

Multi-Service: A Service Aware Routing Protocol for the Next Generation Internet

António Varela, Teresa Vazão, Guilherme Arroz

IST, Inesc-ID, IST

Email: {antonio.varela, teresa.vazao, guilherme.arroz}@tagus.ist.utl.pt

Abstract—Quality of Service support plays a major role in the Next Generation Internet. QoS routing protocols must cope with service differentiation to enhance this support. This paper proposes a service aware QoS routing protocol, the Multi-Service routing, which is an extension to traditional intra-domain routing protocols. It proposes a new path selection policy that guides higher priority traffic through the shortest path and diverts lower priority traffic through longer paths when service performance degradation is foreseen. Simulations results shows that the proposed routing performs better than existing QoS routing and link-state protocols.

I. INTRODUCTION

Quality of Service (QoS) plays a major role in the Next Generation Internet (NGI), as new services and applications arise based on multimedia traffic with special requirements, demanding new service models and routing approaches [1].

The Internet Engineering Task Force (IETF) attempts to solve Internet's lack of QoS, by defining new services models. The first model proposed - Integrated Service (IntServ) [2] - provides strict QoS guarantees, but does not scale well to large networks. The Differentiated Service (DiffServ) [3] model solved this issue and is able to assure QoS to aggregated traffic flows classified into a restricted set of service classes. Multi-Protocol Label Switching (MPLS) [4] is another solution, which assures QoS support by means of traffic engineering capabilities offered below the network layer. Concerning the QoS all these technologies are expected to coexist on the NGI. [5] [6] [7]. Nevertheless, Diffserv will play a central role, as it offers a scalable network layer solution, being then independent of any kind of access technology or higher layer protocols.

To date, the Internet routing focuses on connectivity: routing protocols, such as the Open Shortest Path First (OSPF) or the Routing Information Protocol (RIP), are able to cope with the network impairments, but are unable to fulfill the service requirements imposed by the new kind of applications, being inadequate for the NGI. Traffic between two end points is forwarded through the same path, which is usually the shortest one, disregarding the network conditions and the QoS requirements of the associated flows. Thus, congestion arises in these paths and service requirements can no longer be met, despite the existence of alternative underutilised paths.

Several QoS aware routing protocols have been proposed to solve these issues [8]. Should data and telecommunication networks converge around the NGI, the QoS routing problems will become very difficult to solve. First of all, this convergence leads to the existence of traffic with diverse QoS constraints in the same network and, according to [9], this may increase routing's complexity, as finding a feasible path with two independent constraints is an NP complete problem. Second, as the network state changes very often it may be difficult to gather up-to-date state information, specially in large scale environments. The use of outdated information by a routing protocol may degrade the network performance. And finally, a network where resources are shared among priority and Best Effort (BE) traffic is difficult to manage. Although performance guarantees can be assured in priority traffic, by means of resource reservation, the throughput of BE traffic will suffer, if the network capacity is under optimised, by wasting paths that may be used at least by BE traffic. Most of the QoS routing proposals are able to deal with the network state' information, but do not cope with service differentiation.

This paper aims at defining and studying an extension to intra-domain routing protocols, named **Multi-service routing**, which supports service differentiation. The proposal is targeted at IPv6 networks and complaint to MPLS traffic engineering mechanisms, being particularly foreseen to the NGI.

Previous work of the authors comprises specific aspects of the proposal: in [10] the architecture is described and, in [11], the main focuses is the conceptual model and the simulation results. This paper enhances the previous work by providing a global overview of the architecture comprising a functional and a conceptual model, which is defined according to the main QoS routing phases. Additional simulation results are also included.

The paper is organized as follows: section 2 presents several approaches for QoS routing; section 3 describes the routing architecture; section 4 contains the simulation results and, finally, section 5 presents the conclusions and future work.

II. QoS ROUTING IN THE NGI

The QoS routing in the NGI supports the three main tasks depicted in the framework illustrated in figure 1:

- **State Maintenance** which is responsible for gathering state information and keeping it up-to-date.

- **Route Calculation** where the set of *feasible paths* is identified.
- **Path Selection** that defines the path selection strategy that will be used to choose one path among the set of feasible ones.

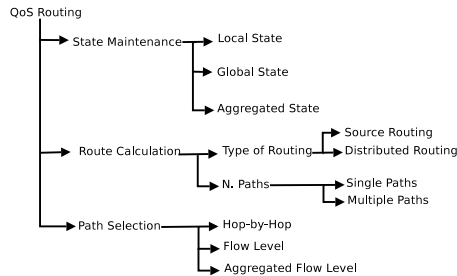


Fig. 1. QoS Routing Framework

There is a wide variety of proposals, characterised by different ways of carrying out these tasks. The most relevant issues to be considered have been also identified in the framework and will be analysed in the following sections.

A. State Maintenance

State Maintenance is supported by local measurements that are performed at each node to evaluate its own state, regarding a single or multiple performance indicator. It can comprise link occupancy, residual bandwidth, delay or the availability of other resources.

A **Local State** strategy is used whenever each node only uses the information it gathers to compute the routes. Nclakuditi et al [12] uses such approach by selecting the path, that will be used to forward a flow, among a set of candidate ones, based on local information. Despite its simplicity, routing decisions are based on an inaccurate view of the network, as remote network conditions are not known.

The use of a **Global State** strategy can solve this problem, as the local state information is disseminated through the network and used to compute the routes. As stated in [13], this kind of approach is particularly adequate for real-time traffic. Although the network state changes very often, routing updates should be bound to reflect the longterm behaviour of the network. Thus, instead of advertising instantaneous performance indicators, quantified metrics must be used. A simple solution was proposed within the ARPANet scope and consists in calculating the average value of the performance indicator [14]; alternatives are also used based on threshold values and hysteresis mechanisms [15] that reduce routing instability and limits the burden of traffic and processing entailed by the routing protocol.

The complexity of this Global State strategy may be compensated by the most accurate view of the network state that can be achieved when compared to the perspective attained by the Local State strategy. However, in large scale networks a less precise view of the network is accomplished, as longer delays are expected to

disseminate and update the routing information. Lack of scalability also arises when the number of metrics to be advertised grows beyond a certain limit. A hybrid strategy based on **State Aggregation** can be used, where nodes are organised hierarchically into clusters; inside a cluster detailed state information is transferred, while among clusters only aggregated information circulates. Private-Network-Network-Interface (PNNI) [16] routing uses such approach, by defining a flexible hierarchical network that can grow up to 104 levels. Scalability gains leads to less optimal paths and complex routing mechanisms.

B. Route Calculation

Route calculation can be performed using two main techniques: source routing and distributed algorithms.

In the **Source Routing** approach each node has a global view of the network and routes are calculated at the source using this information, and piggybacked into every data packet. The entailed overhead precludes its use in large scale networks or under heavy load conditions [17].

The **Distributed Routing** attempts to solve this problem by delegating to each node the task of calculating a part of the path toward the destination. Link-state or distance vectors algorithms can be used. Their use in large networks may introduce a significant overhead, leading to the existence of hierarchical solutions, like the one presented earlier for PNNI or even OSPF.

One of the most important problems in route calculation for QoS routing protocols is related to the fact that routes can no longer be defined based on the number of hops. For instance, if the metric is bandwidth, the best route is the one that maximises bandwidth over the bottleneck link, while if the metric is delay, the best route is the one that minimises it; finally, if both metrics are considered, one needs to maximise bandwidth while reducing delay. In most of the cases the problem can be solved by using modified versions of Dijkstra's [18] or Bellman-Ford algorithms [19].

Another issue that must be considered is the number of paths that are calculated between each pair of source and destination nodes. If a **single path** is used, routing oscillations arise, as long as multi-hop selection is used. This instability problem can be avoided by using load balancing techniques, which can be applied if **multiple paths** are calculated. In [20] it is proposed an algorithm that provides multiple paths of unequal costs to the same destination.

C. Path Selection

Today the Internet uses the datagram service model, where paths are selected in a **hop-by-hop** way, using the network's destination address information contained in the packet; most of the existing routing schemes are based on this principle. According to [21], this kind of approach is not adequate for a network like the NGI, which must be tailored to support services and not just carrying traffic.

Claiming that BE traffic must be routed differently than priority one, new hop-by-hop routing proposals that support service differentiation have recently arisen [22] [23]. Nevertheless, as long as the same routing tasks are performed at both edge and core network elements, a significant burden of information processing is spread across the network. In the NGI, complexity must rely on the edge of the network, in order to allow a faster processing at the core, which means that alternative path selection approaches might be more adequate.

As soon as service differentiation becomes an issue, the notion of flow is fundamental to provide QoS support and it might be used to facilitate the cooperation among routing and resource allocation policies, as a virtual service model can be envisaged [24]. By using **Flow Level** routing traffic may be easily routed according to its class of service. In [25], Nahrstedt and Chen propose a combination of routing and scheduling algorithms where priority traffic is deviated from paths congested by BE traffic. Another proposal was made in [26], where QoS traffic uses less congested paths. However, both of them use source routing paradigm, which is not adequate for NGI, as stated before. IETF has proposed a QoS routing framework [27] that performs the flow level path selection; under this proposal every incoming flow is admitted into the network, only if there are enough available resources; otherwise it is blocked. Despite the accuracy that can be achieved with this type of approach, it is very complex and may not scale well, if individual flows are considered.

Scalability may be achieved if instead of using individual the Flow Level routing, an **Aggregated Flow Level** strategy is used to perform path selection. This strategy is compliant with IPv6 standard that provides a Flow Label field in the IP packet header, and may be supported over MPLS networks. Moreover, more complex routing decisions can be rely on the edge of the network and only when traffic flows initiate their activity.

III. MULTI-SERVICE ROUTING

In this section the main characteristics of the **Multi-Service** routing are described.

A. Main Goals and Policies

The Multi-Service routing architecture was designed having in mind the following goals:

- **Contribution to service fulfilment**, which implies that routing decisions must take into account the network conditions and the established Service Level Agreements (SLAs), meaning that incoming flows should have the agreed performance and must not interfere with the performance of priority flows, already established.
- **Scalable solution**, which means that the mechanism that supports QoS routing should not introduce a significant overhead in the routers, considering both processing time and routing table's size.

- **Optimization of resource usage**, implying that the network must be able to accommodate as much traffic as possible, without penalizing priority traffic.
- **Compatibility** with existing intra-domain routing protocols, meaning that the new routing proposal must introduce a restricted set of changes on the traditional routing protocols, as they are already widely deployed and known by the Network Operators.

B. Architecture Overview

To achieve these goals the Multi-Service routing implements the three tasks that were defined to classify the QoS routing protocols, using the architecture depicted in the figure 2.

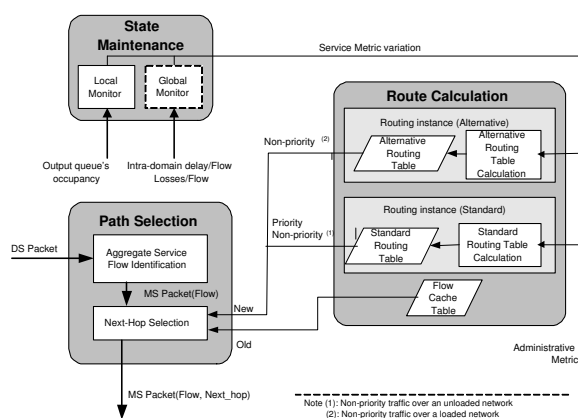


Fig. 2. Multi-Service routing architecture

The Multi-Service routing proposal extends traditional distributed intra-domain routing protocols, by triggering routing table update cycles, whenever service fulfilment may not be accomplished due to the existing network conditions. The evaluation of these conditions is carried out by a set of Local Monitors, situated at each node, and by a distributed Global Monitor. Smooth variant quantified metrics, hysteresis mechanisms and threshold values were used to trigger such updates, using a global network state maintenance strategy. To assure compatibility with existing intra-domain routing protocols, standard mechanisms and messages are used in this updating process.

At each time, each router may have two different routing tables: the **standard table**, describing the set of shortest paths to the destination, and the **alternative table**, describing a set of longer paths to the destination. The selection between these tables must be made according to the following principles:

- The standard table must be used to route all types of traffic when the network is less loaded, in order to minimize the network resource usage.
- The alternative table must be used to route low priority traffic when the network load increases and the level of service of priority traffic may deteriorates if non-priority traffic will keep sharing priority traffic path.

Instead of using an hop-by-hop approach, an aggregated flow level strategy is used, enabling a scalable and efficient solution. Aggregated traffic flows are defined at the edge of the network and stored, in every node, at the Flow Cache Table. Complexity relies on the network's edge, as flow identification and maintenance are performed only at the edge routers. Unless re-routing is needed, routing decisions are taken only once, when a new flow is detected; subsequent packets are routed based on their associated aggregated flow service class.

C. Network State Maintenance

The Multi-Service routing was conceived to meet the service levels, without using very complex monitoring strategies. To achieve an accurate view of the network, global state information is gathered through the monitors depicted in figure 2.

In the our architecture, local information is collected by a **Local Monitor**, situated at each node. To achieve simplicity, a single parameter is used to evaluate the node neighbourhood state: the occupancy of the queues associated with the routers egress interfaces. The information gathered may be used to define the local load trend. Therefore, increasing load conditions can be earlier detected and local congestion situations that arise when the queue occupancy increases very fast can be avoided.

Intra-domain end-to-end performance information is gathered by a service level evaluation component, named **Global Monitor**. It is basically used to monitor if the performance objectives defined for priority traffic are being accomplished. Most of the times, the relevant indicators are delay and losses, as they are very important for the performance of real-time traffic. Both of them can be evaluated at the edge of the network by using information retrieved from the Real Time Control Protocol, which is usually used to carry such type of traffic. If priority traffic has different service requirements, other kind of global indicators must be considered.

By combining both Local and Global information it will be possible to evaluate any kind of SLA in a more accurate way.

Complexity is avoided by using sampling techniques to retrieve the measurements and by restricting the alternative routing table's update cycle to the occurrence of significant traffic variations. The samples are used to compute an **Indicator**, using an exponentially weighed moving average (EWMA) technique. Threshold values are defined and, in order to avoid nasty traffic balance oscillations effects, a hysteresis mechanisms is also considered. Whenever a threshold is reached, a quantified **Service Metric** is modified and the alternative routing table update procedure is triggered.

Three major threshold values were used, as illustrated in figure 3:

- **Deflection Threshold** - it acts like a type of pre-congestion alert; when it is reached, all previous traffic flows keep their paths, while the new incoming

lower priority traffic flows are routed according to the new alternative routing table's paths that will surely not include this pre-congested region.

- **Critical Threshold** - it causes the re-routing of all low priority traffic flows that are currently crossing the critical region. To avoid routing loops, a signalling procedure notifies the the border routers who send traffic to this region, and they trigger the appropriate re-routing actions. Border routers determine new paths to those flows by deleting related old ones.
- **Standard Threshold** - when it is reached, it means that a steady light traffic load condition persists and the corresponding paths will be available, again, to the new low priority traffic flows.

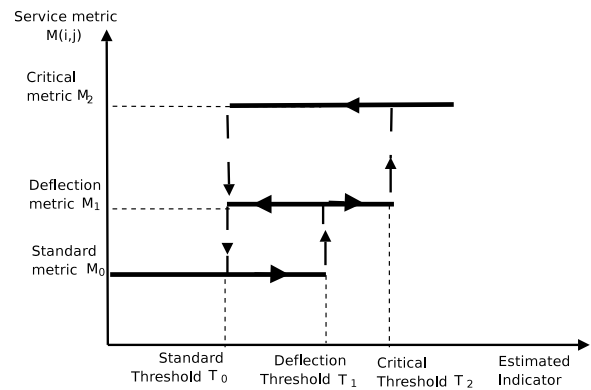


Fig. 3. State maintenance

Considering two adjacent nodes i and j and a link $l_{(i,j)}$ connecting them, a number of samples N , a weight α and the measured output link occupancy $L'_{(i,j)}$; the estimated output link occupancy indicator $L_{(i,j)}$ regarding the connection of node i toward node j , at the sampling time t_i is given by:

$$L_{(i,j)}(t_i) = \alpha * \frac{\sum_{t=t_{(i-1)}-N}^{t=t_i-1} L^{(i,j)}(t)}{N} + (1-\alpha) * L'_{(i,j)}(t_{i-1}) \quad (1)$$

When $M_{(i,j)}(t)$ represents the value of the service metric between node i and j at sampling time t ; T_k represents the k^{th} threshold; H_k the associated hysteresis value and M_k the corresponding metric. At a sampling time $t_i > t$ link $l_{(i,j)}$ changes its service metric, as long one of the two following conditions apply:

$$L_{i,j}(t) < T_k \wedge L_{i,j}(t_i) \geq T_k \Rightarrow M_{(i,j)}(t_i) = M_k \quad (2)$$

$$L_{i,j}(t) \geq T_k \wedge L_{i,j}(t_i) < T_k - H_k \Rightarrow M_{(i,j)}(t_i) = M_0 \quad (3)$$

A similar procedure can be used to calculate and represent the global monitor indicators and their impact on the service metric.

D. Route Calculation

Multi-Service routing is an extension of traditional intra-domain routing protocols, being able to use a link-state or a distance vector approach. Routing information

- **Routing rule 4: Non priority traffic over a local congested network** - in case severe local congestion takes place, existing lower priority aggregate flows may need to be re-routed to the alternative path.

Considering a packet, $pkt_{(i,t)}$, arriving at node i at instant t ; the aggregated service classes $ag_sc(z)$, where z represents a specific class, np any non-priority class and any a class among the existing ones; the DiffServ service classes sc_p , where p represents the priority of the class (*Prio* or *BE*); the network state's conditions, from node's i perspective, $ns_{(i,s)}$, where s represents the network state (low (*L*), medium (*M*), heavy (*H*), global congestion (*C*) or not global congestion (*NC*) load conditions); the standard routing table, std_rt and the alternative routing table, alt_rt ; the selected next hop $hop_{(z,x)}$, where x is the node's selected egress interface (s via the standard path, a via the alternative one and np the path associated to a given non priority class np); and also the flow cache table fl_cache , the routing policy rules can be defined as follows:

- **Routing rule 1: Priority traffic**

$$\begin{aligned} & \text{if } pkt_{(i,t)} \notin ag_sc_{(any)} \wedge pkt_{(i,t)} \in sc_{(Prio)} \Rightarrow \\ & \text{new}(ag_sc_{(z)}, pkt_{(i,t)}) \leftarrow z_1; \\ & hop_{(z_1, x_s)} \leftarrow select(std_rt_{(i,t)}, pkt_{(i,t)}); \\ & \text{new}(fl_cache_{(z, x, t)}) \leftarrow hop_{(z_1, x_s)} \end{aligned}$$

- **Routing rule 2: Non-priority traffic over an unloaded network**

$$\begin{aligned} & \text{if } pkt_{(i,t)} \notin ag_sc_{(any)} \wedge pkt_{(i,t)} \in sc_{(BE)} \wedge ns_{(i,L)} \Rightarrow \\ & \text{new}(ag_sc_{(z)}, pkt_{(i,t)}) \leftarrow z_2; \\ & hop_{(z_2, x_s)} \leftarrow select(std_rt_{(i,t)}, pkt_{(i,t)}); \\ & \text{new}(fl_cache_{(z, x, t)}) \leftarrow hop_{(z_2, x_s)} \end{aligned}$$

- **Routing rule 3: Non-priority traffic over a loaded network**

$$\begin{aligned} & \text{if } pkt_{(i,t)} \notin ag_sc_{(any)} \wedge pkt_{(i,t)} \in sc_{(BE)} \wedge ns_{(i,M)} \Rightarrow \\ & \text{new}(ag_sc_{(z)}, pkt_{(i,t)}) \leftarrow z_3; \\ & hop_{(z_3, x_a)} \leftarrow select(alt_rt_{(i,t)}, pkt_{(i,t)}); \\ & \text{new}(fl_cache_{(z, x, t)}) \leftarrow hop_{(z_3, x_a)} \end{aligned}$$

- **Routing rule 4: Non-priority traffic over a heavy loaded network region**

$$\begin{aligned} & \text{if } pkt_{(i,t)} \in ag_sc_{(z,np)} \wedge ns_{(i,H)} \Rightarrow \\ & \text{modify}(ag_sc_{(z,np)}, pkt_{(i,t)}) \leftarrow z_4; \\ & hop_{(z_4, x_a)} \leftarrow select(alt_rt_{(i,t)}, pkt_{(i,t)}); \\ & \text{modify}(fl_cache_{(z, x, t)}) \leftarrow hop_{(z_4, x_a)}; \end{aligned}$$

Due to the existence of different performance indicators, conflict among routing policies may arise when contradictory routing rules are identified by the local monitors and/or by the global monitor. To solve this problem, a priority is assigned to each routing rule and when different rules are selected for the same aggregate flow, only the most priority one is executed.

Subsequent packets of the same flow are associated with it at the edge of the network. Their forwarding will be based on the flow identifier they carry and on the correspondent routing information, which have been previously stored at a Flow Cache Table, when the first incoming packet of that flow has been routed. The following set of forwarding policy rules are applied:

- **Forwarding Rule 1: Priority traffic** - packets belonging to a priority aggregate flow are always forwarded through the path which has been previ-

ously associated to it when the first packet of this flow entered the network.

- **Forwarding Rule 2: Non priority traffic over an non congested network** - should the network be uncongested and packets belonging to a non priority aggregate flow are forwarded through the path which has been previously associated to it. This path may change during the flow lifetime, if the routing rule 4 is applied to it.
- **Forwarding Rule 3: Non priority traffic over a global congested network** - in case of severe global congestion, packets belonging to lower priority aggregate flows may be dropped at the ingress of the Multi-Service domain, in order to guarantee the level of service of priority traffic.

These forwarding rules determining packets handling can be defined as follows:

- **Forwarding rule 1: Priority traffic**

$$\begin{aligned} & \text{if } (pkt_{(i,t)} \in ag_sc_{(z_1)}) \Rightarrow \\ & hop_{(z, x)} \leftarrow fl_cache_{(z_1, x_s, -)}; \\ & forward(x_s, pkt_{(i,t)}) \end{aligned}$$

- **Forwarding Rule 2: Non priority traffic over an non congested network**

$$\begin{aligned} & \text{if } (pkt_{(i,t)} \in ag_sc_{(z,np)} \wedge ns_{(i,NC)}) \Rightarrow \\ & hop_{(z, x)} \leftarrow fl_cache_{(z,np, x,np, -)}; \\ & forward(x,np, pkt_{(i,t)}) \end{aligned}$$

- **Forwarding Rule 3: Non priority traffic over a global congested network**

$$\begin{aligned} & \text{if } (pkt_{(i,t)} \in ag_sc_{(z,np)} \wedge ns_{(i,C)}) \Rightarrow \\ & drop(pkt_{(i,t)}) \end{aligned}$$

In figure 5 the situations that may arise in a non-congested network are represented. If the network is unloaded both priority and non-priority traffic share the shortest path and routing rules 1 and 2 are applied to select the routes, while forwarding operation uses rule 1. When the network load starts to increase, incoming lower priority traffic is diverted to the alternative path, being routing rule 3 used to select its path and forwarding rule 2 used to expedite subsequent packets.

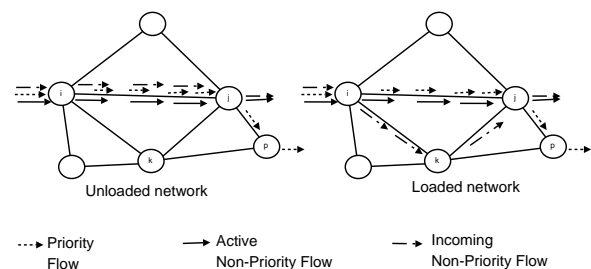


Fig. 5. Path selection strategy: unloaded network

Should the network become congested and one of the two situations depicted in figure 6 arise. In case of local congestion all priority flows must be re-routed to a longer path, and routing rule 4 and forwarding rule 2 are applied. To finalize, when a global congestion occurs non-priority traffic must be dropped at the edge of the network and forwarding rule 3 is used for this kind of traffic.

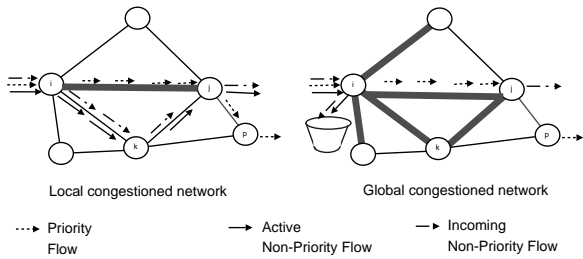


Fig. 6. Path selection strategy: local congested network

IV. SIMULATION STUDIES

A. Simulation Scenario

The proposed routing architecture has been tested through simulations, using the Network Simulator (NS), version 2.27, which has been enhanced with additional capabilities, needed to support this new proposal. Simulations with different network load conditions were performed, using the network scenario described in figure 7 and in table I.

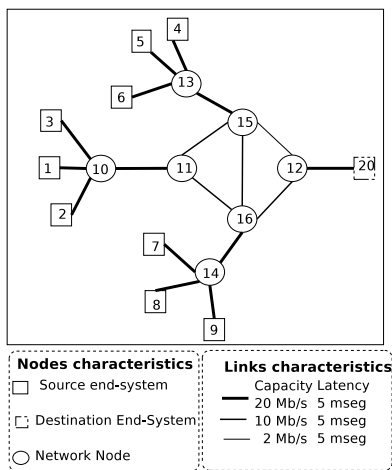


Fig. 7. Network Topology

TABLE I
TRAFFIC CHARACTERISATION

Class	Prio (single)	Prio (aggregate)	Non-prio
Number	1	42	[0..18]
CoS	EF	EF	BE
Traffic	CBR	CBR	CBR
Src	1	4	5
Dst	20	20	20
Rate [Kb/s]	24	24	500
Size[B]	40	40	1500
Total BW [Kb/s]	24	1000	[0..9000]

B. Parameterisation of Threshold Values

A set of simulations were carried out to configure the thresholds of the Multi-Service routing protocol, in order to adjust the performance of the Multi-Service routing protocol.

In the first set of simulations the Multi-Service routing supports only the critical threshold, which means that when it is reached the entire set of non-priority flows are deviated from the shortest path. This kind of situations should happen only when the network is heavily loaded and thus the threshold values tested are high (80% and 90% of the link occupancy). The threshold that offers the best performance is the one that reduces the losses and delay. As stated in figure 8 and 9, although similar results are achieved by both threshold values, fixing the critical threshold at 80% removes the transitory spikes that happened before the path transition occurs and decreases the number of losses in non-priority traffic, which means that a more efficient network utilisation is achieved.

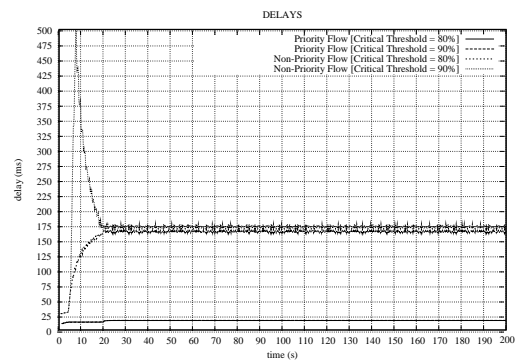


Fig. 8. Critical Threshold parameterisation: evaluation of delay

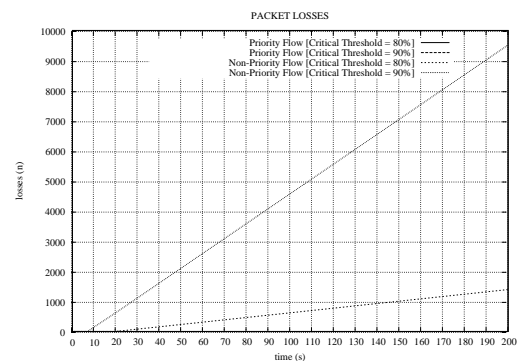


Fig. 9. Critical Threshold parameterisation: evaluation of losses

Should the critical threshold be fixed at 80%, the deflection one may be tuned. Three different values were tested (20%, 50% and 70%) and the results are shown on figure 10 and 11. If the deflection threshold is adjusted to 20% of the link capacity, incoming non-priority flows starts to be diverted too soon and longer delays are achieved for both priority and non-priority traffic. On the other hand, if the 70% value was selected BE losses will be more significant than those achieved when the deflection threshold is defined at 20%, because the modification of the paths happens too late, when the smaller capacity link (15-12) is already heavy loaded. At 50% of link capacity, both priority and non-priority traffic have a good performance, as delay is kept small and no losses occur in BE traffic.

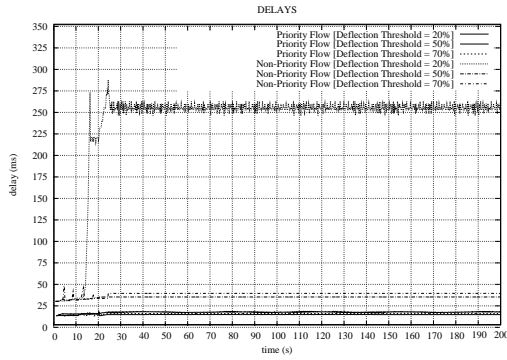


Fig. 10. Deflection Threshold parameterisation: evaluation of delay

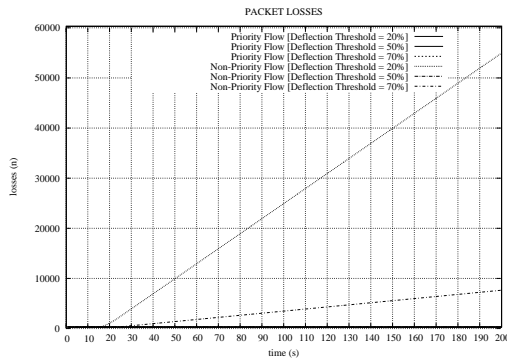


Fig. 11. Deflection Threshold parameterisation: evaluation of losses

TABLE III
NON-PRIORITY TRAFFIC: PERFORMANCE EVALUATION

Type of routing	MS-R	QoS-R	LS-R
Link load	1 Mb/s		
Delay[ms]	-	-	-
Losses[%]	-	-	-
Throughput[Mb/s]	-	-	-
Link load	5 Mb/s		
Delay[ms]	28.71	29.06	10807
Losses[%]	0	0	80.16
Throughput[Mb/s]	3.92	3.42	1.78
Link load	9 Mb/s		
Delay[ms]	31.96	30.25	25895.3
Losses[%]	0	0	76.34
Throughput[Mb/s]	8.60	8.60	1.93

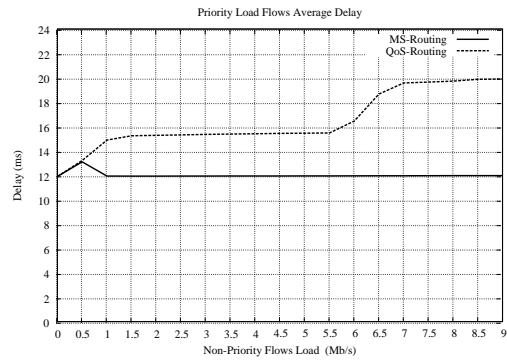


Fig. 12. Delay evaluation of priority traffic under different network load conditions

C. Performance evaluation of Multi-Service routing

The performance of Multi-Service routing (MS-R) was compared to the performance offered by both the traditional link-state (LS-R) and the QoS routing (QoS-R). The results shown in table II and III illustrates the performance of those algorithms under different network load conditions, for both priority and non-priority traffic.

TABLE II
PRIORITY TRAFFIC: PERFORMANCE EVALUATION

Type of routing	MS-R	QoS-R	LS-R
Link load	1 Mb/s		
Delay[ms]	11.91	11.91	11.91
Losses[%]	0	0	0
Throughput[Mb/s]	0.99	0.99	0.99
Link load	5 Mb/s		
Delay[ms]	11.95	15.41	10807
Losses[%]	0	0	80.16
Throughput[Mb/s]	0.96	0.96	0.18
Link load	9 Mb/s		
Delay[ms]	11.98	19.9	25813.4
Losses[%]	0	0	95.60
Throughput[Mb/s]	0.91	0.91	0.04

Figures 12 and 13 represents the delay and throughput achieved by the routing protocols, under different network load conditions.

According to the results depicted in the tables and figures, for both types of traffic, the Multi-Service routing is the one that presents smaller delays; throughput and losses are similar to those achieved by QoS routing, which are much better than the ones achieved by traditional link-

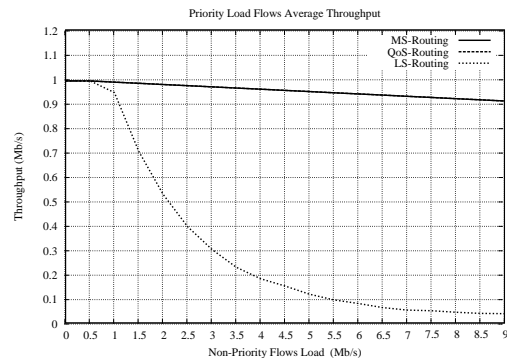


Fig. 13. Throughput evaluation of priority traffic under different network load conditions

state routing.

A more accurate view of the different behaviour of the Multi-Service and the QoS routing protocols is depicted in figure 14 and 15.

As can be stated, the Multi-Service routing also presents a more stable longterm behaviour, as no significant traffic spikes occurs. At time instant 3, the deflection threshold is crossed because the output link of node 15 towards node 12 reaches 50% of its capacity; non-priority traffic presents a slightly better performance than the one it has presented before, as new incoming non-priority flows are diverted through a longer path. As new priority traffic are still being applied to the network after that time instant, the link occupancy (15-12) stays near 80%, but only at time instant 19, it crosses the critical threshold. At

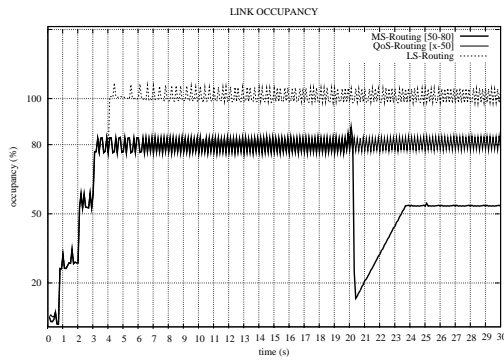


Fig. 14. Longterm behaviour: evaluation of the link occupancy

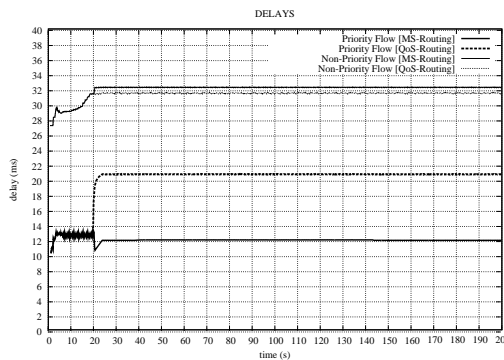


Fig. 15. Longterm behaviour: evaluation of delay

this time, all the non-priority traffic is diverted to a longer path and so the link occupancy and the delay of priority traffic sharply decreases. If QoS routing is used, when the threshold is crossed every incoming new flow (priority or non-priority) is transmitted through a longer path. Thus, link occupancy is kept near 80% and the delay of priority traffic increases approximately 80%.

V. CONCLUSIONS

Existing QoS routing protocols are not able to deal efficiently with service differentiation. The proposed routing protocol provides this kind of support. To perform this, several extensions which provide a solution compatible with traditional routing protocols, with scalability characteristics, have been proposed. Simulation results have shown that priority traffic will achieve better performance and non-priority traffic will suffer less losses. Future work comprises testing the Multi-Service routing in more complex networks; study of other metrics and the integration into an IPv6/MPLS trial platform.

REFERENCES

[1] X. Xipeng and M. N. Lionel, "Internet QoS: A Big Picture," *IEEE Network*, March 1999.
 [2] "Integrated services working group," <http://www.ietf.org/html.charters/intserv-charter.html>.
 [3] "Differentiated services working group," <http://www.ietf.org/html.charters/diffserv-charter.html>.
 [4] "Multi-protocol label switching working group," <http://www.ietf.org/html.charters/mpls-charter.html>.

[5] B. Teitelbaum, S. Hares, L. Dunn, V. Narayan, R. Neison, and F. Reichmeyer, "Internet2 QBone: Building a Testbed for Differentiated Services," *IEEE Network Magazine, Special Issue on Integrated and Differentiated Services for Internet*, September 1999.
 [6] F. L. Faucheur and et al, "Multi-Protocol Label Switching (MPLS) Support of Differentiated Services," IETF, RFC 3270, May 2002.
 [7] Y. Bernet, "The Complementary Roles of RSVP and Differentiated Services in Full-Service QoS Network," *IEEE Network Magazine*, February 2000.
 [8] S. Chen and K. Nahrstedt, "An Overview of Next-Generation High-Speed Networks: Problems and Solutions," *IEEE Network*, November 1998.
 [9] M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-completeness*.
 [10] A. Varela, T. Vazao, and G. Arroz, "Multi-Service Routing: A New QoS Routing Approach Supporting Service Differentiation," in *Proceedings of the AICT'05*, July 2005.
 [11] —, "Multi-Service Routing: A Routing Proposal for the Next Generation Internet," in *Proceedings of the IFIP Networking 2006 Conference*, May 2006.
 [12] S. Nclakuditi, Z. Zhang, and C. H. D. David, "On selection of candidate paths for proportional routing," *IEEE Computer Networks*, 2004.
 [13] M. Abdelhamid, "Quality of Service Dynamic Routing Schemes for Real Time Systems in IP Networks," in *Proceedings of the 5th International Conference on Networking and International Conference on Systems (ICN/ICONS/MCL 2006)*, April 2006.
 [14] A. Khanna and J. Zinky., "The revised ARPANET Routing Metric," in *Proceedings of SIGCOMM'89*, 1989.
 [15] R. Gurin, S. Kamat, A. Orda, T. Prygienda, and D. Williams, "QoS Routing Mechanisms and OSPF extensions," IETF, RFC 2676, August 1999.
 [16] "Private network network interface - PNNI 1.0," ATM Forum, Specification Version 1, March 1996.
 [17] R. Gurin and A. Orda, "QoS Based Routing in Networks with Inaccurate Information: Theory and Algorithms," in *Proceedings of IEEE Infocom'97*.
 [18] E. Dijkstra, *A Note on Two Problem in Connexion with Graphs*, 1959.
 [19] S. Chen and K. Nahrstedt, "On Finding Multi-Constrained Paths," in *Proceedings of IEEE ICC'98*.
 [20] S. Vutukury and Garvia-Luna-Acheves, "A Simple Approximation to Minimum Delay Bridge," in *Proceedings of ACM SIGCOMM'99*, August 1999.
 [21] M. Bruin and et al, "Research challenges in QoS routing," *Computer communications*, 2006.
 [22] M. Oliveira, B. Melo, G. Quadros, and E. Monteiro, "Quality of Service Routing in the Differentiated Services Framework," in *Proceedings of SPIE's International Symposium on Voice, Video and Data Communications (Internet III: Quality of Service and Future Directions*, November 2000.
 [23] J. Wang and K. Nahrstedt, "Hop-by-hop Routing Algorithms for Premium-Class Traffic in DiffServ Networks," in *Proceedings of IEEE INFOCOM 2002*, June 2002.
 [24] L. Cidon, R. Rom, and Y. Shavitt, "," in *Proceedings of of IEEE INFOCOM'97*, April 1997.
 [25] J. Wang and K. Nahrstedt, "Coexistence of QoS and Best Effort Flows - Routing and Scheduling," in *Proceedings of the 10th IEEE International Workshop on Digital Communications: Multimedia Communications*, September 1998.
 [26] Q. Ma and P. Steenkiste, "Support Dynamic Inter-Class Resource Sharing: A Multi-Class QoS Routing Algorithm," in *Proceedings of IEEE INFOCOM'99*.
 [27] E. Crawley, R. Nair, B. Tajagopalan, and H. Sandick, "A Framework for QoS based routing," IETF, RFC 2386, August 1998.

António Varela is currently a PhD candidate in the area of Computers and Electrical Engineering, at Instituto Superior Técnico (IST), Lisbon - Portugal. He was graduated in Informatics and Electrical Engineering, at Rome "La Sapienza" University, Italy. He was lecturer and coordinator of Electrical and Informatics Department

at Instituto Superior de Engenharia e Ciências do Mar (ISECMAR), Mindelo, S.Vicente - Cape Verde Is., 1992-2001.

Teresa Vazão is an Auxiliar Professor of the Department of Informatics (DEI) of Instituto Superior Técnico (IST). Her teaching and research activity is focused in the area of networking. She realized the PhD and MsC in the area of Computers and Electrical Engineering, at IST; she was graduated in Electrical Engineering, at IST. She is a director of the Communications Networks BsC and MsC courses. She is the leader of Communication Network Architecture group of Inesc-ID, at Tagus-park. Her research is mainly focused in Quality of Service and Routing fields. She participated in several international and national projects in the networking area, namely Combine, SFS-II, Peterpan, Moicane and, presently, Sensation. She is the author and co-author of several papers published in international journal, conferences and national conferences; she participates in the submission of drafts to IETF in the area of mobility.

Guilherme Arrozo graduated in 1972 in Electrical Engineering at Instituto Superior Técnico (IST) in Lisbon. He obtained his PhD in Electrical and Computer Engineering in 1986 at the same University. His thesis subject was Local Computer Networks. He is currently Associate Professor and Director of the new campus at Taguspark of Instituto Superior Técnico. His research interests are in the field of networks and also in asynchronous digital circuits.