Rapid Synchronization of RTP Multicast Sessions

Zlatan Begić† and Mladen Kos††

† BH Telecom Joint Stock Company Sarajevo, Directorate Tuzla, Tuzla, Bosnia and Herzegovina
†† Faculty of Electrical Engineering and Computing, University of Zagreb, Zagreb, Croatia

Summary
Quality of Service (QoS) is an important, and admittedly overloaded, concept for emerging services based on the Real-Time Transport Protocol (RTP) to deliver multimedia content. In the context of the Internet Protocol Television (IPTV) services, the synchronization of multimedia components plays an important role. In order to synchronize audio and video components of multimedia content, RTP timestamps have to be related. Periodic Real-Time Transport Control Protocol (RTCP) Sender Report (SR) packets, which associate the RTP timestamps in the stream with a common clock at the transmitter, are associated with each stream. Thus, the frequency and the timing of the RTCP SR packets often contribute to the delay before audio and video are rendered, not just to their synchronization, because many clients will not render anything before the synchronization has been established. Whereas the synchronization delay is problematic in case of network congestion, the retransmission technique is used for the RTP receiver to request the rapid synchronization of RTP multicast sessions from the retransmission server (RS). This paper investigates the concept of isolation of the RTCP from the RTP retransmission packets by assigning them to higher priority class which ensures the synchronization delay of the multicast multimedia sessions sufficient for lip-synchronization without excessive delay regardless of network congestion, with accurate statistics measured by RTCP. A primary use case for this concept is to reduce the channel-change times in IPTV networks where compressed video streams are multicast in different Source Specific Multicast (SSM) sessions and viewers randomly join these sessions.
Key words:
DiffServ, multicast, retransmission, RTP, RTCP, synchronization

1. Introduction

When using RTP to deliver multimedia content, it is often necessary to synchronize playout of the audio and video components of a presentation. This is achieved by using the information contained in the RTCP SR packets. These are sent periodically, and the components of a multimedia session cannot be synchronized until sufficient RTCP SR packets have been received for each RTP flow to allow the receiver to establish mappings between the media clock used for each RTP flow, and the common (Network Time Protocol (NTP) format) reference clock used to establish synchronization. RTP flows are identified by means of the canonical end-point identifier (CNAME) item included in the RTCP Source Description (SDES) packets generated by the sender or signaled out of band [1]. According to [1], in the absence of any packet loss, RTCP SR intervals sufficient for lip-synchronization of multicast RTP sessions without excessive delay are well defined. Recently [2], concern has been expressed that synchronization delay is problematic in case of network congestion.

In order to overcome problems concerning synchronization delay, prior proposals [3] were based on idea to isolate RTCP SR packets from the RTP data stream by assigning them to higher priority class in Differentiated Services (DiffServ) architecture which ensures the average reduced minimum RTCP SR interval according to IETF RFC 3550, regardless of network congestion. This is feasible when it is not possible to offer an upstream channel of RTCP receiver reports (RR). Accordingly, since RTP and RTCP are sent using different ports, any flow classification based upon port number that leads to a differentiation between RTP and RTCP flows could disrupt the statistics because RTCP flows (SR in conjunction with RR) are used to measure, infer and convey information about the performance of an RTP media stream.

In this paper, a proposal is given to assign a higher priority to RTCP than to RTP retransmission packets by using the DiffServ QoS techniques. Accordingly, the RTCP retransmission packets with a higher priority are less likely to be lost than the packets with a lower priority. Since RS sends a relatively small number of RTP retransmission packets before receiver starts to receive the primary multicast stream, statistics measured by RTCP cannot be disrupted [4].

Extensive simulations are performed using the Network Simulator version 2 (ns-2) to determine if intentional prioritization of RTCP over RTP retransmission packets can guarantee the average reduced minimum RTCP SR interval according to IETF RFC 3550, sufficient for inter-media lip-synchronization of the RTP multicast sessions regardless of network congestion without disruption of statistics measured by RTCP. This is the null hypothesis (H0). Statistical calculations were carried out using Analyse-it add-in software for Microsoft Excel.

Manuscript received September 5, 2010.
Manuscript revised September 20, 2010.

Section II describes the retransmission method for rapid synchronization of RTP multicast sessions. Section III introduces a simulation environment, a used simulator, simulation results and statistical analysis. Section IV concludes this paper.

2. RTP Retransmission Method

The difference between the time an RTP receiver joins the multicast multimedia session and the time when visual lip movements of a speaker match the sound of the spoken words is referred to as lip-synchronization delay or simply synchronization delay. The synchronization delay may not be the same for different receivers. It usually varies depending on the join time, length of the RTCP SR interval, size of the synchronization information as well as the application and transport properties. The varying nature of the synchronization delay adversely affects the receivers that frequently switch among multicast sessions.

In this section, a retransmission method for rapid synchronization of RTP multicast sessions that uses the fundamental tools offered by the existing RTP and RTCP protocols [1] is described. In this method, either the multicast source (or the distribution source) retains RTP packets for a period after transmission, or an intermediary network element (RS) joins the multicast session and continuously caches RTP packets as they are sent in the session and acts as a feedback target [5] for the session.

When an RTP receiver wishes to join the same multicast session it sends a request to the feedback target for the session and asks for the RTP retransmission session. The RS starts a new unicast RTP retransmission session and sends RTP retransmission packets to the RTP receiver over that session. If there is spare bandwidth, the RS may burst RTP retransmission packets faster than their natural rate. As soon as the receiver acquires the synchronization information, it can join the multicast session and start processing the multicast data. A simplified network diagram showing this method through an intermediary network element is depicted in Fig. 1.

The described method potentially reduces the synchronization delay [6]. A principle design goal in this solution is to use the existing tools in the RTP/RTCP protocol family. This improves the versatility of the existing implementations, and promotes faster deployment and better interoperability. To this effect, the unicast retransmission support of RTP [7] and the capabilities of RTCP are recommended to handle the signaling needed to accomplish the acquisition. However, it is beyond the scope of this paper.

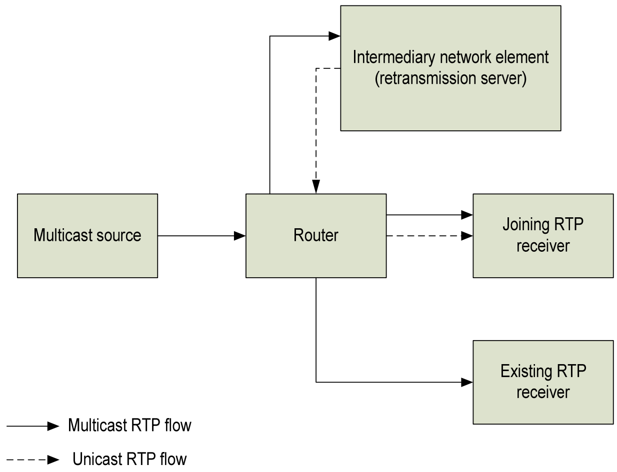

Fig. 1 RTP retransmission method with an intermediary network element.

3. Model Analysis and Simulation Results

The main objective of the following simulation study is to provide recommendations concerning how to implement an adequate RTP/RTCP packet classification scheme based on the DiffServ QoS techniques in order to avoid negative effects of network congestion on synchronization of multicast RTP sessions. Accordingly, the average reduced minimum RTCP SR interval according to IETF RFC 3550, sufficient for inter-media lip-synchronization without excessive delay regardless of network congestion, will be guaranteed.

3.1 Simulation Setup and Protocols

The following simulations are performed using Network Simulator version 2 (ns-2) with adaptations concerning RTP/RTCP protocols [8].

The default implementation of RTP/RTCP in ns-2 is very poor and it is not working according to [1]. In order to achieve the expected timing rules, certain changes are applied in session-rtp.tcl as presented in Fig. 2. The minimum RTCP interval is set to 5 seconds, the fraction of the session bandwidth added for RTCP is fixed at 5%, the timing rules are updated according to [1] and the randomization option is turned on.

The simulations are based on common network topology consisting of eight routers (A, B, C, D, E, F, G, H) with three user domains (D1, D2, D3) attached to them. There are also four servers (A/V streaming server, FTP server, VoD server, RS) attached to routers E, F, G and H. The network topology is shown in Fig. 3.

```
# set minimum interval to 5 seconds
set min_rpt_time_ 5

# update fraction of the session bandwidth added
# for RTCP to be fixed at 5%
set inv_sender_bw_fraction_ [expr 1. /
($sender_bw_fraction * $session_bw_fraction_)]
set inv_rcvr_bw_fraction_ [expr 1. /
($rcvr_bw_fraction * $session_bw_fraction_)]

# update timing rules according to IETF RFC 3550
RTCPTimer instproc adapt { nsrc nrr we_sent } {
mvar inv_bw_ avg_size_ min_rpt_time_
mvar inv_sender_bw_fraction_
inv_rcvr_bw_fraction_
set ibw $inv_bw_
set rint [expr 8*$avg_size_ * $nsrc * $ibw]
if { $rint < $min_rpt_time_ } {
 set rint $min_rpt_time_
}
if { $nrr > 0 } {
 if { $we_sent } {
    set nsrc $nrr
    set rint [expr 360000 * $inv_bw_]
  } else {
    set ibw [expr $ibw * $inv_rcvr_bw_fraction_]
    incr nsrc -$nrr
    set rint [expr 8*$avg_size_ * $nsrc * $ibw]
    if { $rint < $min_rpt_time_ } {
    set rint $min_rpt_time_
    }
  }
}
mvar session_
$session_ report-interval $rint
}

# apply randomization
```

Fig. 2 Changes applied in session-rtp.tcl.

Simulation time is session bandwidth dependent which is sufficiently long to include enough packets of population of interest (RTCP SR packets), and sufficiently short to address the practical aspects of measurement [9]. Simulation time of 200 seconds meets these requirements for 4 Mbps session rate.

Network simulations are carried out by keeping all links except bottleneck links (C→D1,D1→D2,D→D3) constant. Bottleneck links are dimensioned by gradually reducing their capacities to achieve link load of 50%, 80%, 90%, 100%, 110%, 120%, 150%, 200% and 300% (Table 1).

Background traffic is generated by Pareto, Constant Bit Rate (CBR) and File Transfer Protocol (FTP) traffic generators. FTP traffic generators are attached to Transport Control Protocol (TCP), Pareto generators are attached to User Datagram Protocol (UDP) whereas CBR generators are attached to RTP and UDP agents. The traffic pattern choice is driven by the rapid emergence of next generation multimedia services and based on the assumption of video services growing trend [3], [4], [10], [11].

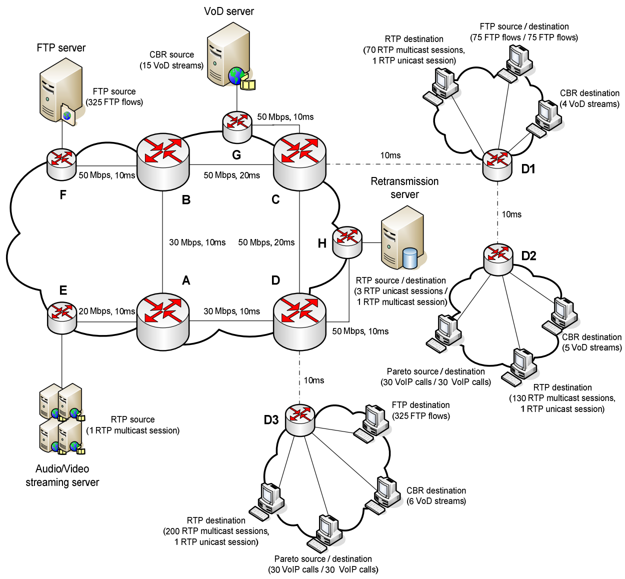

Fig. 3 Network simulation topology.

Pareto generator is chosen to simulate traffic of Voice over Internet Protocol (VoIP) application. It generates 160 byte packets at 100 kbps with burst time of 2 seconds and idle time of 0.5 second.

Video on Demand (VoD) service is simulated with CBR generator attached to UDP agent with packet sizes of 1024 bytes at generation rate of 1024 kbps.

CBR generator attached to RTP agent is used to simulate RTP multicast multimedia component with packets of 512 bytes at 4 Mbps traffic rate with 1 sender and 400 receivers. RTCP timing rules are in compliance with RTCPTimer class.

Unicast RTP retransmission sessions are simulated with CBR generator attached to RTP agent as well. It generates packets of 512 bytes at 4 Mbps. RTCP retransmission packet interval is set to 360 divided by the session bandwidth in kilobits per second [1].

Table 1: Bottleneck link capacities for specified link load

Link load (%)	Bottleneck link capacity (Mbps)		
	C→D1	D1→D2	D→D3
50	51.40	28.00	38.00
80	32.13	17.50	23.75
90	28.56	15.56	21.11
100	25.70	14.00	19.00
110	23.36	12.73	17.27
120	21.42	11.67	15.83
150	17.13	9.33	12.67
200	12.85	7.00	9.50
300	8.57	4.67	6.33

Classification of the retransmission RTP/RTCP packets based on a port number is accomplished by creating two virtual servers and their associated routers (HRTP and HRTCP) to retransmit RTP and RTCP packets from the different sources. Accordingly, expected simulation results are not impaired.

All nodes are configured to contain multicast protocol agents. The time duration for which a prune states are active is configured through the DM class variable, PruneTimeout. It is set to be equal to the simulation time since the number of group members is constant. Additional effort to set up the Protocol Independent Multicast - Source Specific Multicast (PIM-SSM) protocol is not required because it is beyond the scope of this study [12].

FTP traffic consists of 1000 bytes packets generated by Reno version of TCP sources. The average file size of TCP session is 2.5 × 105 bytes with maximum window size of 10 packets. TCP sessions are initialized randomly with an average interval of 0.5 second.

Buffer size for all nodes is set to 60 packets and parameters minth and maxth are set to 25 and 50 respectively. The average queue size is measured in packets. Maximum drop probability (maxp) is set to 0.1 by default.

The following simulation study consists of 6 different scenarios. All 6 scenarios are applied for the 4 Mbps RTP multicast session rate and link capacities dimensioned as specified in Table 1.

Scenario 1 is characterized by the absence of QoS mechanisms, while the specific combinations of them have been considered in scenarios 2, 3, 4, 5 and 6. The assumption is that in scenarios 2, 3, 4, 5 and 6 all routers adopt the same Weighted Random Early Detection (WRED) queue management and the Priority (PRI) scheduling algorithm. The simplest policer available in ns-2, Time Sliding Window (TSW2CM), is configured. A Committed Information Rate (CIR) is defined for Pareto, CBR and RTP with 2 Mbps, 6 Mbps and 4 Mbps, respectively. The null policy model is applied to the FTP and the RTCP packets [13]-[15].

In Scenario 2, priority class 1 is given to Pareto packet flows, class 2 to RTP, RTCP (original and retransmission) and CBR and class 3 to FTP flows.

Scenario 3 is characterized by the RTP/RTCP assignment to a dedicated queue of priority class 2. Here, priority class 1 is given to Pareto packet flows, class 3 to CBR and class 4 to FTP flows.

The highest priority class is given to RTP/RTCP packets in scenario 4, class 2 to Pareto, class 3 to CBR and class 4 to FTP flows.

In scenario 5, retransmission packets are assigned to a separate priority class which isolates them from the other packet types. In case of scenario 5, class 1 is assigned to Pareto, class 2 is assigned to the RTP/RTCP retransmission packets, class 3 to the CBR and original RTP/RTCP packets, and class 4 to FTP flows.

The main idea of this paper is demonstrated in scenario 6. Here, the highest priority class is assigned to Pareto packet flows and, since the importance of the concept of isolation of RTCP from RTP retransmission packets has been recognized, dedicated queue of priority class 2 is assigned to RTCP retransmission packets. Class 3 is assigned to CBR, original RTP/RTCP packets and RTP retransmission packets, and finally, class 4 is given to FTP flows.

3.2 Simulation Results and Discussion

Based on trace files obtained from 54 simulation runs, the results are collected regarding the loss rate and average interval of retransmission RTCP SR packets. The simulation results are shown in Fig. 4. and Fig. 5.

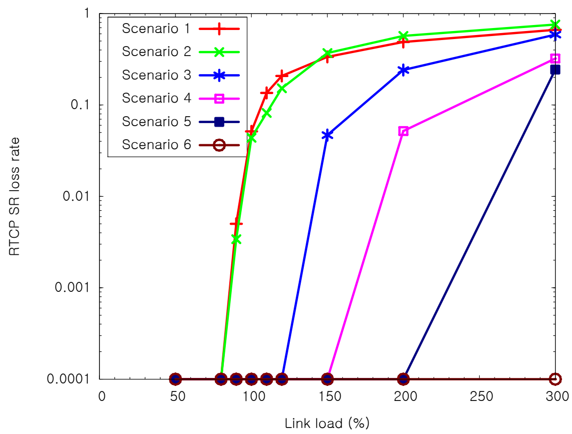

Fig. 4 RTCP SR loss rate.

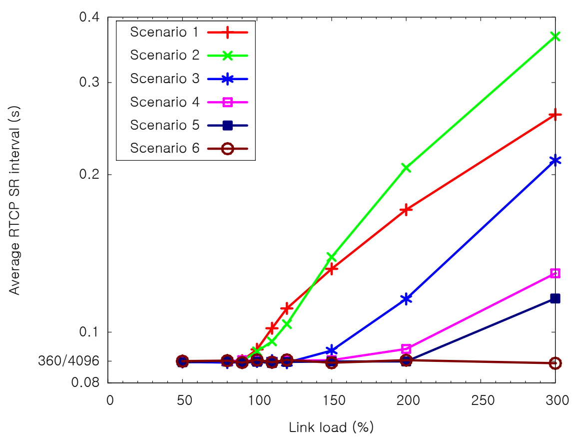

Fig. 5 Average RTCP SR interval.

As it has been expected, when packets are transmitted in the best effort mode as in scenario 1, it is impossible to ensure the average reduced minimum RTCP SR interval defined with 360 divided by the session bandwidth in kilobits per second when bottleneck link load exceeds 80%. Therefore, specific combinations of QoS techniques, which have been considered in scenarios 2, 3, 4, 5 and 6, are anticipated to lead to solution.

In scenario 2, the implementation of QoS mechanisms where RTP/RTCP packets (original and retransmission) are sharing the common queue with CBR packet flows, gives somewhat better results until bottleneck link load exceeds 120%. For highly congested bottleneck links (more than 120%) the obtained results regarding average interval of the retransmission RTCP SR packets are even worse.

In scenario 3, advantages of RTP/RTCP assignment to a dedicated queue of priority class 2 are evident. The obtained results in case of scenario 2 show that RTCP SR packet loss occurs as bottleneck link load exceeds 120%.

The highest priority dedicated queue is assigned to RTP/RTCP packets in scenario 4 which shows improvement compared to previous scenarios although it is not possible to guarantee the average reduced minimum RTCP SR interval defined with 360 divided by the session bandwidth in kilobits per second when bottleneck link load exceeds 150%.

In scenario 5, the retransmission packets (RTP and RTCP) are treated as priority class 2 in DiffServ environment. The CBR packets are sharing the common queue of priority class 3 with the original RTP and RTCP packets. In this case, the obtained results show that the retransmission RTCP SR packet loss occurs when bottleneck link load exceeds 200%.

Scenario 6 demonstrates the main idea of this paper. Results are promising and show that the isolation of the RTCP from the RTP retransmission packets by assigning them to higher priority class ensures the average reduced minimum RTCP SR interval defined with 360 divided by the session bandwidth in kilobits per second, according to IETF RFC 3550, sufficient for the inter-media lip-synchronization of the RTP multicast sessions regardless of network congestion with accurate statistics for the measurements performed by RTCP.

3.3 Statistical Analysis

In this section, statistical calculations were carried out to determine relationship between average RTCP SR interval and link load, and to perform a hypothesis test.

Regression analysis is performed between average RTCP SR interval and link load in form of several available regression models. Based on comparative analysis, linear regression model is selected. The coefficients of determination were found, along with the equations of the regression lines.

The results of linear regression analysis (Table 2) contribute to the recommended principle of this paper – to isolate the RTCP SR packets from the RTP data stream by assigning them to higher priority class in DiffServ architecture, which ensures the average reduced minimum RTCP SR interval according to IETF RFC 3550, sufficient for the inter-media lip-synchronization of the RTP multicast sessions regardless of network congestion.

Low determination coefficient (r2) in scenario 6 is pointing to a very low correlation (r) between average RTCP SR interval and link load, which is not statistically significant at the confidence interval of 95% (p>0.05).

The two-tailed hypothesis test (z-test) of the sample mean is conducted for 6 different scenarios (Table 3). As the computed p-value in case of scenario 6 is greater than the significance level α, one should accept the null hypothesis (H0). The risk to reject the null hypothesis (H0) while it is true is 85.58%.

Table 2: Linear regression analysis

Scenario	Linear fit	r^2	r	p
Scenario 1	0.0265 + 0.0007457x	0.9656	0.9826	<0.05
Scenario 2	- 0.01803 + 0.001192x	0.9349	0.9669	<0.05
Scenario 3	0.03905 + 0.0004966x	0.8246	0.9081	<0.05
Scenario 4	0.07225 + 0.0001573x	0.7582	0.8707	<0.05
Scenario 5	0.07712 + 0.000103x	0.6861	0.8283	<0.05
Scenario 6	$0.08827 - 2.6014 \times 10^{-6}x$	0.1514	0.3891	>0.05

Table 3: Hypothesis test (z-test)

Scenario	n	Mean (s)	SD (s)	two-tailed p	α
Scenario 1	48572	0.11113	0.08310	<0.0001	0.05
Scenario 2	47886	0.11271	0.10527	<0.0001	0.05
Scenario 3	55539	0.09721	0.06044	<0.0001	0.05
Scenario 4	58747	0.09189	0.03628	<0.0001	0.05
Scenario 5	59868	0.09018	0.03386	<0.0001	0.05
Scenario 6	61405	0.08792	0.02535	0.8558	0.05

4. Conclusion

This paper elaborates different RTP/RTCP packet classification schemes based on the DiffServ QoS techniques in order to ensure the average reduced minimum RTCP SR interval sufficient for lip-synchronization without excessive delay regardless of network congestion.

A retransmission method for rapid synchronization of RTP multicast sessions that uses the fundamental tools offered by the existing RTP and RTCP protocols [1] potentially reduces the synchronization delay.

Since RS sends relatively small number of RTP retransmission packets before receiver starts to receive the primary multicast stream, statistics measured by RTCP cannot be disrupted. Furthermore, RTCP SR packets in case of the reduced minimum packet interval constitute

about 0.24% of the RTP session bandwidth and as such cannot cause congestion.
In case of multimedia session with two RTP components, the slower component affects the session synchronization delay in so far as retransmission RTCP SR packet loss occurs. As it was expected, the only sensible approach to overcome problems regarding congestion was to introduce QoS guarantees for these packets.
Based on the extensive simulation analysis, a final conclusion is made that the isolation of the RTCP from the RTP retransmission packets by assigning them to higher priority class ensures the average reduced minimum RTCP SR interval defined with 360 divided by the session bandwidth in kilobits per second, according to IETF RFC 3550, sufficient for inter-media lip-synchronization of RTP multicast sessions regardless of network congestion with accurate statistics for the measurements performed by RTCP.
Future work will focus on synchronization delay problems for RTP multicast sessions caused by packet losses in the core network which disables the RS to cache all RTP/RTCP packets as they are sent in the session and to act as a feedback target for the session.

References

[1] H. Schulzrinne, S. Casner, R. Frederick and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications," IETF, Technical Report RFC 3550, July 2003.
[2] C. Perkins and T. Schierl, "Rapid Synchronisation of RTP Flows," draft-ietf-avt-rapid-rtp-sync-09, IETF, Internet Draft, January, 2010.
[3] Z. Begić, H. Bajrić and M. Kos, "Rapid Synchronization of RTP Multicast Sessions," 6th Euro-NF Conference on Next Generation Internet, NGI 2010, Paris, France, June 2010.
[4] Z. Begić, H. Bajrić and M. Kos, "Rapid Synchronization of RTP Multicast Sessions Using the Retransmission Server," 18th International Conference on Software Telecommunications and Computer Networks, SoftCOM 2010, submitted for publication.
[5] J. Ott, J. Chesterfield and E. Schooler, "RTP Control Protocol (RTCP) Extensions for Single-Source Multicast Sessions with Unicast Feedback," IETF, Technical Report RFC 5760, February 2010.
[6] B. VerSteeg, A. Begen, T. VanCaenegem and Z. Vax, "Unicast-Based Rapid Acquisition of Multicast RTP Sessions," IETF, draft-ietf-avt-rapid-acquisition-for-rtp-08, IETF, Internet Draft, March, 2010.
[7] J. Rey, D. Leon, A. Miyazaki, V. Varsa and R. Hakenberg, "RTP Retransmission Payload Format," IETF, Technical Report RFC 4588, July 2006.
[8] K. Fall and K. Varadhan, "The ns Manual (formerly ns Notes and Documentation)," The VINT Project, A Collaboration between researchers at UC Berkeley, LBL, USC/ISI, and Xerox PARC, January 2009.
[9] International Telecommunication Union – Telecommunication Standardization Sector, Recommendation Y.1540, "IP Packet Transfer and Availability Performance Parameters," November 2002.
[10] J. Baraković, H. Bajrić and M. Kos, "Priority Transmission of SIP Signaling Flows in Case of IP Link Congestion," IEEE International Conference on Communications Theory, Reliability, and Quality of Service, CTRQ 2009, Colmar, France, July 20-25, 2009.
[11] W. Yasin and H. Ibrahim, "Improving Triple Play Services Using Multi Protocol Label Switching Technology," Journal of Computer Science, Vol. 6 No. 3 pp. 269-278, 2010.
[12] R. Wittmann and M. Zitterbart, "Multicast Communication – Protocols and Applications," Morgan Kaufmann Publishers, 2001.
[13] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang and W. Weiss, "An Architecture for Differentiated Service," IETF, Technical Report RFC 2475, December 1998.
[14] J. Babiarz, K. Chan and F. Baker, "Configuration Guidelines for DiffServ Service Classes," IETF, Technical Report RFC 4594, August 2006.
[15] International Telecommunication Union – Telecommunication Standardization Sector, Recommendation Y.1541, "Network Performance Objectives for IP-Based Services," February 2003.

Zlatan Begić graduated from the University of Tuzla, Faculty of Electrical Engineering in 2005. He has been employed by BH Telecom, Joint Stock Company, Sarajevo since 2005. He is currently completing his Ph.D. studies at the University of Zagreb, Faculty of Electrical Engineering and Computing. His research includes a variety of topics in Quality of Service in IP Networks.

Mladen Kos received his B.E., M.E. and Ph.D. degrees from the University of Zagreb, in 1972, 1976, and 1983, respectively. He is a tenured professor in the Department of Telecommunications, Faculty of Electrical Engineering and Computing. He was the Head the Department of Telecommunications (1998-2000), Vice Dean of Academic Affairs (2000-2002), Dean of the Faculty of Electrical Engineering and Computing (2002-2006) and president of the Board of Directors at CARNet (2004-2008). He is a member of IEEE, IEICE, INFORMS, KoREMA and MIPRO.