Resource Allocation and Management in DiffServ Networks for IP Telephony

Maarten Büchli, Danny De Vleeschauwer, Jan Janssen, Annelies Van Moffaert, Guido H. Petit
Alcatel Bell, Network Strategy Group
Francis Wellesplein 1
B-2018 Antwerp, Belgium
+32 3 240 7081

maarten.buchli@alcatel.be

ABSTRACT

This paper discusses resource allocation and management in Differentiated Services (DiffServ) networks, particularly in the context of IP telephony. We assume that each node uses Weighted Fair Queuing (WFQ) schedulers in order to provide Quality of Service (QoS) to aggregates of traffic. All voice traffic destined for a certain output interface is aggregated into a single queue. When a voice flow traverses a node, its packets are placed in this queue; which is drained at a certain rate, determined by a weight associated with that queue. This paper shows how to set this weight such that the edge-router-to-edge-router queuing delay in the DiffServ network is statistically bounded. The nodes are modeled as M/G/1 queuing systems and a heuristic formula is used to compute a quantile of the queuing delay. This formula is compared with results derived from simulations. It is also shown how to use this result to allocate the appropriate resources in a DiffServ network and how to update and process RSVP messages at the edge of a DiffServ network in an IntServ over DiffServ scenario.

Keywords

DiffServ, WFQ, Voice over IP (VoIP), resource allocation.

1. INTRODUCTION

IP is currently one of the enabling technologies for multi-service networks. These networks may, among other services, provide telephony services similar to the current PTSN. Because real-time voice traffic has very strict delay requirements a certain Quality of Service (QoS) is required in the network with regard to bandwidth, delay and packet loss. One of the QoS architectures introduced by the IETF is the Differentiated Services (DiffServ) model [1]. In this architecture traffic requiring the same forwarding behavior is aggregated into a single queue. For example, there may be separate queues for real-time and data traffic. Packets are classified based on their DiffServ Code Point (DSCP) to put them into the correct queue. However, a single queue for all real-time (voice, video) traffic is not likely to be sufficient. In [5] it is shown that mixing voice and MPEG video traffic in the same queue has a serious impact on the delay performance of the voice traffic. Hence, it is advantageous to put voice traffic in a different queue from, for example, highly variable video traffic. A scheduler is needed in order to guarantee throughput and delay to the different aggregate flows in the system. In [10] the maximum load of the voice queue is determined when a Head of Line (HoL) priority scheduler is used, while adhering to a certain statistical delay constraint. However, since the HoL scheduler can completely starve the service given to the best-effort data queue we will consider class-based Weighted Fair Queuing (WFQ). For WFQ the queuing delay will be larger than for HoL priority scheduling but when WFQ is used the best-effort data queue is guaranteed a minimum service rate. This is because the voice queue is only served at a fraction of the link capacity as opposed to HoL where the complete link capacity is in principle available for voice traffic. The queuing delay is strongly impacted by the weight assigned to the voice queue [5]. Dimensioning the bandwidth (i.e. setting the weight) for the voice traffic is therefore of great concern. The dejittering buffer at the receiver has to compensate for the queuing delay. When a static dejittering delay is used it should be chosen equal to the maximum, or a quantile, of the queuing delay.

This paper focuses on (statistical) delay guarantees for voice traffic since telephony is a highly delay-sensitive application. Adaptive coding is not considered in this paper since it can result in a variable voice quality, which is not desirable in many cases. The contribution of this paper is twofold. First, it is shown how to set the weights of a WFQ scheduler in a DiffServ router such that a certain delay bound is met. Secondly, it is shown how this delay analysis can be applied in the context of IntServ/DiffServ interworking scenarios [2] where RSVP signaling is used to collect delay information of the end-to-end path and to make resource reservations.

The rest of this paper is organized as follows. In section 2 the model for the aggregated voice traffic is introduced. The considered router architecture is described in section 3. The analysis to calculate the queuing delay is presented in section 4. Simulation results to verify the formula are presented in section 5. The obtained formula is used in section 6 to calculate the maximum load for different delay constraints. Section 7 shows how these results can be applied for an IntServ/DiffServ interworking scenario where RSVP is used for QoS signaling. Finally, conclusions are drawn in section 8.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

NOSSDAV '01, June 25-26, 2001, Port Jefferson, NY.

Copyright 2000 ACM 1-58113-000-0/00/0000…$5.00.

2. VOICE TRAFFIC MODEL

When voice is transported over packet-based networks the analog voice is first sampled (usually at 8 kHz) and is then (linearly) quantitized. The next step is to encode this digital voice signal. After encoding, one or more codewords are grouped and a packet header is attached. The packetization delay is defined as the time to fill a packet with voice codewords. The header of an IP packet consists of IP/UDP/RTP overhead and has a size of 40 bytes (S_{OH}=320 bit). This overhead makes that the bandwidth needed in the network is larger than the codec bit rate. The voice packet size M is related to the bit rate of the codec R_{cod}, the header overhead S_{OH} and the packetization delay T_{pack} as follows

$$M = (R_{cod} \cdot T_{pack}) + S_{OH} \text{ [bit]} \tag{1}$$

In Table 1 several standardized codecs are listed that may be good candidates for use in Voice over IP (VoIP) applications. The bit rate and the granularity are shown for each codec. The granularity is determined by the duration of the voice signal interval that is encoded in just one codeword. The packetization delay is always a multiple of the granularity of the codec that is used.

CODEC	Bitrate [kb/s]	Granularity [ms]
G.711	64	0.125
G.726	16 / 24 / 32 / 40	0.125
G.728	12.8 / 16	0.625
G.729	8	10
G.723.1	5.3 / 6.3	30
GSM-FR	13	20

Table 1 : Bit rates and granularity of codecs

The bit rate of the listed codecs is fixed. Multirate codecs, e.g. the Adaptive Multi Rate (AMR) codec developed for the use in UMTS networks, are not considered here. The packetizer (IP phone or gateway) has to choose a certain packetization delay. This will result in a packet flow with deterministic packet interdeparture times and fixed packet sizes.

In [3][4] it is argued that an aggregate of Constant Bit Rate (CBR) voice flows is well modeled by a Poisson process. Moreover, they also state that the superposition of (nearly) CBR sources cannot become more bursty than a Poisson process. Hence, a Poisson arrival process assumption is worst case with respect to queuing delay for an aggregate of CBR voice flows. Note that it is assumed that the different voice packet streams are not synchronized such that packets do not all arrive at the same time instant at the router.

Voice sources might use Voice Activity Detection (VAD). In this case voice packets are only sent during the talk spurts. During the silence periods nothing or occasionally (when the characteristics of the background noise change) some packets are sent. Voice sources that use VAD are much harder to characterize. However, in [9] a model for the speech pattern is presented for an average interactive phone conversation. It is an on-off model with exponential distributed length of the on- (speech) and off- (silence) periods. When several on-off sources are aggregated the resulting process depends on the considered time-scale [12]. Over small time scales the process will behave as a Poisson process but over longer time scales it will become more bursty than a Poisson process. The relevant timescale of the arrival process (i.e. whether it behaves as a Poisson process or not) depends on the maximum queuing delay. In case the Poisson assumption holds, the analysis presented here can be applied. Determining over which time scale such a Poisson assumption holds for an aggregate of on-off sources is a study on itself and beyond the scope of this paper.

3. QUALITY OF SERVICE

In this paper we assume that the DiffServ routers use WFQ scheduling to support QoS. Hence, all voice traffic destined for a certain output interface is aggregated into a single queue. Due to the tight delay constraints of telephony it is not recommended to mix voice and video traffic. The delay will increase significantly when voice traffic is mixed with, for example, MPEG video traffic [5][7]. In Figure 1 a system is shown with queues for different applications.

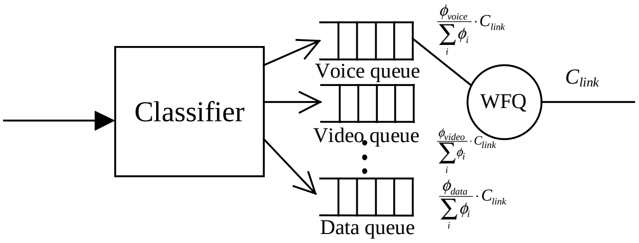
$\frac{\phi_{voice}}{\sum_i \phi_i} \cdot C_{link}$
Classifier
Voice queue
Video queue
Data queue
WFQ
C_{link}
$\frac{\phi_{video}}{\sum_i \phi_i} \cdot C_{link}$
$\frac{\phi_{data}}{\sum_i \phi_i} \cdot C_{link}$

Figure 1: Class Based Weighted Fair Queuing

The classifier reads the DiffServ code point in the IP header to determine whether the packet should be put in the voice or in another queue. Each queue i is assigned a weight ϕ_i such that a certain minimum service rate is guaranteed to that queue. This capacity assigned to voice traffic is denoted by C_{voice} and is given by

$$C_{voice} = \frac{\phi_{voice}}{\sum_i \phi_i} \cdot C_{link} \tag{2}$$

Unused bandwidth will be shared in a fair manner among the backlogged queues. We assume however that there is always data to be processed in the other queues. This assumption can be justified by the fact that data sources are TCP controlled and will always try to utilize the available bandwidth completely. In the next section only the voice queue is considered. The voice queue can be studied in isolation from the other queues because it is assigned a minimum amount of guaranteed bandwidth that is independent of the traffic in the other queues of the system. Hence, with B_{voice} defined as the average bit rate of the aggregate of voice flows the load of the voice queue is given by

$$\rho = \frac{B_{voice}}{\frac{\phi_{voice}}{\sum_i \phi_i} \cdot C_{link}} \tag{3}$$

4. END-TO-END DELAY ANALYSIS

In order to provide IP telephony services with PSTN quality it is very important to keep the mouth-to-ear delay within acceptable bounds. This delay consists of several components that are discussed in §4.1. It is especially important to keep the queuing delay under control since it has to be compensated for in a dejittering buffer. In §4.2 a method is introduced to calculate the queuing delay in a DiffServ network where WFQ schedulers are used.

4.1 Mouth-to-ear delay

An important parameter for interactive voice communications is the Mouth-to-Ear (M2E) delay. This is the time between the moment the sending party has spoken a word and the moment it is heard by the receiving party. The M2E delay consists of a deterministic and a stochastic part. The deterministic part consists of packetization T_{pack}, serialization T_{ser}, propagation T_{prop}, dejittering $T_{dejitter}$ and other (encoding, decoding etc.) delays T_{oth}. The stochastic part consists of the queuing delay of all traversed nodes. Although the queuing delay is a stochastic quantity, we are only interested in the maximum queuing delay, i.e. the delay of the slowest possible packet, because if this one arrives in time for play-out, all others do too. For this maximum queuing delay the absolute maximum or a reasonable quantile (for instance the 99% quantile) can be used since this is the fraction of packets that arrive in time. Packets that arrive too late for play-out are considered to be lost. Packet losses of 1% or lower are acceptable for voice (depending on the type of codec and the use of packet loss concealment algorithms). Taking all delay components into account, the mouth-to-ear delay can be written as

$$\hat{T}_{m2e} = \hat{T}_{queue} + T_{dejitter} + T_{pack} + T_{prop} + T_{ser} + T_{oth} \qquad (4)$$

An example of a delay distribution is shown in Figure 2. The tail distribution of the delay is caused by the stochastic queuing delay.

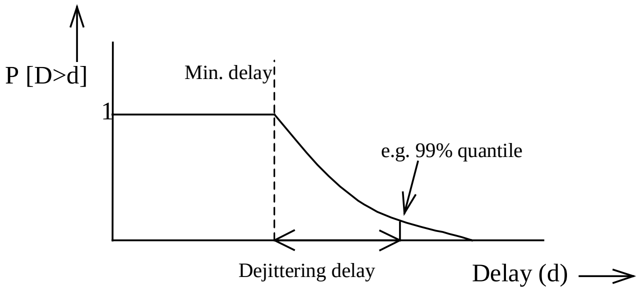

Figure 2: Example of a delay distribution

The queuing delay has to be compensated for in the dejittering buffer of the receiver. Hence, the dejittering delay should be chosen equal to a quantile of the queuing delay in order to prevent packets arriving too late for play-out. This is the delay denoted by the arrow with the caption ‘dejittering delay’ in Figure 2. The packets with a delay larger than the specified quantile may be lost (depending on the queuing delay of the first packet) because they will arrive too late for play-out.

Adaptive dejittering algorithms also exist, they estimate the queuing delay of the first packet. When perfect adaptive dejittering is used the queuing and dejittering delay of each packet is equal to the maximum queuing delay. However, it is questionable whether adaptive dejittering algorithms can converge fast enough during the typical length of a phone call.

4.2 Queuing delay

It has been proven in [11] that the worst case end-to-end queuing delay in a network of WFQ schedulers is deterministically bounded for (r, b) leaky bucket constraint flows, with r the sustainable rate and b the maximum burst size. When the reserved rate R in each node is equal or larger than r, the end-to-end queuing delay is bounded by

$$\hat{D}_{e2e} \le \frac{b}{R} + (N-1)\cdot\frac{M}{R} + \sum_{i=1}^{N}\frac{MTU_i}{C_{link}^{i}} \qquad (5)$$

R	reserved rate
b	maximum burst size
M	maximum voice packet size
N	number of hops
MTU_i	MTU at node i
C^{i}_{link}	Link capacity at node i

The first term denotes the maximum queuing delay caused by a burst b. This term occurs only once because the burst has to be smoothed out just once. The second term denotes the maximum queuing delay in the consecutive nodes. The third term takes into account the non-pre-emptiveness of the system. In other words, a data packet may be in service when a packet of the considered flow is eligible for service.

However, the delay bound defined above applies only when per flow scheduling is used (i.e. IntServ). In DiffServ routers traffic with the same codepoint is aggregated in a single output queue and split again at the next node. This is illustrated in Figure 3. Since aggregation takes place at each router, the arriving traffic will be bursty at each node. Hence, in the worst case the delay due to bursts occurs in each node on the end-to-end path.

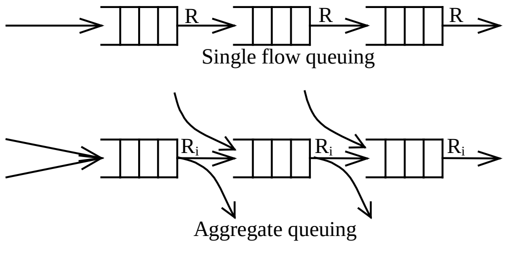

Figure 3: End-to-end queuing scenarios

If the aggregated (voice) traffic is bounded by the traffic envelope (r_i, b_i) at the output queue of each node i the worst case queuing delay is defined by (with $R_i \ge r_i$)

$$\hat{D}_{e2e} \le \sum_{i=1}^{N}\frac{b_i}{R_i} + \sum_{i=1}^{N}\frac{MTU_i}{C_{link}^{i}} \qquad (6)$$

R_i	reserved rate at node i
b_i	maximum burst size at node i

Due to the Poisson assumption we made for the aggregated voice traffic in section 2 we model each DiffServ node as a M/G/1 queuing system. A consequence of mixing the traffic as shown in Figure 3 is that the arriving traffic is not leaky bucket constraint. Because we consider constant bit rate voice sources we assume the arrival process is Poisson. Hence, instead of the delay caused by a maximum burst size at each node, $\Sigma_i\, b_i/R_i$, we use a quantile (e.g. 99%) of the queuing delay of N M/G/1 queues in tandem. Because we use a quantile, the delay bound becomes statistical instead of deterministic, and hence, is tighter. When a voice packet arrives at an empty queue the maximum queuing delay in the WFQ system is M/R. Hence, this delay has to be added for each node on the path. Finally, the maximum error between a WFQ and the Generalized Processor Sharing (GPS) reference system has to be taken into account, this results in the following (statistical) delay bound

$$\hat{D}_{e2e} \le \hat{D}_{N*M/G/1} + \sum_{i=1}^{N}\left(\frac{M}{R_i} + \frac{MTU_i}{C_{link}^{i}}\right) \qquad (7)$$

Where the reserved rate R_i is equal to C_{voice} for all nodes i and $\hat{D}_{N*M/G/1}$ is a quantile of the end-to-end delay of N M/G/1 queues. Note that calculating a quantile of the end-to-end queuing delay provides a much tighter bound than simply adding the delay quantiles of each individual node. For a cascade of M/G/1 nodes the (1-P) quantile of the delay can be approximated with a simple heuristic formula [13]

$$\hat{D}_{N*M/G/1} = \mu_N + \alpha(P)\cdot\sigma_N \qquad (8)$$

The mean μ_N and standard deviation σ_N of the total queuing delay of a network of N identical M/G/1 nodes are respectively

$$\mu_N = N\cdot\frac{\rho}{2(1-\rho)}\cdot\frac{E[S^2]}{E[S]} \qquad (9)$$

$$\sigma_N = \sqrt{N}\sqrt{\left(\frac{\rho}{2(1-\rho)}\cdot\frac{E[S^2]}{E[S]}\right)^2 + \frac{\rho}{3(1-\rho)}\cdot\frac{E[S^3]}{E[S]}}$$

Where E[S] is the (average) service time of a packet in one node, in this case equal to M/R. The load ρ is defined in eq. (3). In case of N identical nodes the factor $\alpha(P)$ can be calculated as $\alpha(P) = \frac{E_N^{-1}(P) - N}{\sqrt{N}}$ where $E_N^{-1}(P)$ is the inverse of the Erlang tail distribution of N stages [8]. The fact that an Erlang distribution is used should be no surprise. The tail distribution of the delay of one node is approximated by an exponential distribution. The sum of N identical exponential distributions results in an Erlang distribution of N stages. Details of the derivation of eq. (8) can be found in [13].

The heuristic formula can be extended for heterogeneous nodes, i.e. nodes with a different load ρ and reserved rate R. However, the factor $\alpha(P)$ and the calculation of μ_N and σ_N have to be modified. Extending the calculation of $\alpha(P)$ for heterogeneous nodes is beyond the scope of this paper.

In the rest of the paper we consider, for illustrative purposes, voice packets with a fixed size. Therefore, the M/D/1 model is used.

5. SIMULATION

In order to verify the analysis of the previous section some simulations have been done with OPNET simulation software.

5.1 Simulation scenario and parameters

The simulation scenario is depicted in Figure 4. The WFQ system contains a voice and a data queue and is connected to an E3 link (34 Mb/s). The aggregate bit rate of the voice traffic is 2 Mb/s. The queue size for voice is dimensioned such that no packet loss occurs. The simulation has been done for a single hop.

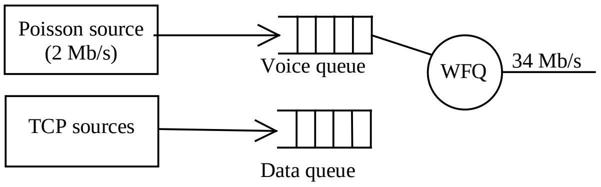

Figure 4: Simulation scenario

In Table 2 the characteristics of both the voice and data source are shown. The data sources are TCP controlled. Since only one hop is considered it will be the bottleneck link for TCP. Hence, the data queue will be almost always backlogged. Two TCP sources have been used. One with short file transfers and one with very long file transfers.

Traffic	Type	Packet size [byte]	Packet rate [pkt/s]	Bit rate [Mb/s]
voice	Poisson	200	1250	2
data	TCP	576	-	-

Table 2: Traffic sources

The simulation has been repeated for different weight settings of the voice queue such that the capacity assigned to voice varied from 2.1 Mb/s to 3.9 Mb/s. During the simulation the queuing delay of each voice packet is collected. From these delay values a Tail Distribution Function (TDF) was constructed in order to determine the (1-10^{-3})-quantile and a maximum value. The simulated time is 60 seconds (i.e. 60*1250 voice packets).

5.2 Simulation results

For each setting of the weight for the voice queue a TDF was constructed from the simulation data. As an example, the TDF for weight setting of 2.2 Mb/s for voice is shown in Figure 5.

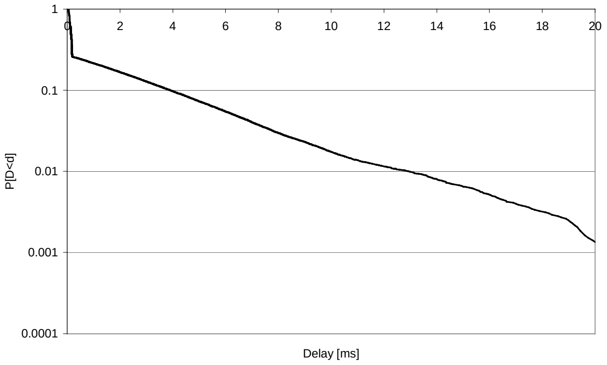

Figure 5: TDF of the queuing delay

Three regions can be distinguished in the TDF depicted in Figure 5. The first one is due to serialization delay of a 200 byte voice packet (0.05 ms). The second region, between 0.05 and 0.18 ms, is due to a data packet that is in service. This is captured by the term $\frac{MTU}{C_{link}}$ in eq. (7). The residual service time of the data packet is uniformly distributed, and hence, there is a straight line in the CDF. The third part, delay larger than 0.18 ms, is the delay due to contention with other voice packets. This delay component is captured by the term $\hat{D}_{N*M/G/1}+\frac{M}{R}$ in eq. (7).

For the different simulations the weight of the voice queue was varied such that the bit rate assigned to voice varied between 2.1 Mb/s and 3.9 Mb/s. Each simulation has been repeated several times with a different seed for the random generator. From each simulation the (1-10^{-3})-quantile and the maximum of the queuing delay was determined. The different obtained values for the (1-10^{-3})-quantile were averaged and from the maximum values of each simulation the largest value was taken. The simulation results are presented in Figure 6 together with the theoretical values calculated with eq. (7). The load of the voice queue is defined by eq. (3) with B_{voice}=2 Mb/s and C_{link}=34 Mb/s.

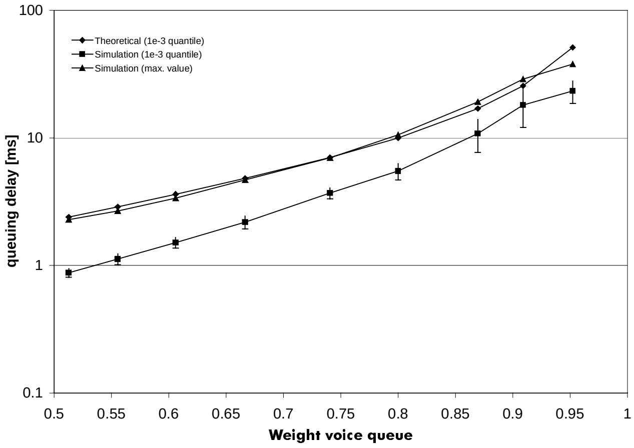

Figure 6: Simulation results

From the comparison in Figure 6 it can be concluded that the calculated quantile gives an upper bound on the delay. Especially for lower loads it is too pessimistic. However, for high loads the calculated bound is tighter. This is also due to the larger confidence intervals.

The analysis and the simulation differ for two reasons. First, the delay measured in the simulation is a direct quantile while in eq. (7) only the first term is a quantile and the other two terms are worst case. Hence, a calculated quantile will always be somewhat higher than the real value. Second, the data queue becomes empty sometimes (for example between two consecutive file transfers). The extra available capacity is then assigned to the voice queue, which will result in a decrease of the effective load. In the simulation the data queue was empty for 1% of the time. This will result in a lower queuing delay.

Summarizing, the analytical method to calculate the queuing delay gives a reasonable delay approximation. For low loads the analysis is too pessimistic. However, it is always an upper bound in the considered cases. For loads larger than 80% the bound is tighter. Hence, the analytic formula provides an easy and efficient way to determine the weight for voice traffic in a WFQ system such that the queuing delay requirement is met.

6. RESULTS

Eqs. (7) and (8) can be applied to determine the maximum tolerable load as a function of the weight assigned to the voice queue under a certain delay constraint. In order to calculate this maximum load eq. (7) has to be inverted (numerically) with respect to ρ, i.e. the tolerable load has to be calculated as function of a given (1-P)-quantile of D_{e2e}. In Figure 7 and 8 the maximum tolerable load is shown as a function of the fraction of the link capacity assigned to the voice queue for respectively N=1 and N=8. These curves are made for a (1-10^{-3})-quantile of 5 ms for the queuing delay, a data MTU of 500 bytes and a fixed packet size for voice of 100 bytes. Curves are shown for link capacities C_{link} up to 30 Mb/s. When the number of hops N is increased the maximum load of the voice queue decreases because the delay budget for queuing has to be shared among more nodes.

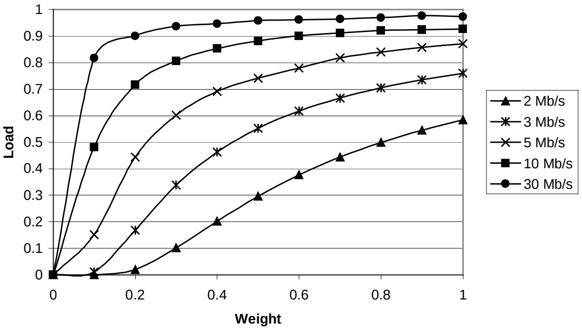

Figure 7: Maximum load as a function of the weight for a queuing delay bound of 5 ms (*N*=1)

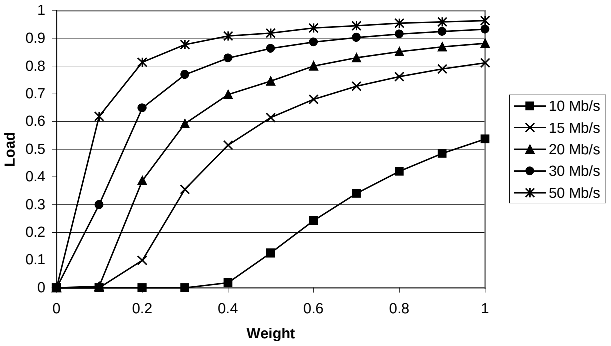

Figure 8: Maximum load as a function of the weight for a delay bound of 5 ms (*N*=8)

The bit rate of the voice aggregate can be calculated with the inverse of eq. (3) when the maximum tolerable load and the weight assigned to the voice queue are known. When the bit rate of a single voice source is known the maximum number of simultaneous calls can be determined.

As an example, we assume that all voice flows use 32 kb/s codecs (e.g. G.726) and use a packetization delay of 30 ms. With a IP/UDP/RTP header of 40 bytes this results in a gross bit rate of approximately 43 kb/s and a packet size of 160 bytes. The MTU of data traffic is assumed to be 1500 bytes. In Figure 9 and 10 the maximum number of calls is shown as a function of the weight assigned to the voice queue in case of a link rate of 10 Mb/s for respectively N=1 and N=8. Curves are shown for different delay bounds. Increasing the capacity reserved for voice increases the number of voice flows that can be supported. From Figure 7 and 8 it is clear that increasing the weight will also cause the maximum tolerable load to increase. This results in curves (Figure 9 and 10) that increase more rapidly than just linearly, especially for small delays. It can also be noted that it is very important to do strict Flow Admission Control (FAC). The curves in Figure 9 for the delay of 9 ms and 15 ms are very close to each other. Hence, this implies that accepting a few calls more in this region will result in a drastic increase of the queuing delay.

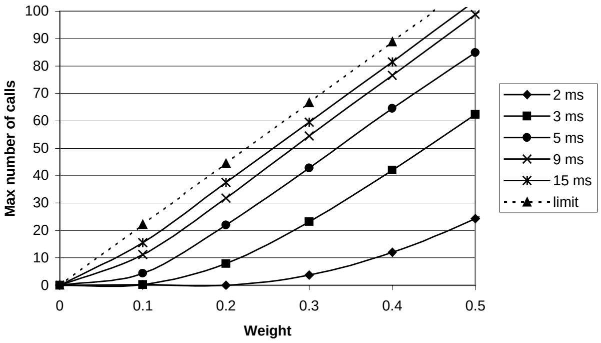

Figure 9: Maximum number of calls as a function of the weight of the voice queue (link capacity 10 Mb/s, *N*=1)

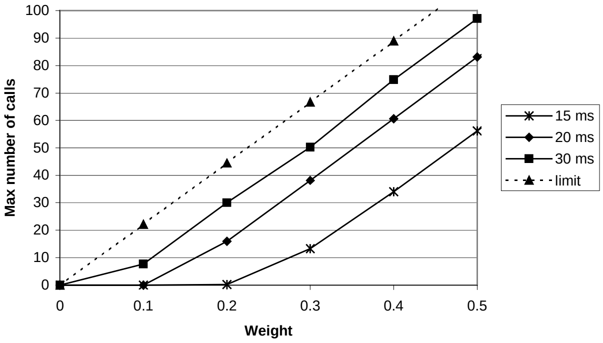

Figure 10: Maximum number of calls as a function of the weight of the voice queue (link capacity 10 Mb/s, *N*=8)

7. VOICE OVER DIFFSERV NETWORKS

This section shows how the results obtained in the previous section can be applied in the context of voice over DiffServ networks. As an example we consider a DiffServ network in an IntServ/DiffServ interworking scenario [2]. In this case the DiffServ network is treated as a single IntServ hop. Hence, when RSVP is used for QoS signaling the edge routers of the DiffServ network should be RSVP aware. They also have to maintain per-flow state information since RSVP makes soft-state reservations. This scenario is depicted in Figure 11.

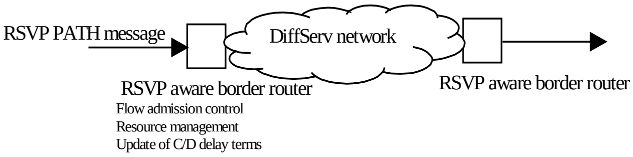

Figure 11: DiffServ network with RSVP aware edges

The edge routers have to perform several functions. First, they have to update the RSVP PATH message with the appropriate delay terms. The rate dependent delay term C should not be increased since the delay in the DiffServ network is not dependent on the rate of a single flow. The rate independent delay term D should be increased with a quantile of the edge-to-edge delay. This delay can be calculated with the method presented in this paper. Second, the router has to perform resource management, i.e. performing flow admission control such that the traffic does not exceed the available resources in the network. This flow admission control is done by accepting or rejecting the RSVP RESV message.

In order to support voice over DiffServ a few steps have to be taken. The first one is to provision the network properly. Between each ingress and egress node a bit pipe has to be provisioned. The capacity of this bit pipe should take delay and the call blocking probability into account. This can be done by using the statistics of the bit rate of voice traffic during busy hour. This bit rate should be increased such that the blocking probability is low enough by applying the Erlang B formula for example. Now, a certain edge-to-edge delay has to be chosen for the bit pipe. By using figures like Figure 9 and 10 the appropriate weight that has to be set can be determined.

Secondly, every edge routers should store the bit rate available for voice traffic and the queuing delays to all other edge routers. The delay information is required to update the D term in the RSVP PATH message and the capacity allocated for voice is needed in order to perform flow admission control. When a RSVP reservation message is received the flow admission criterion is

$$\sum_i p_i + p \leq C^{xy} \qquad (10)$$

Where p_i is the peak rate of the already accepted flows between the edge node x and y and p is the peak rate of the flow for which the resources are requested. C^{xy} is the capacity available for voice traffic (i.e. equal to B_{voice} in eq. (3)). Note that the reserved rate in the DiffServ network (i.e. C_{voice}) is larger than C^{xy} in order to guarantee a certain maximum queuing delay and a certain blocking probability ($C^{xy}=\rho\cdot C_{voice}$). When the criterion of (10) cannot be satisfied, the RSVP reservation request should be rejected.

8. CONCLUSIONS

In this paper the maximum load of a voice queue in a WFQ system was determined under a certain delay constraint. This was done by modeling each node as a M/G/1 queuing system. The formula to calculate the worst case queuing delay in a WFQ system was extended by using a heuristic formula to calculate a quantile of the queuing delay over *N* identical nodes. This results in a much tighter delay bound for two reasons. First, the bound is statistical and not worst case. Secondly, a quantile is calculated from the delay distribution over *N* nodes instead of adding the delay quantile of each node. The formula was compared with results from simulations. The calculated delay quantile is higher than the one obtained from simulation. However, the bound becomes better for larger loads of the voice queue.

The analytical method presented in this paper provides a tool to allocate (i.e. assigning the correct weight to the voice queue) the appropriate resources for voice traffic in order to guarantee a certain delay bound. This can, for example, be used in a DiffServ/IntServ interworking scenario. In this case the edge routers of the DiffServ cloud are RSVP-aware and bit pipes are provisioned between the ingress and egress nodes. When RSVP PATH messages are received the rate independent delay term *D* should be updated with a quantile (e.g. 99%) of the edge-to-edge queuing delay. This delay value can be calculated with the method presented in this paper. The rate-dependent delay term *C* should not be modified. When a certain bit pipe between an ingress and an egress node is saturated it will start rejecting RSVP reservation requests.

Future work will include the extension of the heuristic formula for the heterogeneous scenario, i.e. non-identical nodes. This involves modification of the factor $\alpha(P)$ in eq. (8) and the calculation of μ_N and σ_N. However, once the formula has been extended the method presented here in this paper can still be used.

9. ACKNOWLEDGEMENTS

This work was carried out within the framework of the project LIMSON, sponsored by the Flemish institute for the promotion of scientific and technological research in the industry (IWT).

We would like to thank Stefaan De Cnodder for providing the necessary simulation models.

10. REFERENCES

[1] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang and W. Weiss, “An Architecture for Differentiated Service”, RFC 2475, December 1998.

[2] Y. Bernet et al., “A Framework for Integrated Services Operation over Diffserv Networks”, RFC 2998, November 2000.

[3] T. Bonald, A. Proutière and J.W. Roberts, “Statistical Performance Guarantees for Streaming Flows using Expedited Forwarding”, Proceedings of INFOCOM 2001, Volume 2, pp. 1104-1112, Anchorage (AL), USA, April 2001.

[4] F. Brichet, L. Massoulié, J.W. Roberts, “Stochastic Ordering and the Notion of Negligible CDV”, Proceedings of ITC 15 (Washington), pp. 1433-1444, Washington, June 1997.

[5] M.J.C. Büchli, D. De Vleeschauwer, J. Janssen, A. Van Moffaert, G.H. Petit, “On the Efficiency of Voice over Integrated Services using Guaranteed Service”, Proceedings of the 2nd IP-Telephony Workshop (IPTEL 2001), pp. 6-13, New York City (NY), 2-3 April 2001.

[6] R. G. Garoppo, S. Giordano, S. Niccolini and F. Russo, “A Simulation Analysis of Aggregation Strategies in a WF^{2}Q+ Schedulers Network”, Proceedings of the 2nd IP Telephony workshop (IPTEL 2001), pp. 102-107, New York City (US), 2-3 april 2001.

[7] M.J. Karam and F.A. Tobagi, “Analysis of the Delay and Jitter of Voice Traffic Over the Internet”, Proceedings of INFOCOM 2001, Volume 2, pp. 824-833, Anchorage (AL), USA, April 2001.

[8] L. Kleinrock, “Queueing Systems, Volume I: Theory”, John Wiley & Sons, 1975.

[9] H.H. Lee and C.K. Un, “A Study of On-Off Characteristics of Conversational Speech”, IEEE Transactions on Communications, vol. COM-34, No. 6, pp. 630-637, June 1986.

[10] M. Mandjes, K. van der Wal, R. Kooij, H. Bastiaansen, “End-to-end Delay models for Interactive Services on a Large-Scale IP Network”, Proceedings of the 7th workshop on performance modelling and evaluation of ATM & IP networks (IFIP99), 28-30 June 1999.

[11] A.K. Parekh and R.G. Gallager, “A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Multiple Node Case”, IEEE/ACM Transactions on Networking Volume 2 Nr. 2, pp. 137-150, April 1994.

[12] K. Sriram and W. Whitt, ”Characterizing Superposition Arrival Processes in Packet Multiplexers for Voice and Data”, IEEE journal on selected areas in communications, vol. SAC-4, No. 6, pp. 833-846, September 1986.

[13] D. De Vleeschauwer, G.H. Petit, S. Wittevrongel, B. Steyaert, H.Bruneel, “An Accurate Closed-Form Formula to Calculate the Dejittering Delay in Packetised Voice Transport”, Proceedings of the IFIP-TC6 /European Commission International Conference NETWORKING 2000, pp. 374-385, Paris (France), 14-19 May 2000.