Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 56 (2015) 225 – 232

Procedia Computer Science

The 10th International Conference on Future Networks and Communications (FNC 2015)

Satisfying QoS Requirements in NGN Networks using a Dynamic Adaptive Queuing Delay Control Method

Cherif Ghazel[a], Leila Saïdane[b]

The National School of Computer Sciences,University of Manouba - Tunisia

Abstract

The Next Generation Networks (NGN) are required to support the seamless delivery of voice, video and data with high quality. One of the most important key features of NGN is Quality of Service (QoS), which has been a focal point of NGN research, development and standardization. In this work, we propose and demonstrate an efficient NGN QoS control method, which guarantees the QoS requirements expressed in terms of per-service traffic flow authorized queuing delay. Multiple service levels are used in order to provide differentiated services and prioritize traffic requiring high-level QoS. The advantages of the proposed method are demonstrated through modeling of the QoS queuing delay parameters. Simulation results are generated to evaluate the performance of the proposed method in terms of supporting the QoS and improving the network scalability based on queuing delay control and adjustment.

Keywords: Arrival control, Control and management architecture, NGN, QoS, Queuing delay, QoS constraint delay

1. Introduction

Despite their demonstrated success and numerous practical advantages, NGNs are not without limitations and are the subject of continuous research and development. Over the past decade, several International Standardization and Development Organizations have dedicated their efforts to standardizing NGNs and have achieved significant breakthroughs, such as PacketCable, 3GPP, ETSI, MSF, and ITU-T[1,2,3,4,5,6]. One of the most important key features of NGN is QoS, which has been a focal point in NGN research. In particular, the end-to-end delay is considered as one of the most important factors of the NGN multiservice QoS guarantee. The most significant component of this QoS parameter is the queuing delay which depends especially on the number of packets waiting for service. Some of efforts which focused on developing efficient mechanisms to control this queuing delay, are reported in[7,8,9,10]. These Active Queue Management (AQM) schemes were designed in order to overcome the various drawbacks of the traditional buffer management scheme in packet-switched networks, the Tail-Drop (TD) algorithm[16].

* Corresponding author. Tel.: +216-98-828-877; +216-50-750-722
E-mail address: [a]Cherif.Ghazel@Email.ati.tn; [a] Leila.Saidane@Ensi.rnu.tn

Peer-review under responsibility of the Conference Program Chairs
doi:10.1016/j.procs.2015.07.203

In this work, we build on previous contributions, as in[7,8,9,10,11,12,13] and propose an exhaustive methodology for guaranteeing NGNs QoS and achieving the desired improvement through the queuing delay control and adjustment techniques. The proposed method is applied, under the control of an NGN resource and admission control and management sub-layer, for controlling the queuing delay mainly through adjusting the arrival rate. The aim is to control queuing delay through the buffer at the specified waiting delay value, so that the QoS delay for real time services may be guaranteed. This is required when the measured queuing delay reaches or exceeds the required QoS constraint delay value. In such cases, the traffic flow growth shall be limited in order to stay within the specified QoS constraint delay. The proposed method also extends the NGN architecture functionalities and defines additional rules and policies, which deliver high-level quality session-based services.

For practical purposes, we present modeling and simulation scenarios in order to assess and evaluate the performance of the proposed QoS control method in terms of queuing delay.

This paper is organized in three sections. The first section is this introduction. The details of the QoS control method architecture and application are outlined in section 2. Section 3 presents the experimental results and their analysis. Brief summary and concluding remarks are presented in last section.

2. QoS Control and Management Method

In this section, we outline the proposed queuing delay control and adjustment method and its integration within the NGN resource and admission control and management architecture, as detailed in our previous work[17, 18].

2.1. NGN Resource and Admission Control and Management Architecture

As we detailed in our previous works[17, 18], when an end-to-end connection with specified QoS request is received, the CS processes the demand of service as well as its associated QoS parameters, and sends a QoS resource authorization and reservation request to the Resource and Admission Control Manager (RACM). As illustrated Fig. 1, this entity, which represents the main component in this issue, determines the resource availability, the network topology and performs the CAC procedures and QoS guarantee rules[17,22]. The Service Policy Decision Manager (SPDM) consults the Transport Resource Control Manager (TRCM) in order to determine the network resource availability to satisfy the request. The first TRCM instance interacts with its neighboring TRCMs to determine the requested edge to edge QoS resource in the involved domains. Accordingly, the SPDM calculates the CAC decision based on the resource occupation state received from the TRCMs. It enables an access gate and applies the traffic control rules in the edge level and the bandwidth allocation in the transport level, or it rejects the incoming request.

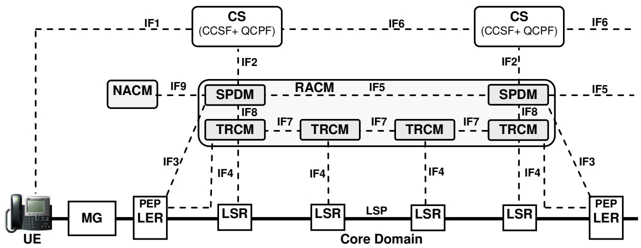

Fig. 1. Peer to peer NGN resource and admission control and management architecture.

2.2. QoS Control Method Architecture

QoS control method architecture consists of two mains parts: the RACM and the QoS-Aware NGN transport Edge routers (Fig. 2). The RACM is composed of a set of SPDMs and TRCMs, where the TRCM communicates with the QoS-Aware edge and core routers[20] and informs the SPDM of the network state. A QoS-Aware router manages data streams as separate flows. As shown in several contributions[17,20,21], flow-aware routers significantly improves the QoS and utilization of an IP-based network by memorizing and analyzing the flow state information and then handling and managing each flow. It achieves a QoS guarantee by discarding packets if the incoming flow exceeds the agreed rate by some burst tolerance. It records the current level of guaranteed traffic and rejects new incoming flows if the agreed bandwidth capacity is in use in order to implement the admission control. Practically, a QoS-Aware router consists of an incoming packets buffer component, a class-based routing component, which serves as a conventional router and a flow-based routing component, which supports QoS for real time traffic flow and controls the available bandwidth. It deploys a "Classifier" to communicate with the TRCM periodically in order to obtain a classification policy and rules and report real information about packets arrival and handling rate, "Flow-Based Processor" to handle high-level priority streams, and "Class-Based Processor" to handle the rest of general data traffic. The traffic quality level and the bandwidth allocation in a QoS-Aware router are decided and carried out by the SPDM based on subscriber and service information. It determines which stream levels should move into class-based or Flow-based Processing Buffers and sends information to the Classifier. Accordingly, the latter controls and manages bandwidth and handles packet streams in the transport level.

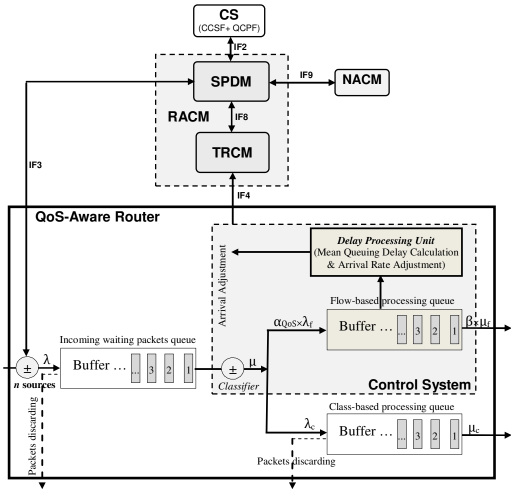

Fig. 2. QoS control and management architecture design

In order to control and reduce the queuing delay and guarantee the QoS for real time traffic, a periodically fine-tuning may be necessary for controlling the queuing delay in the flow-based buffer. Hence, the QoS-Aware edge routers shall determine the flow-based buffer average queuing delay, and dynamically adjust and fix an appropriate arrival rate in order to improve the queuing delay.

Next, we present the queuing model state transition.

2.3. Queuing Model and Analysis

The NGN traffic modeling depends essentially on traffic intensity, waiting delay, priority level and number of available servers. This modeling is used for surveying different traffic carried by NGN networks and dimensioning the corresponding resources. In this work, we assume that the users' traffic arrival process used in the discrete-time queuing model are based on Bernoulli process and the service process follows an Exponential distribution. Under these conditions, the system behavior can be described using Markov Transition Model[11,12,13].

Analysis and State Transition Model

In this subsection, we focus on the traffic state transition queuing model analysis. The flow-based queue of the proposed control system is modelled by a simple discrete-time queuing model that contains one server. For this model, α represents the arrival probability according to a Bernoulli process and β denotes the service probability over each Time Slot (TS) (cf. Fig. 2). This queuing model is in equilibrium state, and the queue length process is a Markov chain in discrete-time, with a finite state space. The system may contain a maximum of k customers and the queuing discipline is first-come first-served (*FCFS*). Indeed, every TS, the system checks the incoming packets waiting queue in order to serve the next packet waiting in the queue. If the server is busy and the number of packets in the system has reached the *determined (the threshold for)* flow-based processing queue length, any incoming new packets may be delayed in the incoming packet waiting queue or rejected if there is no enough free space, and the RACM will be signalled to adjust the overall arrival rate. The discrete-time queuing process is analysed to derive the relationship between the target queuing delay and the corresponding arrival rate. This relationship must meet the constraint that mean value of the queuing delay can be set to the required value as a function of arrival rate adjustment. Therefore, the mean queuing delay is periodically (every Time Slot) determined and the resulting information is used, for adjusting the value of the arrival rate, and limiting the delay at the required value. This largely improves the provided QoS in the NGN transport level.

Furthermore, the general state transition diagram as illustrated in the Fig. 3, is generated given that the packet arrival probability in a Time Slot is α and the packet departure probability in a Time Slot is β.

The balance equations of the finite-state discrete time Markov Chain are given by:

$$\pi_i = \sum_{x\in E} \pi_x Q(x,i) \qquad \text{for } i = 0,\dots,k \tag{1}$$

These equations yield the following:

$$For\ i = 0:\quad \pi_0 = \pi_0(1-\alpha) + \pi_1\beta(1-\alpha) \text{ which yields: } \pi_1 = \frac{\alpha}{\beta(1-\alpha)}\pi_0 \qquad \text{for } i=1 \tag{2}$$

$$For\ i = 1:\ \pi_1 = \pi_0\alpha + \pi_1(\alpha\beta + (1-\alpha)(1-\beta)) + \pi_2\beta(1-\alpha) \text{ which yields: } \pi_2 = \frac{\alpha^2(1-\beta)}{\left(\beta(1-\alpha)\right)^2}\pi_0 \ \text{ for } i=2 \tag{3}$$

$$For\ i = 2, 3, \dots, k-1:\ \pi_i = \pi_{i-1}\alpha(1-\beta) + \pi_i(\alpha\beta + (1-\alpha)(1-\beta)) + \pi_{i+1}\beta(1-\alpha) \tag{4}$$

$$For\ i = k:\ \pi_k = \pi_{k-1}\alpha(1-\beta) + \pi_k(\alpha\beta + (1-\beta)) \tag{5}$$

Rearranging equation (5) yields the following iterative difference equation:

$$\left(\pi_{i+1} - \pi_i\right) = \frac{\alpha(1-\beta)}{\beta(1-\alpha)}(\pi_i - \pi_{i-1}) \qquad \text{for } i = 2,\dots,k-1 \tag{6}$$

Or, equivalently:

$$\left(\pi_{i+1} - \pi_i\right) = \rho(\pi_i - \pi_{i-1}) \qquad \text{for } i = 2,\dots,k-1 \text{ Where: } \rho = \frac{\alpha(1-\beta)}{\beta(1-\alpha)} \tag{7}$$

- **Case 1: $\alpha=\beta$**

For the special case of $\alpha=\beta$, we get:

$$\pi_0 = \frac{1-\alpha}{1+k-\alpha} \tag{8}$$

$$\pi_i = \frac{1}{1+k-\alpha} \qquad \text{for } i = 1,\dots,\ k \tag{9}$$

The mean number of client in this special case is given by:

$$E_\pi(X) = \frac{k(k+1)}{2(1+k-\alpha)} \tag{10}$$

- **Case 2: $\alpha<\beta$**

For the case of $\alpha<\beta$, equation (7) is a discrete time difference equation with general solution of the form:

$$\pi_i = c\rho^i \qquad \text{for } i = 1,\dots,\ k \tag{11}$$

Thus, the stationary equilibrium probabilities, π_i for $i=1, 2,\dots,k$, can be expressed in terms of π_0, as follows:

$$\pi_i = \frac{\pi_0}{(1-\beta)}\rho^i \qquad \text{for } i = 1,\dots,\ k \tag{12}$$

The normalizing equation at steady state of the system may be expressed as follows:

$$\sum_{n=0}^{k} \pi_n = 1 \tag{13}$$

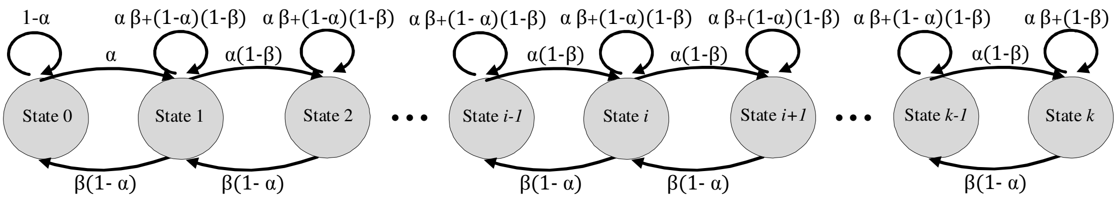

Fig. 3. State transition-rate diagram

By using the normalizing equation (13) as well as the stationary equilibrium probabilities, π_0 may be expressed, as follows:

$$\sum_{i=0}^{k} \pi_i = \pi_0 \left[1 + \frac{\rho}{(1-\beta)} \left(\frac{1-\rho^k}{1-\rho} \right) \right] = 1 \tag{14}$$

Consequently, π_0 can be expressed as follows:

$$\pi_0 = \left[1 + \frac{\rho}{(1-\beta)} \left(\frac{1-\rho^k}{1-\rho} \right) \right]^{-1} \tag{15}$$

Thus, the general form may be written as:

$$\pi_i = \frac{\rho^i}{(1-\beta)\left[1 + \frac{\rho}{(1-\beta)} \left(\frac{1-\rho^k}{1-\rho} \right) \right]} \quad \text{for } i = 1, \ldots, k \tag{16}$$

Once the stationary probabilities and π_0 are determined, we can compute the mean number of clients in this finite queue at the steady state, which is defined as follows:

$$E_\pi(X) = \frac{\pi_0}{1-\beta} \left[\rho \frac{1-\rho^k}{1-\rho} + \rho^2 \frac{\rho^k (k-1) - k\rho^{k-1} + 1}{(1-\rho)^2} \right] \tag{17}$$

- **Case 3: α>β**

For this case, the mean queuing length converges steadily to *k*. In this case, the number of packets in the system will gradually reach the determined *threshold* for the flow-based processing queue. The proposed system will react, and any newly arriving packets will be delayed in the incoming packet waiting queue or rejected if there is not enough space. Also, the RACM will be instructed to adjust the overall arrival rate.

Once the average queuing length is determined, the throughput [14,15] and the mean queuing delay may be given by:

$$Throughput = (1-\pi_0) \times \beta \tag{19}$$

$$MQD = \frac{G_\pi^{(1)}(s)\big|_{s=1}}{Throughput} = \frac{\sum_{i=0}^{k} i \times \pi_i}{Throughput} \tag{20}$$

The throughput may be estimated as the number of packets that have been successfully passed through the queuing system. Equations (10), (17) and (20), yield the mean number of clients in the finite queue and the corresponding queuing delay. The latter can largely be improved by adjusting the arrival rate, as illustrated in the simulation results presented in the next section.

Note that in order to ensure the existence of a unique stationary probability distribution; it is important to show that the modeled Markov chain is irreducible, recurrent non-null. This is in fact the case since $\alpha>0$ and $\beta>0$, and the *state space* is finite (of size $k+1$ states). The irreducibility is ensured by the fact that $\alpha> 0$ and $\beta> 0$, which implies that each state i communicates with neighboring states $i-1$ and $i+1$. In addition, this irreducibility $|E| < +\infty$ implies that the chain is positive recurrent.

3. Experimental Results and Discussion

In this section, we present and discuss simulation results, which illustrate the method, used for controlling the queuing delay mainly through adjusting the arrival probability. We will illustrate also additional simulation scenarios, for evaluating the performance of an NGN edge node in terms of queuing delay. We will outline and discuss experimental results that show the effects of the proposed method in terms of performance gain and QoS benefits.

3.1. Queuing Delay as a Function of Arrival Probability

This sub-section examines how the average queuing delay can be controlled and the required QoS may be reached by comparing a controlled queuing delay method with an uncontrolled one. For our simulation, we wrote MATLAB programs for implementing the proposed method presented in the theoretical model, as derived in (10), (17) and (20) above. We simulated the treatment ensured by an NGN edge node under the control of a RACM, which receives packets from different traffic sources. We examined the performance of the proposed NGN QoS control method based on case study comparisons.

Measurements of the queuing delay are obtained from simulation experiments. The proposed method is applied in a discrete-time environment, which is based on a specified basic time slot length. For each Time Slot (TS), packets arrive and then dispatched according to the packet arrival probability α and the packet departure probability β. Accordingly, the queuing system computes the current average queuing delay, and adjusts the arrival probability for the forthcoming time slot.

For the simulation parameters, we varied the value of the arrival probability, α, as illustrated by Fig.4.(a). Also, we initialized the other parameters' values as listed in Table 1.

Table 1. Simulation Parameters

Parameter	Definition	Variable/Fixed	Value(s)
α	Arrival probability	variable	Interval [0,1]
β	Service probability	Fixed	0.9
k	The queue size	Fixed	150
QoS_{Delay}	The QoS constraint delay	Fixed	100

The Uncontrolled Case:

First we examine the variation of the mean packet queuing delay in terms of the arrival probability, α, in the uncontrolled case. As illustrated in Fig.4.(a), mean queuing delay Q_{Delay} increases at higher rates for larger values of the arrival probability (α>0.80). It then exceeds the required QoS_{Delay} as more packets enter the queue. As the probability of arrival α exceeds the probability of service β, the Q_{Delay} converges to a steady and stationary state. This is the case because as the queue becomes full, newly arriving packets are delayed in the incoming packet waiting queue or rejected.

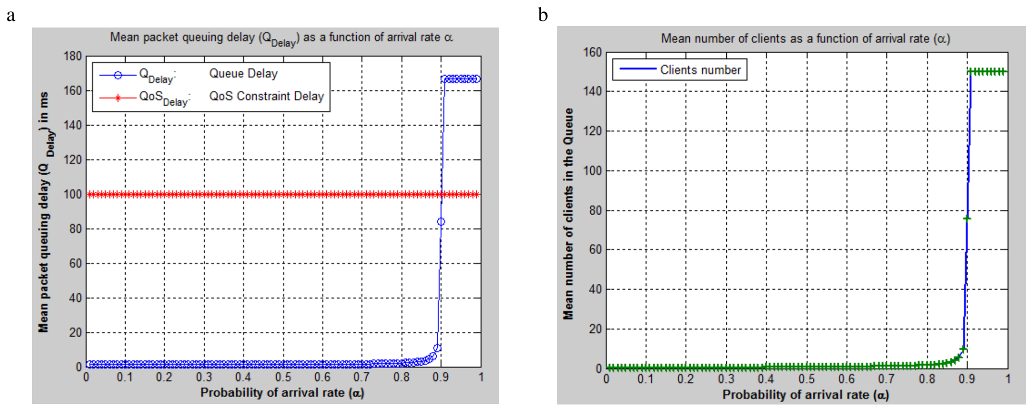

Fig. 4. (a) Queuing delay as a function of the arrival probability; (b) Number of clients as a function of the arrival probability

Fig.4.(b) illustrates how the mean number of packets in the queue increases at higher rates for larger values of the arrival probability (α>0.80) until the queue becomes full. Beyond that point, any incoming packets are delayed or rejected.

In this uncontrolled case, Fig.4.(a) shows that the mean queuing delay increases and exceeds the required QoS_{Delay} as the probability of the arrival rate, α, exceeds 0.9. Next, we shall examine the controlled queue case.

The Controlled Case

For this case, the input parameters are kept the same, as illustrated in Table 1. However, we shall now implement a control system, which reacts if and when the mean queuing delay Q_{Delay} exceeds the QoS constraint delay QoS_{Delay}. In such case, the control system shall adjust the arrival rate by adjusting the arrival probability, α, as explained next.

Fig.5.(a) illustrates the average queuing delay Q_{Delay} variation as a function of the arrival probability α. Note how the average queuing delay Q_{Delay} increases smoothly until it reaches the required QoS constraint delay QoS_{Delay}. Beyond this point, the control system computes the mean number of clients in the queue and the resulting average queuing delay by using the value of α, denoted by α*, for which $Q_{Delay} = QoS_{Delay}$. Basically, the control systems adjust the probability of the arrival rate α:

$$If \ \ \alpha \geq \alpha^{*} then\ choose\ \alpha = \alpha^{*}$$

As such, the control system adjusts the arrival rate, as needed, in order to maintain the required average queuing delay in the queue. Accordingly, this ensures that the variation of the average queuing delay will always meet the required QoS_{Delay} specification: $Q_{Delay} \leq QoS_{Delay}$

a b

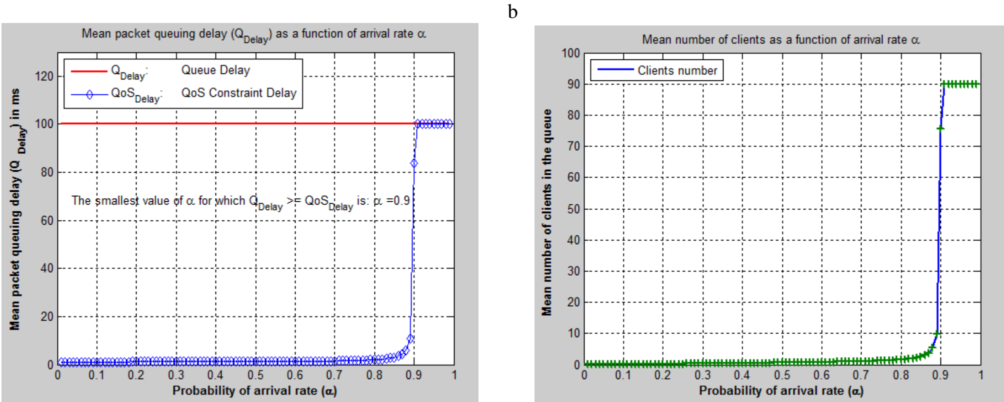

Fig. 5. (a) Queuing delay as a function of the arrival probability; (b) Number of clients as a function of the arrival probability

In view of Fig.5.(b), we observe that as the probability of the arrival rate, (α), increases, the mean number of packets in the queue increases progressively until it reaches a controlled limit. Thereafter this variation becomes well maintained and limited to a fixed value. This imposed queue length limit may be considered as a threshold installed in the queue and used for controlling and adjusting the required average queuing delay for real time services. We also observe that the simulated variation of the mean number of packets in the queue as a function of α are consistent with the expected results as studied in the theoretical model. This ensures that the simulated delay is constrained by the required QoS waiting value, as illustrated in Fig.5.(a).

Fig.5.(a) and Fig.5.(b) illustrate that the mean number of packets in the queue and the corresponding average queuing delay are low when $\alpha<\beta$. This implies that, in this case, the system is in a stationary state and it delivers a higher QoS for clients.

For the controlled case, the simulation results show that the average waiting delay can be adjusted in order to meet the QoS_{Delay} by employing a suitable queuing delay control system.

In this controlled case, it is important to note that for each QoS constraint delay (QoS_{Delay}) and a fixed service rate, for example $\beta=0.9$, the proposed method may set an appropriate and corresponding arrival probability and average queue length (threshold) in order to respect the requested QoS constraint delay, as illustrated Table 2.

Table 2. Arrival Probability and Queue Length as a Function of the QoS Constraint Delay

QoS_{Delay} (ms)	20	40	60	80	100
Arrival probability (α)	0.89472	0.89763	0.89884	0.89974	0.89995
Queue length (Packets)	18	36	54	72	90

In this section, we presented and discussed simulation results, which illustrate the importance of the proposed queuing delay control method through adjusting the arrival rate (α).

4. Conclusion

In this work, we proposed a queuing delay control and adjustment method, which guarantees the required QoS for the transport of real-time multi-service traffic in NGN networks. It controls the queuing delay at the specified waiting delay value mainly by adjusting the arrival probability, so that the QoS delay for real-time services may be always achieved and guaranteed. This is applied when the measured queuing delay reaches or exceeds the required QoS constraint delay value in order to limit the traffic flow growth and stay within the specified QoS constraint delay. The proposed method also extends the NGN architecture functionalities and defines additional rules and policies, which deliver high-level quality session-based services. It is applied at the edge level under the control of the RACM as a basis component for the proposed large scale QoS guarantee method.

In order to evaluate the performance of the proposed method, we derived theoretical models and generated and discussed simulation results from these models. The performance of the proposed method is assessed and evaluated in terms of queuing delay over an NGN transport network. The determined results illustrate the advantages of the proposed method in terms of performance gains and QoS benefits and provision. They present the relationship between the arrival rate and buffer capacity control and QoS requirement and guarantee.

References

1. PacketCable™ 1.5 Specifications PK-TSP-DQOS1.5-I02-050812, "Dynamic Quality-of-Service", PacketCable, August 2005.
2. 3GPP TS 23.207, "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; End-to-end Quality of Service (QoS) concept and architecture", 3GPP, 2004.
3. ETSI ES 282 003 V1.1.1, "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Resource and Admission Control Sub-system (RACS); Functional Architecture", ETSI, 2006.
4. Gallon C, Schelén O. "Quality of Service for Next Generation Voice Over IP Networks", MultiService Forum, 2003.
5. Gallon C, Schelén O. "Bandwidth Management in Next Generation Packet Networks", MultiService Forum, 2005.
6. ITU-T Recommendation Y.2012, "Functional requirements and architecture of the NGN", 2006.
7. Floyd S, Jacobson V. "Random early detection gateways for congestion avoidance", Networking, IEEE/ACM Transactions on Volume 1, Issue 4, pp.397-413, 1993.
8. Bonald T, May M, Bolot J. "Analytical Evaluation of RED Performance", IEEE Infocom 2000, Vol. 3, p.1415-1424, 2000.
9. Lapsley D, Low S. "Ransom Early Marking for Internet Congestion Control", Proceeding of GlobeCom'99, p1747-1752, 1999.
10. Athuraliya S, Lapsley D, Low S. "An enhanced random early marking algorithm for Internet flow control", IEEE Infocom 2000, Vol. 3, p.1425-1434, 2000.
11. Etbega-Mohamed M, Woodward ME., "Adaptive Drop Tail: An Algorithm for maintaining delay bounds in a buffer with time varying arrival rate", Proceedings of ICCTA'08 IEEE, p 625-226, Syria, 2008.
12. Wang J, Guan L, Wang XG, Grigg A, Awan I, Phillips I, Chi X. "QoS Enhancements and Performance Analysis for Delay Sensitive Applications", Advanced Information Networking and Applications (AINA 2009), England, 2009.
13. Etbega-Mohamed M, Woodward ME. "Adaptive Early Random Drop: An algorithm for controlling queueing delay in a buffer", 16th Telecommunications Forum TELFOR 2008, Sarbia, 2008
14. Chuah MC. "Computer Networks - Little's Theorem/Formula Basic Queuing Model", Fall 2004, USA, 2004.
15. Willig A. "Short Introduction to Queueing Theory", Technical University Berlin-Telecommunication Networks Group, Berlin, 21, 1999.
16. Braden B et al. "Recommendations on Queue Management and Congestion Avoidance in the Internet", RFC2309, 1998.
17. Ghazel C, Saidane L. "An Improved Model for Next Generation Networks with Guaranteed End-to-End Quality of Service", The 15th IEEE International Conference on Networks Proceeding, Australia, p. 112-117, 2007.
18. Ghazel C, Saidane L. "A Novel QoS-Aware Method Based on Resource Control and Management in NGN Networks", The 22th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing-PDP2014, Italy, p. 288-291, 2014.
19. Bonald T, Boulahia SO, Roberts J. "IP traffic and QoS control: the need for a flow-aware architecture", World Telecommunications Congress, Paris France, 2002.
20. Roberts LG. "The Next Generation of IP-Flow Routing", SSGRR 2003S International Conference, Italy, July 2003.
21. Oueslati S, Roberts J. "A new direction for quality of service: flow-aware networking", Next Generation Internet Networks 2005, Rome Italy, pp. 226-232, 2005.
22. Ban SY, Choi JK, Kim HS. "Efficient End-to-End QoS mechanism using egress node resource prediction in NGN network", ICACT 2006 Proceeding, pp. 480-483, 2006.