Workshop on Computer Networks

Differentiated Service Queuing Strategies for Internet Telephony[*]

Luigi Alcuri
Department of Electric Engineering – University of Palermo
Viale delle Scienze 9, 90128 Palermo
luigi.alcuri@tti.unipa.it
Tel. +39-0916465250, Fax. +39-091488452

Francesco Saitta
Department of Electric Engineering – University of Palermo
Viale delle Scienze 9, 90128 Palermo
francesco.saitta@tti.unipa.it
Tel. +39-0916465269, Fax. +39-091488452

Contact author: Francesco Saitta

Abstract: **This paper evaluates two different strategies for configuring Edge-Router queues in Telephony over IP scenario. In such scenario the introduction of DiffServ aimed at differentiate traffic streams in order to achieve different levels of Quality of Service. The relationship between DiffServ-class IP streams and edge-router queues seems to be a critical point for the end-to-end performance of the IP transport network. So, we make a simulation-based study in order to discover the advantages and disadvantages of different edge-router configurations in a multi-provider and multi-service IP backbone transport network. We have considered two different queue configuration strategies, called "queue-to-service" and "queue-to-provider". They are characterized by a different way of assigning DiffServ codepoint to edge-router queues. The "queue-to-service" considers aggregate packet streams on the basis of the service class, while "queue-to-provider" make the same on the basis of TOIP Service Provider. The simulation results show the convenience of using "queue-to-service" in order to get better performance both in Quality of Service and network efficiency.**

Key Words: Differentiated Service, IP Telephony, Edge Router, Queue management

[*] This research has been supported by Italtel, Carini (PA), in the frame of a collaboration project with University of Palermo.

I. Introduction

This work aims to investigate the performance of IP transport networks, when they are used to transport data streams of applications characterized by different real-time and quality of service (QoS) needs in access network configurations in which many Service Providers (SP) share the same IP backbone to transport their services. In more details, we have focused our study on Telephony over IP (TOIP [3, 4 and 5]) services, since they are considered the first step for developing a new multi-media IP transport network with QoS support. Moreover the migration of telephony from a circuit switching network to a packet switching network will bring economic benefits to telephony SPs and new services to the users. From this point of view, the importance and relevance of implementing QoS support in packet switching networks directed our work in order to investigate the problems and possible solutions for running real-time applications, such as phone calls, over IP network.

In the future, we should have a TOIP scenario made of various SPs, which will collect their user service requests by means of many Points of Presence (POP) disseminated throughout the territory. Each of these POP is linked to the others through an IP backbone network in order to serve QoS guaranteed real-time applications.

The introduction of DiffServ on this backbone causes that every packet introduced into the network must be marked with an appropriate codepoint (CP), which will be used by the router for addressing the packet towards the destination POP. The access to the transport network is controlled by an edge-router, which has the functions of classifying, aggregating, marking and routing the packets received by all POPs, which are allowed to use the network. The edge-router work is made difficult by the presence of packets which have different priorities and must be served sooner than other.

If different SPs have POPs linked to the same edge-router, then the configuration of the edge-router results to be a very important and critical point to behave fair towards each SP. In this way the quality of services received by the users will not be dependent from the SP chosen for reaching the transport network.

The remainder of this paper is organized as follows. In the next section we describe the network scenario used during the simulation to test edge-router configurations. In Section III we describe the service classes that were used during the simulation and how they were marked with DiffServ class codepoints [10]. Section IV is dedicated to edge-router configuration and it describes the different strategies implemented during software simulations. In section V we illustrate the results and performances of simulations that are classified on the basis of different edge-router configuration strategies. Finally, section VI summarizes the results and conclusions achieved, and in addition describes the potential application and future evolution of this study.

II. Simulation Scenario

We have chosen to implement a simulation scenario as close as possible to a probable implementation of future TOIP networks. For this reason we have simulated the access to the backbone trough POPs of three different SPs. Each SP has a POP and can configure it for giving to the SP's users as many services as it wants; besides, every service are managed in an independent manner and users can choose which service utilize for every call that they do. In order to simplify the study, however, we have limited or grouped the services in three different main classes. Each SP can choose how many kind of service wants to offer to its users from these three main classes. POPs have the purpose of receiving user calls, associating them to a specific service and checking the state of the network for evaluating the possibility of accepting or refusing the call. If there are not the conditions for sustaining the Service Level Agreements (SLA) associated to the call then the call is refused else the sender start transmitting packets to the edge router for the routing toward the destination POP.

The access to the backbone is managed by an edge-router, which has the purpose of guaranteeing at each provider a fair service time in order to give at each SP's service an adequate quality in a compatible manner with the SLA associated to that service. Moreover the edge-router should be able to maintain high the occupancy level of the network resources optimizing the efficiency. This means that network resources reserved to a temporary not active provider, should be available for further requests, until the "owner" provider does not need to use these resources.

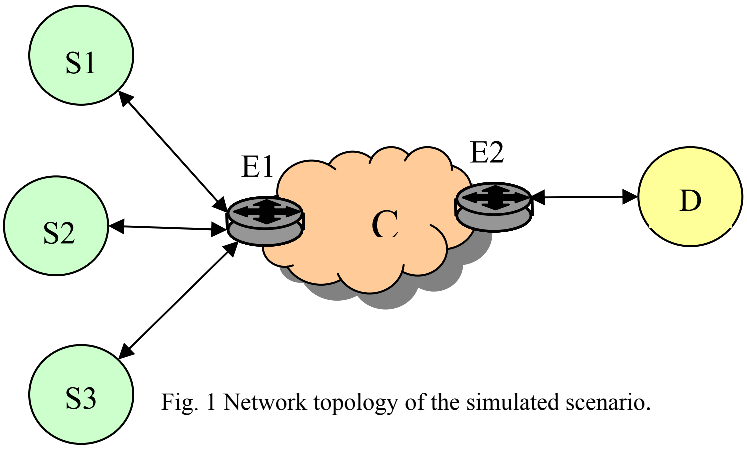

Fig. 1 Network topology of the simulated scenario.

The trade-off between network efficiency and quality of service obtained by the end-user was the base of our study, which aimed at the research of an efficient assignation method for service time between SPs and their applications. From this point of view we looked for a work-conserving solution, which was able to allocate resources with priorities skipping the services that do not have packets to send.

We have implemented the network topology illustrated in Fig. 1. In this network topology there are 3 sources of traffic indicated by nodes S_1, S_2, S_3, one node destination D, 2 Edge routers (E_1 and E_2), and finally a backbone network constituted by a single core router C. Every link between two nodes is a duplex-link with fixed propagation time and bandwidth.

The link between E_1 and C is set as a bottleneck for all the traffic originated by the sources with just 2.4 Mbit/sec of maximum capacity. As described in [9], this bottleneck can be used to model a whole transport network aggregating every causes of congestion on a single point of the network. During the simulation every data and voice connections start from nodes S toward the node D, while on the other direction there are control packets generated by the receiver containing the measured statistics of direct connections. Obviously, with this model we are evaluating the performance of the network in a one way only case; when considering phone calls the same structure has to be implemented on the other way and quality-check are performed on both ways before accepting a new call. Because in both directions there are implemented the same control mechanism, the results obtained with one-way simplification are sufficient to estimate the mean overall performance of the transport network.

Besides, in this scenario we assume that on the backbone there is implemented a differentiated services support so that the work of packet classification and marking done by SP on the network-edges can be used in order to get different transport performance and to control the QoS of user calls. For this reason we have assumed that the backbone network was implemented by using one or more DiffServ Domains. The destination POP D in fig. 1 is the target of every call during the simulation. This POP is independent from the SP and it has also the function of feedback for the quality level achieved by each call. For this purpose it collects some statistics related to the packet delivery of each call and transmit them to the source POP by means of control packets at regular time interval; when the Source POP (S_1, S_2, S_3) receives this statistics, it utilize them for upgrading the thresholds on which is based a Call Admission Control (CAC) method. Each POP in order to give QoS control on its service uses the call admission control method described in [1], which is based on an estimate of bandwidth availability by evaluating the QoS received by the already accepted calls.

III. Service Description

A. Service Classes

Even though each SP could choose the services in an independent manner according to needs of their users, we have chosen of analyzing the performance of the backbone network in presence of differentiated packet streams grouping all the services in three main classes.

The first class, called “Premium”, could be used to implement real-time services, in which it is important a low latency and a small jitter in packet delivery times. For getting this scope, the Premium class is allowed to have a packet loss rate higher than other classes; however this rate should be low enough to allow an acceptable development of phone conversations through services of this class.

The second class, called “Basic”, could be also used for real-time applications but on the contrary of the first it is configured for minimizing the packet loss rate with a few penalizations of latency and jitter. This solution is mainly focused for the transport of compressed streams with real-time needs, such as the video-conference. Even though latency and jitter of the Basic class are more penalized than those of the Premium class, they must be sufficient to allow an understanding conversation between two users.

The convenience of Premium class, from a telephony operator point of view, is that it can be used for the transport of voice flows which are encoded in a very simple way with redundant information, such as G.711. This requires low computational resources and above all it avoids introducing any encoding delays, which could contribute to make the response times of the phone conversation unacceptable. In conclusion, Premium class should be optimized to reduce delays on the transport network, while Basic class should be optimized for reducing packet losses.

Finally, the third class, called "Data", should be used by all the other application without real-time needs, i.e. to transfer e-mail, file, or fax between two or more computers. In fact this class is characterized by having just a minimum guaranteed throughput without any requirement on the delivery time or packet loss rate. This throughput should be sufficient to avoid the starvation of the applications, which use this class for the transport of the packets, protecting them from the resource consumption of the more privileged real-time applications which use not connection oriented streams.

Besides, the transmission of control packets between different POP for feedback control should be considered as a different class. This class should have a preferential channel with very low latencies and small packet loss rates, which are the essential requisites for having a prompt control reaction whenever the network experiences packet transmission delays or low quality performance due to congestion.

B. DiffServ Classes

The DiffServ structure is constituted of four *"Assured Forwarding (AF)"* classes, an *"Expediting Forwarding (EF)"* class and *"Best Effort"*.

The AF classes are usually used for differentiating IP streams inside diffserv domains, in which is subdivided the transport network; each of these classes is independent from the others and there are not differences or privileges between the AF classes. However each class has three packet drop levels; where the first level, “Gold”, is better than the second level, “Silver”, and similarly the second drop level is better than the third drop level, “Bronze”. This means that during network congestion the router will begin to drop firstly the bronze level packets, secondly the silver drop level packets, and lastly the gold level ones. In this way the SP can implement three different kinds of service characterized by different packet drop probability.

The EF class is especially suitable for carrying real-time packet streams because it is designed for reaching very low latency with a guaranteed bandwidth. The negative side of having such excellent performance is that each router has to reserve (with a consequent reduction of network efficiency) for EF class a predetermined amount of network resources, so the traffic profile of expediting forwarding should be well determined a priori and affected by minor variations.

At the end, the Best effort class encloses all the other packet streams, which belong to services without real-time requirements. For these streams it is very important the delivery to destination of the packets in the best way throughout the transport network without affecting the performance of other privileged streams.

C. Matching between service classes and DiffServ classes

After having described the possible kind of services and how DiffServ can differentiate various packet streams, the next step consist of associating at each service class a diffserv codepoint (CP) so that routers on the backbone network can manage the different IP streams obtaining a quality of service compatible with service level specifications.

One of the possible strategies to associate codepoint to service class could be implemented by associating at each SP an AF class and differentiating its services by means of drop levels. In more detail, we chose to associate the gold drop level with the services of Basic class; in fact this service class, which use compressed streams, require above all a small packet loss rate in addition to a low latency time. In this way, using this diffserv codepoint we protect from dropping packets during network congestion and differentiate the streams from other traffics, such as best effort, for a better performance in delivery time.

On the other hand, the packets of Premium class services could be associated with diffserv Silver drop level. In this way, we have the disadvantage of a higher packet loss rate than that of the previous class, but it is compensated by the advantage of having mean lower queue occupancy than previous class in the routers of backbone networks. This causes as a consequence that, during network congestion periods, there is a reduction of the packet mean trip time and a reduction of packet delivery jitter in confront to the previous class, which are two desired effects in the transport of real time streams. However, it is important for the end-to-end quality of the services to limit the loss rate under a maximum threshold and, above all, to maintain the packet burst losses as small as possible in order to not affect the understanding of the phone conversation between users. For this reason we configured the queue management system so that the number of consecutively dropped packets for a single stream was minimized, and the negative effects of packet drops easily recovered by means of error correction codes. It is important to note that when the network is not overloaded the performance obtained by the two classes are similar both in packet loss rate and propagation delay, because the class-based optimizing performance control solution start to work automatically when the network is not more able to serve all the packets and router queues begin to fill up.

Finally, the diffserv Bronze drop level could be used to mark packets of Premium and Basic classes which have gone out of the traffic profile fixed by the SP for these real-time service classes. The aim of this further classification consists of improving the transport network efficiency admitting more packets than foreseen during low network load periods, without affecting the guaranteed QoS streams if suddenly a network congestion breaks up; in fact these out of profile packets marked with Bronze CP will be dropped before of the others.

For the Data service class we used the Best effort CP because there are not real time applications served by this class. Minimum guaranteed bandwidth is the only requisite for having a correct working mode for this kind of services avoiding any starvation due to the presence of privileged UDP streams, which, when there is an high network load, tend to block the throughput of TCP connection mainly used by Data services.

To complete the description of this strategy, we have to associate a diffserv CP to the control packets exchanged by POPs. These packets are essential for setting up phone calls and monitoring the quality of services with low latency feedback information transmissions. Because of the importance of these packets and the easy estimate of control traffic profile the diffserv EF class should be a good choice for these packet streams. In this way we could guarantee the delivery of control packet with a low delay even though there is network congestion, this is a very useful element for implementing a prompt feedback control system able to give quality of services control.

A second strategy of association between diffserv CP and service classes for TOIP not use the drop levels for differentiating the services. In this case, the services are differentiated by using the AF and EF classes grouping inside the same class the packets of similar services sent by different SPs. In more detail, we could choose to aggregate the services of each SP, which have similar service level specifications, in a single packet aggregate and mark it with an AF CP for the transmission throughout the backbone network.

Before going on in the confront between advantages and disadvantages of both strategies, we need to give more details about the configuration of the queues on the routers and the CAC method adopted in the simulations.

IV. QUEUE MANAGEMENT SETTINGS.

A. Configuration of Edge-router queues

To have differentiated the services is not sufficient for obtaining a satisfactory QoS; it should be needed to set the routing through the transport network and to use an admission control policer in order to advantage the transport of real-time streams in order to match their service level specifications.

Under the hypothesis that core routers and related links are well dimensioned for avoiding bottlenecks in traffic stream transports, we have to focus our attention on the point of access to the backbone, which are the edge routers. This hypothesis can be justified by the following arguments:

a) According to the statistics collected by the major telecommunications operators, the backbone network seems to be under utilized.

b) The introduction of new routing protocols, which are able to recognize marked streams, such as MPLS, can be used to create virtual circuits between source and destination POPs through low load network links with a consequent reduction of router queue delays.

c) Backbone links affected by high traffic loads with long congestion times can be easily upgraded with low cost by means of optical fibres.

d) By limiting and shaping the traffic at the ingress of transport network we can reduce traffic variations and improve the estimate of traffic loads on the network; in this way it is possible to avoid potential network congestion announcing them to the router before it happens through service messages in a similar way of Explicit Congestion Notification.

e) Linking the core routers in several ways can help to reduce the traffic load on singular links in a fast way moving privileged streams on alternative routes which have fewer loads than main routes.

Edge routers have the functions of connecting POPs to the transport network, aggregating different streams on the basis of their service class and destination, and labelling the aggregated stream so that it will be transmitted along the best route.

Because all packets have to pass through an edge router, it is evident that it may be considered as the main cause of performance decaying during the transmission of real-time streams usually used for phone applications. Using DiffServ we can differentiate the packet streams and in a similar way we can differentiate the queues of the router, so that all the services will be independent between them.

In particular for each edge router we can distinguish between physical queues, which are served in an independent way, and virtual queues, which are inside a physical queues and are used for distinguish packet drop levels.

We have chosen to use a Weighed Interleaved Round Robin (WIRR) algorithm to distribute the service time between edge-router physical queues.

The choice of WIRR was based on the fact that it is an algorithm which allows to associate a weight for each queue and then serving them in a round robin way with a service time proportional to the queue weight. Moreover it works in work-conserving mode, if a queue does not have packets to transmit during its service period, then the router pass to serve the next queue with the possibility of returning suddenly to the previous queue if meanwhile some packets arrive. The advantage of this scheduler consists of having a minimum guaranteed bandwidth for each physical queue avoiding the starvation of services which use these queues. Besides avoiding to spent service time for empty queues, we do not affect the efficiency of the network, and by using interleave we guarantee that privileged queues are served with maximum priority interrupting the service of less privileged queues until their service time is not exhausted. So by using WIRR we have the possibility to promote the queue with real-time streams over the best effort queue simply by attributing different and convenient weights to each queue.

The management of virtual queues was realized using Random Early Drop (RED) algorithm in order to reduce the mean queue length and to avoid burst losses during network overloads. Each virtual queue is configured, in an independent manner, on the basis of the service class associated to the queue; besides RED works only in function of the number of packets waiting in the virtual queue.

B. Description of queue strategies

During our study, we have tested the performance obtained by two queue configuration strategy, on the scenario described previously by means of Network Simulator [8] software simulations.

The first strategy, called “Queue-to-provider”, consists of assigning at each provider a single edge-router physical queue for TOIP service classes. Instead the Data service classes of all SPs are merged in another physical queue independent by the previous. In this way we can manage the best-effort class specifications in a very easy way assigning to this queue a weight sufficient for obtaining the required outgoing bandwidth; moreover packets of real-time streams are well separated from best effort packets. So using this strategy and the simulation scenario described above we have configured four physical queues at edge router: three were dedicated to each SP for their Premium and Basic service classes, while the last is in common to three SP and receive all the Data traffic without difference based on the source POP.

Each of the three physical queues reserved to SP include three virtual queues, which were respectively reserved to Premium, Basic and out-profile packets.

In each virtual queue there is a RED mechanism, which works in a decoupled manner with the other virtual queues, for reducing queue mean length and as a consequence the mean queue waiting time of the packets. The RED configuration parameters change on the basis of the service in queue, they are more aggressive to drop packets for Premium services than Basic ones; besides the out-profile virtual queue is configured in a very aggressive way so that this kind of packets will not affect the quality of service for in-profile packets during network congestion period.

In order to make a confront on the fairness of services between the various SPs, we have assumed that each SP will generate the same amount of traffic and will have the same warranties on the outgoing bandwidth from the edge router. In this way we were able to evidence from the simulation results and statistics any kind of differences between SP services.

So after having reserved for the best effort physical queue the 27% of total edge router service time (which is equivalent to have a minimum guaranteed bandwidth of 640Kbit/sec), the remaining time was subdivided equally to the SP physical queues.

The second strategy, called “Queue-to-service”, consists of assigning at each service class a different physical queue and joining the packets of different providers on the basis of their class in a single queue. This strategy should be applied only if the SP services are quite homogeneous and share the same service level specifications for each kind of class.

Again in our simulation we have configured four physical queues: one reserved to Premium class packets, one to Basic class packets, and one to Data packets marked as best effort; the last queue was used by out-profile packets of both Premium and Basic classes.

In this case we do not need to configure any virtual queue because there are no reasons for promoting or distinguish SP using different drop levels; so the RED methods has been applied directly to the physical queue with the same parameter configurations of the previous strategy for each kind of service.

The attribution of the weights to the physical queues is more complex than the previous case, because it is necessary to know the traffic profile for each kind of service and SP in order to allocate the right service times and bandwidth at the edge router. In fact the weights have to be chosen so that, in the presence of a normal network load, all the physical queues have to be served according to the level specifications of the services in each queue without any packet dropping.

The configuration used during the simulation assigned the 27% of service time to best effort queue, the 28% to the Premium queue, the 40% to the Basic queue and the remaining 5% to the last out-profile packets queue.

In order to be able to confront the results of the simulations based on the two different strategies we have used the same traffic profiles and call-arrive probability distribution in both cases. In particular we offered to the network a traffic load of 200% by means of both CBR and Poisson flows; the probability of a new call arrive for each type of service follows an exponential distribution with a mean value of 0.5 sec, and the data traffic was simulated by using three always-on FTP connections. To summarize, passing from a type of simulations to the other, we have switched only the queue weights and the way of assigning packets at edge router queues.

Each simulation has been running until 95% confidence interval of measured parameters was less than 10 % of mean value and at least for 3600 seconds. Moreover, many different configurations of the CAC method were tested in order to evaluate the validity of queue strategies in a wide range of the QoS Weighted Bandwidth index for different trade-off between QoS and Efficiency. In more details this CAC method estimates the network load and bandwidth availability for admitting new calls using some statistics on the mean latency time collected by the destination POP using the following formula:

$$QWB = \frac{Bytes\_received}{Time - \alpha * Latency}$$

The α parameter is used for configuring the bandwidth margin due to QoS. In fact, when the network is experiencing a period of heavy load, the packets have to wait more time in the router queue before being served, so the latency grows and in a similar way the *QWB*. The greater is the *QWB,* the less is the bandwidth available and as a consequence a few new calls will be admitted. In this case α determines how much the mean latency influences the QWB. With Time we indicate the interval between two evaluations of *QWB*.

V. RESULTS

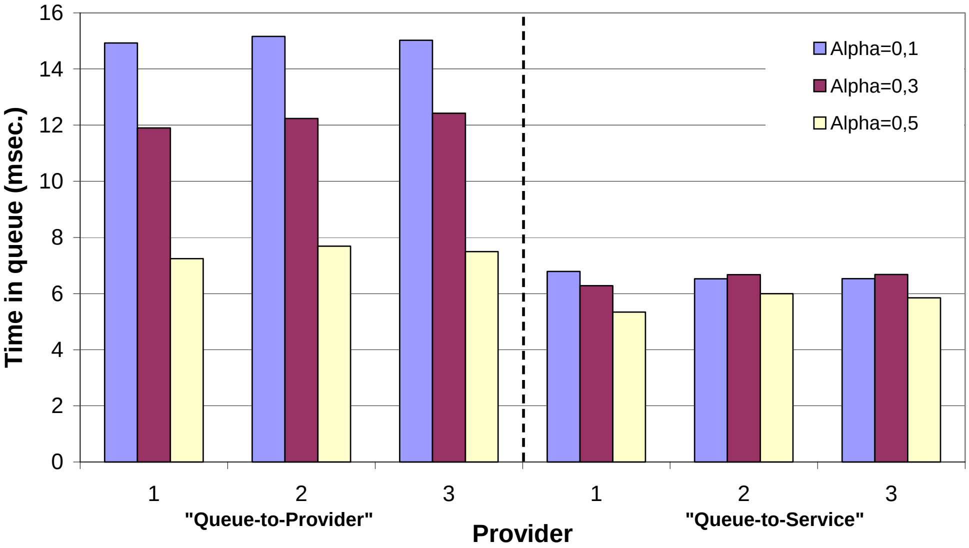

Fig. 2: Comparison between the mean times spent in queue for Premium services with simulation based on different queue configuration strategies. During the simulations was used a QoS Sampling Time T equivalent to 1 second.

In the bar-graphs of figs. 2 and 3, there are reported the mean time values spent in edge router queues by the packets of real time services respectively Premium and Basic. These values have been collected by changing the edge-router queue configuration strategy and alpha parameter, which is used for obtaining different performance in the trade-off between QoS and network efficiency.

The first positive aspect that can be inferred from these results is that each of the three SP obtains in a mean way the same treatment by the edge router during the phase of access to the transport network. In fact the values of mean time spent in queue are practically the same for each provider with the same alpha index and strategy adopted, such as it is evidenced by the first column of each SP in queue-to-provider strategy.

This result could have been considered obvious for queue-to-provider strategy, because we have set the service time between the provider physical queues in a fair way. Anyway it was difficult to predict for queue-to-service strategy, because, in this case, the traffic of each SP is divided on the basis of service type and aggregated in different physical queues, which are served according to their service level specifications. So after these simulation results we can assure that both strategies have a fair behaviour towards SPs, when they produce a similar kind of traffic. This result can be extended even in the real case of different offered-load profiles expecting to get a service behaviour proportional to the service class load for each SP.

A second observation linked to this graph is about the performance obtained from the two strategies regarding the mean latency times. In fact, by using the queue-to-service strategy we experienced a significant reduction of waiting time in edge router queue, which was greater than 50% with a small alpha parameter of 0.1 (it means in a

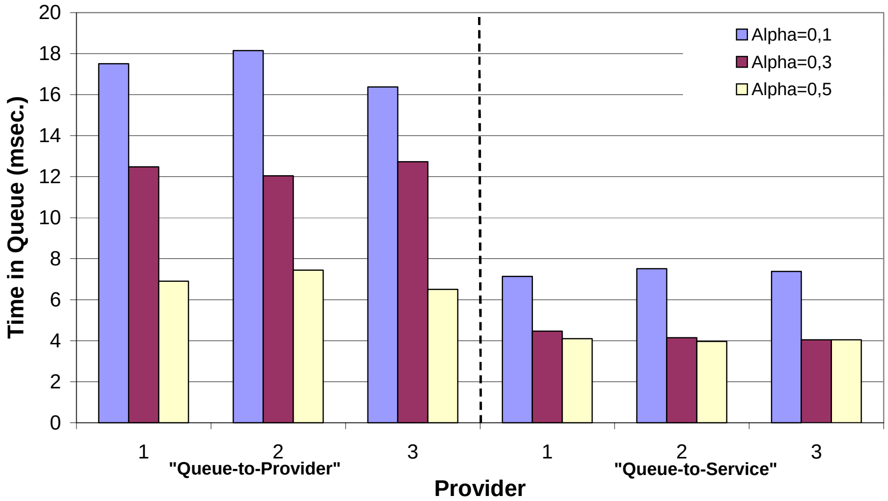

Fig. 3: Comparison between the mean times spent in queue for Basic services with simulation based on different queue configuration strategies. During the simulations was used a QoS Sampling Time T equivalent to 1 second.

more loaded link), in the confront of the queue-to-provider strategy. If we multiply this reduction for the number of routers, which are located on a typical route between two POPs, then we obtain a significant improvement in latency for the real-time services which are served by this queue configuration strategy. However, it is important to evidence that the more we increment the alpha value the less difference we obtain between latency performances of two strategies. This effect is explained whenever we consider that incrementing the alpha value means to increment the margin on QoS bandwidth reserved for real-time services; in this way we reduce the load of privileged packets on the edge-router queues with a consequent reduction of mean waiting time, but the network efficiency is also reduced.

For this reason, we have to evaluate the advantage of reducing the time spent in queue with the diagrams in fig.4 and 5, which show the changes in network efficiency for both strategies by using different values of alpha. Because in all the configurations the overall bottleneck level of utilization is very close to the 100%, we can assume that adopting these strategies along with a good CAC method, such as that based on QoSWeighted index, is the best solution to have a good trade-off between QoS and efficiency. Such result along with the better latency performance obtained, promote the utilization of the "queue-to-service" strategy in future implementation of diffserv edge-router. In fact, this strategy seems to be able to get the best trade-off between QoS and efficiency. However this strategy is also more difficult to configure than queue-to-provider strategy unless to knowledge a priori the profile of traffic offered by each SP. Moreover assigning a diffserv AF class to each kind of service could be a limit in future differentiation of the services, because actually there could be only 4 main groups of services.

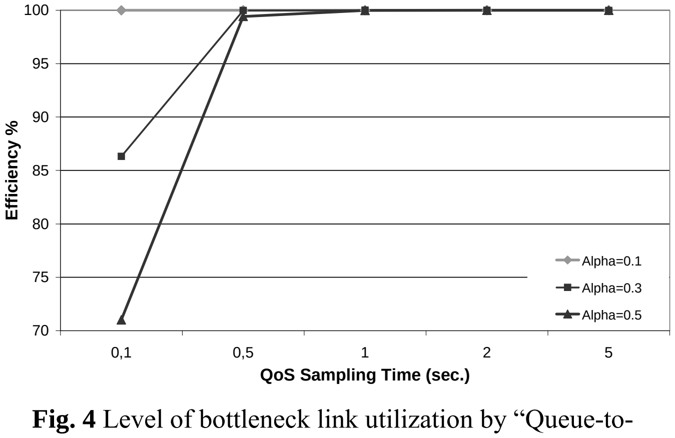

Fig. 4 Level of bottleneck link utilization by "Queue-to-Provider" configuration varying the period of time between QoS statistic control packets.

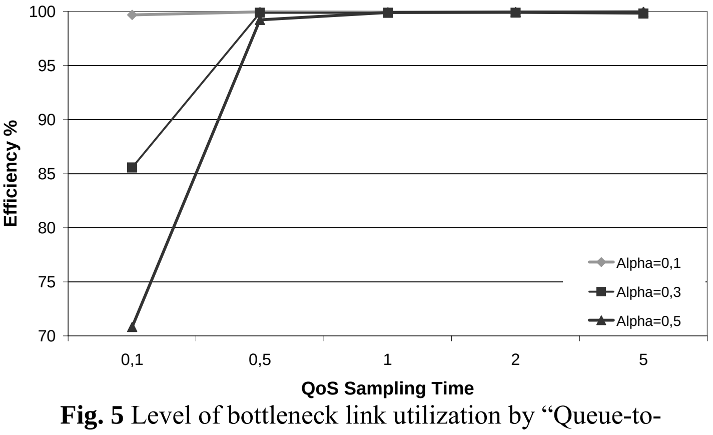

Fig. 5 Level of bottleneck link utilization by "Queue-to-Service" configuration varying the period of time between QoS statistic control packets.

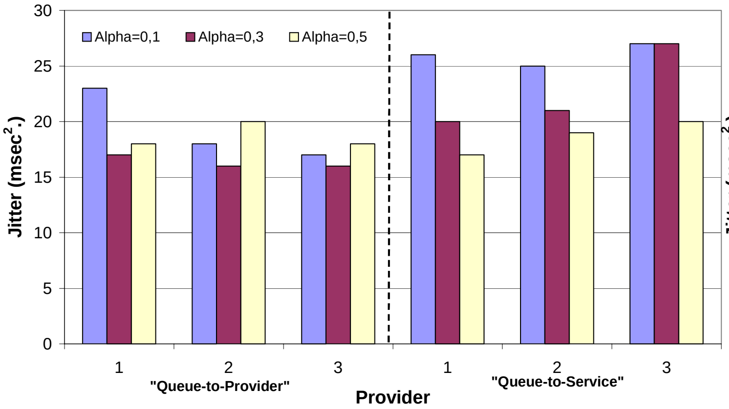

Fig. 6: Comparison between mean jitter values for Premium services obtained by using both queue configuration strategies. During the simulations was used a QoS Sampling Time T equivalent to 1 second.

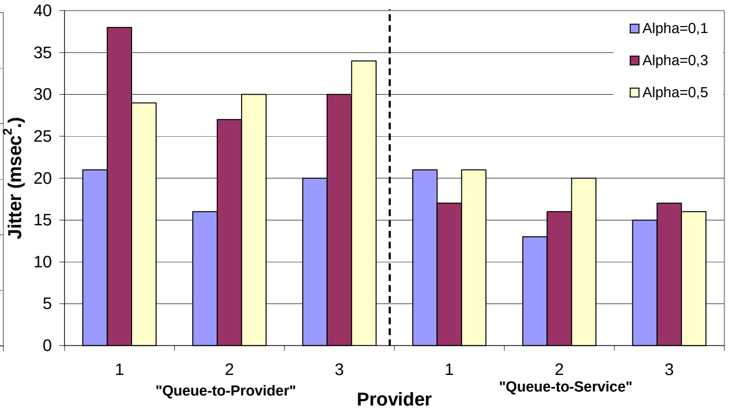

Fig. 7: Comparison between mean jitter values for Basic services obtained by using both queue configuration strategies. During the simulations was used a QoS Sampling Time T equivalent to 1 second.

The packet delivery jitter is also a useful parameter, which can be used to evaluate the QoS of real-time streams. For this reason we have collected statistics about this parameter and realized the diagrams in Fig. 6 and 7. The results show a quite similar behaviour for both configurations with a little advantage of queue-to-service strategy during the service of Basic streams. Anyway, the mean values of jitter are barely limited and they could be easily corrected by means of a buffer at the receiver side of the connection.

Finally, to complete the description of simulation results, we have added the diagrams in Fig. 8 and 9, which

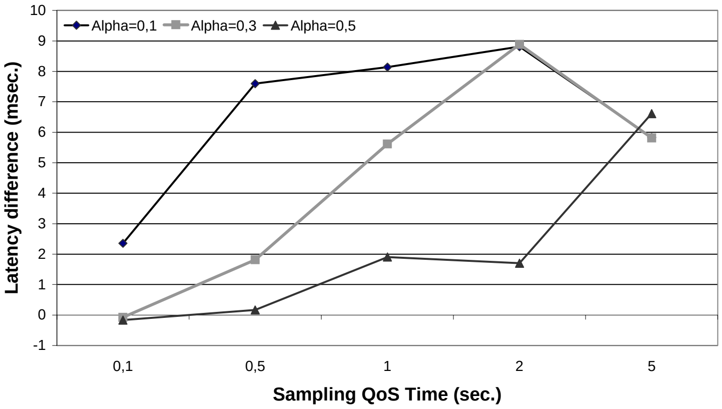

Fig. 8: Difference between mean latency time between queue-to-provider and queue-to-service at changing of QoS Sampling Time with the statistic related to a single SP for Premium class services

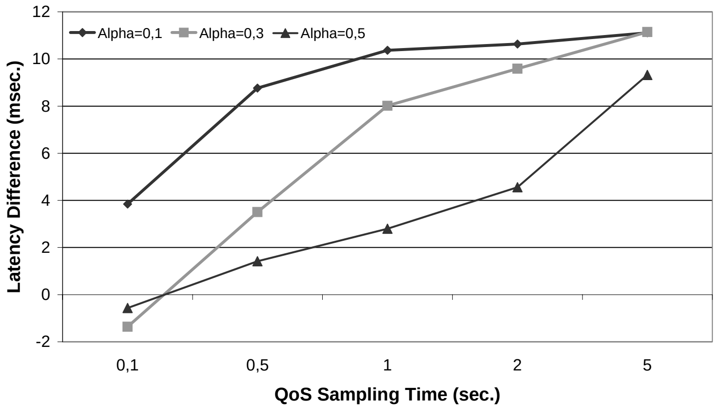

Fig. 9: Difference between mean latency time between queue-to-provider and queue-to-service at changing of QoS Sampling Time with the statistic related to a single SP for Basic class services

show the difference between "queue-to-provider" and "queue-to-service" mean latency time varying the interval between QWB sampling. An increment of QoS sampling time implies an increment of time between two control packets, which transport statistic about the QoS of active calls; this fact causes a minor number of control packets sent along the backbone network but also a delay in the reaction of the CAC to QoS degradation. The diagrams show an increment of the difference between the two strategies at growing of the QoS Sampling Time.

VI. Conclusion

Thanks to a simulation based study, it was possible to make a comparison between two different edge-router queue configuration strategies, which could be used for regulating the access to the backbone network from many providers with differentiated services. The access to the network throughout the edge router could be obtained configuring in different ways the physical and virtual queues in order to get different service level specifications for class of service.

In particular we have tested two different edge router queue configuration: the first called "queue-to-provider" is characterized by the fact that it is used a single physical queue for each service provider; and the second called "queue-to-service" is characterized by assigning a physical queue to each class of service. The "queue-to-service" strategy has obtained results better than the queue-to-provider strategy on real-time services performance with a good trade-off between QoS and network efficiency. However this strategy is more difficult to configure, because it needs a well a priori defined traffic profile for each kind of service with small variations from the mean values in order to get a fair behaviour in serving the different queues. Moreover the solution of assigning an Assured Forwarding class code point to each kind of service could be a bottleneck in differentiating the services with different specifications.

The main contribute of the study was to prove that even a simple change in the association between the class of service and the physical router queues can bring significant improvements in the performance of the differentiated service network. We believe that a more theoretical analysis of the queue-to-service strategy should be conducted in the future. In order to find a simple way for assigning the right weights to the queues in an adaptive way so that the management of edge router will became easier than now.

REFERENCES

[1] L. Alcuri, F. Saitta, "An End-to-End Call Admission Control Scheme for TOIP Applications", http://www.tti.unipa.it/articoli/AS-QWB.pdf

[2] P. Senesi, P. Ferrabone, G. Gritella, R. Rinaldi, M. Siviero, "Telephony over IP: theoretical modelling and lab experiments", [Universal Multiservice Networks, 2000. ECUMN 2000. 1st European Conference on , 2000Page(s): 262 -271]

[3] Computer Networks, Special Issue on Internet Telephony, Vol. 31, No. 3, Feb 1999.

[4] IEEE Network, Special Issue on Internet Telephony, Vol. 13, No. 3, May/Jun. 1999

[5] A. Vugrinec, S. Tomažič, "IP telephony from a user perspective", 10th Mediterranean Electrotechnical Conference, MEleCon 2000, vol. I

[6] D. Houck and G. Meempat, "Centralized Call Admission Control and Load Balancing for Voice Over IP," Pert, and Cont. of Network Sys., SPIE 2000.

[7] H. Knoche and H. de Meer, QoS Parameters: A Comparative Study for Mapping Purposes, Tech. REpt.,Computer Science Department, University of Hamburg, August 1998.

[8] UCB, LBNL, VINT Network Simulator – NS, http://www-mash.cs.berkeley.edu/ns/ns.html

[9] B. Ahlgren, A. Andersson, O. Hagsand, I. Marsh, "Dimensioning links for IP telephony"

[10] D.Black et al., An architecture for Differntiated Services, IETF RFC 2475, Dec. 1998

[11] V. Jacobson, K. Nichols and K. Poduri, An Expedited Forwarding PHB, IETF RFC 2598, Jun. 1999.

[12] A. Tyagi, J. K. Muppala and H. de Meer, VoIP Support on Differentiated Services using Expediting Forwarding, Proc. IPCCC 2000, Phoenix, AZ, USA, Feb. 2000, pp. 574-580

[13] P. Busschbach, D. Houck, and G. Meempat, "QoS for IP Telephony", Networks 2000

[14] A. D. Clark, "Modeling the effects of burst packet loss and recency on subjective voice quality", IPTEL 2001 IP – Telephony workshop, Columbia University

[15] S. Floyd, V. Jacobson, "Random Early Detection gateways for congestion avoidance", IEEE/ACM Transactions on Networking n. 1

[16] S. Blake et a/., "An Architecture for Differentiated Services," RFC 2475, Dec. 1998.

[17] N. Greene, M. Ramalho, B. Rosen, "Media gateway control protocol architecture and requirements" IETF RFC 2805, April 2001

[18] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson,"RTP: a transport protocol for real-time applications", IETF RFC 1889, Jan. 1996

[19] V. Finenberg, "A Practical Architecture for Implementing End-to-End QoS in an IP Network", IEEE Communications Magazine, January 2002