Jungle Issues in Class-based Routing

Marília Curado, Orlando Reis, João Brito, Gonçalo Quadros, Edmundo Monteiro

University of Coimbra
Laboratory of Communications and Telematics
CISUC/DEI, Pólo II, Pinhal de Marrocos, 3030-290 Coimbra – Portugal
Phone: +351239790000 Fax: +351239701266
`http://lct.dei.uc.pt`
`{marilia, oreis, jbrito, quadros, edmundo}@dei.uc.pt`

Abstract. The main contribution of this paper is the evaluation, on a prototype, of a class-based intra-domain routing strategy using traffic patterns and workloads based on actual Internet traffic measurements. The particular case addressed is the evaluation of the impact of the lifetime of the flows on routing and traffic performance. The importance of this issue is due to the fact that class-based routing, naturally used in the Differentiated Services framework, does not include flow handling capabilities. This study is conducted on the Quality of Service routing protocol for the Differentiated Services framework developed at the University of Coimbra. A set of mechanisms is used to control the performance, including the quantification of metrics and distribution of routing information subject to a relative threshold and to a hold-down timer. The results obtained show that class-based routing is able, in the test network used, to reduce the problems that long-lived flows experience without violating the class based paradigm.

Keywords: Class-based routing, Quality of Service.

Introduction

Quality of Service plays a major role in the deployment of communication system for applications with special traffic requirements, such as video-conferencing or Internet telephony. The need to support these types of traffic has motivated the communication research community to develop new approaches. Some of this work resulted in the Differentiated and Integrated Services architectures proposed by the Internet Engineering Task Force (IETF) [1, 2].

Current routing protocols used in the Internet lack characteristics for QoS provision to support emerging new services. All traffic between two endpoints is typically forwarded on the same path, even if there are other alternative paths with more interesting properties for the requirements of a specific flow or traffic class. Usually, the shortest path is selected, based on a single static metric that does not reflect the availability of resources. In these situations, congestion easily occurs on the shortest path, with the corresponding degradation of traffic performance, despite the underutilization of network resources on alternative paths. This scenario has motivated the development of QoS aware routing protocols.

The most significant developments on QoS routing are aimed at communication systems where traffic differentiation is done per flow, as in the Integrated Services [1]. The Differentiated Services framework does not explicitly incorporate QoS routing. It is, thus, essential to develop QoS routing protocols for networks where traffic differentiation is done per class. The Quality of Service Routing strategy of the University of Coimbra (UC-QoSR) was conceived to fulfill this purpose.

The UC-QoSR strategy selects the best path for each traffic class based on information about the congestion state of the network. This strategy extends the Open Shortest Path (OSPF) routing protocol [3] in order to select paths appropriate for all traffic classes as described in [4, 5]. A prototype of UC-QoSR was implemented over the GateD[1] platform, running on the FreeBSD operating system [4].

The behavior of a QoS routing proposal depends greatly on the types of traffic that exist in the network. Traffic patterns consist of sets of flows generated by several applications, mostly using UDP and TCP transport protocols. While traffic patterns vary according to numerous factors, there are some well known characteristics, namely concerning the workload and lifetime of flows. One characteristic that has been widely studied is called the elephants and mice phenomenon, where a small amount of the flows carry the biggest part of the information transmitted in the Internet [6]. Besides being classified by the amount of information carried, flows can be distinguished by their duration. In this terminology, flows that last less than two seconds are called Short Dragonflies, flows that last between two seconds and fifteen minutes are called Dragonflies and flows that last above fifteen minutes are Tortoises [7]. The workload and lifetime classification of flows are two independent ratings that influence routing behavior and traffic performance.

QoS routing can show poor performance due to the instability created by the dynamic behavior of the traffic in the network. Particularly, when short-lived flows enter the network, they origin the re-computation of paths and cause path shifts to long-lived flows, which will see their performance degraded. Saikh *et al.* proposed a combined routing strategy, where short-lived flows are forwarded in static paths and long-lived flows on dynamically computed paths [8]. With this approach the instability caused by short-lived flows is avoided, increasing the overall traffic performance. However, this approach requires the detection of long-lived flows.

In the UC-QoSR strategy, instability caused by traffic dynamics is avoided by using a set of mechanisms that control the emission of routing updates [9]. With this approach, the frequency of diffusion of link-state messages is reduced, and with an adequate tuning, routing inaccuracy is kept at a level that does not cause excessive route flapping.

In this paper the UC-QoSR strategy is evaluated under different workloads based on actual Internet traffic measurements, concerning both the type of applications and the lifetime of flows. Particularly, the behavior of long-lived flows under class-based routing is assessed. The rest of the paper is organized as follows: Section 2 summarizes the UC-QoSR strategy; test conditions and analysis of results concerning routing dynamics and traffic performance are presented in Section 3; the main conclusions and issues to be addressed in future work are presented in Section 4.

---

[1] <http://www.gated.org>

2 UC-QoSR Strategy

In this section the main characteristics of the routing strategy UC-QoSR are briefly described. A more detailed description can be found in previous publications of the authors [4, 5]. The mechanisms used to control the overhead imposed in the network and in routers are discussed with some detail.

2.1 UC-QoSR System Model

The UC-QoSR strategy was designed for hop-by-hop QoS routing in networks where traffic differentiation follows the class-based paradigm. It was conceived for communication systems where traffic characterization is based on class sensitivity to delay and loss. In this work three classes are considered: the best-effort class, a delay sensitive class and a loss sensitive class. The objective of the routing proposal is to select the best possible path for each traffic class, but without giving any guarantees, since this approach would require more signaling messages and the maintenance of additional state by the routers. This strategy is composed of three main components, as follows:

a) A QoS metric that represents the availability of resources in the network;
b) Traffic class requirements in terms of QoS parameters;
c) A path computation algorithm to calculate the most suitable path for each traffic class, according to the dynamic state of the network expressed by a QoS metric.

The availability of resources in the network is measured through a QoS metric that represents the congestion state of the routers interfaces. This metric consists of two congestion indexes, one relative to packet delay (*DcI*) and other to packet loss (*LcI*) that are continuously measured at each router interface. These indexes evaluate the impact that delay and loss at the router, have on application performance [10, 11]. The delay and loss congestion indexes are computed as a linear function of delay and loss, respectively. Supported by the congestion indexes, each traffic class is characterized by a degradation slope concerning loss and other concerning delay. These slopes determine the class sensitivity to the degradation of the corresponding QoS parameter. The scheduling and queue management mechanisms distribute resources among classes, so that all classes have the same delay and loss indexes, despite the congestion state of the interface. Since different classes have different slopes (sensitivities), the same congestion index reflects different delay/loss behaviors for each class.

The congestion indexes are distributed to all routers in the domain through modified OSPF routing messages (Router Link State Advertisements – R-LSA).

The paths for each traffic class are computed by the Dijkstra algorithm, as in original OSPF. Three shortest path tree are computed, one for each traffic class. The path for the delay sensitive class is the shortest path according to *DcI* and the path for the loss sensitive class is the shortest path according to *LcI*. The path for the best-effort class is computed in a way where this type of traffic is digressed from the path used by the delay class. Equation 1 shows the cost function to compute the cost of a link *i* for the best-effort class. The objective of this approach is to protect delay sensitive traffic from best-effort traffic, following an approach similar to the approach presented by Wang and Nahrstedt [12].

$$\cos t_i = \frac{1}{(DcI + 1)} \tag{1}$$

This strategy ensures that the most suitable paths are chosen for each of the two different types of traffic with QoS requirements, however without giving guarantees or bounds on the performance of each traffic class. In the tests presented in this paper, the paths for delay and loss sensitive classes are individually computed. However, the algorithm can be easily modified to select paths in a service model where traffic classes have simultaneously delay and loss sensitivity. In this situation, the delay and loss congestion indexes can be combined in a single cost, weighted according to the corresponding sensitivity of the class to each QoS parameter.

2.2 Mechanisms for scalability

QoS routing protocols must contribute to a significant improvement in traffic performance and network resource usage to compensate for the burden they introduce on the network. This overhead is twofold, comprising an increase in the communication load due to routing traffic and a raise in the processing capacity of routers caused by the frequency of path computations. In UC-QoSR, these overheads are controlled by a policy that controls the emission of link state updates. This policy combines metrics quantification, threshold based diffusion and a hold-down timer. A similar approach was followed by Apostolopoulos *et al.* but in the flow establishment context [13] and evaluated by Lekovic and Mieghem [14].

The metrics quantification is formed by a Moving Average of the congestion indexes resulting from Equation 2, with a variable window size (*N*). The congestion indexes are monitored every second (the lowest time granularity provided by GateD) and the samples are taken continuously. In Equation 2, $MA_d(k)$ is the moving average of *N* values of the delay congestion indexes at sample *k*. This function is used to filter the peaks of the QoS metric.

$$MA_d(k) = \sum_{i=k-N}^{k} \frac{d(i)}{N} \tag{2}$$

The filtered values, resulting from the application of Equation 2, are then presented to the diffusion control module. In this module, the new value is compared with the one that was previously advertised, and will be diffused only if it significantly different. The decision to issue the advertisements is controlled by the value of a defined threshold.

Combined with the mechanisms described above, a Hold-down Timer is used to impose a waiting period between the emission of routing updates. This timer is activated in situations of high instability in the network and avoids routing oscillations that would degrade the overall performance of the system. The definition of these parameters is very important to establish the tradeoff between the overhead of distributing routing updates and keeping the state of the network accurate enough, as addressed by Masip-Bruin *et al.* [15].

3 Experimentation

In this section the experimentation made to evaluate the UC-QoSR behavior under different traffic patterns is presented and its results are analyzed.

3.1 Test Conditions

The test-bed used for the experiments presented in this section is depicted in Figure 1. The *endpoints* 1 to 3 are traffic sources and *endpoints* 4 to 6 are traffic destinations. Each endpoint only generates or receives traffic of a single class to avoid the influence of endpoint processing on traffic patterns. Traffic was generated and measured with the traffic analysis tool Chariot from NetIQ[2].

The routers are INTEL machines with FreeBSD operating system. The kernel is modified, at the IP level, to include the delay and loss metric modules and to schedule and drop packets according to class sensitivity to these parameters [11]. The monitoring of the delay and loss congestion indexes is needed for the routing decision. The kernel is also modified to interact with the UC-QoSR protocol embedded in GateD. It keeps the routing table with paths for all traffic classes and makes packet forwarding decisions based on destination IP address and Differentiated Services Code Point (DSCP) [2].

The interfaces between endpoints and routers are configured at 100 Mbps. Interfaces between routers are configured at 10 Mbps to introduce bottlenecks. In the results presented, the moving average window size is 60 samples and the threshold that controls the diffusion of R-LSAs is 30%. This value means that a new advertisement is issued only when the new measured metric value differs 30% from the last advertised value. These values resulted from the tuning that was done by extensive experimentation with combinations of configurations [5].

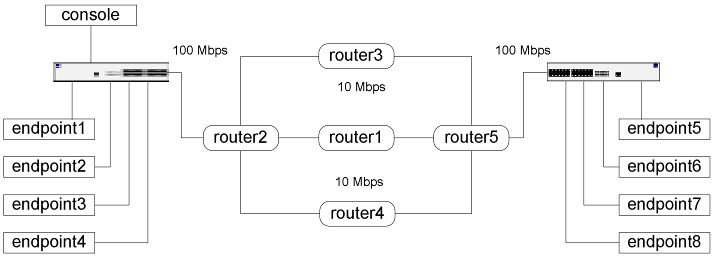

Fig. 1. Test-bed network used in the experiments

---

[2] <http://www.netiq.com>

The applications used were chosen according to the Internet traffic analysis presented by McCreary and Claffy [10]. The most common applications using transport protocols TCP and UDP were used, and the relative contribution of each one to the total load of the respective transport protocol was followed. TCP traffic is responsible for 85% of all traffic, while the rest is mostly UDP. The TCP share is composed of 80% web traffic (HTTP) and the remaining 20% is FTP, NNTP, POP3 and other less representative applications. UDP traffic is mainly audio and video streaming.

In the experiments made the above traffic distribution was closely followed. The only exception concerns the contribution of TCP traffic that is reduced to around 50% of the total load. This was due to the need to have significant loads of the other traffic classes which would require a very large amount of TCP traffic, imposing a burden on the processing of endpoints. This change does not affect the overall behaviour since an increase in TCP traffic would not affect UDP traffic as it is well known.

In order to generate a blend of Short Dragonflies (SDF), Dragonflies (DF) and Tortoises (T), the lifetimes of flows from the different applications were configured according to the results presented by Brownlee and Claffy [9]. Therefore, the traffic workload is composed as follows: 50% of Tortoise flows (with the duration 15 minutes), 30% of Dragonflies flows (with a duration of 5 minutes) , and 20% of Short Dragonflies flows (with a duration of 2 seconds). The number of flows is distributed in the following way: 44% for Short Dragonflies, 50% for Dragonflies and 6% for Tortoises. This distribution reflects the Internet traffic where there is a small number of long lived flows that are responsible for most of the network load.

The mapping between applications and traffic classes took into consideration the characteristics of the generated traffic. FTP and HTTP are mapped into the best-effort traffic class. The video streaming application uses the delay sensitive class. The audio application is mapped into the loss sensitive class due to the capability of this type of traffic to accommodate to small delay variations with buffering taking advantage of the inter-frame gap time.

The experiments were done with three levels of load (low, medium and high) and with the hold-down timer size of 1 and 10 seconds. Table 1 show the resulting traffic mix used in the experiments.

Summarizing the description of the experimentation discussed in this paper:

- Two sets of test were conducted using the traffic mix described in table 1 - one set with a hold-down timer of 1 second and the other set using a hold-down timer of 10 seconds;
- In each group of experiments, three load levels were used: light, medium and high;
- The number and duration of flows was defined according to the characteristics of Internet traffic.

Table 1. Traffic mix for the experiments

Traffic Class	Application	Duration	Num. of Flows	Low load [Mbps]	Med. load [Mbps]	High load [Mbps]
Best-effort	HTTP-SDF	SDF	22	0,163	0,218	0,272
	FTP-DF	DF	8	0,216	0,288	0,36
	FTP-T	T	1	3	4	5
Delay Sensitive	IPTVV-DF	DF	4	0,216	0,288	0,36
	IPTVV-T	T	1	3	4	5
Loss Sensitive	IPTVA-DF	DF	4	0,216	0,288	0,36
	IPTVA-T	T	1	3	4	5
	RAudio-DF	DF	9	0,216	0,288	0,36
TOTAL			50	18	24	30

The evaluation of the UC-QoSR strategy was accessed by indicators of protocol dynamics, path dynamics and traffic performance. Protocol dynamics was evaluated by the measurement of the following indicators:

a) Number of routing messages issued (Router-LSA);
b) Number of times the Shortest Path First (SPF) algorithm is applied;
c) Number of Path Shifts (PathS).

Path dynamics is evaluated measuring the path changes that occurred for each traffic class. The evaluation of traffic performance was made by the average throughput of the traffic classes. The values depicted on the graphics show the relation between the average throughput of traffic classes and the load generated in each class. This performance is evaluated by the ratio of the average throughput over the total generated load, for each traffic class. This approach was employed to allow for the comparison of the performance of traffic belonging to classes that were generated with different levels of load.

Each experiment was carried out for fifteen minutes. The inspection of protocol and path dynamics was done in all routers using the OSPF-Monitor tool included in GateD and modified to collect the new parameters mentioned. The results discussed concern Router 2, where the bottleneck exists. These values were measured by the application Chariot. The plotted results have a degree of confidence of 95%.

3.2 Protocol Dynamics

The indicators used to evaluate protocol dynamics in the experimental test-bed used in these experiments show that the worst behavior is observed when the network is less congested. Figure 2 shows that, in this situation, more routing updates are issued and paths are computed with higher frequency causing more path shifts. The reason for this behavior is that with light load the flows of different traffic classes tend to share the same path (since the total throughput is below the link capacity). However, the link becomes less attractive than the other links, and traffic of all classes will simultaneously shift to a new path, causing instability. The protocol dynamics behavior under low load is considerably better when the traffic of each class is generated over time, instead of being generated simultaneously.

The problem of QoS routing overhead and instability is more critical when the network load is high. These results show that the parameters used are effective in controlling unwanted routing behavior that would degrade traffic performance in a congested network.

The protocol dynamics behavior can be controlled by increasing the size of the Hold-down Timer (HDT). When this timer is used there is a substantial decrease in the number of updates issued, causing a reduction in the number of path computations and path shifts.

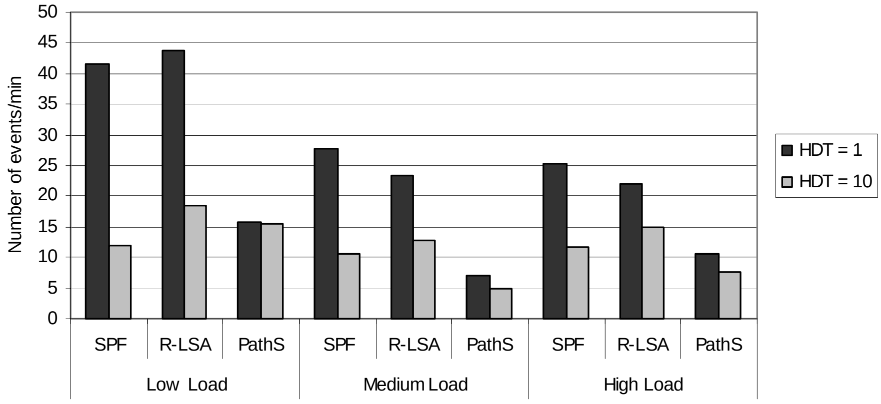

Fig. 2. Protocol dynamics for different levels of load and values of the hold-down timer

3.3 Path Dynamics

Path dynamics is evaluated by the number of path shifts that occur during each experiment. Figure 3 shows the path dynamics of the three traffic classes considered in this evaluation in a lightly loaded network. The available paths of the test network (see Figure 1) are identified as Path 1, 2 and 3. The results are consistent with the previous sub-section. The increase in the number of path shifts causes the computation of new paths and traffic will shift frequently. The best-effort class remains in the same path, while the other classes usually share a link. Even though there are many route flaps of the delay and loss sensitive classes application traffic performance is not excessively damaged, as seen in Figures 5 and 6.

The path dynamics of the UC-QoSR strategy for high load is depicted in Figure 4. The delay sensitive and best-effort classes have stable paths during all the experiment. The loss sensitive class suffers some instability, shifting between paths 2 and 3. This is due to the traffic mix in the loss sensitive class: it has a higher number of short flows and a higher total load. The injection of short flows during the experiment naturally causes instability when the load exceeds 9 Mbps, causing the search for a better path. There is naturally the link capacity limitation that can only be avoided by using admission control mechanisms or routing per flow.

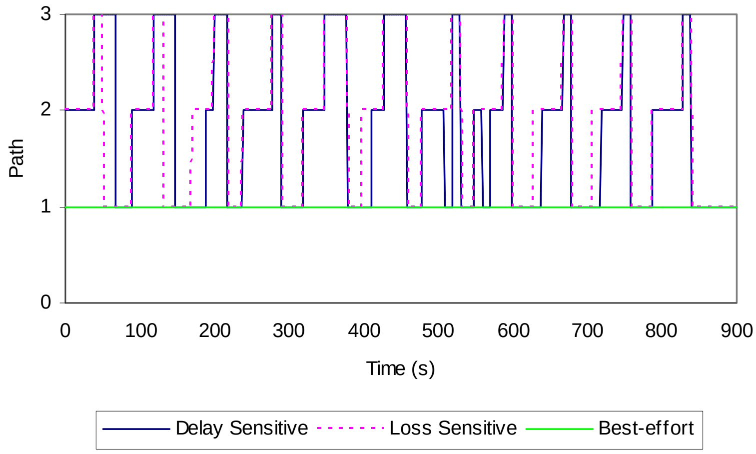

Fig. 3. Path dynamics with low load with a hold-down timer of 10 seconds

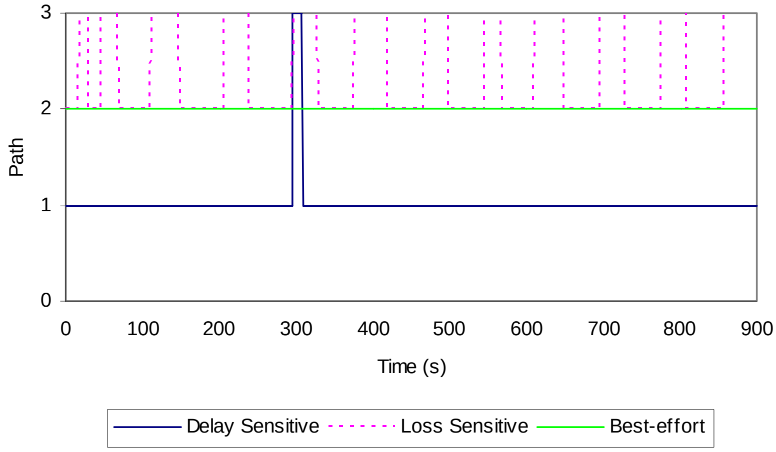

Fig. 4. Path dynamics with high load with a hold-down timer of 10 seconds

The above analyzed results are indicators of protocol dynamics concerning traffic mixes with different lifetimes, but need further evaluation in larger networks in order to be adequately validated. This work is presently being done by simulation.

3.4 Traffic Performance

In this sub-section the performance of long-lived (Tortoises) and short-lived flows (Short Dragonflies and Dragonflies) is addressed. As shown in Table 1, three Tortoise flows, each belonging to a traffic class, were generated and several very short and short flows of all traffic classes were generated throughout the experiment. The results depicted in Figure 5 show that Tortoises and Dragonflies flows have comparable performance and, therefore, that long-lived flows do not see their performance degraded by the introduction of short-lived flows. This behavior is achieved by the combination of mechanisms used in the UC-QoSR strategy and whose results were analyzed in the previous sub-sections.

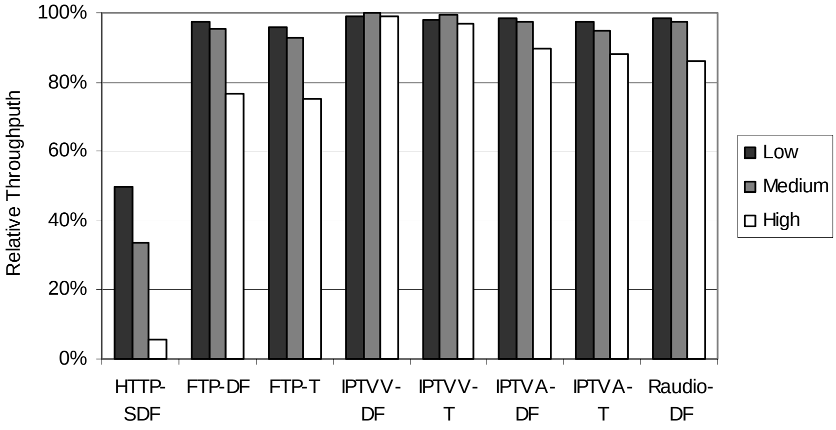

Fig. 5. Relative throughput of application traffic with hold-down timer of 1 second

In Figure 6, Tortoises have generally worst behavior than Dragonflies. In these tests, a Hold-down Timer of 10 was used.

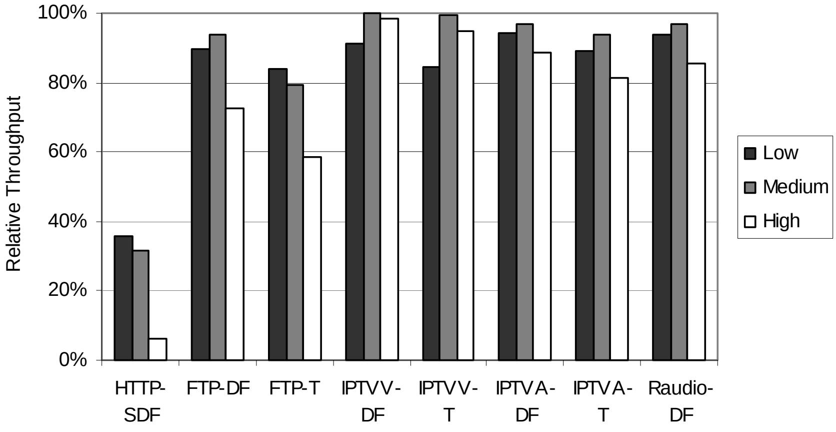

Fig. 6. Relative throughput of application traffic with hold-down timer of 10 seconds

The increase in the size of the hold-down timer damages the performance of long-lived flows because the paths do not adapt fast enough to react to the entrance of the new short flows. This is an observation that contradicts the situations where long-lived flows are damaged by excessive instability caused by short-lived flows. The hold-down timer contributes to reduce QoS routing protocol overhead but is responsible for a reduction of traffic performance. This degradation of performance is patent in all types of traffic, but it is more noticeable in long-lived flows. This behavior can be caused by the test-bed topology used and should be analyzed in larger networks, with variable capacity links.

Short Dragonflies have poor performance because they use TCP as a transport protocol. Due to their short duration, TCP slow start mechanism does not reach the maximal attainable throughput. As the load increases, these flows suffer a higher reduction in the achieved throughput due to the overhead of TCP response time in short flows.

4 Conclusions and Future Work

At the University of Coimbra a QoS routing strategy (UC-QoSR) was developed to support class-based traffic differentiation. The paths suitable for traffic classes are computed based on a QoS metric that evaluates the state of the network in terms of delay and losses. The proposed mechanisms were implemented as an extension to OSPF routing protocol on the GateD platform. Mechanisms for scalability and stability were embedded in the UC-QoSR strategy in order to overcome the common problems associated with QoS routing.

This work addressed the evaluation of the impact of the lifetime of flows on routing and traffic performance of the UC-QoSR. The results show that class-based routing was able to protect and accommodate the QoS needs of Tortoises in a network populated with Dragonflies. Furthermore, the delay sensitive and loss sensitive traffic classes are preserved from the best-effort traffic load, independently of the duration of the flows.

Future work will address the optimization of the proposed mechanisms in order to reduce its instability and some excessive burden on HTTP like traffic Short Dragonflies with high load.

Acknowledgements

This work was partially supported by the Portuguese Ministry of Science and High Education (MCES), under program POSI (Project QoS II and IPQoS) and under the PhD grant PRAXIS XXI/ BD/13723/97.

References

1. R. Braden, D. Clark, S. Shenker, “Integrated Services in the Internet Architecture: an Overview”, Request for Comments 1633, Internet Engineering Task Force, June 1994.
2. S. Blake, D. Black, M. Carlson, E. Davies Nortel, W. Weiss, “An Architecture for Differentiated Services”, Internet Engineering Task Force, Request for Comments 2475, December 1998.

3. J. Moy, “OSPF Version 2”, Internet Engineering Task Force, Request For Comments 2328, April 1998.
4. M. Oliveira, J. Brito, B. Melo, G. Quadros, E. Monteiro, “Quality of Service Routing in the Differentiated Services Framework”, Proceedings of SPIE’s International Symposium on Voice, Video, and Data Communications (Internet III: Quality of Service and Future Directions), Boston, Massachusetts, USA, November 5-8, 2000.
5. M. Oliveira, J. Brito, B. Melo, G. Quadros, E. Monteiro, “Evaluation of a Quality of Service Routing Strategy for the Differentiated Services Framework”, Proceedings of the 2001 International Conference on Internet Computing (IC'2001), Monte Carlo Resort, Las Vegas, Nevada, USA, June 25-28, 2001.
6. Y. Joo , V. Ribeiro , A. Feldmann , A.. Gilbert , W. Willinger, “TCP/IP traffic dynamics and network performance: a lesson in workload modeling, flow control, and trace-driven simulations”, ACM SIGCOMM Computer Communication Review, Volume 31 Issue 2 April 2001.
7. N. Brownlee and K. Claffy, “Understanding Internet Traffic Streams: Dragonflies and Tortoises”, IEEE Communications, vol. 40, num. 10, pp. 110-117, October 2002.
8. A. Shaikh, J. Rexford, K. Shin, “Load-Sensitive Routing of Long-Lived IP Flows”, Proceedings of ACM SIGCOMM’99, Harvard University Science Center, Cambridge, Massachusetts, USA, August 31-September 3, 1999.
9. S. McCreary and K. Claffy, “Trends in wide area IP traffic patterns - A view from Ames Internet Exchange”, ITC Specialist Seminar, Monterey, California, USA, September 2000.
10. E. Monteiro, G. Quadros, F. Boavida, “A Scheme for the Quantification of Congestion in Communication Services and Systems”, in Proceedings of the Third International Workshop on Services in Distributed and Networked Environments (SDNE’96), IEEE Computer Society, Macau, June 3-4, 1996.
11. G. Quadros, A. Alves, E. Monteiro, F. Boavida, “An Approach to Support Traffic Classes in IP Networks”, Proceedings of the first International Workshop on Quality of future Internet Services (QofIS 2000), Berlin, Germany, September 25-26, 2000.
12. J. Wang, K. Nahrstedt, “Hop-by-Hop Routing Algorithms for Premium-class Traffic in DiffServ Networks”, Proceedings of IEEE INFOCOM 2002, New York, NY, June, 2002.
13. G. Apostolopoulos, R. Guerin, S. Kamat, and S. Tripathi. "Quality of Service Based Routing: A Performance Perspective", Proceedings of SIGCOMM'98, Vancouver, British Columbia, USA, September 1998.
14. B. Lekovic, P. Van Mieghem, “Link State Update Policies for Quality of Service Routing”, Proceedings of Eighth IEEE Symposium on Communications and Vehicular Technology in the Benelux (SCVT2001), Delft, The Netherlands, October 18, 2001.
15. X. Masip-Bruin, S. Sánchez-López, J. Solé-Pareta, J. Domingo-Pascual, “A QoS Routing Mechanism for Reducing the Routing Inaccuracy Effects”, Proceedings of the 2nd International Workshop on QoS in Multiservice IP Networks (QoS-IP 2003), Milano, Italy, February 24-26, 2003.
16. M. Curado, O. Reis, J. Brito, G. Quadros, E. Monteiro, “Stability and Scalability Issues in Hop-by-hop Class-Based Routing”, Proceedings of the 2nd International Workshop on QoS in Multiservice IP Networks (QoS-IP 2003), Milano, Italy, February 24-26, 2003.