]-[10], the length of the measurement time was 5 minutes in each case. The measurements were recorded in every second. Easy to observe in the traffic generation code, that the generated voice traffic average was 513 packets per second. As in previous articles (see e.g. [8]-[11]) the following areas were examined: packet loss, end-to-end delay and jitter (delay variation).

Concerning the packet loss of voice packets (see Fig. 4) LLQ and PQ algorithms have proven to be most effective, followed by the CBWFQ. It can be observed that while the previous works based on simulation results concluded that WFQ was the best congestion management algorithm, our measurement results showed, that the WFQ performance was behind the performance of LLQ, PQ and CBWFQ. The CQ has the next poor algorithm performance, while the least efficient queuing scheduler was the FIFO.

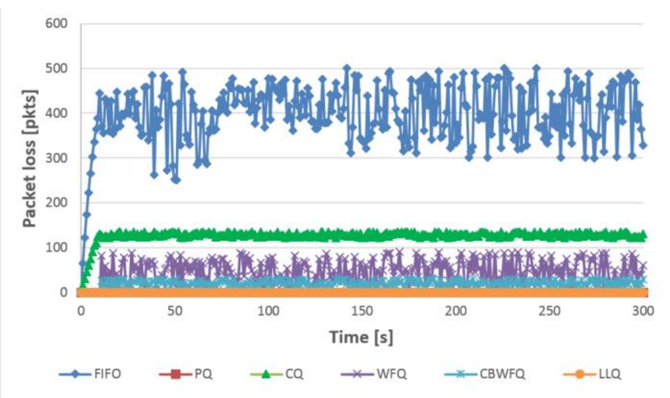

Fig. 4. VoIP packet loss

Fig. 5 shows the same content as Fig. 4, except that the former does not include the efficiency representation of the two least efficient congestion management algorithm. Thus it is prominently observable the difference of performance of PQ, WFQ, CBWFQ and LLQ in packet loss. Easy to see, that in the case of LLQ and PQ there was no packet loss due to the absolute priority queue, in which the real-time voice was classified. Subsequently, CBWFQ performance was the most effective, and finally the WFQ's.

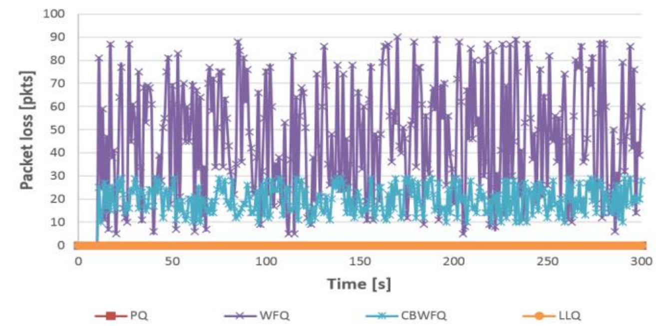

Fig. 5. VoIP packet loss for PQ, WFQ CBWFQ and LLQ

With respect of voice packet delay (see Fig. 6 and Fig. 7), LLQ and PQ algorithms managed to squeeze those values below 100 ms, while the CQ has under 255ms, which already exceeds the threshold set by the QoS requirement. It is clear that in the case of WFQ and CBWFQ the delay is a little more than 1 second, while in the case of FIFO than can reach up to 8.5 seconds, which are unacceptable values provided by the QoS requirements.

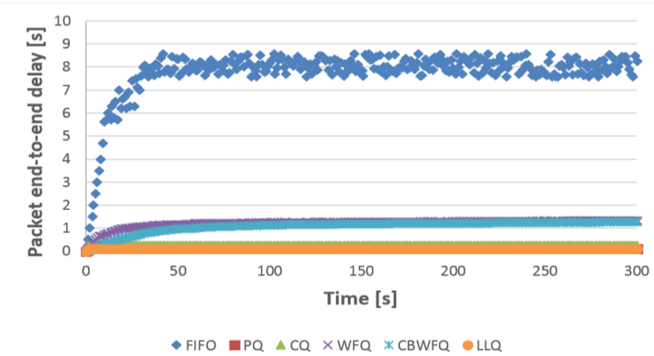

Fig. 6. VoIP traffic delay

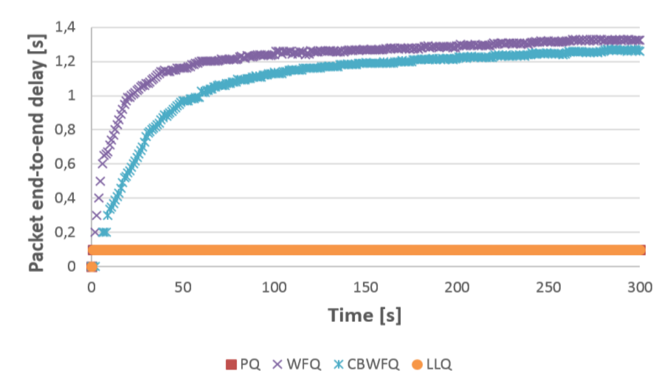

Fig. 7. VoIP traffic delay for PQ, WFQ, CBWFQ and LLQ

As for the delay variation (jitter) (see Fig. 8 and Fig. 9) the LLQ, PQ and CQ has managed to keep the measured values below 30ms. Subsequently, the WFQ and CBWFQ ensured around 150ms and 210 ms jitter, while the FIFO has finally managed to stabilize its delay variation around 1 second. It should be noted that in terms of jitter PQ and LLQ congestion management algorithms managed to meet under the requirements of the QoS threshold requirement.

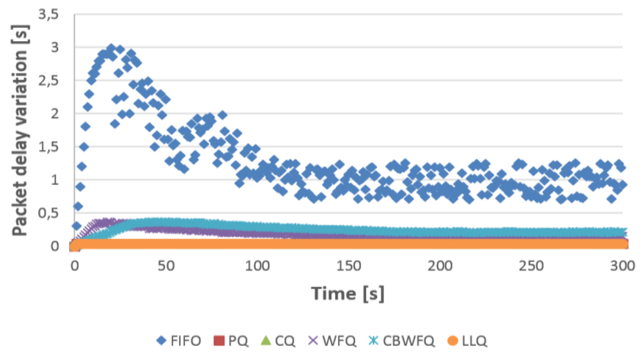

Fig. 8. VoIP jitter

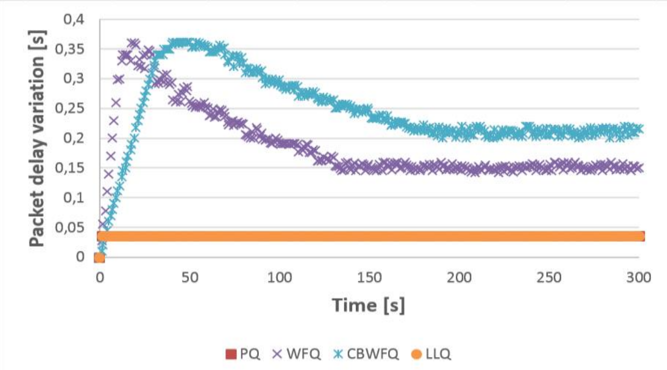

Fig. 9. VoIP jitter for PQ, WFQ, CBWFQ and LLQ

IV. CONCLUSION

This paper compares the performance of the main congestion management algorithms based on a measurement environment. The laboratory environment was implemented at the Faculty of Informatics University of Debrecen. In all cases the measurement result shows that the FIFO scheduling principle is the most inconvenient algorithm for handling of interactive voice packets in case of congestion. In the case of voice transmission the PQ and the LLQ algorithms were the two most appropriate algorithms, in terms of packet loss rate, end-to-end delay and jitter. Using these algorithms no packets suffered packet loss. However, knowing the principle of the PQ, namely that it serves the maximum priority queue, but produce packet starvation for other tree queues, based on the literature and on the measurement results the conclusion is, that for the interactive real-time voice traffic, taking all in consideration, the LLQ congestion management algorithm is most appropriate. Further research topic is to support the results and test the algorithms presented in the current article by mathematical modeling.

APPENDIX

These router configuration settings were used for implementing the congestion management algorithms:

FIFO

```
int s0/1/1
no fair-queue
end
```

PQ

```
access-list 101 permit tcp 172.16.10.0 0.0.0.255
172.16.20.0 0.0.0.255 eq 21
access-list 102 permit tcp 172.16.10.0 0.0.0.255
172.16.20.0 0.0.0.255 eq 1720
access-list 103 permit udp 172.16.10.0 0.0.0.255
172.16.20.0 0.0.0.255 eq 5060
priority-list 1 protocol ip high list 103
priority-list 1 protocol ip medium list 102
priority-list 1 protocol ip normal list 101
priority-list 1 default low
priority-list 1 queue-limit 20 40 60 80
int s0/1/1
priority-group 1
end
```

CQ

```
access-list 101 permit tcp 172.16.10.0 0.0.0.255
172.16.20.0 0.0.0.255 eq 21
access-list 102 permit tcp 172.16.10.0 0.0.0.255
172.16.20.0 0.0.0.255 eq 1720
access-list 103 permit udp 172.16.10.0 0.0.0.255
172.16.20.0 0.0.0.255 eq 5060
queue-list 1 protocol ip 2 list 103
queue-list 1 protocol ip 3 list 102
queue-list 1 protocol ip 4 list 101
queue-list 1 default 1
queue-list 1 queue 1 limit 4
queue-list 1 queue 2 limit 10
queue-list 1 queue 3 limit 10
queue-list 1 queue 4 limit 4
int s0/1/1
custom-queue-list 1
end
```

WFQ

```
int s0/1/1
fair-queue
end
```

CBWFQ

```
access-list 101 permit tcp 172.16.10.0 0.0.0.255
172.16.20.0 0.0.0.255 eq 21
access-list 102 permit tcp 172.16.10.0 0.0.0.255
172.16.20.0 0.0.0.255 eq 1720
access-list 103 permit udp 172.16.10.0 0.0.0.255
172.16.20.0 0.0.0.255 eq 5060
class-map VOICE
match access-group 103
exit
class-map VIDEO
match access-group 102
exit
class-map FTP
match access-group 101
exit
policy-map R1-Serial
class VOICE
bandwidth percent 30
exit
class VIDEO
bandwidth percent 30
exit
class FTP
bandwidth percent 10
exit
class class-default
bandwidth percent 5
exit
exit
int s0/1/1
no fair-queue
service-policy output R1-Serial
```

```
end
```

LLQ

```
access-list 101 permit tcp 172.16.10.0 0.0.0.255
172.16.20.0 0.0.0.255 eq 21
access-list 102 permit tcp 172.16.10.0 0.0.0.255
172.16.20.0 0.0.0.255 eq 1720
access-list 103 permit udp 172.16.10.0 0.0.0.255
172.16.20.0 0.0.0.255 eq 5060
class-map VOICE
match access-group 103
exit
class-map VIDEO
match access-group 102
exit
class-map FTP
match access-group 101
exit
policy-map R1-Serial
class VOICE
priority 384
exit
class VIDEO
bandwidth percent 30
exit
class FTP
bandwidth percent 10
exit
class class-default
bandwidth percent 5
exit
exit
int s0/1/1
no fair-queue
service-policy output R1-Serial
end
```

REFERENCES

[1] QOS, "Implementing Cisco Quality of Service, Student Guide", Ver. 2.2, vol. 2, Cisco Systems Inc., 2006.

[2] A. S. Tanenbaum and D. J. Wetherall, "Computer Networks", 5th Edition, Pretince Hall, ISBN-13: 978-0132126953, 2013.

[3] T. Svensson and A. Popescu, "Development of laboratory exercices based on OPNET Modeler", Blekinge Institute of Technology, Department of Telecommunications and Signal Processing, 2003.

[4] C. Mancas and M. Mocanu, "QoS Optimization in Congested Multimedia Networks," Proceedings of the 36th TSP Conference, pp. 38-42, 2013.

[5] "Cisco IOS Quality of Service Solutions Configuration Guide", Release 12.4T, Cisco Systems Inc., 2008.

[6] W. Odom, J. Geier and N. Mehta, "CCIE Routing and Switching Official Exam Certification Guide", 2nd Edition, Cisco Press, 2006.

[7] A. S. Ranjbar, „CCNP ONT Official Exam Certification Guide", Cisco Press, 2007.

[8] M. M. G. Rashed and M. Kabir, "A Comparative Study off Different Queuing Techniques in VoIP, Video Conferencing and FTP," Daffodil International University Journal of Science and Technology, vol. 5, no. 1, 2010.

[9] S. Szilágyi, "The Effects of Different Queuing Techniques over VoIP Performance: A Simulation Approach," in Abstracts & Pre-Proceedings of the 9th International Conference on Applied Mathematics, Baia Mare, Romania, 2013.

[10] S. Szilágyi, "Analysis of the Algorithms for Congestion Management in Computer Networks," Carpathian Journal of Electronic and Computer Engineering, vol. 6, no. 1, pp. 3-7, 2013.

[11] S. Akhtar, E. Ahmed, A. k. Saka and K. S. Arefin, "Performance Analysis of Integrated Service over Differentiated Service for Next Generation Internet," JCIT, vol. 1, no. 1, 2010.

[12] S. Szilágyi and B. Almási, "A Review of Congestion Management Algorithms on Cisco Routers," Journal of Computer Science and Control Systems, vol. 5, no. 1, 2012.

[13] A. Kuki, T. Bérczes, B. Almási, J. Sztrik, "A Queing Model to Study the Effect of Network Service Breakdown in a CogInfoCom System", Cognitive Infocommunication (CogInfoCom) 2013, 4th IEEE International Conference on Cognitive Infocommunications, IEEE Publisher, pp. 205-2010, ISBN: 978-1-4799-1543-9, 2013.

[14] B. Almási, G. Bolch, D. Tutsch, "Stochastic Modeling of Multistage Interconnection Networks with MOSEL", Journal of Mathematical Sciences, vol. 121, no. 5, 2004.

[15] L. L. Peterson and B. S. Davie, "Computer Networks: A System Approach - Network Simulation Experiments Manual", 3rd Edition, Elsevier Inc., 2011.

[16] A. S. Sethi and V. Y. Hnatyshin, "The Practical User Guide for Computer Network Simulation", NW: CRC Press Taylor & Francis Group, 2012.

[17] A. Cisco Networking, „Pagent IOS Tutorial," Cisco Systems Inc., 2009.

[18] A. Cisco, "CCNP: Optimizing Converged Networks v5.0 - Instructor Lab Manual," Cisco Systems, Inc., 2007.

[19] C. Cisco, „TGN (Traffic GeNerator) User Manual," Cisco Systems Inc., 2007.