Ann. Telecommun. (2009) 64:225–237
DOI 10.1007/s12243-008-0054-1

Quality of service for voice over IP in networks with congestion avoidance

Vitalio A. Reguera · Evelio M. G. Fernandez · Felix A. Paliza · Walter Godoy Jr. · Eduardo P. Ribeiro

Received: 14 May 2008 / Accepted: 22 September 2008 / Published online: 22 October 2008

Abstract This paper assesses the impact of active queue management schemes on the quality of service of voice over Internet protocol applications. A new analytical method based on a fixed point approach to estimate the end-user satisfaction is proposed. The results obtained were validated using discrete event simulation techniques. In all the studied cases, it was observed a great deal of agreement between the analytical results and the results obtained through simulation. The theoretical predictions, as well as the presented empirical evidences confirm, as demonstrated in previous works, that the use of active queue management offers better quality of service than the traditional queue control mechanisms used in Internet. From these results, we may reasonably conclude that the presented method can be used for network design in the presence of voice traffic.

V. A. Reguera · F. A. Paliza
Telecommunications Department,
Central University of Las Villas,
Santa Clara, Cuba

V. A. Reguera
e-mail: vitalio@uclv.edu.cu

F. A. Paliza
e-mail: fapaliza@uclv.edu.cu

E. M. G. Fernandez (✉) · E. P. Ribeiro
Electrical Engineering Department,
Federal University of Parana,
Curitiba, Brazil
e-mail: evelio@eletrica.ufpr.br

E. P. Ribeiro
e-mail: edu@eletrica.ufpr.br

W. Godoy Jr.
Electrical Engineering Department,
Federal University of Technology-Parana,
Curitiba, Brazil
e-mail: godoy@utfpr.edu.br

Keywords VoIP · Congestion avoidance · AQM · Quality of service · Internet

1 Introduction

Active queue management (AQM) has been proposed as a tool for congestion avoidance in Internet [1]. Several studies have been dedicated in the past few years to design and evaluate AQM algorithms [2–5]. AQM mechanisms interact at network routers sending back information to the traffic sources about the imminence of congestion. Transport layer protocols like the transmission control protocol (TCP) have the capacity to react to these signals reducing data transmission rate to avoid network congestion. In many cases, the use of AQM helps to reduce network impairments such as delay, delay variation (jitter), and packet loss that are produced during the congestion periods. These lead to an increase in the quality of service (QoS) of those applications that are affected for the degradation of these parameters. The complex interaction between congestion control mechanisms and the applications makes it analytically difficult to evaluate the impact of AQM on the perception that the end users have on the offered QoS. A previous work [5] verified through computational simulation that adaptive version of active queue management algorithms increases in a significant way the customer-perceived quality of phone calls transmitted through congested connections.

It is clear that in those scenarios where it is possible to establish a differentiation of flows and to separate voice applications from other applications, a better choice is to use queues with priority (e.g., priority queue, low latency queue) [6]. This guarantees for expedited sending of voice packets with a superior QoS than it would have with any other mechanism. However, in heterogeneous networks, it is not always possible to establish end-to-end QoS warranties for voice applications. The results of this research are applicable to those network segments where voice traffic is not differentiated from remaining traffic. These scenarios are very common in Internet with the proliferation of voice over Internet protocol (IP) services (VoIP) based on peer to peer architectures as it is the case of Skype whose software has been widely diffused [7, 8].

In this work, an analytical procedure is presented based on a fixed point approach to determine the impact of adaptive random early detection (ARED) and adaptive virtual queue (AVQ) in the quality of service of VoIP applications. The random early detection (RED) algorithm is the most spread and implemented AQM proposal in the Internet [2]. This algorithm signals the traffic sources about congestion imminence by marking or discarding packets with a certain probability depending on the average queue size. However, the overall performance of the original version of RED algorithm has shown to be very dependent on the configuration parameters. The desired results may not be obtained in some network scenarios [9, 10]. Its adaptive version solves this problem adjusting the control parameters of the algorithm in a dynamic way. The only requirement is to set the desired average queue size or equivalently the average packet delay in the queue.

The AVQ mechanism [11] is based on the idea of maintaining a virtual queue with the same size of the real queue, but with less outflow capacity. For each input packet, a fictitious packet is put in the virtual queue whenever it is not full; otherwise, the packet is marked or discarded.

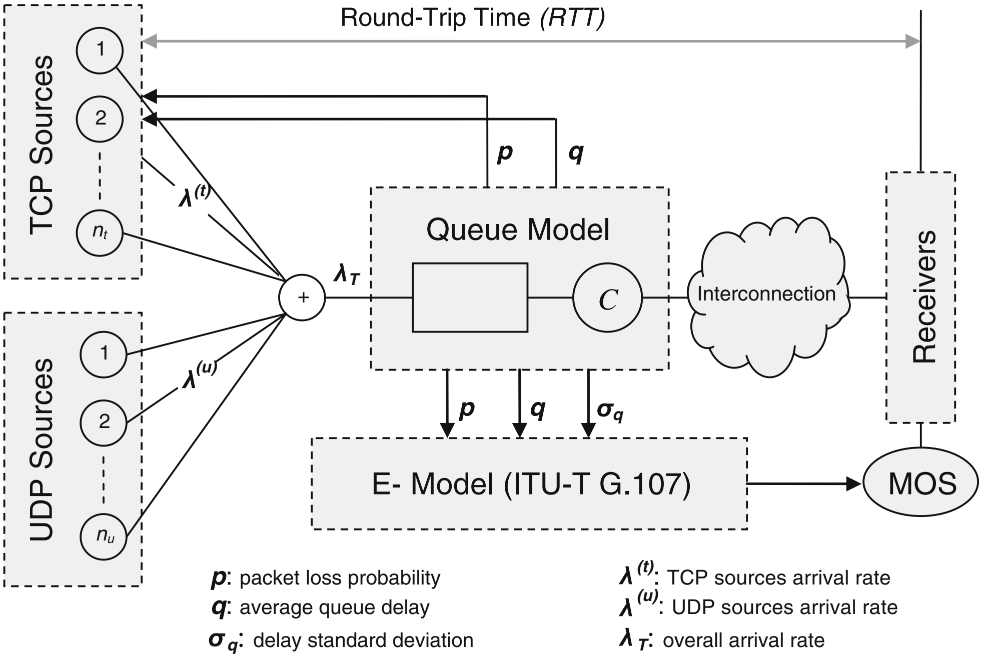
The model proposed in this paper allows capturing the interaction between AQM algorithms and sources of traffic, estimating the values of the delay, jitter, and packet loss rate. Once obtained, these network metrics are used as inputs to the computational algorithm proposed in [12] that allows to estimate the amount of satisfaction experienced by users of a VoIP system.

A detailed description of the proposed analytical method appears in the next section. In Section 3, the results obtained by the analytical method are contrasted with the results obtained through simulation. Finally, Section 4 comments in the form of conclusions some criteria about the validity and usefulness of the obtained results.

2 Analytical models

The starting point for the development of the mathematical model is a congested link where some AQM mechanism is implemented. Considering an Internet router in which TCP and user datagram protocol (UDP) flows are mixed, the interaction between these

Fig. 1 Network diagram



flows and the router AQM mechanism can be represented as illustrated in Fig. 1. TCP sources regulate the generated traffic depending on the packet loss rate and on the round trip time (RTT) experienced by the packets. On the other hand, UDP sources are unresponsive to the degree of congestion that the network presents.

From the point of view of queuing theory, the router can be considered as a queue whose service discipline and service time will be determined by the acting AQM mechanism and by the capacity of the output link. Different from the analysis carried out in [13], in this work it is taken into account the feedback that exists between the queue and the traffic sources. The steady-state solution of the system (fixed point) resulted from its inherent interaction is considered. This idea was first described in [14] and later it has frequently been used to model the behavior of different TCP implementations [15, 16]. Starting from an accurate model of the sources and the queue, the fixed point corresponds to the operating point of the real network. The obtained values of loss rate, average queue delay, and jitter serve as inputs to the E-Model [12] to estimate customers' mean opinion score (MOS) for voice applications.

While the analytical models themselves are not new, its combination and application to VoIP traffic in networks with congestion avoidance had never been addressed in previous works to the best of our knowledge.

2.1 Fixed point approach

Let $\lambda_i^{(t)}$ and $\lambda_i^{(u)}$ be the arrival rate generated by the i-th TCP and UDP source, respectively (see Fig. 1). The total arrival rate, λ_T, is

$$\lambda_T = \sum_{i=1}^{n_t} \lambda_i^{(t)} + \sum_{i=1}^{n_u} \lambda_i^{(u)} \tag{1}$$

where n_t is the number of TCP sources and n_u is the number of UDP sources. Let $q_i(\lambda_T)$ and $p_i(\lambda_T)$ the average queue delay and $t_i(q_i, p_i)$ the packet loss rate determined by the queue model and the throughput of the i-th TCP source. Assuming that the arrival rate of the UDP sources is constant and is not affected by network congestion, then the operating point can be found in such a way that satisfies

$$\sum_{i=1}^{n_t} t_i\left(q_i\left(\lambda_T\right), p_i\left(\lambda_T\right)\right) + \sum_{i=1}^{n_u} \lambda_i^{(u)} = \lambda_T \tag{2}$$

Equation 2 is of the form $f(x) = x$ where λ_T is a fixed point. Depending on the queue model used, it is not always possible to find a compact expression for $q_i(\lambda_T)$ and $p_i(\lambda_T)$. The solution of Eq. 2 can be found using computational methods (e.g., using Brent's method).

2.2 Source model

Since UDP protocol does not have the ability to adapt the transmission rate to the predominant network conditions, the arrival rate from sources associated to this protocol will be considered constant. This can be the case of a voice codec generating equal size packets at a constant rate (e.g., G.711). On the other hand, sources based on TCP protocol will adapt the transmission rate depending on the amount of congestion actually experienced.

In what follows, a widely used model for steady-state throughput estimation of TCP sources proposed in [17] will be considered. This model is applicable to TCP Reno sources, one of the most widespread in Internet [18]. Here the throughput is a function of the loss rate and the round trip time experienced by the link. Using the same previous notation and in accordance with [17], the throughput of a TCP source can be approximated by the following expression

$$t_i(q_i, p_i) = \min\left(\frac{W_m}{\mathrm{RTT}_0 + q_i}, \frac{1}{(\mathrm{RTT}_0 + q_i)\sqrt{\frac{2bp_i}{3}} + T_o \min\left(1, 3\sqrt{\frac{3bp_i}{8}}\right) p_i\left(1 + 32p_i^2\right)}\right) \tag{3}$$

where RTT_0 is the round trip time without taking into account the delay produced by the queue, W_m is the maximum window size configured at the receiver (measured in segments), T_o is the waiting time before the first retransmission and b is the number of packets that are acknowledged by ACK (typically two).

2.3 Drop-tail queue model

In order to consider the benefits that can be obtained by using AQM in the presence of voice traffic, firstly it is necessary to establish the network behavior when a traditional queue mechanism such as drop-tail (DT) is used. A DT queue can be analyzed with classic models of queuing theory. In this case, the model can be described using the abbreviated Kendall notation [19] as a $G/G/K+1$ queue. That is, a queue with arrival process and service time characterized by a general distribution, only one server and capacity for K packets in the queue. Considering a link with constant capacity, the service time distribution will be determined by the distribution of the packets size. Studies accomplished in the Internet [20, 21] show that this distribution is predominantly trimodal due to the combination of the TCP acknowledgments (ACK) and the prevalence of two maximum transmission units (MTU). However, this distribution can vary as a consequence of the continuous appearance of new applications [21]. In the simplified model of Fig. 1, this distribution will be characterized by the combination of the TCP maximum segments size (MSS) and by the size of the frames generated by the voice codecs and other sources of unresponsive traffic.

The arrival process distribution is much more difficult to be described due to the bursty nature of the data packets that characterizes many of the traffic sources in the Internet. Some studies [22] evidence that Internet traffic exhibits a self-similar behavior, in other words, the statistical traffic patterns have similar characteristics when they are analyzed in different time scales. These observations have led many researchers to avoid the use of classic models of queuing theory. However, recently, these models have been revisited with acceptable results in the analysis of some Internet systems [14, 15, 23]. Specifically, in [2], it is demonstrated that the traffic produced by the overlap of a considerable number of sources tends to behave as a Poisson arrival process. Considering a high multiplexing level in the input of the congested link, the aggregated traffic can be approximated by an exponential distribution. As a result, this system becomes an $M/G/K+1$ model.

For an $M/G/K+1$ queue, the probability of existing j packets in the system in an arbitrary steady-state time instant, P_j^{DT}, can be computed (using classic results of queuing theory, [19]) as

$$P_j^{\mathrm{DT}} = \frac{H_j}{1+aH}, \qquad j = 0, 1, \ldots K \tag{4}$$

where a is the offered traffic and H_j and H are obtained as follows:

$$H_{j+1} = \left(H_j - \psi_j - \sum_{i=1}^{j} \psi_{j-i+1} Hi \right) \psi_0^{-1}, \quad j = 0, 1, \ldots K-1 \tag{5}$$

$$H = 1 + \sum_{i=1}^{j} H_j \tag{6}$$

being $H_0 = 1$ and ψ_j the probability that j packets arrive during the service time. The loss probability is obtained as:

$$p = P_{K+1}^{\mathrm{DT}} = 1 - \sum_{j=0}^{K} P_j^{\mathrm{DT}} = 1 - \frac{H}{1+aH} \tag{7}$$

Using Eq. 4 and Little's formula [19], the average queue waiting time is:

$$q = \frac{\sum_{i=1}^{K+1} (j-1)\, P_j^{\mathrm{DT}}}{\lambda_{\mathrm{T}}\,(1-p)} \tag{8}$$

In the above model, the loss probability and the average queue waiting time are the same for all traffic sources so, in Eqs. 7 and 8 the subscript i has been omitted ($p_i = p$ and $q_i = q$). In practice, there can be differences between the values of loss and delay experienced by packets sent in burst from some sources. The validity of this and other approaches included in the model will be analyzed and contrasted with experimental results in the next section.

Jitter is estimated from the standard deviation of the packet delay. Let σ_{q}^2 denote the system queue waiting time variance. Considering that a packet that enters in the queue should wait until all previous packages in the system were served, and using the results from [24] leads to

$$\sigma_{\mathrm{q}}^2 = Q\sigma_h^2 + \sigma_Q^2 h^2 \tag{9}$$

where Q is the average number of packets in the system, σ_Q^2 the variance of the number of packets in the system, h the average service time, and σ_h^2 the variance of the average service time.

From Eq. 9, the jitter experienced by the i-th traffic source can be computed as

$$\sigma_i = \sqrt{\sigma_{\mathrm{q}}^2 + \sigma_{h_i}} \tag{10}$$

where σ_{h_i} is the average service time variance of the i-th traffic source.

The network equilibrium point can be found using Eqs. 2, 3, 7, and 8. Once the fixed point is determined, the values obtained through Eqs. 7, 8, and 10 are used to estimate the degree of end-user satisfaction of VoIP systems.

2.4 ARED queue model

When ARED is implemented in the router queue, the packet loss rate will be conditioned by the discard probability function of the algorithm, in accordance with

$$p_{\mathrm{RED}}\left(l_{\mathrm{q}}\right)=\begin{cases}0, & l_{\mathrm{q}}<\min_{\mathrm{th}}\\ 1, & l_{\mathrm{q}}>\max_{\mathrm{th}}\\ \max_{\mathrm{p}}\left(\frac{l_{\mathrm{q}}-\min_{\mathrm{th}}}{\max_{\mathrm{th}}-\min_{\mathrm{th}}}\right), & \text{elsewhere}\end{cases} \tag{11}$$

where $\min_{\mathrm{th}}$ and $\max_{\mathrm{th}}$ are the maximum and minimum average queue size threshold between which the discard probability has a linear behavior with positive slope, $\max_{\mathrm{p}}$ is the maximum probability value in that interval and l_{q} is the average queue size, updated from the instantaneous queue size, I_{q}, in the following way

$$l_{\mathrm{q}}\longleftarrow\left(1-w_{\mathrm{q}}\right)l_{\mathrm{q}}+w_{\mathrm{q}}I_{\mathrm{q}},\qquad 0<w_{\mathrm{q}}<1 \tag{12}$$

where w_{q} is a weighting factor that indicates the weight that the most recent sample (I_{q}) has in relation to the cumulated average l_{q}.

The steady-state probability of queue occupation can be obtained using a Markov chain and approximating the arrival process and service time distributions by exponential distributions. In this case, the process becomes a birth–death process. By approximating the average queue size by the instantaneous queue size, the steady-state probability of having j packets in the system can be computed as

$$P_j^{\mathrm{ARED}}=\frac{a^j\prod\limits_{l=0}^{j-1}\left(1-p_{\mathrm{RED}}\left(l\right)\right)}{1+\sum\limits_{i=1}^{K+1}a^i\prod\limits_{l=0}^{i-1}\left(1-p_{\mathrm{RED}}\left(l\right)\right)} \tag{13}$$

where a is the offered traffic.

One needs to take into account the adaptive behavior of ARED in the computation of Eq. 11. The parameter $\max_{\mathrm{p}}$ is adjusted according to the arrival rate following the guideline presented in [25].

The ARED algorithm can be configured to discard all the packets with equal probability (packet mode) or to discard packets depending on its size in bytes (byte mode). In the first case, the packet discard rate will be the same for all the traffic sources and can be obtained as

$$p=\sum_{j=0}^{K+1}P_j^{\mathrm{ARED}}p_{\mathrm{RED}}\left(j\right) \tag{14}$$

When ARED is configured in byte mode the loss rate will depend on the size of the sent packets so the loss rate for the i-th source is computed as

$$p_i=\frac{\lambda_{\mathrm{T}}h_i}{a}\sum_{j=0}^{K+1}P_j^{\mathrm{ARED}}p_{\mathrm{RED}}\left(j\right) \tag{15}$$

where h_i is the average service time for the i-th source.

The queuing delay experienced by packets from the i-th traffic source can be computed from Little's formula as

$$q=\frac{\sum\limits_{j=0}^{K+1}\left(j-1\right)P_j^{\mathrm{ARED}}}{\lambda_{\mathrm{T}}\left(1-p\right)} \tag{16}$$

2.5 AVQ queue model

For the case of a queue with AVQ mechanism, the model was obtained in a different way due to the specific characteristics of this AQM algorithm. In Fig. 2, a block diagram of an AVQ queue model is shown. Here the system is composed of two queues, the first one representing the virtual queue which determines the loss probability. Traffic that is not discarded will wait on the second queue until being served. Given that the link capacity of the first queue is less than the link capacity of the whole system, the second queue practically does

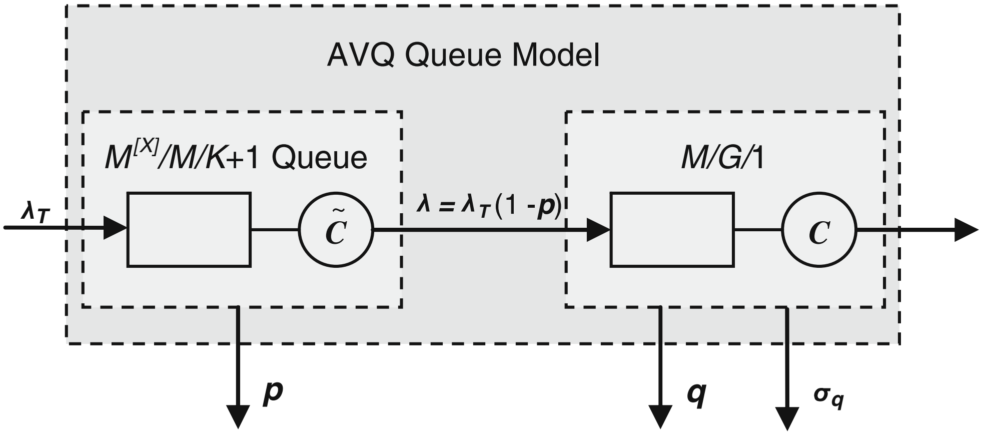


Fig. 2 AVQ queue diagram

not experience congestion; therefore, the delay is small in comparison with a DT queue.

When the AVQ algorithm is configured to measure the size of the queue in packets, it can be used (Eq. 7) to determine the loss probability with the difference that, in this case, the link capacity is, $\widetilde{C} = \rho C$ where ρ is the desired utilization. However, the predetermined configuration of the algorithm measures the size of the queue in bytes (note that a steady-state analysis is being made and the adaptation mechanism developed in AVQ ensures that the link utilization fulfils the desired utilization). This implies that bigger packets will experience high loss rates when the queue begins to saturate. In order to take into account this effect, the queue is represented using a batch arrival model of the form $M^{[X]}/M/1/K+1$, where X is the random variable for the number of octets in the packet and K is the maximum queue size, both expressed in bytes. In this case, the probability distribution of the steady-state system, $P_1^{(1)\mathrm{AVQ}}$, can be found through the following expressions:

$$\begin{aligned}
P_1^{(1)\mathrm{AVQ}} &= P_0^{(1)\mathrm{AVQ}} \sum_{i=1}^{K+1} a_i \\
P_j^{(1)\mathrm{AVQ}} &= P_{j-1}^{(1)\mathrm{AVQ}} \left(1 + \sum_{i=1}^{K+1} a_i\right) \\
&\quad - \sum_{i=0}^{j-2} P_i^{(1)\mathrm{AVQ}} a_{j-i-1} \\
P_{K+1}^{(1)\mathrm{AVQ}} &= \sum_{i=1}^{K} P_i^{(1)\mathrm{AVQ}} a_{K+1-i}
\end{aligned}$$

$$P_0^{(1)\mathrm{AVQ}} + \sum_{j=1}^{K+1} P_j^{(1)\mathrm{AVQ}} = 1 \tag{17}$$

where, $a_n = \lambda_n/\mu$, λ_n is the arrival rate of the n-size packets and μ is the reciprocal of the time required for the transmission of 1 byte. Assuming that all packets produced by the i-th traffic source have the same size n, the loss probability will be

$$p_i = \sum_{j=K+2-n}^{K+1} P_j^{(1)\mathrm{AVQ}} \tag{18}$$

The traffic that is not discarded enters on the second queue with an arrival rate $\lambda = \lambda_{\mathrm{T}}(1-p)$. The packet average waiting time can be calculated substituting in Eq. 8 the steady-state system probability distribution by that of an $M/G/1$ model which is a particular case of DT for $K = \infty$. In the same way, the jitter can be estimated using Eq. 10.

2.6 Multiple congested links and other considerations

In an Internet voice communication, packets travel through multiple links. Depending on the route, congestion can occur in more than one link. The previously described model for a congested link can be easily extended to a topology with multiple links. In this case, the network can be represented by a set of cascaded queues where the traffic experienced by a particular link will enter the next queue being mixed with the traffic generated by another set of sources. Applying for each queue the procedures described in the previous subsections, the operating point of the whole system in steady state can be found. The loss probability, the average waiting time, and the end-to-end jitter considering m queues can be computed, respectively, as

$$p = 1 - \prod_{\forall m} (1 - p_m) \tag{19}$$

$$q = \sum_{\forall m} q_m \tag{20}$$

$$\sigma = \sqrt{\sum_{\forall m} \sigma_m^2} \tag{21}$$

The proposed model assumes a reliable data link layer (i.e., packet loss and significant delay variations are located at the queue of the networks' nodes due to traffic congestion) as well as stable routing mechanisms. The above is true for most wired networks but it could be different for certain types of wireless networks. Particularly, wireless multi-hop ad hoc networks (e.g., IEEE 802.11) can suffer significant packet drops at data link layer and re-routing instability. In such cases, the model should be modified to reflect these behaviors.

2.7 MOS calculation

Mean opinion score (MOS) [26] has been traditionally used to measure subjective perception of voice communications. MOS is given on a scale of 1–5, where a higher value corresponds to better quality. Since MOS is a subjective test difficult to be carried out in practical situations, some other objective tests have been developed. ITU recommendation G.107 describes E-model, a computational algorithm that incorporates impairment factors present in modern transmission networks. The output of E-model is a scalar quality rating value, R, which is computed as

$$R = R_{\mathrm{o}} - I_{\mathrm{s}} - I_{\mathrm{d}} - \mathrm{Ie}_{\mathrm{eff}} + A \tag{22}$$

where R_{o} represents, in principle, the basic signal-to-noise ratio, I_{s} is a combination of all impairments that occur more or less simultaneously to the voice signal, I_{d} represents the impairments caused by the delay, $\mathrm{Ie}_{\mathrm{eff}}$ is the effective equipment impairment factor and A is an advantage factor. From the calculation of R based on network metrics, the correspondent MOS value, $\mathrm{MOS}_{\mathrm{CQ}}$, for conversational quality can be derived as

$$\mathrm{MOS}_{\mathrm{CQ}} = 1 + 0.035R + R\,(R-60)\,(100-R)\,7\times10^{-6} \tag{23}$$

Equation 23 is valid for $0 < R < 100$. For $R < 0$, $\mathrm{MOS}_{\mathrm{CQ}} = 1$ and for $R > 100$, $\mathrm{MOS}_{\mathrm{CQ}} = 4.5$. Table 1 shows the relationship between the R-value of the E-model, the MOS score and the equivalent degree of user satisfaction.

With the use of standard parameters [12], the results that characterize a high quality voice transmission are obtained with $R = 93.2$. Taking into account only network impairments such as delay, jitter, and packet loss, Eq. 22 can be rewritten as

$$R = 93.2 - I_{\mathrm{d}} - \mathrm{Ie}_{\mathrm{eff}} \tag{24}$$

Assuming that perfect echo cancellation exists in the network, the delay degradation factor is reduced to zero when one-way delay is below or equal to 100 ms ($T_{\mathrm{a}} \leq 100$ ms). For $T_{\mathrm{a}} > 100$ ms, I_{d} is computed as

$$I_{\mathrm{d}} = 25\left\{\left(1+x^6\right)^{\frac{1}{6}} - 3\left[1+\left(\frac{x}{3}\right)^6\right]^{\frac{1}{6}} + 2\right\} \tag{25}$$

where

$$x = \frac{\log\left(\frac{T_{\mathrm{a}}}{100}\right)}{\log 2} \tag{26}$$

The parameter T_{a} is composed by the sum of all the delays that occur from the initiation of a voice signal until the reception and identification of the identical signal at the received output speaker. It includes delay introduced by the codec and the time necessary to accomplish all operations until a packet is sent. Once a voice packet is put in the network, it will suffer delays due to the propagation time of the physical communication channels, the transmission delay associated to channel capacity, and the delay introduced by network nodes. Packets which arrive at its destination still have to experience delays produced by the unpacking and decoding processes and one more additional delay caused by the jitter buffer at the receiver. In the model presented in this paper, the value of T_{a} is computed as

$$T_{\mathrm{a}} = T_0 + q + f(\sigma) \tag{27}$$

where T_0 is the end-to-end delay without including the delays in the queue (which depends basically on the topology and the links transmission parameters) and $f(\sigma)$ is a function of the jitter that depends on the algorithm used in the jitter buffer at the receiver. Throughout this paper it will be used

$$f(\sigma) = 4\sigma \tag{28}$$

Table 1 Relationship between the R factor and MOS score

User satisfaction	R factor (lower limit)	MOS (lower limit)
Very satisfied	90	4.34
Satisfied	80	4.03
Some user dissatisfied	70	3.60
Many user dissatisfied	60	3.10
Nearly all user dissatisfied	50	2.58

In all the studied scenarios, the use of Eq. 28 guarantees a sufficiently wide margin to absorb the jitter produced in the queues of the routers with a negligible discard rate.

The degradation factor introduced by the use of low bit rate codecs, Ie, is tabulated in ITU-T recommendation G.113 [27]. The value of the effective equipment impairment factor when the encoders operation is under conditions of random packet loss is computed as

$$\mathrm{Ie}_{\mathrm{eff}} = \mathrm{Ie} + (95 - \mathrm{Ie})\,\frac{p}{p + B_{\mathrm{pl}}} \tag{29}$$

where the packet loss robustness factor, B_{pl}, is defined as a codec-specific value in [26] and p is computed through Eqs. 7, 14, or 18 depending on the AQM algorithm under consideration.

3 Results and discussion

In this section, the results obtained by applying the analytical procedure proposed above to obtain the mean opinion score for voice applications are presented. Three network scenarios are considered providing elements about the validity of the proposed theoretical model and about the performance of some AQM schemes in the presence of voice traffic.

3.1 Experiment setup

The network scenario was composed by a congested connection where voice applications were mixed with a variable number of TCP sources as shown in Fig. 3 The round trip time for TCP connections were varied in the range of 20 to 400 ms. The maximum TCP's segment size was set to 1,000 bytes which is approximately the average MSS value commonly observed

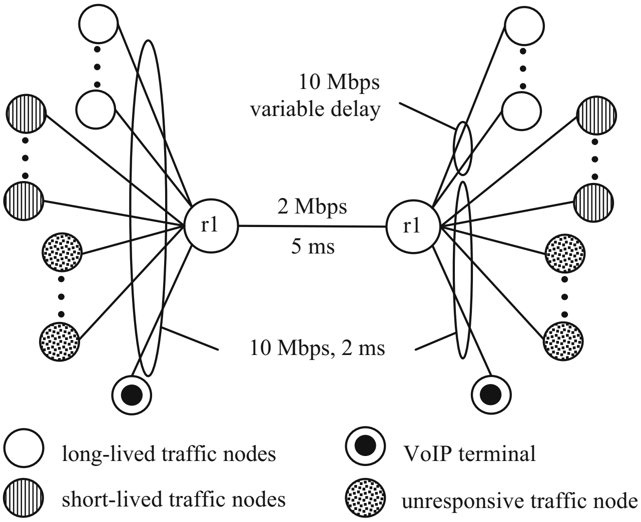


Fig. 3 Network simulation topology

in the Internet [23, 26]. The AQM mechanism under study was activated in routers having a queue capacity of 120 packets. The target queue size was set to 20 packets. This represents an average queue delay of approximately 40 ms considering a mean packet size of 500 bytes and a link capacity, *C*, of 2 Mbps. The voice calls were simulated using constant bit rate traffic sources. In doing so, UDP sessions were established with a payload of 92 bytes per packet, 80 of them corresponding to the audio samples and 12 to the real-time transport protocol header. Packets were sent at a rate of 100 packets/s to simulate a G.711 codec.

Simulations were conducted using the network simulator. Each simulation was replicated starting calls sequentially by intercalating idle periods of 10 s between the end and the beginning of consecutive calls. The number of replications was adjusted to obtain estimations with a confidence interval of less than 5% of the mean value with 95% confidence.

3.2 Predominant TCP background

The MOS values obtained for different AQM mechanisms are shown in Fig. 4. The number of TCP sources was increased to observe the performance of the algorithms in conditions of severe congestion. The results obtained using the fixed point approach are represented in continuous lines while the dashed lines show the simulation results.

In all cases, a great similarity is observed between the analytical results and the results of the simulation. For the case of a DT queue, the quality of the voice calls degrades with the increase of the number of TCP sources;

Fig. 4 Analytical and simulated MOS values

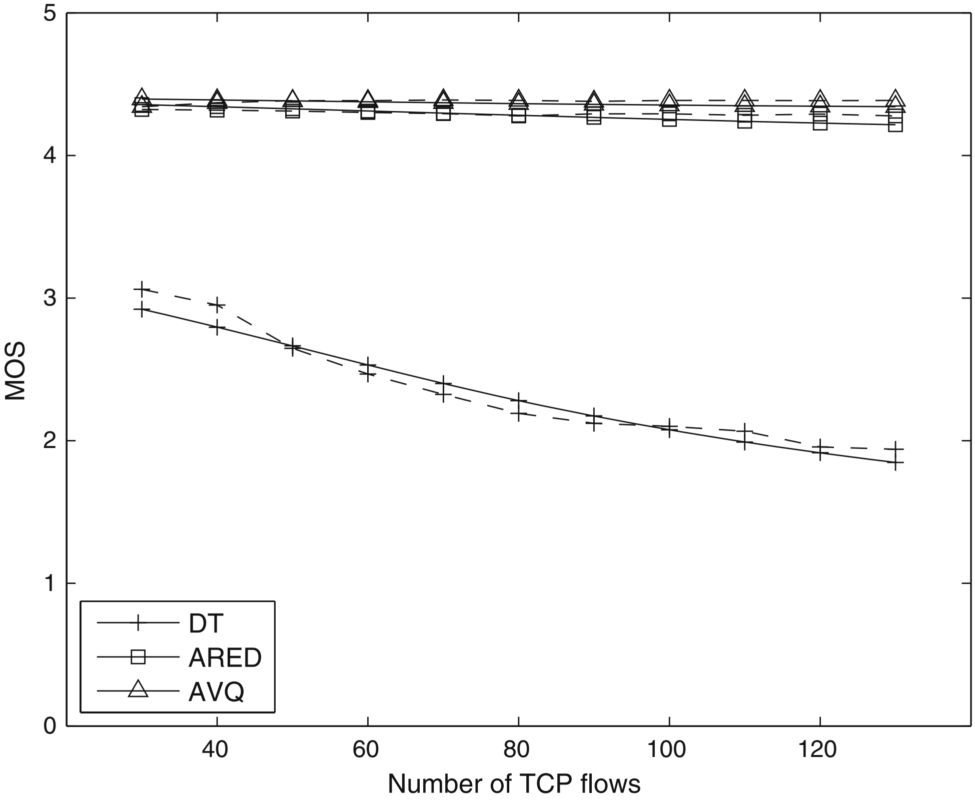
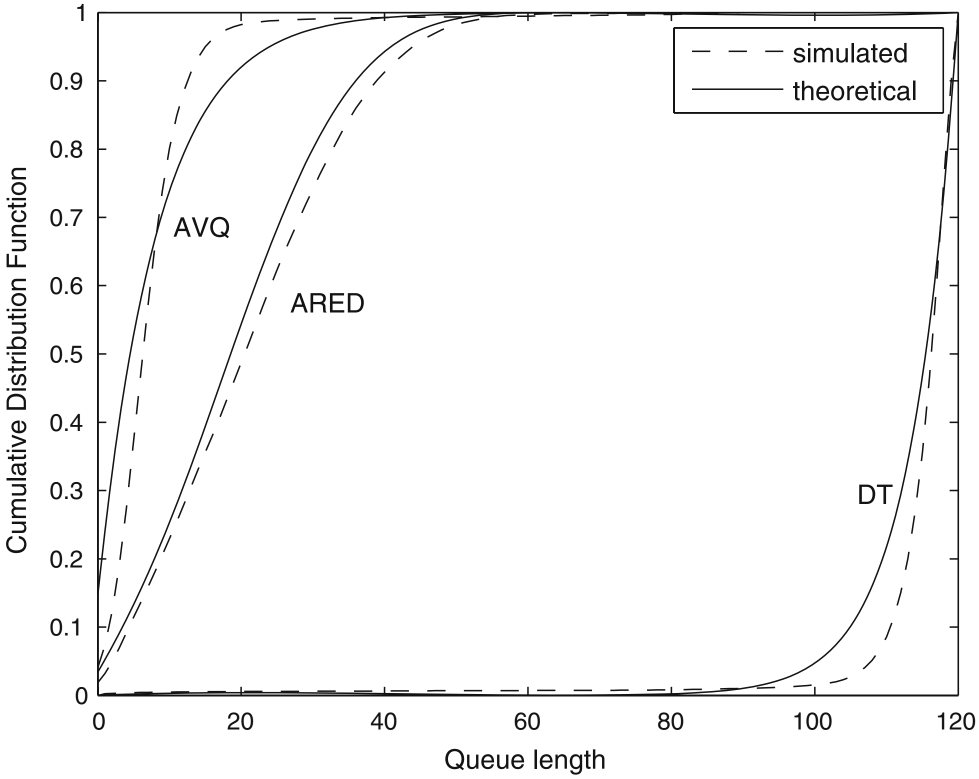


Fig. 5 Cumulative probability function of the queue



it is predictable that with the increasing congestion the packet loss rate and the delay will increase too and, therefore, the MOS values will decrease. However, when ARED is implemented and configured in byte mode, the loss rate experienced by the voice packets is low when compared with that of the TCP packets that have bigger size (MSS = 1,000). Furthermore, as it is shown in Fig. 5 the mean value of the queue occupancy distribution is near the desired value of 20 packets. As a result, the MOS achieved with ARED corresponds to a high user satisfaction and, as it is verified in Fig. 4, this result holds when the number of responsive traffic sources is increased.

When the AVQ algorithm is implemented in the routers, the results are even better because the loss rate and the queue average waiting time are smaller in comparison with ARED. The AVQ configuration used measures the size of the queue in bytes so, bigger packets will be more affected for the loss probability while the queue occupation is maintained in low levels. As in ARED, with the use of AVQ the quality of voice calls does not suffer with the increase of the number of TCP sources. These results confirm the statements done in [3] about the benefits that real time applications receive from AQM mechanisms.

The above-mentioned is valid for the analyzed network scenario, which reflects some of the typical characteristics of the global network links. However, due to the heterogeneity of the Internet, these characteristics can vary in some network segments. In the next subsection, the performance of the AQM algorithms is analyzed when the traffic pattern is altered.

3.3 Variations in TCP MSS

In the previous subsection, it was considered a TCP maximum segment size of 1,000 bytes which corresponds to the average MSS value observed in the Internet [21]. Since apparently the benefits obtained by using AQM are much related to the bias of the packet loss rate due to bigger packets, it would be reasonable to hope that the end user-perceived voice

Table 2 MOS for different values of maximum segment size

	DT			ARED			AVQ		
MSS	1,000	500	80	1,000	500	80	1,000	500	80
Analytical	2.07	3.17	4.28	4.25	4.16	4.24	4.35	4.37	4.24
Simulation	2.10	3.12	4.19	4.29	4.16	4.24	4.38	4.37	4.12

Fig. 6 Occupancy probability function of the DT queue for different values of MSS

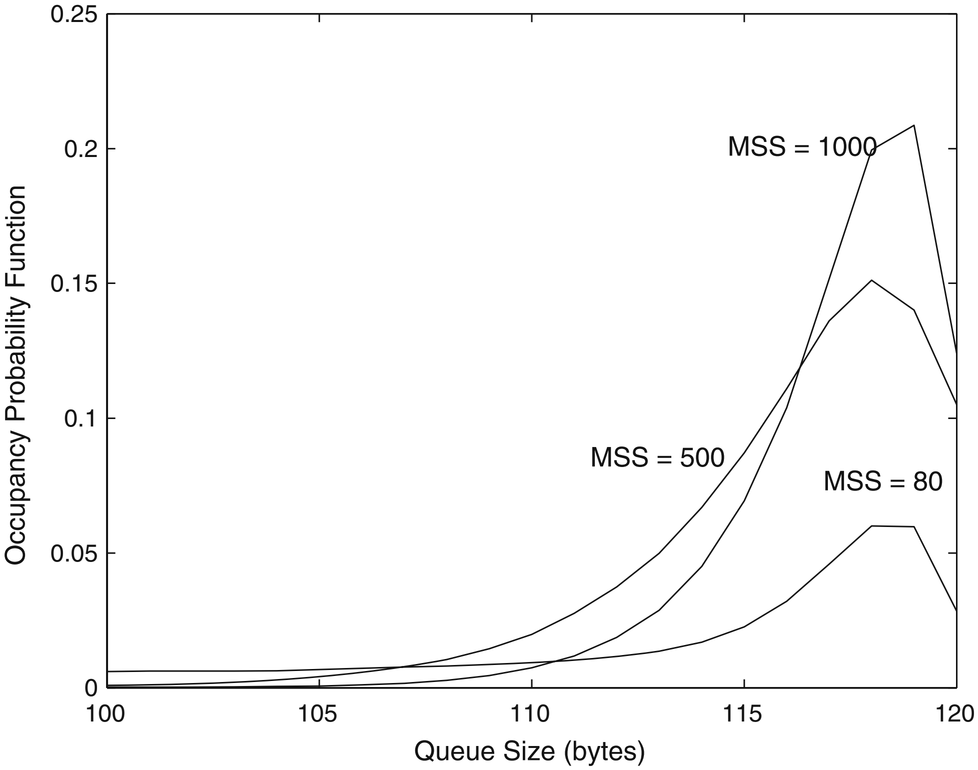
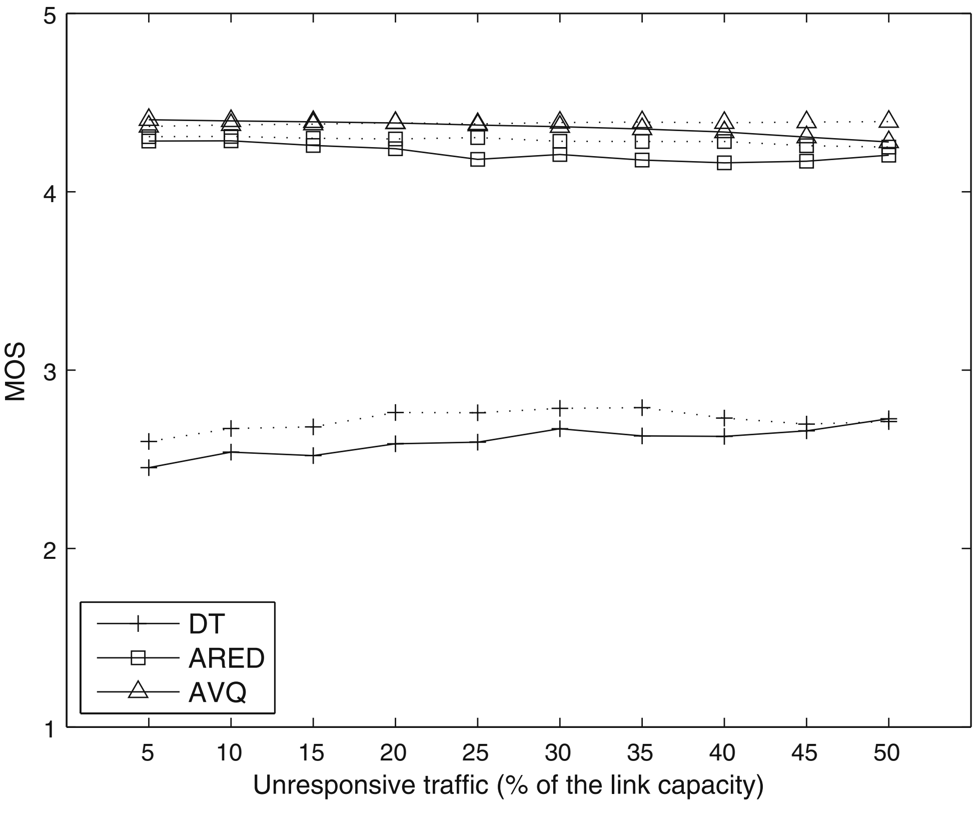


Fig. 7 MOS value with unresponsive traffic (analytical and simulated)



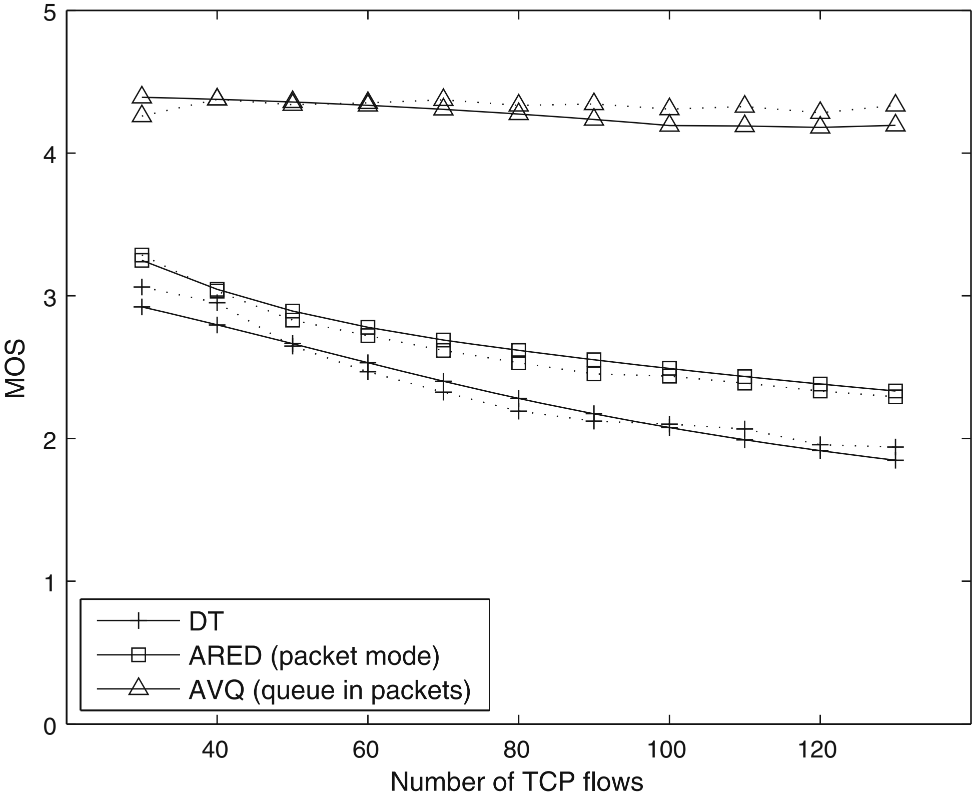


Fig. 8 MOS values when ARED is in packet mode and the AVQ queue is measured in packets

quality may degrade as the size of the TCP packets becomes comparable to the size of the voice packets. Table 2 shows the values of the mean opinion score for different maximum segment size values obtained with 100 homogeneous TCP sources (i.e., with identical configuration parameters).

When the size of the segments is decreased, the transmission time is also decreased. This leads to an increase in the service rate. As a result, a decrease in the queue average waiting time is produced in all cases. In particular, for a DT queue, this decrease is significant and, together with the displacement of the queue occupation distribution moving away from its physical limit and causing less packets loss (see Fig. 6), it produces a gradual increment of the MOS as the value of the TCP maximum segment size decreases.

The AQM schemes continue offering a satisfactory quality of service for the VoIP applications. This means that ARED as well as AVQ are able to satisfy their design parameters without imposing a noticeable packet loss rate (inferior to 1%) for the users of the voice services. Clearly, if the number of sources is increased it is more difficult to regulate its transmission rate and, consequently, a deterioration of the quality of service will take place in all the cases.

3.4 Increasing the unresponsive traffic

In the previous example, the traffic that was mixed with the voice applications was composed exclusively for TCP-based traffic flows. In this sense, a good performance of the AQM algorithms is predictable since they were projected basically to work with this kind of traffic. The next experiment includes the presence of unresponsive traffic in conjunction with the TCP-based traffic flows.

Table 3 Relative error for different experiments

Experiment	DT		ARED		AVQ	
	Mean	Max	Mean	Max	Mean	Max
Predominant TCP background (Fig. 4)	0.0315	0.0532	0.0068	0.0147	0.0062	0.0122
Variations in TCP MSS (Table 2)	0.0173	0.0215	0.0031	0.0093	0.0120	0.0291
Increasing the unresponsive traffic (Fig. 7)	0.0444	0.0634	0.0163	0.0283	0.0090	0.0261
Changes in AQM operation (Fig. 8)	0.0315	0.0532	0.0219	0.0401	0.0189	0.0323

Figure 7 shows the MOS values in the presence of a growing number of UDP flows that represents an amount of traffic up to half the available capacity. The number of TCP sources was fixed to 50. Once again, an appropriate correspondence between the analytical results and the results of the simulation is noticed. ARED and AVQ provide a satisfactory quality of service while DT exhibits a bad performance. In all cases, quality degradation is not observed with the increase of the unresponsive traffic. This is due to an unequal distribution of the available capacity, scenario in which the UDP sources are benefited. TCP connections reduce their transmission rate allowing unresponsive traffic to occupy a larger bandwidth.

3.5 Changes in AQM operation

All the previous results were obtained operating ARED in byte mode and expressing the size of the AVQ queue in bytes. These parameters were then altered in the configuration of the AQM algorithms to observe their effect in the quality of service of the voice applications. Figure 8 shows the QoS degradation when ARED is configured in packet mode. Now the chances for a flow to have a packet discarded is related to the number of transmitted packets and not the number of transmitted bytes. The loss rate experienced by the voice traffic is comparable to that of the TCP packets and rises with the increase of the traffic payload. As a result, ARED exhibits a similar performance to that observed with DT. In the case of AVQ, the modification in the queue measurement unit did not significantly alter the quality of service of VoIP.

In all conducted experiments, it was verified that the theoretical predictions match the simulation results with acceptable accuracy. Table 3 shows the values of the mean and maximum relative error. The worst case was observed using DT while increasing the number of unresponsive traffic with a mean and maximum relative error of 4.44% and 6.34%, respectively. The curves in Fig. 7 show that the analytical model tends to be pessimistic and therefore in this case the approximation errors could be considered as a safety margin for design purposes.

4 Conclusions

Simulation tools have been traditionally used to evaluate AQM performance in complex scenarios. In this paper, a new analytical method was proposed to evaluate the impact of active queue management on the quality of service of voice over IP systems. The numeric examples above described evidence that the proposed analytical model for estimating the user satisfaction gives similar results to that obtained through computer simulation. The results agree with previous works evidencing that the use of some AQM schemes provide a better voice quality than traditional drop-tail queues. The proposed method can assist in the deployment of VoIP systems, providing useful information to network designer with a high degree of accuracy in a cost-effective manner.

References

1. Network Working Group (1998) RFC 2309—recommendations on queue management and congestion avoidance in the internet. http://www.faqs.org/rfcs/rfc2309.html. April
2. Floyd S, Jacobson V (1993) Random early detection gateways for congestion avoidance. IEEE/ACM Trans Netw 1(4): 397–413
3. Hollot C, Misra V, Towsley D, Gong W (2002) Analysis and design of controllers for AQM routers supporting TCP flows. IEEE Trans Automat Contr 47(6):945–959
4. Huggard M, Robin M, Bitorika A, McGoldrick C (2004) Performance evaluation of fairness-oriented active queue management schemes. In: Proceedings of IEEE international symposium on modeling, analysis, and simulation of computer and telecommunication systems (MASCOTS). Volendam, The Netherlands, pp 105–112
5. Reguera VA, Paliza FA, Fernandez EMG, Godoy Jr W (2008) On the impact of active queue management on voice over IP quality of service. Comput Commun 1(1):73–87
6. Szigeti T, Hattingh C (2004) End-to-end QoS network design: quality of service in LANs, WANs, and VPNs. CISCO, Indiana
7. Bu T, Liu Y, Towsley D (2006) On the TCP-friendliness of VoIP traffic. In: Proceedings of IEEE Infocom. Barcelona, Spain, pp 1–12
8. Tao S, Xu K, Estepa A, Fei T, Gao L, Guerin R, Kurose J, Towsley D, Zhang Z (2005) Improving VoIP quality through path switching. In: Proceedings of IEEE Infocom. Miami, FL
9. Floyd S, Kohler E (2003) Internet research needs better models. SIGCOMM Comput Commun Rev 33(1):29–34
10. May M, Bolot J, Diot C, Lyles B (1999) Reasons not to deploy RED. In: Proceedings of the 7th int. workshop on quality of service (IWQoS '99). London
11. Kunniyur S, Srikant R (2004) An adaptive virtual queue (AVQ) algorithm for active queue management. IEEE/ACM Trans Netw 12(2):286–299
12. The E-model (2005) A computational model for use in transmission planning. ITU-T Recommendation G.107
13. May M, Bonald T, Bolot J (2000) Analytic evaluation of RED performance. In: Proceedings of IEEE Infocom, vol 3. Tel Aviv, Israel, pp 1415–1424
14. Casetti C, Meo M (2000) A new approach to model the stationary behavior of TCP connections. In: Proceedings of IEEE of Infocom, vol 1. Tel Aviv, Israel, pp 367–375
15. Casetti C, Meo M (2001) An analytical framework for the performance evaluation of TCP Reno connections. Comput Netw 37(5):669–682

16. Bu T, Towsley D (2001) Fixed point approximations for TCP behavior in an AQM network. In: Proceedings of ACM Sigmetrics. Cambridge, Massachusetts, pp 216–225
17. Padhye J, Firoiu V, Towsley D, Krusoe J (1998) Modeling TCP throughput: a simple model and its empirical validation. In: Proceedings of ACM SIGCOMM '98. Vancouver, British Columbia, pp 303–314
18. Padhye J, Floyd S (2001) On inferring TCP behaviour. In: Proceedings of ACM SIGCOMM. Ago San Diego, California, pp 287–298
19. Gross D, Harris CM (1998) Fundamental of queuing theory, 3rd edn. Wiley, NY
20. Thompson K, Miller G, Wilder R (1997) Wide area internet traffic patterns and characteristics. IEEE Netw 6: 10–23
21. Fraleigh C (2003) Packet-level traffic measurements from the sprint IP backbone. IEEE Netw 17(6):6–16
22. Chakraborty D, Ashir A, Suganuma T, Keeni GM, Roy TK, Shiratori N (2004) Self-similar and fractal nature of internet traffic. Int J Netw Manag 14:119–129
23. Cao J, Ramanan K (2002) A Poisson limit for buffer overflow probabilities. In: Proceedings of IEEE INFOCOM. New York
24. Tijms HC (1994) Stochastic models: an algorithmic approach. Wiley, New York, NY
25. Floyd S, Gummadi R, Shenker S (2001) Adaptive RED: an algorithm for increasing the robustness of RED's active queue management. http://www.icir.org/floyd/papers/adaptiveRed.pdf
26. Methods for subjective determination of transmission quality, ITU-T Recommendation P.800 (1996)
27. Provisional planning values for the equipment impairment factor Ie and packet-loss robustness factor Bp, ITU-T Recommendation G.113, Appendix I (2002)