INTERNATIONAL JOURNAL OF COMMUNICATION SYSTEMS
Int. J. Commun. Syst. 2015; **28**:1037–1052
Published online 22 January 2014 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/dac.2742

An OAM function to improve the packet loss in MPLS-TP domains for prioritized QoS-aware services

Francisco-Javier Rodríguez-Pérez[1,*,†], José-Luis González-Sánchez[2], Javier Carmona-Murillo[1] and David Cortés-Polo[2]

[1]*Department of Computing and Telematics System Engineering, University of Extremadura, Cáceres, Spain*
[2]*Research, Technological Innovation and Supercomputing Center of Extremadura (CénitS), Cáceres, Spain*

SUMMARY

The emergence of new kinds of applications and technologies (e.g., data-intensive applications, server virtualization, and big data technology) has led to a higher utilization of network resources. These services imply increased bandwidth consumption and unexpected congestions, especially in backbones. In this article, a novel proposal is studied with the aim of improving the performance of prioritized forwarding equivalence classes in congested Multiprotocol Label Switching Transport Profile (MPLS-TP) domains. The congestion impact on those QoS-aware services that require high reliability and low delay is analyzed. A new policy has been implemented on MPLS-TP, which is a technology that provides QoS by means of flow differentiation in the Internet backbones. The proposal is known as Gossip-based local recovery policy and is offered as an operation, administration, and management function to allow local recovery of lost traffic for MPLS-TP privileged forwarding equivalence classes. In order to fulfill the requirements for implementation on MPLS-TP, a minimum set of extensions to resource reservation protocol traffic engineering has also been proposed to provide self-management capable routes. Finally, we have carried out a performance improvement measurement by means of an analytical model and simulations. Copyright © 2014 John Wiley & Sons, Ltd.

Received 9 July 2013; Revised 1 November 2013; Accepted 15 December 2013

KEY WORDS: MPLS-TP; congestion; RSVP-TE; Gossip; local retransmissions

1. INTRODUCTION

Extensive research has been carried out on active queue management (AQM) and network resources dimensioning. This work has influenced the quality in provisioned services required by the expedited forwarding traffic in production networks established over Multiprotocol Label Switching (MPLS) enabled domains [1]. However, congestion will always occur because of the unpredictability of traffic, resulting in overloaded switches and routers [2–4]. In fact, new kind of data-intensive applications have dramatically increased network utilization, as well as the amount of consumed bandwidth between communicating hosts. Thus, nodes are experiencing congestion, which has a direct impact on the end-to-end delay and on the performance of reliability and latency-sensitive applications. Furthermore, bandwidth demand increase is stimulated by the rapid growth and penetration of new packet-based communications and multimedia services, with severe bandwidth and QoS requirements. Because of this, Internet service providers and customers expect more efficient traffic engineering schemes and more satisfactory QoS techniques [5, 6].

This movement toward packet-based services means transport networks are evolving in order to encompass the provision of packet-aware capabilities [7], thus enabling carriers to leverage their installed, as well as planned, transport infrastructure investments [8], because traffic and congestion

*Correspondence to: Francisco-Javier Rodríguez-Pérez, Department of Computing and Telematics System Engineering, University of Extremadura, Spain.
†E-mail: fjrodri@unex.es

control plays always an important role in QoS provision, in order to address policing, shaping, scheduling, and resourcing allocation. Consequently, it is needed to implement efficient management schemes that are able to have a better bound on end-to-end packet delay for these packet transport services.

In a congestion control context, AQM mechanisms are used for congestion avoidance by proactively dropping packets, in order to provide an early congestion notification to the relevant sources. Random early detection is an algorithm that predicts the congestion before it occurs and sends feedback to the senders by dropping their packets with the appropriate probability. Nowadays, AQM mechanisms aim at improving the dynamic performance, as well as the stability and the robustness of congestion control systems [9, 10]. However, they are able neither to stabilize the queue size nor to maximize the throughput inside the network. They provide feedback to the senders by discarding packets before overload and, in addition, Transmission Control Protocol (TCP) usually responds to these small increases in loss rate with large decreases in its sending rate. Furthermore, if dropped packets belong to latency-sensitive applications, those end-to-end retransmissions would imply additional delay. Finally, under this feedback-based congestion control, we must also keep in mind that the duration of congestion at a bottleneck is directly related to the bandwidth-latency product. Therefore, the larger the end-to-end delay in a network, the longer it takes until the ingress endpoint can determine that the domain has become congested. Moreover, the higher the bandwidth of the network, the larger the amount of data the sender node may put into the network in the time that it takes to detect the congestion [11]. The MPLS Transport Profile (MPLS-TP) domains are an example of networks with a large bandwidth-delay product. In essence, the aim of MPLS-TP is to develop MPLS extensions where necessary in order to meet classical transport network requirements, such as scalability, multi-service, cost efficiency, high level of availability, and extensive operations, administration, and maintenance (OAM) capabilities [12]. Packet loss measurement (PLM) is one of these OAM MPLS-TP functions, being a key challenge for many service providers, as the service level agreements depend on the ability to measure and monitor performance metrics for packet loss or delay [13].

In this context, the Gossip-based Local Recovery Policy (GLRP) management is proposed to minimize packet loss for QoS-aware services [14, 15]. It is a policy that uses Gossip-based mechanisms in order to distribute the information, standing in contrast to centralized schemes, in which only head ends are responsible for disseminating information about the non-successful reception of packets at the receiver. Gossip-based algorithms have the advantage that they are very easy to implement with each node following a simple local rule in each event of interest, and they are highly fault-tolerant, because communication will happen in aggregate despite a fairly high level of packet loss or node failures [16]. The premise underlying our Gossip-based policy is very simple: when a packet of a particular service is lost, intermediate nodes in the route select another node as a communication partner and exchange information about the loss with it. Over time, loss information can travel through the route in an epidemic fashion.

The GLRP proposal can be defined as a new OAM function for MPLS-TP, in order to improve the reliability of QoS-aware forwarding equivalence classes (FEC) with stringent delay and reliability requirements, when PLM is detecting and counting lost packets. Thus, the GLRP for MPLS-TP provides, to a limited number of intermediate nodes, the ability to cooperate with each other in order to recover lost traffic of prioritized MPLS-TP FEC from upstream intermediate neighbors. It is signaled with a limited extension of the resource reservation protocol traffic engineering (RSVP-TE) protocol, when the label switched path (LSP) is being configured [17].

Thus, GLRP also cooperates with RSVP-TE to obtain local retransmissions of lost traffic when an LSP failure occurs, in conjunction with the fast reroute point-to-point technique [18]. For this purpose, only the packets from prioritized QoS-aware services are temporally stored in a buffer called GLRP Buffer (GBuffer) in parallel to the forwarding operation of the MPLS-TP node [19]. However, observe that a particular packet must be buffered for only a short interval of time. This is because the time elapsed since a packet is forwarded, until a hypothetical local retransmission request for that packet is received, is very limited due to the low packet delay in MPLS-TP domains [20]. Therefore, a Gossip node only needs to store a very low number of packets, which implies fast searches, in which the new incoming packets overwrite the oldest. Summarizing, our GLRP proposal is an MPLS-TP OAM function adequate for privileged services, in order to manage faster retransmissions of lost traffic. Thus, the objective in this paper is to analytically study the GLRP feasibility and performance

DOI: 10.1002/dac

when QoS-aware services are prioritized in congested MPLS-TP domains, where the number of packets counted by PLM is high.

The remainder of this article is structured as follows: First, in Section 2, we discuss the usefulness of packet buffering for FEC sensitive to data loss. Then, in Section 3, we define the GLRP as an OAM function for MPLS-TP domains and how is signaling of the local recovery messages and, in Section 4, the RSVP-TE extensions are detailed. Next, an analysis of the performance improvement is shown, followed by several charts in order to compare some interesting parameters. Finally, we draw some conclusions and contributions from our research.

2. BOOTSTRAPPING DISCUSSION

The temporal locality is the property whereby, during congestion, a packet loss indicates that other packets will very likely be lost soon; due to the fact that if a packet from a FEC is lost, other packets of the same FEC could also be discarded soon. Consider this situation: Suppose that a node i is along a route from a source node to a destination node. Suppose further that whenever a node fails to forward a data packet to its next hop, it drops the packet. The upper protocols that bring reliability to the data transmission would start the retransmission of the packets from the source when it detects the loss. If it is a FEC with high requirements of delay and reliability, the end-to-end retransmission of lost packets would negatively affect the delay-related metrics of this service. However, if the node that drops a packet sends a local retransmission request toward a previous neighbor using the reverse route, the time elapsed to recover the packet from a closer node would be substantially reduced. Summarizing, these low-level retransmissions could avoid, in part, the requests to the head-end (initiated by the upper layers protocols), resulting in lower increment in the global bandwidth consumption in the congested domain.

In this context, when the node i receives a local retransmission request (*Gossip Request* or *GReq*) from a downstream node j, the message indicates that a packet recently sent by the source and recently forwarded by i toward the destination across j is lost. This behavior fulfils the property of temporal locality, as in the case of congestion, a recently dropped packet is a recently sent packet by a close upstream neighbor. Therefore, if node i had a memory to buffer data packets, even if the buffer is small, there would be a high probability that the packet could still be found in the buffer. In this case, it could recover the packet, avoiding the forwarding of the local retransmission request backwards. In classical reactive protocols, only the source node can retransmit a lost data packet. However, although additional storage is required in nodes, a local retransmissions technique can significantly reduce the packet loss, because packet buffering enables more nodes to salvage a lost packet or, in essence, packets retransmission is cooperatively distributed.

Observe that GLRP works in parallel with TCP and if a lost packet cannot be found in the buffer of any Gossip node, GLRP stops. In this case, the end-to-end retransmission timeout of TCP is triggered, and the packet is retransmitted end-to-end from the sender. Therefore, GLRP operation is always made before the retransmission timeout of TCP is triggered. GLRP implies that congestion is detected later by endpoints due to the acknowledgement (ACK) reception of the locally recovered packets, but higher bursts of dropped packets have not been found when GLRP is used. However, the performance metrics, such as throughput or packet delay, show the trade-off nature with local retransmissions in different bandwidth, delay, and loss settings, at the cost of spurious timeouts. However, the appearance of these timeouts is rare when using GLRP because of the conservative nature of the modern TCP timeout algorithms [21–23].

3. GLRP FOR COOPERATIVE PACKET LOSS CONTROL

Observe that, when a packet from a privileged FEC is lost, GLRP needs to know the set of previous nodes that have forwarded the lost packet. This set of Gossip nodes that have switched the packets of a prioritized FEC is called the GLRP Plane (*GPlane*). The number of necessary hops to achieve a successfully local recovery is called diameter (d). In Figure 1, GLRP operation is shown when a packet of a prioritized FEC is discarded in an intermediate node X_4 and three feasible diameters can be selected to recover locally the lost packet.

DOI: 10.1002/dac

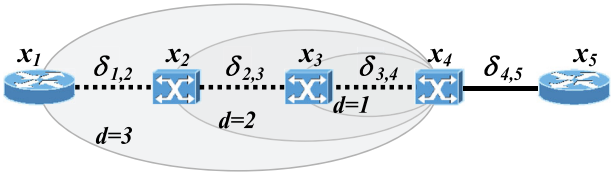

Figure 1. *GPlane* from node X_4, with diameter size up to three hops.

The node that drops the packet knows in which nodes is buffered the lost packet. However, obtaining the GPlane from a Gossip node is not trivial. Let consider a domain $G(U)$, with a set of nodes U and a FEC $\varphi(G) = \varphi(x_i, x_n)$ in $G(U)$ across a path $LSP_{i,n}$, with the origin in node x_i and destination in node x_n, with $\{x_i, x_n\} \subset U$. Maybe x_n only knows incoming port and incoming label of any arrived packet of the FEC $\varphi(G)$, that is, x_n only knows that x_{n-1} is the sender of $\varphi(x_i, x_n)$. It would know which node the sender of a packet is, by means the label information. However, this is not a reliable strategy because, in case of flow aggregates, an RSVP-TE aggregator could perform reservation aggregation to merge k flows, in the form:

$$\varphi(x_{n-1}, x_n) = \sum_{i=1}^{k} \varphi_i(x_{n-1}, x_n)$$

In this context, in case of congestion, x_n may not be able to satisfy the Flow Conservation Law:

$$\sum_{i=1}^{k} p_{il} > \sum_{j=1}^{k} p_{lj}$$

Therefore, in order to request local retransmissions when a packet of a privileged FEC is lost, GLRP explicitly needs to know the set of Gossip nodes that forward the packet. With this purpose, RSVP-TE has been extended not only to create the GPlane, but also to enable the retransmission requests, even, across non-Gossip nodes, because the deployment of GLRP only implies the enabling of the GLRP capability in a bottleneck nodes. It could be activated when the bandwidth reserved by RSVP-TE exceeds a predefined threshold in a particular node.

3.1. A Connection-Oriented GLRP Plane

Observe that in the MPLS-TP Control Plane, at the same time as the LSP is being signaled by RSVP-TE, the GPlane is configured at Gossip nodes. This integration of the GLRP with the MPLS-TP Control Plane allows that each Gossip node of the LSP knows its previous GLRP enabled node (i.e., its partner). The GLRP characterization info (*Gossip Level* and *Gossip Previous Hop*) is only sent when the LSP is being signaled, adding a new row in a table of the Gossip nodes, which is called the *GTable*. Observe that the FEC with higher Gossip levels will use routes across more GLRP enabled nodes and, in addition, these routers will reserve more capacity to buffers for them. This way, packets from the most privileged FEC (i.e., with higher Gossip Level) have higher probability to be recovered faster, because it will probably be retransmitted from a closer node.

Therefore, the GPlane can be considered as a connection-oriented subset of nodes, which have GLRP capabilities. This LSP that configure a GPlane in order to enhance the performance of a FEC with high requirements of delay and reliability is called *privileged LSP*.

In Figure 2, the Gossip node architecture is shown. The MPLS-TP architecture has been extended, integrating the *GTable* and the *GIndex*. GLRP messages of Control and Forwarding planes are also showed. Moreover, GLRP extends the RSVP-TE protocol in order to configure the GPlane as a subset of nodes of a privileged LSP. In the MPLS-TP Control Plane, when an ingress node receives an RSVP-TE message requesting for a new LSP, it inserts a new row in the *forwarding information base*, with information about how to forward data packets across nodes of the LSP that is being signaled. This is the info to be used by a router in the MPLS-TP Forwarding Plane when it receives a labeled packet and have to make the label swapping and forward the packet to the next hop. In this context, when RSVP-TE signals a new LSP for a privileged FEC, the GLRP

DOI: 10.1002/dac

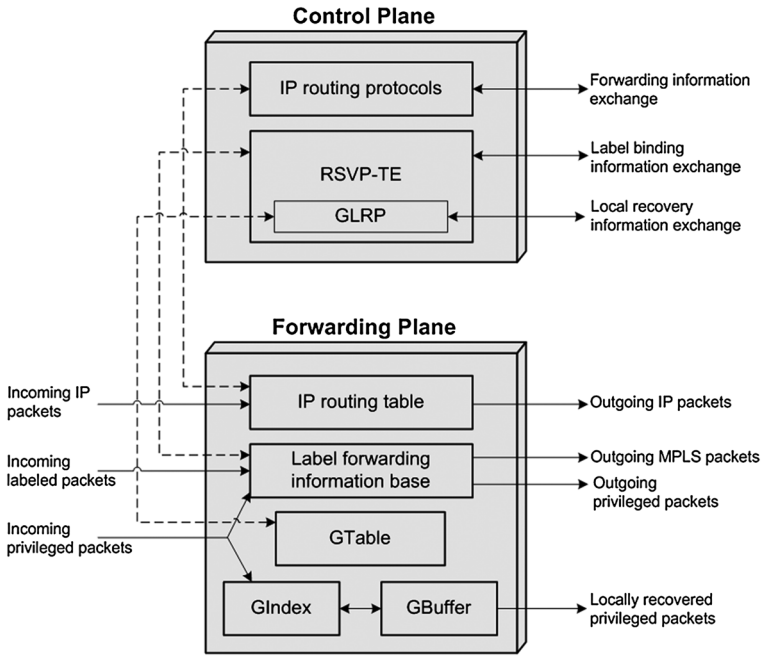

Figure 2. Multiprotocol Label Switching Transport Profile node architecture, with GLRP capabilities.

enabled nodes of the LSP to add a new row to the forwarding information base table, but in parallel they also insert into the GTable the info about the GPlane. Thus, this connection-oriented GPlane allows that when a packet from a privileged FEC is lost, the Gossip node already has all it needs to initiate a local retransmission request (*GReq*).

3.2. GLRP data structures

As described in the previous text, there is no need to carry GLRP information into data packets, because this task is only carried out in the Control Plane, that is, when the path is being configured by the RSVP-TE. This way, GLRP characterization info (Gossip Level and Gossip previous hop) is only sent when the LSP is being signaled, avoiding the need to forward GLRP information with each data packet.

Table 1 shows an example of a GTable of a node that forwards packets of four FEC. Each row contains a first column that identifies the FEC (by means of incoming and outgoing labels combination), a second column with the Gossip level of the FEC and, finally, a third column is used to record the address of the previous GLRP enabled hop (to send it a retransmission request in case of packet loss). Observe, for instance, that packets of FEC 36/68 and 108/44 are forwarded across the same previous Gossip node, with address x.x.160.17. However, they will be managed differently, because of their different Gossip levels.

The Index Table is used to enhance the access to the packet buffer. It allows optimizing the search for a packet in the GBuffer when a local retransmission request is received. It allows random access the buffer, because, instead of searching in the entire buffer for the requested packet, the Index Table

Table I. Format of the GTable.

Incoming label/outgoing label	Gossip level	Gossip PHOP address
4/32	11	x.x.160.12
36/68	1	x.x.160.17
108/44	18	x.x.160.17
74/60	4	x.x.160.35

DOI: 10.1002/dac

allows access just to the position in the GBuffer where the packet is located. To do this, an index key, which is the pointer to the position in the buffer where a packet is located, is used for each packet that is stored. This way, it is retrieved more quickly and can be retransmitted more efficiently, regardless of the size of the buffer, as there is no need to search in the whole buffer. If the packet is not stored in the buffer, the index also indicates this without accessing the buffer. In order to obtain this, a perfect hash function is used for the Index Table in order to obtain efficient lookup operations. It is a hash function that maps distinct elements in the buffer to a set of integers, with no collisions. Perfect hash is a very efficient function in terms of processing overhead, allowing for constant-time lookups and managing up to 1 million keys in a few seconds of CPU time if were needed [24].

Moreover, some keys of the Index Table can be periodically erased to minimize the size of the index. Based on a timestamp value associated with each packet in the table, the rows that have already reached the maximum waiting time for a possible retransmission of the associated packet can be erased (i.e., if a key of the index has not been requested for more than the value of *timestamp*, it is deleted). This value depends on the RTT_d that is the round trip time between the recovery node and the Gossip node that detects the loss. It is also used to choose which packet must be replaced by the new incoming packets of the prioritized FEC when the buffer is full. In this case, when a packet has been overwritten, future retransmission requests for that packet will be forwarded toward another upstream Gossip node. Note that this situation is more probable if it had been assigned a low Gossip level to the FEC and, therefore, a lower reservation in the buffer.

Summarizing, a particular packet is only deleted from the buffer when it is overwritten by a new incoming packet. Nevertheless, the fact that would rather a closer upstream node has deleted a packet do not implies that further upstream nodes have deleted the packet too. For instance, if the closer node is more congested due to cross traffic than the previous nodes, the closer one deletes the packet earlier than further upstream nodes, which can store the packet in the buffer for longer. However, the record in the Index Table associated with a packet is only deleted from the table when there are no possibilities to receive a local retransmission request. The value *timestamp* is used to calculate the time-margin, and the objective is only avoided that the Index Table grows indefinitely.

3.3. GLRP states diagram

In Figure 3, a states diagram of the operation of a Gossip node is shown. In the MPLS-TP Forwarding Plane, the state of a Gossip node is *Data Forwarding*, switching labels and forwarding data packets to the next hop. There are only two events that change this state in a Gossip node. The first is the detection of a packet loss from a privileged FEC. In this case, the GLRP enabled node obtain FEC and GPlane information and change its state to *Local recovery request*, sending a local retransmission request (*GReq*) to its GPlane partner (the closest upstream Gossip node). When a response (*GAck*) is received, it changes back to the initial state.

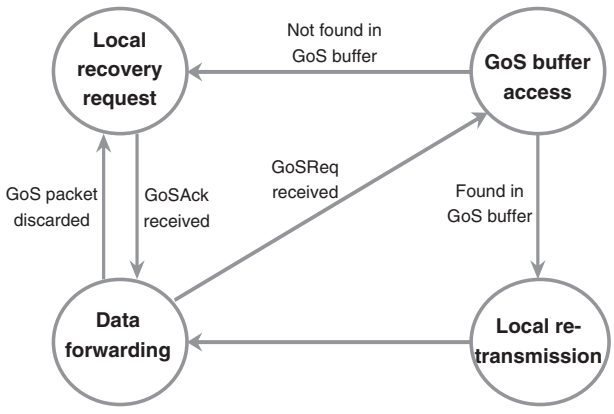

Figure 3. States diagram of a GLRP enabled node.

DOI: 10.1002/dac

The other event that changes the state is the reception of a *GReq* from any downstream Gossip node. In this case, the node changes its state to *Buffer Access*, in order to look for the requested packet, according to the information received in the *GReq*. If the packet is stored in the GBuffer, an optional *GAck* can be sent in response to the *GReq*, indicating that the requested packet was found and it will be retransmitted locally. Next, it changes to *Local Retransmission* state to obtain the packet from the GBuffer and reforward it. After that, it will return to initial *Forwarding* state. In case of not finding the packet in GBuffer, it can send a *GAck* message to inform that packet was not found. Next, it changes to the *Local Recovery Request* state, in order to forward the *GReq* to its previous Gossip node in the GPlane, if there is any more.

4. GLRP MESSAGES

Gossip levels can easily be mapped to MPLS-TP FEC that is commonly used to describe a packet-destination mapping. A FEC is a set of packets to be forwarded in the same way (e.g., using the same path or QoS criteria). One of the reasons to use the FEC is that it allows grouping packets in classes. It can be used for packet routing or for efficient QoS supporting too; for instance, a high priority FEC can be mapped to a healthcare service or a low priority FEC to a Web service.

4.1. Signaling the GPlane

The label is used by MPLS to establish the mapping between FEC and packet, because an incoming–outgoing labels combination identifies a particular FEC. With different classes of services, different FEC with mapped labels will be used. In our proposal, *privileged FEC* concept is used to classify the different Gossip levels, giving more priority to the most privileged FEC. Thus, a privileged FEC gives different treatments to packets from FEC with different privileges, although they are being forwarded across the same route. In order to optimize the GLRP signaling in the MPLS-TP Forwarding Plane, GLRP characterization info (Gossip Level and GPlane) can be signaled by RSVP-TE in the MPLS-TP Control Plane. When a privileged LSP is being configured, extended RSVP-TE *Path* and *Resv* messages forward the info about Gossip Level and GPlane (see Figures 4 and 5, respectively).

When a new LSP tunnel is being signaled in the Control Plane, a Gossip node that receives a GLRP-extended *Path* message will access this GLRP info in order to add a new row in its GTable.

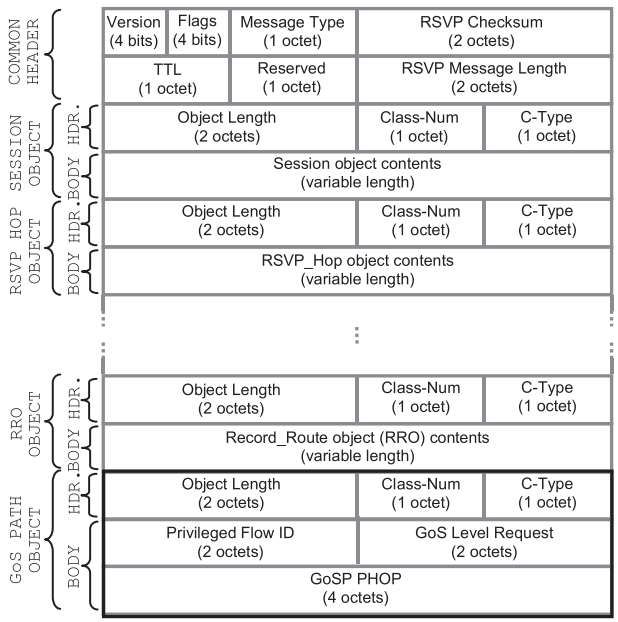

Figure 4. Extended message resource reservation protocol traffic engineering path, with *GPath* object.

DOI: 10.1002/dac

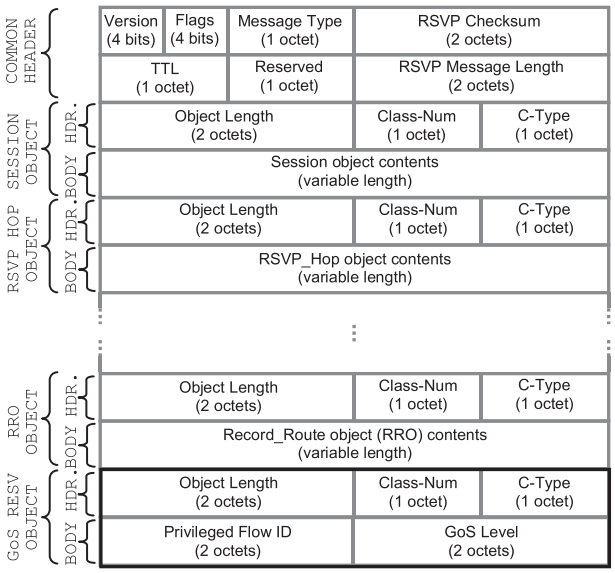

Figure 5. Extended message resource reservation protocol traffic engineerin0067 Resv, with *GResv* object.

Then, it records its Internet Protocol (IP) address into the *GPlane PHOP* field of the *GPath* object, because it is the previous hop of the next downstream Gossip node in the LSP. This way, if a Gossip node detects a packet lost, it only sends a local retransmission request to its previous hop in the GPlane. If that LSR cannot find the requested packet, it could forward the request to its Gossip previous hop, and so forth. Finally, following the RSVP-TE operation way, when an LSP signaling is being confirmed, Gossip information will also be confirmed with the reception of a GLRP-extended *Resv* message, confirming the requested Gossip level.

4.2. Signaling the GLRP local retransmissions

It is not needed to send the entire GPlane in every *GReq* message, because Gossip nodes have an entry in the GTable with the GPlane in the previous hop for each FEC. Thus, in case that a GPlane in the previous hop cannot satisfy a local retransmission request, it reads from its GTable to obtain the next upstream Gossip neighbor, to resend it the received GReq. Therefore, the node that initiates a GReq never needs to send requests to all the nodes of the GPlane, but only to its previous Gossip neighbor. For this reason, only one address is needed to be inserted in the GPlane PHOP column of the GTable, instead of the entire GPlane. With this purpose, RSVP-TE Hello message has been extended (Figure 6).

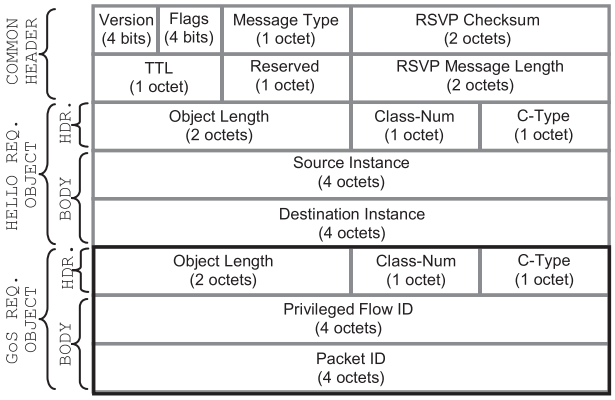

Figure 6. GLRP extended Hello message format, with *GReq* object after the *Hello* object.

DOI: 10.1002/dac

In particular, *Hello Request* message has been extended with a *GReq* object, in order to request to the upstream GPlane in the previous hop the retransmission of a lost packet of the FEC (specified in *Privileged FEC ID* field). Upstream Gossip node that receives the *GReq* message optionally sends a response in an extended *Hello Ack* message (Figure 7), with a *GAck* object, in order to notify if requested packet has been found in the GBuffer. Furthermore, following the RSVP-TE operation way, *Source Instance* and *Destination Instance* of the *Hello object* are used to test connectivity between GPlane neighbor nodes.

In Figure 8, operation of the GLRP when a packet that is being forwarded from X_1 to X_5 (with latency $\delta_{1,5}$) is discarded in the intermediate node X_4 is shown.

For instance, in this simple topology, three GPlane diameters ($d=1$, $d=2$, and $d=3$) can be used to achieve a successfully local retransmission from X_4. First, X_4 sends a local retransmission request (*GReq*) to the first node of the GPlane (X_3). Then, that node will send a response (*GAck*) to indicate whether it is located or not in the GBuffer. If it is found (this fact implies *diameter* = 1), it will send that locally recovered packet (*LRP*) toward its destination. But if it is not found, X_3 will send a new *GReq* message to its PHOP in the GPlane (X_2). If it finds the packet, the successfully diameter would be $d=2$. Finally, if X_1, which is the last node of the GPlane, finds the lost packet, then a

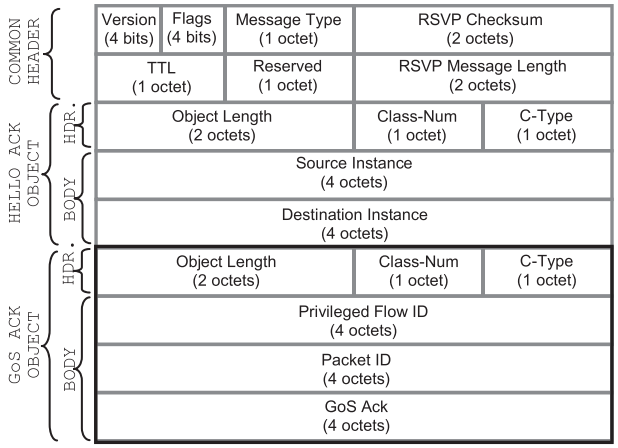

Figure 7. GLRP extended Hello message format, with *GAck* object after the *Hello* object.

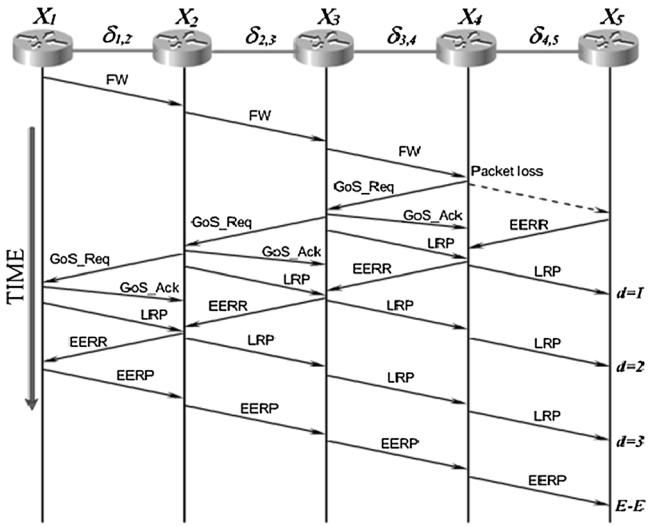

Figure 8. Local retransmission operation when a packet is discarded in an intermediate node.

DOI: 10.1002/dac

diameter $d = 3$ would achieve a successfully local retransmission. Furthermore, this local recovery process is compared in the figure with an end-to-end retransmission request of an end-to-end retransmission packet.

5. NETWORK MODEL AND ASSUMPTIONS

A QoS backbone network is represented as a graph $G = (R, L)$, where R is the set of routers, and L is the set of edges or links. Let δ_{ij} be the delay of the link $(r_i, r_j) \in L$, and let $\delta(r_i, r_n)$ be the delay of a path $LSP_{i,n}$ between an ingress node of the network, r_i and an egress node of the network, r_n. Dijkstra algorithm allows us to optimize the packet delay when packets are forwarded between any two routers, r_i and r_j, of $LSP_{i,n}$:

$$min\ \ \Delta(r_i, r_j) = \sum_{i=1}^{n} \sum_{j=1}^{n} \Delta_{ij} x_{ij}$$

subject to:

$$\sum_{l=2}^{n} x_{1l} = 1$$

$$\sum_{i=1}^{n} x_{il} - \sum_{j=1}^{n} x_{lj} = \ 0,\ \ l = 2, 3, \ldots, n-1$$

$$\sum_{l=1}^{n-1} x_{ln} = 1$$

where:

$$\Delta_{i,i} = 0, \forall i \in R$$

$$x_{i,j} = \ 1, \forall (r_i, r_j) \in \mathrm{LSP}_{i,n},$$

and

$$x_{i,j} = 0, \forall (r_i, r_j) \notin LSP_{i,n}$$

For instance, in the case of a congested egress node r_n without GLRP capabilities, the retransmissions of dropped packets would be performed end-to-end (E-E) by the upper layers. In this case, when a packet is dropped in r_n, this loss is detected at the source node when the sink does not send the *acknowledgement* toward the source. This way, the function *Loss Detection Time* ($\mathrm{LDT}_{E\text{-}E}$) of $LSP_{i,n}$ is:

$$\mathrm{LDT}_{E-E}(r_i, r_n) = \sum_{l=i}^{n-1} \delta_{l,l+1} x_{l,l+1} \tag{1}$$

In the best case, if upper layers perform the end-to-end retransmission of lost data using the *Fast-Retransmit* mechanism, then it would need to wait for two more disordered packets, and the delay of the retransmitted packet would be:

$$\delta_{E-E}(r_i, r_n) = 2 \sum_{l=i}^{n-1} \delta_{l,l+1} x_{l,l+1} \tag{2}$$

Therefore, the total delay $\Delta_{E-E}(r_i, r_n)$ to retransmit a packet toward r_n is derived from Eq. (1) plus Eq. (2):

DOI: 10.1002/dac

$$\Delta_{E-E}(r_i, r_n) = 3 \sum_{l=i}^{n-1} \delta_{l,l+1}\, x_{l,l+1} \tag{3}$$

5.1. End node r_n with GLRP capabilities:

However, if r_n is a Gossip egress router and in case a packet is lost in r_n, the (LDT_d) between source and sink nodes of the path $LSP_{i,n}$ is:

$$\mathrm{LDT}_d(r_i, r_n) = \sum_{l=i}^{n-1} \delta_{l,l+1}\, x_{l,l+1}, \tag{4}$$

where d is the diameter or the number of hops of the local retransmission.

In this case, the minimal delay (δ_d) of the local retransmission is the delay of the *GReq* message from the egress router to the node at d hops upstream, plus the delay of the retransmitted packet:

$$\delta_d(r_i, r_n) = 2 \sum_{l=n-d}^{n-1} \delta_{l,l+1}\, x_{l,l+1}, \tag{5}$$

subject to: $0 < d < n–i$, because if diameter in Eq. (5) is $n-i$, then $l = n$, $d = n$, $(n–i) = n$, and $n+i = i$, we would find that:

$$2 \sum_{l=n-d}^{n-1} \delta_{l,l+1}\, x_{l,l+1} = 2 \sum_{l=i}^{n-1} \delta_{l,l+1}\, x_{l,l+1}, \tag{6}$$

That is, it would be an *e2e* retransmission.

However, if in Eq. (5) *diameter* is bigger than $n-i$, then it would be trying to obtain a retransmission from a previous node to r_i, but this one is the sender and a retransmission from a previous node of the sender is unfeasible. Thus, the total delay $\Delta_d(r_i,r_n)$ to retransmit a packet in r_n is derived from Eq. (4) and Eq. (5):

$$\Delta_d(x_i, x_n) = \sum_{l=i}^{n-1} \delta_{l,l+1}\, x_{l,l+1} + 2 \sum_{l=n-d}^{n-1} \delta_{l,l+1}\, x_{l,l+1} \tag{7}$$

At this point, we can test if $\Delta_d(r_i,r_n) < \Delta_{E-E}(r_i,r_n)$:

$$\sum_{l=i}^{n-1} \delta_{l,l+1}\, x_{l,l+1} + 2 \sum_{l=n-d}^{n-1} \delta_{l,l+1}\, x_{l,l+1} < 3 \sum_{l=i}^{n-1} \delta_{l,l+1}\, x_{l,l+1} \tag{8}$$

$$2 \sum_{l=n-d}^{n-1} \delta_{l,l+1}\, x_{l,l+1} < 2 \sum_{l=i}^{n-1} \delta_{l,l+1}\, x_{l,l+1} \tag{9}$$

Therefore, according to Eq. (5) and Eq. (2), we only need to verify in Eq. (9) that $\delta_d(r_i, r_n) < \delta_{E\text{-}E}(r_i, r_n)$. The only condition that differentiates the members of Eq. (9) is the set of values of the variable l. It only needs to be demonstrated that l takes a lower number of values in $\delta_d(x_i, x_n)$ than in $\delta_{e2e}(x_i, x_n)$:

$$\begin{aligned} &n–1–(n–d) < n–1–i \\ &n–1–n + d < n–1–i \\ &-1 + d < n–1–i \\ &d < n–i, \end{aligned} \tag{10}$$

We find that the problem remains in the feasibility zone of the problem, because Eq. (10) is one of the restrictions of Eq. (5).

DOI: 10.1002/dac

Thus, it has been demonstrated that $\Delta_d(r_i,r_n) < \Delta_{E-E}(r_i,r_n)$. Therefore, local retransmissions perform with delay benefits, that is, $\Delta_{E-E}(r_i,r_n) - \Delta_d(r_i,r_n) > 0$. In the case that the egress router had GLRP capabilities, the delay improvement for every lost packet is:

$$\Delta_{E-E}(r_i, r_n) - \Delta_d(r_i, r_n) = 2 \sum_{l=i}^{n-d-1} \delta_{l,l+1}\, x_{l,l+1} \tag{11}$$

5.2. Intermediate node r_{DD} with GLRP capabilities:

Let r_{DD} be a core Gossip node. In the case a packet is dropped by x_{DD}, the (LDT_d) between the source and congested node r_{DD} would be:

$$\mathrm{LDT}_d(r_i, r_{DD}) = \sum_{l=i}^{DD-1} \delta_{l,l+1}\, x_{l,l+1} \tag{12}$$

The delay of the local retransmission with a diameter d is the delay of the *GReq* message plus the delay of the retransmitted packet from the node $DD-d$ hops upstream:

$$\delta_d(r_i, r_{DD}) = 2 \sum_{l=DD-d}^{DD-1} \delta_{l,l+1}\, x_{l,l+1}, \tag{13}$$

subject to: $0 < d \le DD-i$,

If diameter in Eq. (20) is bigger than $DD-i$, then it would be trying to obtain a retransmission from a previous node to r_i, but this one is the source of data and it would be unfeasible. In this case, the retransmission from the source node r_i, with $d = DD–i$, performs better than the *E-E* case, because r_{DD} is a previous node to r_n: $DD < n \Rightarrow DD-i < n–i$), that is, the packet is retransmitted in a lower number of hops.

Therefore, total delay $\Delta_d(r_i,r_n)$ to retransmit a packet in r_n is derived from Eq. (12) and Eq. (13):

$$\begin{aligned} \Delta_d(r_i, r_n) &= LDT_d(r_i, r_{DD}) + \delta_d(r_i, r_{DD}) + \sum_{l=DD}^{n-1} \delta_{l,l+1}\, x_{l,l+1} = \\ &= LDT_{e2e}(r_i, r_n) \;+\; \delta_d(r_i, r_{DD}) \end{aligned} \tag{14}$$

In this case, we can test again if $\Delta_d(r_i,r_n) < \Delta_{E-E}(r_i,r_n)$:

$$\sum_{l=i}^{n-1} \delta_{l,l+1}\, x_{l,l+1} + 2 \sum_{l=DD-d}^{DD-1} \delta_{l,l+1}\, x_{l,l+1} \;<\; 3 \sum_{l=i}^{n-1} \delta_{l,l+1}\, x_{l,l+1} \tag{15}$$

Optimizing, we obtain:

$$2 \sum_{l=DD-d}^{DD-1} \delta_{l,l+1}\, x_{l,l+1} \;<\; 2 \sum_{l=i}^{n-1} \delta_{l,l+1}\, x_{l,l+1} \tag{16}$$

Therefore, according to Eq. (5) and Eq. (2), once again, we only need to verify in Eq. (16) that $\delta_d(r_i, r_n) < \delta_{E\text{-}E}(r_i, r_n)$. As in Eq. (10), in this case, we also find that the problem remains within the feasibility zone. Therefore, Eq. (14) performs with an improved delay: Eq. (11–14 >0:

$$\Delta_{E-E}(r_i, r_n) - \Delta_d(r_i, r_n) \;=\; 2 \left(\sum_{l=i}^{DD-d-1} \delta_{l,l+1}\, x_{l,l+1} + \sum_{l=DD}^{n-1} \delta_{l,l+1}\, x_{l,l+1} \right) \tag{17}$$

This proof can easily be extended to include other metrics or to the case in which an intermediate node is requesting local retransmissions.

DOI: 10.1002/dac

6. PERFORMANCE EVALUATION

In this section, we present extensive performance evaluation through simulations in order to determine the behavior of GLRP as a function of different protocol parameters and under different scenarios or conditions. We have carried out a series of simulations focused on AT&T backbone network topology (Figure 9), which is MPLS enabled to provide QoS for customers who require value-added services. In our simulations, AT&T core topology is characterized by 120 LER nodes, 30 LSR nodes, and 180 links, with capacities in the range of (45 Mbps and 2.5 Gbps). A GLRP enabled node has been located at the eight routers with the biggest connectivity. In scenarios, signaled LSP are unidirectional and the bandwidth demanded for each FEC is drawn from a distribution over the range of (64 Kbps and 4 Mbps). In order to analyze the effect that GLRP retransmissions have on transport layer protocols, several privileged services over TCP/IP that use LSP across a different number of Gossip nodes have been compared with not privileged TCP/IP services across the same paths. LSP congestion has also been considered in the range of (0.01%, 4%).

In order to determine the effectiveness of the buffer, we will measure in charts the *buffer hit ratio*, that is, the number of successful buffer reads divided by the total number of buffer accesses. Next, in the performance comparison, the protocol is evaluated analyzing the *packet delivery ratio*, as the total number of delivered packets divided by the total number of packets sent and, finally, the *end-to-end delay* as the delay for every packet delivered at the egress node. Routers with GLRP capabilities maintains exactly one data buffer regardless of the number of connections or destinations that node serves. A larger data buffer, do not imply a more efficient performance of the GLRP policy. Indeed, the optimal buffer size does not depend on the link velocity, but on the percentage of loss. When the link speed is increased, the data packets ratio can also be increased.

However, the delivery time of the *GReq* messages is also reduced. Thus, data packets can be stored for a shorter time until they are overwritten by the new incoming packets. This hypothesis is well supported by the results in the Figure 10, which show the buffer hit ratio when incoming packet ratio and buffer size are increased, with a maximum diameter of eight hops. In general, the data buffer hit ratio shows a significant growth when the buffer size is increased by six in the chart.

Observe that the optimal size of the GBuffer depends on the RTT_d that is the round trip time between the recovery node and the Gossip node that detects the loss. Indeed, RTT_d represents the minimal caching time for the packets at a recovery node. Furthermore, the hit ratio is the highest, although the packet ratio is increased. However, if the buffer size is significantly lower than $Ratio \times RTT_d$, the hit ratio is medium but shows a significant decrease when the packet ratio is increased, with a lower decrease if the diameter of the local recoveries is shorter. For this reason, for a particular buffer size, the graph shows points where there is an increase of the plot, although the packet ratio is increased, because the local recoveries could have been successful with a shorter diameter. In particular, at 20 Mbps, regardless of the data buffer size, more than 65% of the accesses are successful and the hit ratio always remains close to 100% when the buffer size is 24 KB or more. However, with a buffer size of 18 KB or lower, the plots become more dispersed and at 200 Mbps the plots are dispersed further still. These

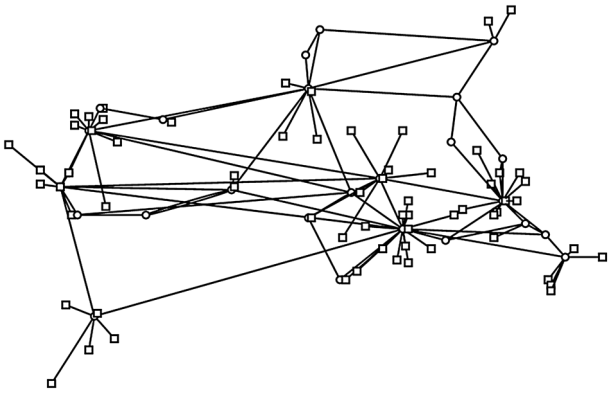

Figure 9. AT&T core topology characterization.

DOI: 10.1002/dac

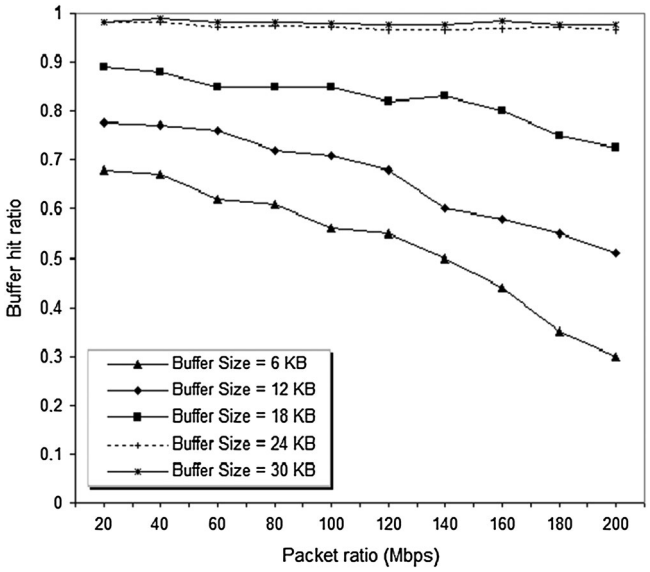

Figure 10. Buffer hit ratio as a function of incoming packet ratio.

results suggest the evidence of temporal locality in the dropped packets. Recall that, whenever a node with GLRP capabilities fails to forward a data packet to its next hop, it sends a request message backwards. The GLRP enabled node that receives this message accesses to its GBuffer trying to recover the packet. Frequently, a recently dropped packet in a downstream Gossip node is still in the GBuffer of the upstream nodes. Hence, a node does not need a large data buffer because, with temporal locality, only the most recently buffered packets are accessed and used for local retransmissions. This implies that it doesn't matter how large the data buffer is because the relatively old packets are never requested and can be overwritten by the new incoming data packets. In the case of a buffer size of 24 KB, the hit ratio already does not depend on the packet ratio, but on the diameter of the local recoveries and level of congestion. Hence, with a shorter diameter, the hit ratio can be high although the buffer size is much too small. Therefore, the buffer size of the Gossip nodes must be optimized bearing in mind not only the maximum diameter allowed, but also the level of congestion in the network domain.

Figure 11 shows the end-to-end delay of data packets. It shows the growth as the buffer of the GLRP enabled nodes is increased. The delay falls as far as 1.42×10^{-6} s from the case where there

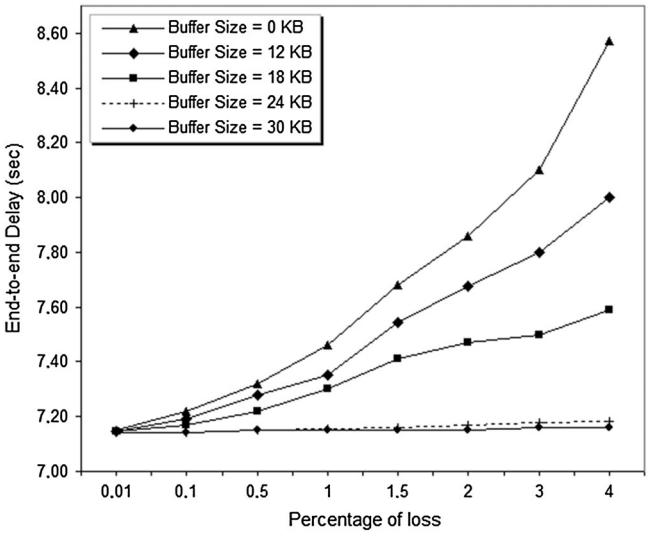

Figure 11. End-to-end delay as a function of packet loss.

DOI: 10.1002/dac

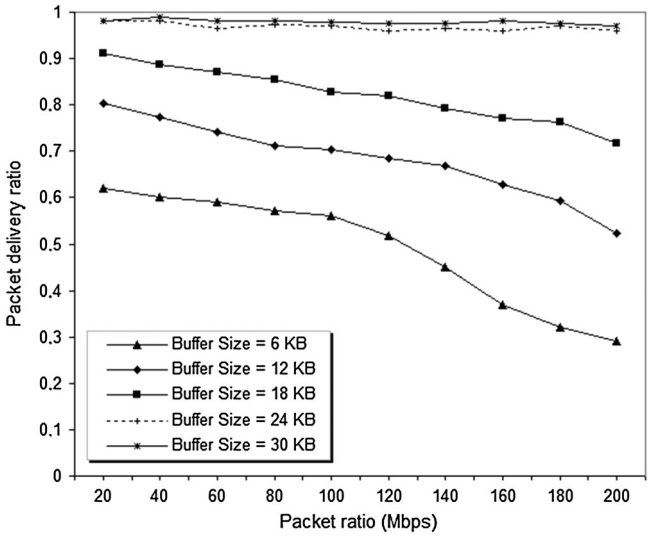

Figure 12. Delivery ratio of locally recovered packets, as a function of the incoming packet ratio.

is no data buffer to the case where the data buffer has a size of 30 KB. This increase is not at all unexpected. Recall that a node invokes a local retransmission when it fails to forward a data packet to its next hop due to congestion. Moreover, when a node fails to salvage a data packet from its data buffer, it propagates the retransmission request message upstream. Without a local retransmissions scheme, this undeliverable packet is simply discarded. However, with local retransmissions, this packet (and other undeliverable packets) could have been stored in the buffer of the previous GLRP enabled nodes. Thus, if the lost packet had been found in any of the previous Gossip nodes, this packet (and other undeliverable packets) could be forwarded toward its destination.

Finally, for the case of a buffer size of 24 KB or 30 KB, the delay remains steadier. Thus, the FECs with high requirements of delay variability would obtain performance benefits too. Figure 12 shows the packet delivery ratio for the simulations with varying incoming buffer size and the incoming packet ratio of the network. The results in the chart show that buffering of data packets can indeed improve packet delivery. Furthermore, from the data buffer size of 24 KB, there is no significant increase in packet delivery as the data buffer size is increased; from this value, there is no evidence suggesting that increasing the data buffer size results in a higher number of delivered packets. However, with smaller buffer sizes, the plots are decreased significantly and a higher number of end-to-end retransmissions are needed.

7. CONCLUSIONS AND FUTURE RESEARCH

This article discusses the GLRP as a congestion control mechanism in MPLS-TP domains with the aim of improving reliability and performance of prioritized QoS-aware services. We have first defined and discussed the requirements for GLRP over MPLS-TP. Then, the proposal has been analytically studied, and finally the benefits derived from local retransmissions of lost traffic have been evaluated. Because of the property of temporal locality in lost packets, a small data buffer has been used to significantly reduce the number of dropped packets, thereby improving packet delivery of QoS-aware services. However, buffering of data packets at the network layer is a new technique for improving robustness and it needs further investigation. Although the simulations performed in this paper were quite extensive, and this protocol is designed to solve real-world congestion problems in backbone MPLS-TP networks, further research is required to determine its behavior with the QoS requirements in real backbone networks, particularly in DiffServ over MPLS-enabled networks when new premium service classes are enabled.

DOI: 10.1002/dac

ACKNOWLEDGEMENT

The authors would like to thank the Government of Extremadura, Spain, for financially supporting this research under contract no GRU10116.

REFERENCES

1. Alshaer H, Elmirghani J. Enabling novel premium service classes in DiffServ over MPLS-enabled network. *International Journal of Communication Systems* 2008; **18**(5):447–464. DOI:10.1002/nem.693.
2. Wan C-Y, Eisenman S, Campbell A. CODA: congestion detection and avoidance in sensor networks. *ACM International Conference on Embedded Networked Sensor Systems*, 2003; 266–279. DOI: 10.1145/958491.958523.
3. Yang H, Chen X, Hu R. An end-to-end content-aware congestion control approach for MPEG video transmission. *International Conference on Future Generation Communication and Networking* 2008; **1**:122–125. DOI: 10.1109/FGCN.2008.76.
4. Ko E, An D, Yeom I, Yoon H. Congestion control for sudden bandwidth changes in TCP. *International Journal of Communication Systems* 2012; **25**(12):1550–1567. DOI: 10.1002/dac.1322.
5. Papastergiou G, Georgiou C, Mamatas L, Tsaoussidis V. A delay-oriented prioritization policy based on non-congestive queuing. *International Journal of Communication Systems* 2011; **24**(8):1065–1086. DOI: 10.1002/dac.1215.
6. Zhao Y, Han D, Zhang J, Xing J. QoS sensitive routing in DiffServ MPLS-TP networks. *International Conference on Computer Application and System Modeling (ICCASM)*, Vol. **1**, October 2010; 726–730.
7. Cao C. Packet-level optimization for transmission performance improvement of internet-bound traffic in a MPLS-TP network. *IEEE/OSA Journal of Optical Communications and Networking* 2010; **2**(11):991–999.
8. Winter R. The coming of age of MPLS. *IEEE Communications Magazine* 2011; **49**(4):78–81.
9. Zhou X, Zhang D, Yang Y, Obaidat MS. Network-coded multiple-source cooperation aided relaying for free-space optical transmission. *International Journal of Communication Systems* 2012; **25**(11):1465–1478. DOI: 10.1002/dac.2450.
10. Zhani MF, Elbiaze H, Kamoun F. Analysis and prediction of real network traffic. *Journal of Networks* 2009; **4**(9):855–865.
11. Li Y, Leith D, Shorten R. Experimental evaluation of TCP protocols for high-speed networks. *IEEE/ACM Transactions on Networking* 2007; **15**(5):1109–1122.
12. Oishi T. Implementation of packet transport system using MPLS-TP technologies. 8th Asia-Pacific Symposium on Information and Telecommunication Technologies (APSITT). June 2010; 1–6.
13. Kim M, Mutka MW, Kim H-Y. ESC: estimation of selecting core for reducing multicast delay variation under delay constraints. *International Journal of Communication Systems* 2011; **24**(1):40–52. DOI: 10.1002/dac.1137.
14. Bagula AB. On achieving bandwidth-aware LSP//spl lambda/SP multiplexing/separation in multi-layer networks. *IEEE JSAC* 2007; **25**(5):987–1000.
15. LF Zhou, L Chen, Pung HK, Ngoh LH. Identifying QoS violations through statistical end-to-end analysis. *International Journal of Communication Systems* 2011; **24**(10):1388–1406. DOI: 10.1002/dac.1273.
16. Pruteanu A, Iyer V, Dulman S. FailDetect: gossip-based failure estimator for large-scale dynamic networks. *20th International Conference on Computer Communications and Networks (ICCCN)*, August 2011, 1–6. DOI: 10.1109/ICCCN.2011.6006082.
17. Chen J-L, Liu S-W, Wu S-L, Chen M-C. Cross-layer and cognitive QoS management system for next-generation networking. *International Journal of Communication Systems* 2011; **24**(9):1150–1162. DOI: 10.1002/dac.1218.
18. Li L, Buddhikot MM, Chekuri C, Guo K. Routing bandwidth guaranteed paths with local restoration in label switched networks. *IEEE Journal on Selected Areas in Comm* 2005; **23**(2):437–449.
19. Rodríguez-Pérez FJ, González-Sánchez JL, Gazo-Cervero A. RSVP-TE extensions to provide guarantee of service to MPLS. *6th IFIP Networking*, May 2007, 808–819.
20. Ashour M, Tho L-N. Delay-margin based traffic engineering for MPLS-DiffServ networks. *Journal of Communications and Networks* 2008; **10**(3):351–361. DOI: 10.1109/JCN.2008.6388356.
21. Ma L, Barner KE, Arce GR. Statistical analysis of TCP's retransmission timeout algorithm. *IEEE/ACM Transactions on Networking* 2006; **14**(2). DOI: 10.1109/TNET.2006.872577.
22. Gurtov A, Ludwig R. Responding to spurious timeouts in TCP. *INFOCOM 2003*. Vol **3**, April 2003. DOI: 10.1109/INFCOM.2003.1209251.
23. Cho I, Han J, Lee J. Enhanced response algorithm for spurious TCP timeout (ER-SRTO). *International Conference on Information Networking*, January 2008. DOI: 10.1109/ICOIN.2008.4472748.
24. Pao D, Wang X, Lu Z. Design of a near-minimal dynamic perfect hash function on embedded device. *15th International Conference on Advanced Communication Technology*, January 2013; 457–462.