Extending the QoS provisioning in GRNET

Christos Bouras Vaggelis Kapoulas Dimitris Primpas Leonidas Poulopoulos

Research Academic Computer Technology Institute,
N. Kazantzaki str., University Campus, GR-26500 Rio (Patras), Greece, and
Computer Engineering and Informatics Dept., University of Patras,
GR-26500 Rio (Patras), Greece
Email: {bouras,kapoulas,primpas,leopoul}@cti.gr

Abstract: **This work presents the extension of QoS provisioning in GRNET, so as to include a "second chance" mechanism for rejected premium-traffic requests. Regularly, accepted requests can be satisfied even if there is a single failure in the network and no specific path is allocated within the network core. The extended mechanism allows for rejected requests to be reconsidered for acceptance over a specific path, under the condition that if a failure occurs in this path, then the corresponding traffic will lose the premium handling and will be then considered as best effort. The extended scheme allows for more flexibility and gives a balanced alternative to the "either full guarantees or no QoS at all" situation of the existing scheme.**

1. INTRODUCTION

Quality of service (QoS) is a crucial ingredient of today's multi-service packet networks. QoS-enabled networks can accommodate simultaneously various differing traffic types,such as data, voice and video by handling time-critical traffic appropriately at congestion points. DiffServ [1] is becoming the prevalent QoS architecture in today's IP-based packet networks.

GRNET [2] is the Greek National Research and Education Network (NREN). GRNET is a mixed IP- and Ethernet-based network operating at Gigabit speeds. Together with the high-speed LANs of its subscribers (universities and research institutes) and the European academic and research backbone, Géant [3], GRNET forms a set of hierarchically-federated networks.

The QoS services of GRNET are based on the traffic classes and the queuing mechanisms defined in [4]. We can characterize the dimensioning of the QoS service in GRNET by means of the maximum priority-traffic load share a_l on each link l in the set L of core links, $0 < a_l < 1$. By keeping a_l below a certain limit, we can provide maximum delay and jitter guarantees for the *Premium* family of services.

Providing guarantees in large-scale DiffServ networks can be a complex task mainly due to the heterogeneity of the participating subscriber and provider networks, whose capacity can range from several Gbps to under one Mbps, as in the case of low-speed DSL links. This capacity mismatch can result in congestion at the network core and more likely at the network edge.

One approach to address this problem is to limit the amount of premium traffic that can be carried over the access links and implement an appropriate dimensioning and provisioning scheme for ensuring the guarantees for premium traffic. Hence, subscribers with the same access link capacity can be allowed to use up to the same maximum percentage of the access link's capacity for carrying premium traffic, ensuring fairness in using the premium IP service.

Thus, the priority-traffic limits at the network perimeter define the aggregated priority-traffic at the network core.

In this paper, we will present an extension of the current QoS provisioning scheme. We will focus on the part that concerns priority-traffic requests that are being rejected because they violate the network's perimeter limitations, while, under some conditions they could be granted. We will overcome the current limitations imposed by the existing provisioning scheme and give the requests a so called "second chance".

2. PREVIOUS WORK

There has been enough work proposed in literature concerning the estimation of premium-traffic demand over a network's core in order to dimension network capacity. The traffic demand is usually expressed with a two-dimension traffic matrix that quantifies the demand between all source and destination pairs in the network as stated in [5], [6] and the references included there. The methods described in [6], [7] require measurements of the actual traffic.

In [7], traffic demand originates from an ingress router to a set of egress routers. The model described, is appropriate for transit domains, where a destination can be reached from multiple egress routers. But this approach cannot be applied to Premium IP services that are offered for the first time, as there is no previous usage data available..

Another approach for network dimensioning of premium IP services involves the formulation of an optimization problem involving link costs. However, consideration of delay and loss constraints leads to an NP-complete problem, whose solution requires various heuristic assumptions [8]. Optimization objectives involving link costs can also be considered when premium IP traffic co-exists with best-effort traffic [9], [10].

Unlike the above works the dimensioning approach adopted by GRNET, which is based on the third algorithm described in [11], does not require knowledge of a traffic matrix between source-destination pairs, but only the maximum percentage of the access link capacity that can be used for premium IP traffic.

On the other hand, the works of [8], [9], [10] consider the capacity dimensioning problem in conjunction with routing. Such joint consideration would lead to more optimal dimensioning; however, in current networks that typically over-dimension some of the core network links carrying both best-effort and premium traffic, the added complexity required to modify the routing algorithm is likely to outweigh the increased efficiency.

In [4] and [11], there is a detailed description of the algorithm adopted by GRNET describing the existing dimensioning scheme of GRNET's network. The existing mechanism uses a worst-case algorithm with link failures.

3. THE EXISTING QOS HANDLING IN GRNET

GRNET manages a modern backbone network that connects all the universities, research institutes as well as the school networks and many public (governmental) services. In the scope of GRNET's virtual NOC, we designed and applied a Quality of Service solution. The design covered the QoS services IP Premium service as well as the LBE service presented in [11] and [12].

The whole network has been dimensioned in such a way that each access link has declared a given percentage of its capacity that can be used by IP Premium traffic. This portion is secure in any case and can provide the IP Premium's guarantees even if all access links in GRNET's topology are full and there is a link failure.

The existing dimensioning scheme of GRNET's network is based on an algorithm that performs network dimensioning when only the maximum amount of priority-traffic that can be carried across each access link is known. The algorithm, takes as input the maximum bandwidth percentage that can be reserved for priority traffic at each access-line and calculates the maximum priority-traffic load that each backbone link can afford at worst-case with link failures (WC-LF). The algorithm assumes alternate routes exist; this is the case when the network contains loops, i.e. a set of links whose endpoints form a closed loop.

Since the GRNET network does not contain loops with common links, we apply this WC-LF algorithm for estimating the worst-case premium IP bandwidth requirements in the case of link failures. Also, because loops do not share common links, there are two routes between routers belonging to loops.

In GRNET, requests for traffic-priority connections are made through a web-based tool, Advanced Network Services tool (ANSTool) [13], designed and implemented by GRNET that takes admission control based solely on the access line capacities, their maximum percentage that can be used for priority-traffic and the active requests.

4. THE PROBLEM AND RATIONALE

What seems to be the drawback in the current mechanism is the fact that, once an access link is filled with premium-traffic requests to its limit it is impossible to make additional requests even if the core network could afford them.

As a result, requests that violate the bandwidth percentage reserved at access lines are being rejected. Even if the request can be implemented over a certain path at the core network without exceeding the worst-case bandwidth limits, the request is still rejected.

Our target is to expand the current mechanism used at GRNET so as to avoid the total rejection of subscribers' requests. Once a request is rejected due to violation of the access line premium bandwidth percentage our mechanism will determine whether there can be a certain path at the backbone network that can serve the request. If such a path exists, the request will be accepted and implemented but only over the pre-mentioned path.

Eventually, for those requests that will utilize our mechanism it is preferable to serve them even under no guarantees in case of failure, than totally rejecting them.

5. THE PROPOSED EXTENSION TO THE ADMISSION CONTROL

The proposed extension of the QoS provisioning scheme of GRNET mainly extends the admission control part of the system.

Our initiative to extend the QoS provisioning scheme arose from the observation that with the current (policy imposed) limits for priority traffic in the access links, the maximum priority-traffic load share a_l (where l is a core link), for some (or all) of the core links, cannot be reached.

Therefore, the core network is able to handle more priority traffic in some (or all) of the core links.

The main idea is then that one can allow usage for priority traffic of this portion of the core links that would otherwise remain available only to normal traffic.

We can then summarise the idea as follows:

1. Find out how much capacity in each core link is allowed by policy to be used for priority traffic, but cannot be used because of the limits in access lines.
2. If a request is rejected by the current admission control scheme, examine if there is a path with enough of the above mentioned capacity to satisfy the request, and if possible accept it (over this path only).

We refer to the above extension to the admission control scheme as the “second chance”.

In essence we partition the maximum priority-traffic load share a_l, for all core links l into two parts. The first part is as big as the current QoS provisioning scheme can use and the remaining second part is used by the “second chance” extension.

Obviously, there is no conflict between the two chances for acceptance as they are considering different parts of the capacity allowed to be used for priority traffic.

As mentioned above the proposed solution needs an initialisation phase to determine the capacity in each link to be used by the “second chance” for requests. The algorithm for this phase is simple and it is shown in Figure 1.

```
FOR every link $l$ in $L$
    SET maximum bandwidth for priority traffic
        TO value imposed by policy
END FOR
EXECUTE the dimension algorithm of the current scheme
FOR every link $l$ in $L$
    SET the bandwidth reserved
        TO value calculated by the dimensioning step
END FOR
FOR every link $l$ in $L$
    SET the bandwidth for “second chance”
        TO maximum bandwidth for priority traffic
            MINUS bandwidth reserved by dimensioning
    STORE the result in persistent storage (database)
END FOR
```

Figure 1 - The initialisation algorithm

Essentially, the initialisation algorithm follows the steps:

1. For every link we define the maximum bandwidth (or better for every type of link we define a maximum percentage of that bandwidth) that will be available for premium traffic.

   For each link, this maximum should be enough to cover the bandwidth calculated to be used by premium traffic in the worst case by the dimensioning algorithm used in the existing QoS provisioning scheme (otherwise policy may be violated).
2. We run the dimensioning algorithm that calculates the maximum premium traffic that will pass through each link in the worst case, given the maximum allowance for premium traffic in the access links.
3. We calculate for each link the difference of the amounts defined in step 1 and calculated in step 2. The differences define the “part” of the network that can be used by the extension of the admission control.

The calculated values are stored in the database that models the GRNET.

In case there is a change in the network, steps 2 and 3 should be run again to get the new values.

Then the extension to the admission control part of the QoS provisioning system is done by adding another phase (a “second chance”) for requests that are being rejected by the current system. These requests are then re-examined for acceptance over a specific path (if there is availability all over this path for carrying the added premium traffic) and with automatic fallback to best-effort handling for this traffic if this path becomes (e.g., by link or node failure) broken.

The admission control algorithm for the “second chance” part is shown in Figure 2.

```
FOR every link $l$ in $L$
    READ the bandwidth for “second chance”
        FROM persistent storage
END FOR
CONSTRUCT the graph $G$ representing the overlay network
FOR every link $l$ in $G$
    IF bandwidth($l$) < requested bandwidth THEN
        REMOVE $l$ from $G$
    END IF
END FOR
EXECUTE a Shortest Path Finding algorithm
IF a path $P$ is found THEN
    FOR every link $l$ in $P$
        SET the bandwidth for “second chance”
            TO bandwidth for “second chance”
                MINUS bandwidth requested
        STORE the result in persistent storage (database)
    END FOR
    ACCEPT the request
ELSE
    REJECT the request
END IF
```

Figure 2 - Extension of admission control algorithm

More specifically, this extension of the admission control follows the steps:

1. We read from the database the current available bandwidth for this second chance step (as calculated by the initialisation phase and updated by the acceptance of other requests). From these data we construct the corresponding graph where each graph link has a weight

equal to the available bandwidth for this second change step. The constructed graph is a sub graph of the graph representing the complete network.

2. We run an algorithm to find a path between the nodes defined in the request were all the links in this path have weight greater than the requested bandwidth for premium traffic. This can be done easily if for example we delete all links with weight less that the requested bandwidth and then apply a pathfinding algorithm.

   If, in step 1 of the initialisation phase, we assign a large part of the links to premium traffic then there will probably be several paths between the two nodes of the request. In this case the pathfinding algorithm can be a variation of a shortest path finding algorithm taking into account the weights of the links.

   If we do not assign a large enough part to premium traffic, it is possible that the graph becomes disconnected when we remove the links with weight smaller than the requested bandwidth and the pathfinding algorithm may not find a suitable path.

3. If a path is found in the previous step then the request is accepted. In such a case, the requested bandwidth is subtracted from the links of the path and the database is updated (essentially reserving this bandwidth over this path for this request). The subtracted bandwidth will "return" to these links once the request expires (essentially making it available again to other requests).

The graph constructed in step 1 represents the current state of an overlay network that is handled by the "second chance" part of the admission control algorithm and can be used for priority traffic to be granted by it.

Finally for each accepted request we must implement a mechanism for handling in the network the packets of the traffic relating to the accepted request. This mechanism is as follows:

1. For each accepted request we establish an MPLS-TE tunnel from the source node to the destination node (and possibly another one in the reverse direction, if the request is for bidirectional traffic)

2. The IP packets that relate to the accepted request are "captured" (e.g., through an extended access list), policed for conformance to the requested bandwidth, remarked as normal traffic (!) and routed over the established MPLS-TE tunnel (by being suitably encapsulated).

   If the tunnel is non-existing (e.g. because of some failure during the course of time) then this traffic is routed through the normal routes (i.e. like the rest of the traffic). In this case this traffic received no priority treatment (this is why we marked this traffic as normal upon entry to the network).

3. In the logical interface related to the MPLS-TE tunnel the packets are marked as premium traffic (but only in the outer level header) and are handled with priority thereof (because of the already existing setup in GRNET).

   However, the encapsulated packets remain marked as best-effort.

   In case there is a link or node failure in the path, then the MPLS-TE tunnel goes down. Packets that are already in the tunnel exit the tunnel prematurely and are then handled as normal traffic, i.e. travel the rest of the network through the normal routes and receive no priority treatment (this is why the packet was marked as premium traffic only at the outer level header).

   In case of path failure, the traffic related to the accepted request must receive downgraded priority because the path that it follows may not have enough allowance for premium traffic (as no relevant reservation has be made).

4. At the last router, as the packet exits the tunnel, some special handling must be done in order to forward it with priority in this last hop. Upon exiting the tunnel this packet lost the premium marking.

The specific commands for implementing the above mentioned mechanism, in the network of GRNET, are given in the next section.

6. IMPLEMENTATION ISSUES

Let us suppose that a QoS request for premium traffic is not accepted by the existing setup and it is given a second chance. The admission control extension finds a specific path to satisfy this request.

We will refer to the router of GRNET that the source is connected to, as PE1, the intermediate routers in the core path as P1, P2, P3, etc., and router that the destination is connected to, as PE2.

In order to satisfy the request we need to do the following setup in the routers (which are CISCO routers and the proposed setup is based on information found in [14-20], although similar configuration setup should be possible for any other made of routers):

1. Setup the access list that "captures" the traffic that pertains to the request:

   This can be done by using the user supplied access list (users provide the necessary information to a wizard, and the wizard produces the access list). We define a class-map that captures the traffic define by the access list and we apply a policy-map at the input of the pertinent interface of PE1 where we police the traffic to the requested bandwidth and set the dscp to 0 (best effort).

2. Setup the MPLS-TE tunnel (using the path identified by the pathfinding algorithm):

   This can done by explicitly defining the path (using the ip explicit-path identifier {path id} construct) and setting up the tunnel, not forgetting to set the mode to mpls traffic-eng, set the bandwidth, and the path-option as explicit.

3. Make sure that the request that pertains to the request goes through the defined tunnel:

   This can done by setting up a route-map that matches only the traffic that pertains to the request and sets the next interface to the tunnel interface, and then applying this route-map to the input interface (the interface at PE1 that connects to the source).

   In case of tunnel failure the default behaviour is to fallback to the normal routing.

4. Mark the traffic in the tunnel as premium:

   This can done by setting up a policy-map that sets the mpls exp topmost to 5 (IP Premium) for all traffic and applying it at the tunnel interface.

   Please notice that only the topmost label is marked as premium, and only traffic in the tunnel receives that marking.

5. Make sure that the traffic get premium handling at the last hop:

   This can done by setting up a policy-map that sets the dscp to 46 (IP Premium) for all traffic (class-default) and applying at the tunnel interface at router PE2, where traffic exits the tunnel.

   This ensures that the traffic will get premium handling at the interface that connects PE2 to destination

The above configuration steps must be repeated for the reverse direction.

Please note that the above configuration steps, assumes that the configuration for the existing QoS provisioning scheme is already in place.

8. CONCLUSIONS AND FUTURE WORK

We have presented an extension in the QoS provisioning of GRNET which allows for failed requests to be reconsidered and possibly accepted under the limitation that they will be routed over a specific network core path. In case this path fails, they will loose any premium handling.

Instead of not accepting the request at all, this option becomes a compromise as it provides additional flexibility to the QoS provisioning schemes.

Future work will initially focus on performing experiments so as to achieve a fair balance between the two parts ("chances") of the admission control algorithm and compare the performance under different pathfinding algorithms.

Also, future work may focus on discovering alternate routes (even for parts of the path) and provide the option, if possible, to reroute the request over an alternative in case of a failure.

REFERENCES

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

[2] GRNET home page, (http://www.grnet.gr/?language=en)

[3] Géant home page, (http://www.geant.net/)

[4] A. Varvitsiotis, V. Siris, D. Primpas, G. Fotiadis, A. Liakopoulos and C. Bouras, "Techniques for DiffServ-based QoS in Hierarchically Federated MAN Networks – the GRNET Case", The 14th IEEE Workshop on Local and Metropolitan Area Networks (LANMAN 2005), Chania, Island of Crete, Greece, 18-21 September 2005.

[5] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg, "Fast accurate computation of large-scale IP traffic matrices from link loads," in Proc. of ACM SIGMETRICS'03.

[6] A. Gunnar, M. Johansson, and T. Telkamp, "Traffic Matrix Estimation on a Large IP Backbone - A Comparison on Real Data," in Proc. Of Internet Measurement Conference 2004, October 2004.

[7] A. Feldmann, A. Greenberg, C. Lund, N. Reingold, J. Rexford, and F. True, "Deriving Traffic Demand for Operational IP Networks: Methodology and Experience," in Proc. of ACM SIGCOMM'00.

[8] P. Trimintzios, T. Bauge, G. Pavlou, L. Georgiadis, R. Egan, and P. Flegkas, "Quality of Service Provisioning for Supporting Premium Services in IP Networks," in Proc. of IEEE Globecom 2002.

[9] K. Wu and D. Reeves, "Capacity Planning of DiffServ Networks with Best-Effort and Expedited Forwarding Traffic," Telecommunication Systems, vol. 25, pp. 193–207, 2004.

[10] K. Wu and D. Reeves, "Link Dimensioning and LSP Optimization for MPLS Networks Supporting DiffServ EF and BE traffic classes," 18th International Teletraffic Congress, 2003.

[11] V. Siris and G. Fotiadis, ""Network Dimensioning Based on Percentage of Access Link Capacity used for Premium IP Traffic". In Proc. of IEEE Int'l Conference on Commun. (ICC'06), Istanbul, June 2006

[12] C. Bouras, A. Karaliotas, M. Oikonomakos, M. Paraskevas, D. Primpas and C. Sintoris, "QoS issues in the Research and Academic Networks: The case of Grnet", Industrial Conference on Multi-Provider QoS/SLA Internetworking (MPQSI 2005), Tahiti, French Polynesia, 23 - 28 October 2005

[13] V. Haniotakis, D.Primpas and A.Varvitsiotis, "GRNET Advanced Services Tool", 18 th TERENA TF-NGN meeting, Jul 2005, Paris (tool URL: http://anstool.grnet.gr/

[14] Cisco Systems, Inc., Modular Quality of Service Command-Line Interface Overview

[15] Cisco Systems, Inc., Cisco IOS Quality of Service Solutions Configuration Guide, Release 12.2

[16] Cisco Systems, Inc., Cisco IOS Switching Services Command Reference, Release 12.2 T

[17] Cisco Systems, Inc., Deploying Guaranteed-Bandwidth Services with MPLS, White paper

[18] Cisco Systems, Inc., MPLS Traffic Engineering - DiffServ Aware (DS-TE)

[19] Cisco Systems, Inc., DiffServ Tunneling Modes for MPLS Networks, Document ID: 47815

[20] Cisco Systems, Inc., Quality of Service for Multi-Protocol Label Switching Networks, Q&A