End-to-end Proportional Loss Differentiation

Alok Kumar Jasleen Kaur Harrick M. Vin

Technical Report TR-01-33
Department of Computer Sciences
University of Texas at Austin
TAY 2.124, Austin, TX 78712-1188, USA

{alok,jks,vin}@cs.utexas.edu

Feb 26, 2001

Abstract

Service differentiation is at the core of designing next-generation Internet. In this paper, we present a buffer management framework for achieving end-to-end proportional loss differentiation in networks. There are two main facets of our buffer management framework. First, it decouples the decisions of *when* to drop a packet from *which* packet to drop. This allows the framework to utilize existing single-class buffer management techniques—such as RED—to determine *when* to drop a packet; in fact, when instantiated with RED, the framework extends the primary advantages of single-class RED—namely, early notification of congestion and maintenance of average buffer occupancy at low, configurable levels—to a multi-class workload. Second, at each router, the framework governs the selection of *which* packet to drop based on the number of packets of a flow transmitted by its source, rather than the number of packets that arrive at a router. The framework achieves this by encoding information about the losses observed by a flow at a router in packet headers. This allows the framework to provide end-to-end proportional loss differentiation, unlike most existing schemes that provide loss differentiation only on a per-hop basis. We evaluate the efficacy of this approach under various network settings. We describe an implementation of our framework and discuss its complexity.

1 Introduction

The Internet has traditionally supported the *best-effort* service model in which the network offers no assurance about when, or even if, packets will be delivered. With the commercialization of the Internet and the deployment of inelastic continuous media applications, however, the best-effort service model is increasingly becoming inadequate. Hence, development and deployment of network mechanisms that provide different levels of service to different application classes is at the core of designing next-generation Internet.

Over the past few years, several network architectures for providing service differentiation have been proposed [14, 12]. The Differentiated Services (DiffServ) framework is one such architecture [12]. This architecture achieves scalability by implementing complex classification and conditioning functions only at network boundary routers (which process lower volumes of traffic and lesser numbers of flows), and providing service differentiation inside the network for flow aggregates rather than on a per-flow basis [12]. To provide service differentiation to traffic aggregates, the DiffServ architecture defines a small set of behavior (or flow) aggregates (also referred to as Per Hop Behaviors—PHB). Recently, several PHBs—such as the Expedited Forwarding (EF) and the Assured Forwarding (AF) PHB—and several end-to-end services—such as the Virtual Leased Line service [10, 12], Assured service [5], and the Olympic service [8]—have been defined.

Consider, now, the Assured service [5] and the Olympic service [8] definitions. These services are based on the Assured Forwarding (AF) PHB [8]; further, they require the network to provide differentiation in the *loss rates* experienced by packets of flows subscribing to different levels of service. For instance, the Assured service definition partitions the packets of flows requesting the Assured service into two classes—IN and OUT, and require the network to ensure that packets belonging to the IN class always experience lower loss rate than packets belonging to the OUT class. The Olympic service, on the other hand, defines three classes—gold, silver, and bronze. Within each class, three levels of drop precedence—low, medium, and high—may be defined such that packets marked with the low drop precedence experience lower loss rates than those marked with medium, which in turn experience lower loss rates than those marked with high. Observe that both of these service definitions require the network to provide only a *qualitative* differentiation in the loss rates experienced by different classes of packets. Recently, it has been argued that service definitions that export richer semantics—such as quantitative loss differentiation—may be more desirable for network subscribers. Unfortunately, most existing buffer management mechanisms either do not provide any quantitative loss differentiation or do so only on a per-hop basis. In this paper, we propose a buffer management framework for providing end-to-end proportional loss differentiation in networks.

There are two main facets of our buffer management framework. First, it decouples the decisions of *when* to drop a packet from *which* packet to drop. This allows the framework to utilize existing single-class buffer management techniques—such as Random Early Detection (RED)—to determine *when* to drop a packet; in fact, when instantiated with RED, the framework extends the primary advantages of single-class RED—namely, early notification of congestion and maintenance of average buffer occupancy at low, configurable levels—to a multi-class workload. Second, at each router, the framework governs the selection of *which* packet to drop based on the number of packets of a flow transmitted by its source, rather than the number of packets that arrive at a router. The framework achieves this by encoding information about the losses observed by a flow at a router in packet headers (see for example, Dynamic Packet State [16]). This allows the framework to provide end-to-end proportional loss differentiation, unlike most existing schemes that provide loss differentiation only on a per-hop basis. We evaluate the efficacy of this approach under various network settings. We describe an implementation of our framework in routers based on the Intel's IXP1200 network processors. Our preliminary evaluation indicates that EPLD can be implemented in high-speed routers without any degradation in router performance.

The rest of the paper is organized as follows. In Section 2 we formulate the problem of end-to-end proportional loss differentiation and derive our design principles. EPLD is described in Section 3 and an approximation is proposed in Section 4. A comprehensive experimental evaluation is presented in Section 5. In Section 6 we present our implementation design of EPLD in the IXP1200 routers and discuss the storage and computation overheads. We discuss related work in Section 7. Section 8 summarizes our contributions.

2 Problem Formulation

Consider a network that supports N traffic classes, $J_1, J_2, \ldots, J_N$. Let $\alpha_i \neq 0$ be the *loss differentiation parameter* associated with class J_i. Let $\alpha(f)$ denote a function such that $\alpha(f) = \alpha_i$ *iff* f *belongs to class* J_i. Consider two flows, f_j and f_k that share a network path P of one or more hops. Let T be a time-interval of observation during which both the flows are *active* (i.e., both flows are transmitting packets). Let $l_j(P,T)$ and $l_k(P,T)$, respectively, denote the percentage of packets lost on path P during time-interval T, by flows f_j and f_k. Then, we say that the network provides *end-to-end proportional loss differentiation*, if for all j, k, P, and T:

$$\frac{l_j(P,T)}{l_k(P,T)} = \frac{\alpha(f_j)}{\alpha(f_k)} \tag{1}$$

To motivate the design of a buffer management mechanism that can achieve such end-to-end proportional loss differentiation, consider, first, the design of Random Early Detection (RED)—the most popular buffer management scheme for today's networks (that support a single class of service). Routers that implement RED manage the available buffer space in the router as follows. The router maintains a running estimate of the average buffer occupancy.

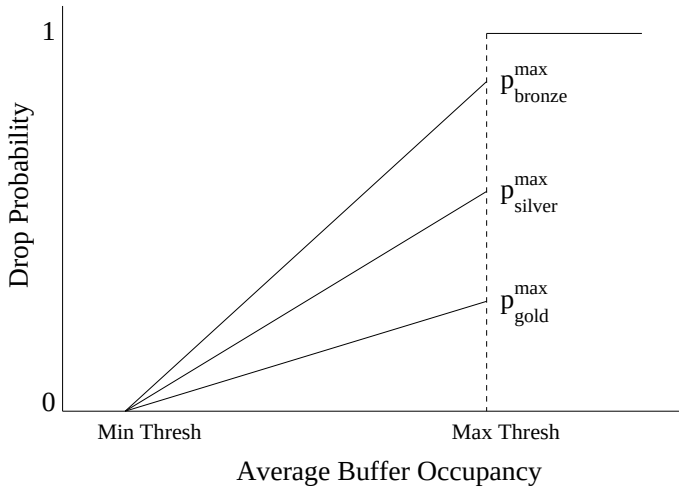

Figure 1: Proportional Loss Differentiation in WRED

Each incoming packet is dropped with a probability that increases linearly from 0 to p_{max} as the buffer occupancy increases from a minimum threshold (MinThresh) to a maximum threshold (MaxThresh). If the average buffer occupancy exceeds MaxThresh, then all incoming packets are dropped unconditionally. If, on the other hand, the average buffer occupancy is below MinThresh, none of the incoming packets are dropped. The relationship between the drop probability that a packet may observe and the average buffer occupancy is described in a *RED curve*.

There are two primary benefits of RED (over a simple tail-drop buffer management policy). First, by selecting appropriately the parameter values, RED facilitates early congestion notification to the applications. Second, RED enables a router to maintain its average buffer occupancy at low, configurable levels.

Recently, extensions of RED—referred to as multi-class RED or Weighted RED (WRED) [13]—have been proposed to extend the benefits of RED to network environments that support multiple classes of service. These extensions use different RED curves for dropping packets belonging to different service classes [13]. Figure 1 depicts one such setting with three service classes—gold, silver, and bronze. In this configuration, when the average buffer occupancy is in the range [MinThresh, MaxThresh], the probabilities for dropping packets of the gold, silver, and the bronze class are in the ratio $p_{gold}^{max} : p_{silver}^{max} : p_{bronze}^{max}$. Hence, by selecting the values of p_{gold}^{max}, p_{silver}^{max} and p_{bronze}^{max} based on the desired loss ratios, the above configuration can achieve proportional loss differentiation in each router.

Unfortunately, such extensions of RED have the following two limitations.

1. *Inconsistent congestion notification*: Ideally, loss differentiation should be provided by determining the class from which a packet should be dropped, in the event that a router needs to drop a packet. In the above scheme, however, loss differentiation is achieved by determining *when* an incoming packet is dropped; the selection of the packet to be dropped is implicit. This approach has two undesirable effects.

   - Consider two routers R_1 and R_2 with the WRED buffer management scheme (as depicted in Figure 1). Let R_1 and R_2 receive packets at the same rate. Now, if all of the traffic entering R_1 and R_2, respectively, belongs to the bronze and the gold class, then router R_1 will drop a much larger number of packets as compared to router R_2 (even though both routers are receiving packets at the same rate, and hence are experiencing the same level of congestion). Since packet losses indicate congestion to applications, the WRED configuration described in Figure 1 delivers inconsistent congestion notifications to the applications.
   - With the WRED configuration described in Figure 1, the average buffer occupancy in routers is not just a function of the RED parameters, but also the traffic composition. For instance, in the above

example, since router R_2 drops a much smaller number of gold packets, it experiences a much higher level of average buffer occupancy (as compared to router R_1, which, by virtue of dropping a larger number of bronze packets, maintains the average buffer occupancy at lower levels). Hence, WRED does not preserve a key property of RED—by selecting appropriately the RED parameters, average buffer occupancy can be *configured* to remain below certain levels [7].

These observations lead us to our first design principle.

Principle 1 *Buffer management schemes should decouple the decisions of when to drop a packet from which packet to drop* [1].

2. *Lack of proportional differentiation in multi-hop networks*: A network of WRED routers does not provide the desired loss differentiation in a multi-hop network. To demonstrate this, consider two flows f_j and f_k that share a network path P consisting of two routers. During an observation interval of length T, let flows f_j and f_k be active simultaneously. Further, let the loss rates experienced by packets of the flows at each router be proportional to the loss differentiation parameters of the classes the flows belong to. Thus, for some k_i (which depends on the level of congestion at router i), packets of flow f_j experience a loss rate—with respect to the incoming traffic at that router—of $\alpha(f_j)k_i$ at the i^{th} router on the path. Then, the ratio of packet losses incurred by the two flows in a two-hop network is given by:

$$\begin{aligned}
\frac{l_j(P,T)}{l_k(P,T)} &= \frac{\alpha(f_j) \times k_1 + (1 - \alpha(f_j) \times k_1) \times \alpha(f_j) \times k_2}{\alpha(f_k) \times k_1 + (1 - \alpha(f_k) \times k_1) \times \alpha(f_k) \times k_2} \\
&= \frac{\alpha(f_j)\ (k_1 + k_2) - k_1 k_2\ \alpha(f_j)^2}{\alpha(f_k)\ (k_1 + k_2) - k_1 k_2\ \alpha(f_k)^2} \qquad (2) \\
&\neq \frac{\alpha(f_j)}{\alpha(f_k)} \qquad (3)
\end{aligned}$$

This illustrates that proportional loss rates at individual routers fails to translate to the same ratio for end-to-end loss rates. Observe that the end-to-end loss ratio is a function of the *sum* and *product* of the loss rates observed at individual routers (Equation (2)). While the *sum* of loss rates over all routers on a shared path is in the desired ratio, the *product* is not.

Figure 2 depicts the expected deviation from the per-hop loss ratios with increase in the number of hops in a shared path. Each line in Figure 2 represents a level of congestion in the routers (which in turn translates to a certain percentage of packets dropped at the routers); further, the graph considers a setting with multiple routers along the path experience the same level of congestion. Figure 2 shows that if the level of congestion, indicated by the packet loss rate, is high, then the end-to-end loss differentiation deviates significantly from the desired ratio even when flows share path with a small number of hops. On the other hand, for networks with small packet loss rates, the deviation is significant only if flows share a path with a large number of congested routers. Thus, the deviation from the desired ratio of proportional loss differentiation becomes significant when (1) flows encounter multiple congested links on their path, and (2) the loss rates experienced at the congested links are high.

A recent study reported in [4] measures loss rates of around 10% on sample paths through the Internet. Thus the Internet today satisfies condition (2), but it is not known whether these losses occur at a single link or

[1] The need for decoupling the decision of *when* to drop a packet and *which* packet to drop was also identified in [6]. In [6], the authors motivate the need for decoupling by arguing that with WRED, the ratio of losses for observed by two different classes depends on the traffic composition. However, the WRED configuration shown in Figure 1 does not have this limitation. Even for this configuration, we have argued that decoupling is necessary for a completely different set of reasons.