On the Reduction of Congestion for Multimedia Streaming in Diffserv Networks

Panayotis Fouliras and Christos K. Georgiadis

Department of Applied Informatics
University of Macedonia
Egnatias 156, 540 06 Thessaloniki, GREECE
{pfoul, geor}@uom.gr

***Abstract* - In this paper, we propose a simple meter and classification scheme, specifically designed for real-time multimedia streaming applications in Diffserv QoS networks. This is based upon the observation that conformance deterioration takes place on such networks for all traffic of variable-size packets. We show that it is simple to detect, shape and drop packets which experience exceeding delay, thus reducing the incurring congestion and providing better network resource utilization.**

Index Terms* - *Video-on-Demand, multimedia, streaming, QoS, Diffserv.

I. Introduction

The Internet has been based on the model of a packet-switched network, providing a best-effort service. Provided that the traffic is non-real-time, this is a valid choice with the additional advantage of better utilization of the available resources. Typically, the network management employed by an ISP has been to provide statistical assurance by tracking the amount of traffic tagged as inbound when crossing various links over time and provide enough capacity to carry this traffic, even at times of congestion [1].

The last few years, however, technology advances and cost reduction have promoted the introduction of applications with real-time constrains, such as audio and video. Multimedia applications with real-time constrains need a guaranteed delivery of packets in terms of packet loss and delay which is not assured by the best-effort service provided by the Internet [17]. The necessity of new Quality of Service (QoS) models with better network utilization became evident. For this reason IETF developed two additional models for QoS: Intserv and Diffserv [2].

The Intserv (Integrated Service) approach is a framework under which applications can choose among multiple, controlled levels of delivery service for their data packets, individualized for each session. A participating router must know the resources reserved for each session (buffers, link bandwidth) and thus maintain individual state information per flow. Furthermore, each session must setup a call; here each router in the path from source to destination must be examined whether it is possible to reserve the necessary resources locally, in order to meet the QoS requirements in a guaranteed way. Typically the RSVP protocol is used for this purpose.

The problem with Intserv though is that per flow reservation is not scalable, apart from the lack of many prespecified service classes that would offer more flexibility.

The Diffserv (Differentiated Service) approach employs a small set of building blocks from which a variety of aggregate behaviors may be built. In this respect, edge routers classify, mark, measure and shape or drop incoming packets according to some pre-negotiated traffic profiles [16]. However, internal Diffserv-aware routers within a domain or ISP forward packets only according to those markings. The later is called PHB (Per-Hop Behavior) and uses the DS-field of the old TOS field of the IP header, thus maintaining full compatibility with existing standards. Thus, Diffserv offers service differentiation among the traffic aggregates and not among individual flows.

Due to the flexibility of the marking scheme, as well as the fact that no flow states need to be maintained, Diffserv is considered as a reasonable candidate for the offer of services with QoS characteristics beyond best-effort on a large scale [14], [15]. Currently, two PHBs are standardized and under active discussion within the Diffserv working group: EF (Expedited Forwarding) [11] and AF (Assured Forwarding) [3] PHB. The former provides for each traffic class the abstraction of an isolated link with a minimum guaranteed link bandwidth. AF divides traffic into four classes, with three drop preference categories [9]. Each class however is guaranteed a minimum amount of bandwidth and buffering.

Nevertheless, there is considerable criticism for Diffserv, since an “expensive” class might be so popular that congestion could cause more delay compared to a “cheaper” but unpopular class, given that each class is guaranteed a minimum amount of bandwidth. In all cases, policing is costly and Diffserv could run into problems if traffic passes through more than one administrative domain with different class guarantees [13]. Furthermore, policing does not provide fairness between TCP and UDP traffic as pointed out in the case of AF PHB [12].

So far, one important point about shaped flows has been that they keep their conformance consistently once accepted under some specific profile. This may be true when packets are of constant size. In the case of variable size packets, however, conformance consistency may not be maintained at egress points, even without any network interference as pointed out in [4] and [5].

Furthermore, multimedia traffic with real-time constraints is quite sensitive to jitter and any congestion can cause the information conveyed by the respective packets to become obsolete by the time they reach their destination. Video, in particular, typically requires a large amount of packets to be sent. It is, therefore, important to examine whether significant savings in terms of network resources can be achieved by detecting and dropping such packets as early as possible at intermediate nodes – especially in Diffserv-aware networks.

Based on these assumptions, we propose the use of a meter and classifier at the egress of each successive DS region, which detects and drops RTP packets with excessive delays, thus freeing valuable network resources and reducing congestion.

The rest of the paper is organized as follows. In section 2 we formulate the problem. In section 3 we present our proposal and a preliminary analysis of its performance. Our conclusions follow in section 4.

II. Problem Formulation

We assume that there is at least one Diffserv ISP domain between a source and a destination host pairs. Furthermore, there are Diffserv-aware routers at both ingress and egress boundaries of each domain with appropriate classifiers not only of the BA (Behavior Aggregate), but also of the MF (Multi Field) type [6]. In this way, it is possible for them to examine more packet fields than just the DS code.

We also assume that there are at least two video/audio flows that pass through the nodes mentioned above over the same time period. These flows may be both unicast and/or multicast broadcasts, with multicast groups being dynamic in nature [7]. Thus, there are n senders and m receivers, where $n \leq m$. RTP [8] over UDP is mostly used for carrying video and audio.

Given these assumptions, we want to determine the following:

- Whether conformant RTP packets can become non-conformant to jitter restrictions, while traversing subsequent ISP domains. Such packets would be probably dropped by the receiver, so it is best to drop them as soon as possible in order to free network bandwidth.

- If the above is correct, then we need to estimate the amount of such traffic which becomes non-conformant. In this way we can establish whether appropriate measures are worth taking.

- If the previous hold, then we need to find a solution for detecting and dropping RTP packets. This solution must be simple, efficient and compatible with the existing Diffserv framework, so that it is easy to implement and deploy.

III. Problem Solution

A. *Conformance Deterioration*

RTP packets can be of variable size, since the multimedia content they convey depends on the actual codec(s) used. In general, Diffserv QoS regions carry shaped traffic which is conformant to the respective profile(s) at ingress points. Also, it is understood that video transmission to large, dynamic multicast groups is possible to generate at least transient connections [9]. Although this may be largely true for fixed size packets, it is important to determine whether it holds for variable sized packets, as well.

As shown in [5], it is possible to have conformance deterioration for any type of variable size packets. Let us assume a Diffserv region composed of three routers which all have a TB (Token Bucket) mechanism as depicted in Fig. 1. The first two routers (R1 and R2) are ingress and the third (R3) an egress node. All routers use a TB shaper with parameters (r, b), where r is the rate of generated tokens and b the size of the bucket. Packets are generated at nodes R1 and R2 which are then regulated by the TB included in each of them. This traffic is sent over the respective link at a constant rate C and regulated at R3, before ending at the Sink.

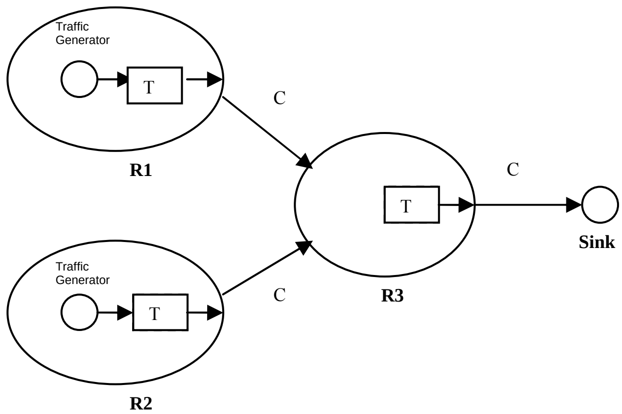

Fig.1 Simple topology for conformance deterioration

Assume that L_1 and L_2 are two initially conformant packets of respective size in bits, that the TBs are full with b tokens at both the ingress and egress nodes when L_1 arrives (in full) and that $TBC_{n,\,t}$ is the TB capacity at node n at time t, expressed in tokens. If L_1 is sent from an ingress node at time t_1 and subsequently at the same node packet L_2 arrives at time t_2, the latter is conformant with $TBC_{ing,\,t2} = L_2$. This packet departs the ingress at time t_2 and arrives at the egress at time t_3. Thus, the transmission time for packet L_2 is $\Delta_2 = t_3 - t_2 = L_2/C$.

At the egress node, L_1 arrives at time t_2 and leaves $TBC_{egr,t2} = b - L_1$ tokens. Since the transmission period of L_2 is Δ_2, as many as $r \bullet \Delta_2 = r \bullet L_2/C$ tokens are accumulated. By the time t_3 when L_2 arrives at the egress node, there are $TBC_{egr,t3} = b - L_1 + r \bullet L_2/C$ tokens available. Therefore, if $TBC_{egr,t3} < L_2$ then L_2 becomes non-conformant as shown in Fig. 2.

Since the latter is quite possible, we reach the conclusion that conformant packets can indeed become non-conformant. The direct consequence of this is that non-conformant packets will either have to be dropped or reshaped. The first action would lead to retransmissions for reliable traffic, whereas the second would require buffering of packets and transmission at some later time when the TB will have enough tokens. The latter course of action leads to some degree of congestion, delay and, therefore, jitter that is introduced for RTP packets. Should this prove to be large enough, the respective RTP packets will have to be dropped by the time they arrive at their destination.

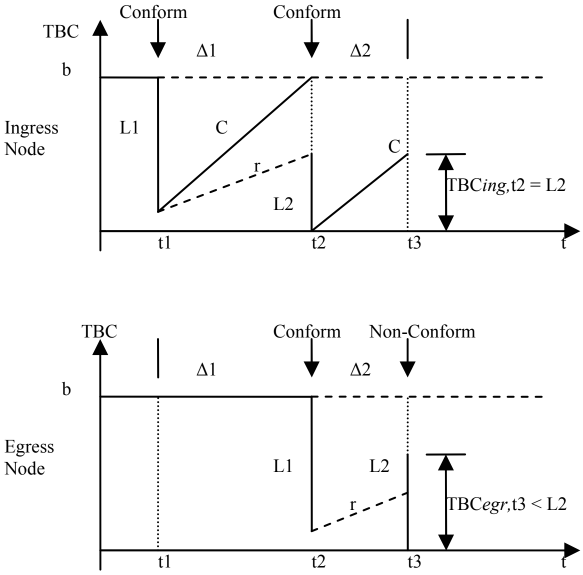

Fig.2 How conformance deterioration takes place

B. *Estimating non-Conformant Traffic*

As pointed out earlier, it is necessary to estimate the amount of non-conformant traffic in order to establish whether adding a filter at each appropriate network node for RTP packets is justified. It has already been shown [5] that the probability of conformance deterioration η is:

$$\eta = \frac{\alpha - \alpha^2}{2 + \alpha - \alpha^2} \qquad (1)$$

where $a = \frac{r}{C}$. Thus, a packet is conformant at both nodes only if the link speed is equal to the token rate r, or infinity. The worst possible case for conformance deterioration takes place when we have: $d\eta/d\alpha = 0$, from where we get that $\alpha = ½$, in which case $\eta = 11.1\%$. Moreover, conformance deterioration becomes worse if more nodes of the same type are included in series after R3 in Fig.1, without any traffic shaping taking place in between. Increasing the size of the TB reduces non-conformant traffic at the expense of longer delays, assuming that there is enough buffer space.

C. *A Practical Solution*

Having established a good estimate on the percentage of non-conformant packets after passing through a Diffsev aware network, we now turn our attention to a possible practical solution along the lines described earlier.

Given that in Diffserv networks internal nodes typically check only the DS field for their PHB (Per Hop Behavior), the only points at which any possible solution can be applied are the edge routers [9], [10], if the maximum degree of compatibility is required. More specifically, these are the ingress and egress routers of such a Diffserv region, where an appropriate MF classifier must be employed. The question now is what such a classifier should detect and classify in terms of real-time multimedia traffic.

Typically, real-time multimedia traffic uses UDP and the tendency is to use RTP on top of UDP for this purpose. UDP is unreliable, hence non-responsive to congestion and other problems that may arise.

In [1] it is proposed that the routers concerned look at the history of out packet drops, in which non-responsive connections (e.g., those using UDP) from such sources are overrepresented. In this way non-responsive connections are isolated from the rest and some fair-queuing mechanism is employed in order to “punish” them. Nevertheless, such a scheme would be quite expensive; furthermore, it must be pointed out that not all UDP traffic conveys real-time multimedia traffic.

Instead, we propose the adoption of a filter that checks for the following fields:

- Whether it is an RTP packet
- The source IP address
- The RTP timestamp

The filter employed has to be combined with an Average Rate Meter [9] which checks whether successive RTP packets i, i+1, from the same source experience a delay $\delta_{i,i+1} \geq 150$ msec; should this happen, the respective RTP packet is dropped, since it will be too late by the time it reaches its destination and would be discarded there anyway.

Nevertheless this scheme cannot be applied directly, since it is possible that some of the RTP packets may have been lost or dropped before reaching an edge router. It is also possible, that RTP packets are not generated at constant rate (e.g., periods of silence in voice applications). For this reason, the RTP timestamp ts_i of successive packets should be used in order to compare the timestamp difference to the actual delay; obviously an RTP packet i+1 should be dropped only if both conditions below are true:

- $ts_{i+1} - ts_i < 150$ msec (2)
- $\delta_{i,i+1} \geq 150$ msec (3)

Although RTP is not responsive, there is some provision for feedback on behalf of the receivers through the use of

RTCP reports. Therefore, it is expected that the sender(s) in such cases will eventually detect a higher rate of undelivered RTP packets, thus lowering their transmission rate if possible. This occurs even in the special case of a class with low data rate but many packet bursts, since RTCP RR packets report back using the *loss fraction* field whose value is calculated per interval.

There is no need to use other fields, such as destination address, since jitter is the only determining factor for each flow coming from the same source. Furthermore, when calculating the difference of timestamps of successive packets, only the last 16 bits of timestamps need to be used in most cases [8].

D. Performance

Let us assume that, on average, the total RTP traffic has to cross k successive DS regions. In the case of light traffic, there will be no congestion due to non-conformance or any other reason and all packets will reach their destination.

At some time though, congestion will take place, either because of TCP traffic, which is aggressive, or due to other factors, inclusive of conformance deterioration as discussed above. In this case, there will be a conformance deterioration η_i introduced by each successive DS region, where $1 \leq i \leq k$. Therefore, the total amount of conformance deterioration is:

$$\eta_{total} = \sum_{1}^{k} \eta_i \qquad (4)$$

By the time the RTP traffic reaches its destinations, the maximum conformance deterioration will be $k \bullet \eta_{max}$. If p is the percentage of the whole traffic which is of the RTP type, then in the worst case a total of $p \bullet k \bullet \eta_{max}$ RTP traffic will reach the respective destinations too late and will be dropped.

If, instead, we apply the proposed solution, and a percentage q_i of RTP traffic is dropped at every egress node due to its exceeding delay, we get that the RTP traffic proceeding to the next DS region now becomes $(1 - q_i) \bullet p$. Hence, at the egress of the *k*th DS region we have: $\sum_{1}^{k} (1 - q_i) p$ and obviously, $q_i \leq \eta_i$.

A direct consequence of the above is that the remaining traffic is up to $\sum_{1}^{k} [p(1 - q_i) + (1 - p)]$ and that at each stage up to $p \bullet q_i$ bandwidth is now freed, reducing the probability of conformance deterioration.

IV. Conclusion and Future Work

We have proposed a simple, yet practical solution to the problem of conformance deterioration in Diffserv networks when faced with real-time multimedia traffic. The solution proposed is compatible with the existing framework and no additional modifications are needed for its implementation and deployment. By applying this scheme, up to $p \bullet q_i$ of the total bandwidth is freed at each successive DS region, leading to better network resource utilization.

Further work is needed for a more thorough analysis in terms of the achievable gains since the total conformance deterioration was calculated as an aggregated sum in all regions, as well as to determine the actual cost of implementing this scheme at egress nodes.

Acknowledgement

We would like to thank the anonymous referees for their constructive comments.

References

[1] David Clark, Wenjja Fang, Explicit Allocation of Best-Effort Packet Delivery Service. IEEE/ACM Transactions on Networking, Vol. 6, No. 4, August 1998, pp. 362-373.

[2] J. Kurose, K. Ross, Computer Networking: A Top-Down Approach Featuring the Internet, Third Edition. Pearson Education, Inc., 2004.

[3] Assured Forwarding PHB Group. RFC 2597.

[4] Roch Guérin, Vicent Pla, Aggregation and Performance in Differentiated Service Networks: A Case Study, ACM Communication Review, vol. 31, no. 1, 2001, pp. 21-32.

[5] Frank Yong Li, Norvald Stol, A Study on Traffic Shaping, Policing and Conformance Deterioration for QoS Contracted Networks. Proceedings of the IEEE Global Telecommunications Conf. (GLOBECOM), Taipei, Taiwan, 2002, pp. 17-21.

[6] An Architecture for Differentiated Service. RFC 2475.

[7] Roland Bless, IP Multicast in Differentiated Services Networks, IETF, Internet Draft, draft-bless-diffserv-multicast-07.txt, August 2003.

[8] Colin Perkins, RTP: Audio and Video for the Internet, Pearson Education Inc., 2003.

[9] An Informal Management Model for Diffserv Routers, RFC 3290.

[10] Zheng Wang, Internet QoS, Morgan Kaufmann Publishers, 2001.

[11] An Expedited Forwarding PHB. RFC 2598.

[12] Nabil Seddigh, Biswajit Nandy, Peter Pieda. Study of TCP and UDP Interaction for the AF PHB. IETF, Internet Draft, draft-nsbnpp-diffserv-tcpudpaf-00.txt, June 1999.

[13] Overview and Principles of Internet Traffic Engineering. RFC 3272.

[14] A Single Rate Three Color Marker. RFC 2697.

[15] A Two Rate Three Color Marker. RFC 2698.

[16] IP Multicast in Differentiated Services (DS) Networks. RFC 3754.

[17] Ashley Stephenson. QoS: The IP Solution, Delivering End-to-End Quality of Service for the Future of IP. White Paper, Lucent Technologies, 1999.