Aggregation and Conformance in Differentiated Service Networks: A Case Study*

Roch A. Guérin
Dept. Elec. Eng., U. Pennsylvania
200 South 33rd Street
Philadelphia, PA 19104, USA
guerin@ee.upenn.edu

Vicent Pla†
Dept. Communications, U.P. de Valencia
Carretera Nazaret-Oliva
46730, Grau de Gandia, Spain
vpla@dcom.upv.es

ABSTRACT

The Differentiated Service (Diff-Serv) architecture [1] advocates a model based on different "granularity" at network edges and within the network. In particular, core routers are only required to act on a few *aggregates* that are meant to offer a pre-defined set of service levels. The use of aggregation raises a number of questions for end-to-end services, in particular when crossing domain boundaries where policing actions may be applied. This paper focuses on the impact of such policing actions in the context of individual and bulk services built on top of the Expedited Forwarding (EF) [7] per-hop-behavior (PHB). The findings of this investigation confirm and quantify the expected need for reshaping at network boundaries, and identify a number of somewhat unexpected behaviors. Recommendations are also made for when reshaping is not available.

1. INTRODUCTION

Support for some form of service guarantees in IP networks is becoming an important requirement, not only because of emerging multimedia applications, but also because of new usages embodied in service level agreements between users and providers. This is one of the motivations behind the Differentiated Services standardization efforts carried out by the Internet Engineering Task Force (IETF) [1]. The Differentiated Services framework relies on a *small* number of service levels, or *Per Hop Behaviors* (PHBs), that each specifies how a router should treat the corresponding packets. At least two types of treatments are being standardized: The Assured Forwarding (AF) group of PHBs [4], and the Expedited Forwarding (EF) PHB [7]. The aggregate model of Diff-Serv is highly scalable, but it also raises questions in terms of the type of services it can provide. For example, because all packets within a PHB are treated the same way, the granularity of "services" that can be offered is unclear. In this paper, we concentrate on the *potential* penalty imposed by aggregating traffic into a small number of packet treatments, in terms of the *conformance* of flows as they exit a Diff-Serv domain. Conformance is measured in relation to a policer (see [5, 6] for two possible examples), that controls the volume and timing of packet transmissions. There are several reasons for why such a measure is of interest. First, it provides a simple reference point for estimating the level of "perturbations" caused by interactions between flows aggregated in the same PHB[1]. Second, it quantifies those perturbations in terms of the "contract" violations they translate into, namely, the number of non-conformant packets they create as seen by an egress policer. Such a contractual comparison is meaningful in environments where flows cross multiple Diff-Serv domains, and where policing actions are applied at (provider) domain boundaries.

In that context, we consider two possible service configurations. The first corresponds to an end-to-end service model, where a customer has established individual contracts with all the providers on its path. In such an instance, policing at domain boundaries is performed on the set of packets associated with each customer, as providers will typically not trust the policing performed by other providers. The second configuration assumes that customers negotiate service contracts only with their local provider, which is then responsible for securing the necessary (peering) agreements with other providers it connects to. In this case, policing at boundaries between provider domains will be based on aggregates corresponding to the bulk contracts negotiated between providers.

In the paper, we investigate the above issues in the context of a service based on the EF PHB. We assume that user EF traffic is *shaped* on ingress to conform to a single token bucket filter that controls long term rate and burst size[2], and we study the extent to which it becomes non-conformant after crossing a Diff-Serv domain. Egress non-conformance is evaluated using another token bucket to determine the number of non-conformant packets, as well as the distribution of the amount of time by which they are non-conformant. This is aimed at assessing the impact of aggregation and the need for egress reshaping. Reshaping has been mentioned as a possible requirement for services based on the EF PHB [7, 10], but where and the extent to which it is needed is still unclear. In particular, because of the need to compute conformance times and

*This work was supported in part by a grant from Nortel Networks and by NSF grants ANI99-06855 and ANI99-02943.

†Part of this work was done while visiting the University of Pennsylvania. This author has been supported in part by the Spanish Ministry of *Educación y Cultura* under project TIC98-0495-C02-02

[1] In this paper, we concentrate on interactions *within* a PHB, and ignore those caused by interactions *across* PHBs.

[2] In conformance with the "spirit" of [10], the burst size is limited to one or two maximum size packets.

hold packets until they become conformant, reshaping adds complexity, especially on high speed adapters. Furthermore, the non-work-conserving nature of shapers also requires additional buffers and contributes to higher delays. Hence, its introduction comes at a cost that must be weighed against the benefits it provides. One contribution of this paper is to quantify the expected importance of reshaping, and when it is not available, identify parameters and alternatives that can be used to mitigate the impact of the non-conformance induced by traffic aggregation. The investigation is carried out by simulation to allow a wide range of scenarios with various user traffic, policer parameters, interfering traffic patterns, and network and service configurations.

1.1 Previous Works

The motivation for this work is to gain a better understanding of the impact of aggregation on conformance checks that may be performed at network boundaries. Several previous works have looked at similar issues, and their answers provided additional incentives for exploring this topic.

A first such work is [2], which explored through simulations the effect of *bunching* on CBR [15] streams as they merge with other such streams and travel through multiple network nodes[3]. The paper assumed fixed size packets (ATM cells), and focused on estimating the size of the egress play-out buffers required to recreate the constant rate of a CBR stream. Both the environment and goals of this paper are somewhat different, as we allow greater variability (within the conformance bounds of the policer) in the user traffic (packet sizes, burst sizes, rates, etc.) and are primarily interested in assessing how the bunching introduced by the network affects egress conformance. Nevertheless, several results in [2] are relevant to our study, in particular the fact that the bunching occurring in the network was found not to significantly affect end-to-end delays. This suggests that it should be possible to remove non-conformance on egress through the use of small reshaping buffers. This is an aspect we investigate further.

Another more recent paper that addressed a similar topic is [14], which extended [2] by allowing different packet sizes across flows. The motivation for this extension was, as in our case, the introduction of services based on the EF PHB in IP networks. As in [2], the study is aimed at constant rate traffic and targets a similar set of measures, namely, end-to-end delay and reshaping buffer sizing. However, it also considers additional measures of the level of distortion introduced by aggregation, and evaluates the distribution of inter-packet separation at the network egress. Such a measure is closer to the conformance measure we are interested in, although there can be substantial differences between the two. For example, smaller variations in inter-packet spacing need not translate into fewer non-conformant packets on egress. There are also differences between [14] and this paper in terms of the scenarios being considered, e.g., extensions that consider variable rate streams and packet variability within streams. Those differences together with our focus on conformance lead to different conclusions in several cases. For example, while both [14] and this paper note that the bandwidth of an individual flow affects the variation of inter-packet spacing it experiences, conclusions differ when it comes to the impact of network utilization.

A last work that partly influenced and motivated the study undertaken in this paper is [8]. This paper establishes a result, which despite some limiting assumptions, i.e., feed-forward networks, saturated queues, and fixed size packets, has interesting implications for the support of fixed rate traffic in packet networks. It basically states that when input traffic is *smoother than Poisson* and it only interacts with similar traffic, it retains that smoothness when crossing a network[4]. In other words, the bunching of traffic that the network can potentially introduce will not increase the burstiness[5] of the traffic above and beyond that of a Poisson process. However, the use of ingress and egress policers together with variability in packet sizes and network loads go beyond the model of [8], and were one of the motivations for this study.

The rest of this paper is structured as follows. Section 2 describes the simulation environment and its parameters together with the various performance measures being investigated. Section 3 is devoted to a set of basic experiments, that rely on homogeneous traffic sources. Section 4 carries out a similar investigation, but allows heterogeneity in traffic sources and assesses how such variations affect the findings of Section 3. Finally, Section 5 summarizes the findings of the paper.

2. MODEL AND METHODOLOGY

In order to investigate the issues described in the previous section, a simulation environment was developed using the NS-2 simulator [11], to which a number of modifications and extensions were applied in order to accommodate the requirements of this study. In all scenarios, 90% confidence levels were targeted, and the corresponding confidence intervals are shown in the figures even if they are occasionally difficult to discern due to their small size. Figure 1 shows the generic topology and setup used in most of the experiments we report on in this paper. The topology is a linear multi-hop topology which has been widely used in previous works, e.g., see [14] and references therein. In this topology, a variable number of "tagged" EF streams enter the network at the ingress (first) node R_1, and traverse all the other nodes until they reach the egress node R_n. The tagged streams are those whose performance is monitored at both egress and intermediate nodes. In order to emulate interactions with cross-traffic, external traffic is injected at every node on the path. In our investigation, this cross-traffic consists only of EF traffic as our focus is on the impact of aggregation, i.e., interactions between like streams. We assume that the impact of other traffic classes is minimal, e.g., because of the use of priority queues (EF traffic is assigned to the higher priority) or a scheduling mechanism such as weighted fair queueing that isolates traffic classes.

Note that while the linear topology of Figure 1 is rather simple, it can be configured to allow a wide range of interfering traffic patterns. In the experiments we report on in the paper, cross-traffic entering at a given node is restricted to interfering with the tagged streams for only *one* hop, i.e., it leaves the network at the next hop. This is meant to limit the number of parameters that need to be specified and simplify the discussion of the results. Several other configurations were also investigated, which varied the number of hops over which cross-traffic interfered with tagged streams and the number of interfering streams. No substantial differences were observed between those scenarios and the single hop interference one, which was therefore chosen because of its simplicity. In the context of this scenario, there are nevertheless a number of other parameters that can be varied. These include the number of hops traveled by the tagged streams, the number of tagged and cross streams, and the total link load. Most experiments involve streams

[3] A similarly motivated study but focusing on interactions with a wider range of cross traffic can be found in [3].

[4] It should be noted that this result was first conjectured in [12], and that there had been a number of previous works, e.g., [9, 13] aimed at quantifying the perturbations experienced by a CBR stream as it crosses a network.

[5] The results of [8] are for a specific definition of traffic ordering, and the interested reader is referred to the paper for details.

(tagged and cross-traffic) with similar characteristics, i.e., a homogeneous environment, so that link load and the relative weight of tagged traffic and cross-traffic is simply a function of the number of streams in each category. However, the impact of heterogeneous stream characteristics, i.e., different bandwidth and/or packet sizes, is also investigated in a small number experiments.

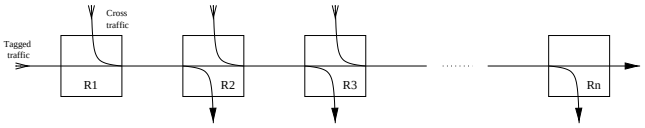

Figure 1: Generic scenario.

Traffic sources are based on a packet generator driven by a renewal packet arrival process, and connected to a traffic conditioner in the form of a token bucket filter with parameters (R, b) (see figure 2). The token bucket is configured to act as a shaper, and ensures that all packets forwarded *into* the network are conformant with the specified traffic contract. A variety of traffic sources were used as inputs to the token bucket, e.g., sources with inter-packet times distributed according to Pareto or exponential distributions, or based simply on a jittered constant time. Because of the traffic conditioning effect of the token bucket, the choice of a particular distribution was found to have only a minor impact on performance, at least in terms of its impact on egress conformance. As a result, and for the sake of simplicity, the results reported in this paper rely only on a Poisson input. Clearly, this is not meant to be representative of "real" traffic patterns in IP networks, but its single parameter characterization and ease of generation make a Poisson input a convenient tool to systematically explore the range of possible traffic mixes and intensities. Again, the justification for such a choice is that because of the normative effect of the shaper, only minor differences were observed in terms of egress conformance between Poisson and, say, Pareto, inputs.

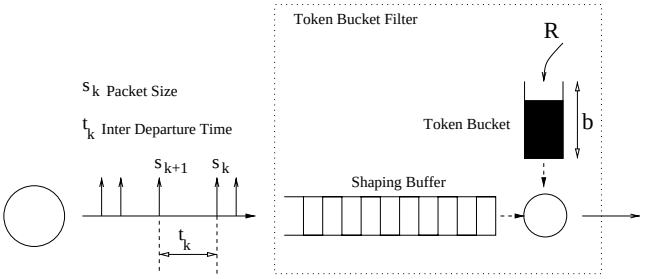

Figure 2: Source model.

In most scenarios, the bucket size (b) was set to twice the average packet size, and the token rate (R) was usually set at a value 25% higher than the average input rate. This margin was chosen to avoid consistently reshaping the traffic into a constant rate stream, that would essentially eliminate differences between input streams. Furthermore, it is to be expected that users will specify traffic contract that exceed by some amount, e.g., 25%, their average traffic volume, in order to accommodate most traffic patterns without excessive reshaping delays. Finally, a last parameter that we vary in our experiments is the link load, as it is expected to influence the level of interactions between tagged streams and cross-traffic. In particular, the parameter ρ^* is used to indicate the ratio between the link bandwidth and the sum of the token rates of the streams it carries. Note that varying ρ^* allows us to capture different implementations of the EF PHB. For example, a setting of $\rho^* = 1$ would "emulate" a network where the EF PHB is supported using some form for Fair Queueing scheduler, i.e., the EF traffic only sees the bandwidth it has been allocated. Alternatively, a setting of, say, $\rho^* = 0.5$ could correspond to a network using a priority scheme to implement the EF PHB. It should also be noted, that because the actual packet generation rate is only 80% of the token rate, the actual link utilization ρ is only $0.8 \cdot \rho^*$.

Upon reaching the egress router R_n, packets from tagged streams are again passed through a token bucket filter (TBF) and checked for conformance. As mentioned before, the purpose of this egress check is to assess the extent to which interactions within the network transform a conformant stream into a non-conformant one. This check is performed using a range of TBF configurations aimed at evaluating different approaches to handling this non-conformance. In particular, both the token rate and the amount of buffering at the egress TBF are varied, e.g., egress buffering can be set anywhere between zero (strict policing) and infinity (full reshaping). The level of egress non-conformance is measured using various statistics such as the number of non-conformant packets, and the distribution of the amount of time by which those packets are non-conformant. Both measures are useful not only to quantify the impact of the network on a stream's conformance, but also to gain some understanding into how conformance can be reestablished. In configurations that involve a reshaping buffer, the distribution of buffer occupancy and reshaping delay are also monitored as both are useful indicators of the cost and efficacy of reshaping.

Measurements are carried for two settings. In the first setting, each individual stream is associated with its own egress TBF. This corresponds to an environment where *individual* (per user) service contracts are extending across multiple providers. In the second setting, multiple (all) tagged streams are mapped onto a common egress TBF. This is representative of an environment where user level contracts are mapped onto *provider level* contracts when crossing provider boundaries. In other words, all the EF traffic leaving provider A and entering provider B is mapped onto the aggregate EF contract passed between providers A and B.

Before discussing the paper's findings, we briefly pause to comment on whether the experimental setting of the paper is sufficiently "realistic" to allow conclusions that are applicable to *real* networks. This is a question that is always difficult to answer in the context of simulation based studies, and this paper is no exception. As mentioned earlier, many more experiments were conducted than are reported in the paper, and one of the reasons that led us to omitting them was the relative lack of sensitivity of the results across variations in topologies, interference patterns, and incoming traffic characteristics. We feel that this provides a reasonable level of confidence that the paper's conclusions and findings should hold in more general and different settings.

3. HOMOGENEOUS SOURCES

In this section, we consider scenarios that consist only of homogeneous sources, i.e., sources with identical rates and packet sizes. This obviously wont be the case in practice, but focusing on homogeneous sources helps isolate the effect of individual parameters. Heterogeneous sources are considered in Section 4.

3.1 Basic configuration

The basic configuration used in these initial investigations consists of a single tagged stream ($n_{ts} = 1$) with a *fixed* packet size (500 bytes). As mentioned earlier, the traffic source associated with the tagged stream feeds packets to its ingress TBF according to a *Poisson* process. The performance of this basic configuration is ex-

plored by varying the number of hops (N), the number of cross streams (n_{cs}), the relative network load (ρ^*), the egress TBF token generation rate (R_e), and the size of the egress reshaping buffer (B_e).

The results of this first set of experiments are reported in Figures 3 to 8, which plot over a range of scenarios the probability P_d that a packet is found non-conformant at the egress TBF. Except for Figure 7 that allows shaping, the egress TBF is configured to drop all non-conformant packets. Note that the maximum load value of $\rho^* = 1.2$ shown in most figures corresponds to an actual average link load of only 1. This is because the token generation rate is 25% higher than the average source data rate.

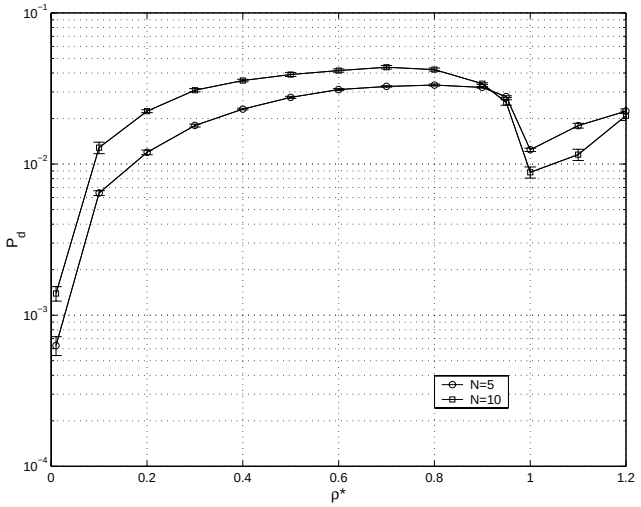

Figure 3: Drop probability P_d for $n_{cs} = 1$.

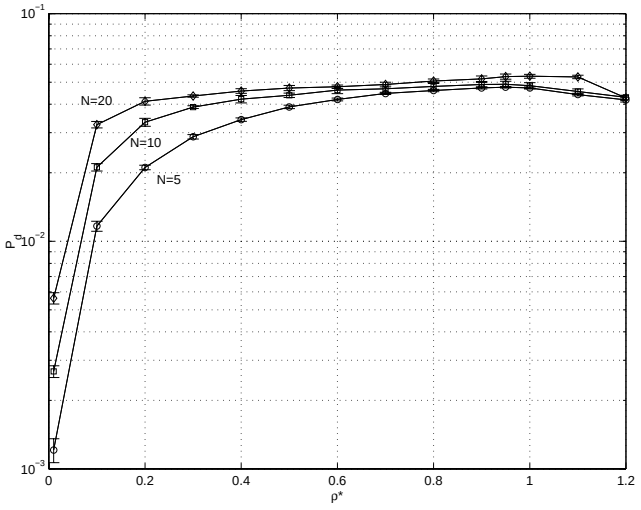

Figure 4: Drop probability P_d for $n_{cs} = 20$.

A first general conclusion that can be drawn from the the figures is that simply throwing network bandwidth at the problem does not appear to be very effective at ensuring egress conformance. In particular, we observe in Figures 3 and 4 that as the network load decreases, the probability of non-conformance (P_d) hardly changes until we reach very low network loads. One of the reasons for this maybe counter-intuitive behavior, is that while lowering network

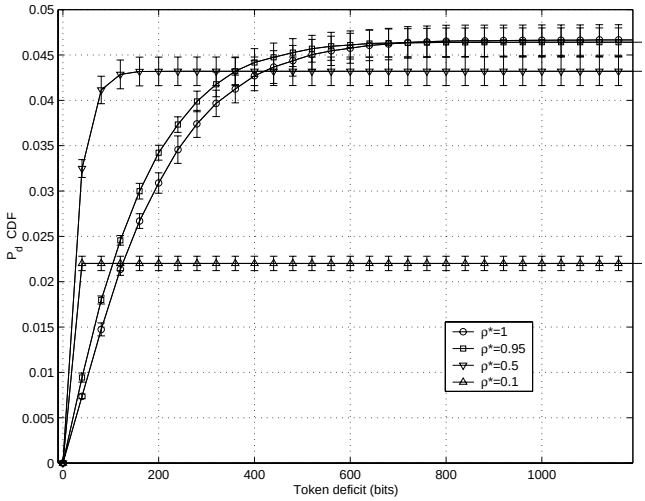

Figure 5: Token deficit cumulative distribution on the arrival of a non-conformant packet. $N = 10$, $n_{cs} = 100$.

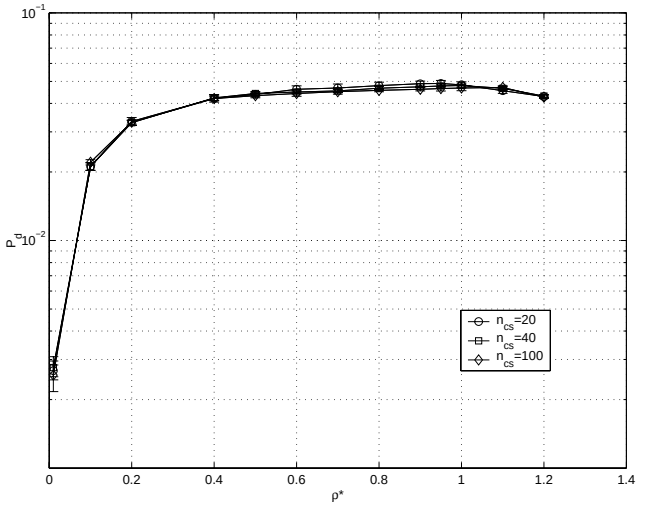

Figure 6: Drop probability P_d for $N = 10$.

load reduces the *magnitude* of the perturbations experienced by individual packets, it has little or no effect on the *number* of packets that experience some level of perturbation. On second thoughts, this is not a completely unexpected result, as illustrated in Figure 5, which plots the distribution of the amount of time by which packets are non-conformant for different network loads. From the figure, we see that there is indeed a shift in distribution as network load decreases, but it does not translate into a transfer of probability mass to zero (probability of being conformant) until very low link loads, i.e., $\rho^* = 0.1$.

Figure 3 also illustrates a somewhat unintuitive behavior, namely, that *lowering* network load can *increase* the probability of egress non-conformance. The specific scenario of Figure 3 where this phenomenon is observed, is somewhat "extreme" and consists of two streams, one tagged and one cross-traffic, that share network links. In such a setting, when the network load is in the neighborhood of $\rho^* = 1$, a decrease in network bandwidth yields an increase in the number of non-conformant packets. This rather artificial behavior is caused by the combination of the specific rela-

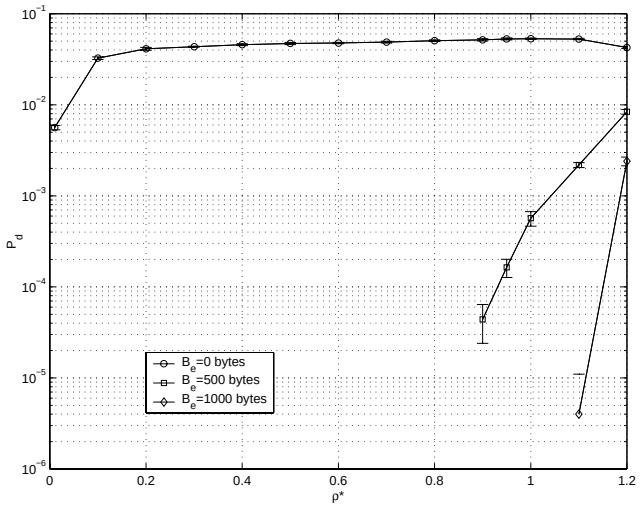

Figure 7: Impact of reshaping buffers. $N = 20$**,** $n_{cs} = 20$

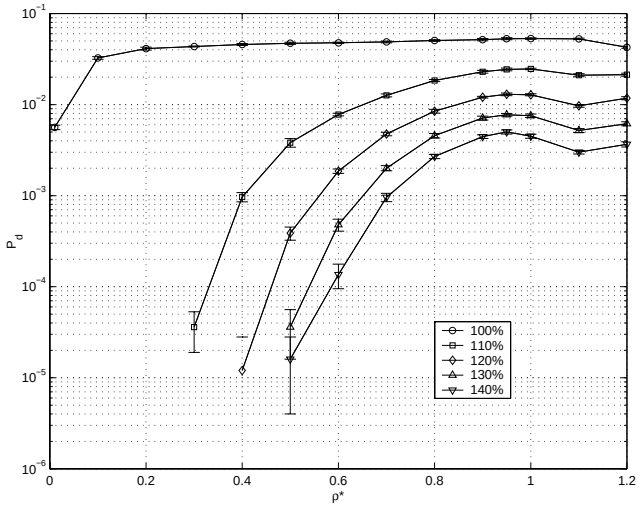

Figure 8: Impact of egress token rate. Percentages refer to the ingress conditioner token rate. $N = 20$**,** $n_{cs} = 20$

tion that exists between the link speed and the token rate, and the high likelihood of interleaved packet transmissions from the tagged and cross-traffic streams. Specifically, it is likely that at high loads queues will form that consist of an alternating sequence of packets from the two streams. In such a scenario, when we have exactly $\rho^* = 1$, packets of the tagged stream are properly spaced apart so that they arrive at the egress at the exact same rate as tokens are generated. However, when the network load decreases slightly, i.e., the link speed increases, the likelihood of such interleaved patterns hardly changes, but because packets are delivered faster, the likelihood that some of them reach the egress policer too early increases. Further confirmation of this interpretation was obtained by simulating a number of additional scenarios, which showed that when the token rate of the egress TBF is increased, the point where the non-conformance probability P_d stops decreasing as the load ρ^* increases shifts towards the left. In other words, the network load and, therefore, link speed that corresponds to the ideal interleaving point, tracks the increase in token rate. Similarly, as illustrated in Figure 4, increasing the number of streams (cross-traffic or tagged) sharing the link, all but eliminates this behavior. This is because realizing the desired interleaving pattern becomes highly unlikely as the number of streams increases.

Another conclusion that can be drawn from Figures 4 and 6, is that neither the number of cross-traffic streams nor the number of network hops traversed by the tagged stream appear to have a major influence on egress conformance. The only noticeable impact is that, as illustrated in Figure 4, increasing the number of hops traversed by the tagged stream, increases slightly the likelihood of non-conformance at low loads. This is because interferences between streams are then rare, and crossing more hop proportionally increases the odds of such events. In contrast, at higher loads, interferences at any given hop are relatively common and, therefore, less sensitive to potential increases due to larger hop counts.

A key, if not unexpected finding, is that the introduction of a small reshaping buffer drastically reduces the probability of non-conformance (Figure 7). This was found consistently across scenarios, and especially for "typical" scenarios with relatively low link loads (use of a priority scheduler) and large numbers of aggregated EF streams. This is because, when link loads are low and/or the number of aggregated streams is high, the *time scale* of the perturbations induced by network interferences is relatively small. Hence, they can be easily removed through a small amount of buffering. In particular, we see that in normal conditions, i.e., a token rate that is 25 % higher than the actual traffic intensity, a buffer size of about 2 packets is sufficient to essentially eliminate egress non-conformance. In more stressful scenarios, i.e., when users saturate their ingress TBF, slightly larger buffers are required, but even then the buffer sizes remain small. We explores this issue further in Section 3.3.

Another approach that we explored as a means to absorb perturbations introduced by network interferences, was to increase the egress token rate. Clearly, a higher egress token rates replenishes the token bucket faster and can, therefore, tolerate a higher level of non-conformance. The main question is the efficacy of such an approach, i.e., by how much to increase the egress token rate to absorb network induced perturbations. From Figure 8, we see that increasing egress token rate is much less effective than buffering. For example, lowering the probability of non-conformance by one order of magnitude requires increasing the egress token rate by at least 40%, while a reduction of several orders of magnitude can be achieved using only a couple of reshaping buffers.

3.2 Variability In Packet Sizes

In this section, we investigate the impact of variable size packets. Variations in packet sizes can affect egress conformance in a number of ways. Variations in the sizes of packets from cross-traffic streams can translate into greater external network interferences, and therefore contribute to a higher level of egress non-conformance. In addition, because packets consume a number of tokens proportional to their size, the presence of variable size packets introduces internal variation that can lead to behaviors different from those seen with only fixed size packets. As a matter of fact, variable packet sizes alone can introduce egress non-conformance even without *any* network induced perturbations. This is best understood through a simple example that illustrates how the presence of packet of different sizes can cause egress non-conformance.

Consider a scenario where two back-to-back packets of sizes S_1 and S_2, respectively, arrive simultaneously at the ingress TBF. Assume further that the ingress TBF has been configured with a rate of r and a token bucket size of b, and that it is connected to a link of speed C, which is in turn directly connected to an egress TBF identical to the ingress TBF. In other words, the

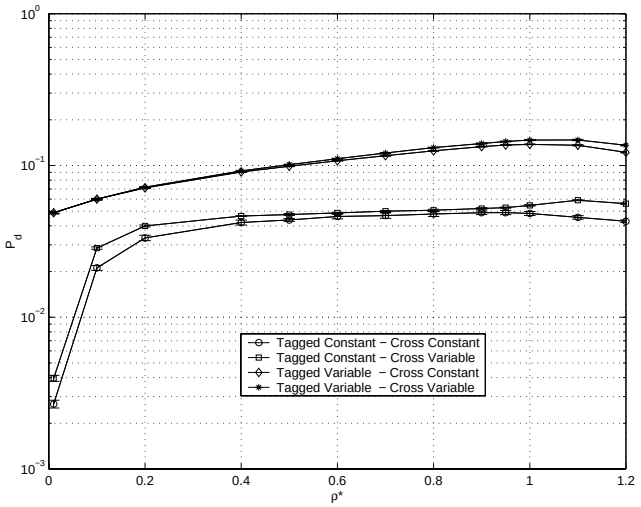

Figure 9: Impact of packet size variability.

network consists of a single link of speed C connecting the ingress and egress TBF's and dedicated to the traffic exiting the ingress TBF, i.e., there is no interfering traffic. Without loss of generality, assume that upon its arrival, the first packet finds enough tokens in the bucket. As a result, the first packet immediately exits the TBF and starts its transmission on the link at time $t = 0$. The first packet leaves behind a total of $b_1 \geq 0$ tokens, and this value together with the token rate of r determines when the second packet of size S_2 will exit the TBF. In general, the spacing Δ_0 between the times at which the second and first packets exit the ingress TBF is given by:

$$\Delta_0 = \begin{cases} 0, & \text{if } b_1 \geq S_2 \\ (S_2 - b_1)/r, & \text{if } b_1 < S_2 \end{cases} \quad (1)$$

Once it exists the ingress TBF, the second packet is then enqueued for transmission on the link. Its transmission will begin either immediately, i.e., at time $t = \Delta_0$, or once the transmission of the first packet completes if it is still in progress at time $t = \Delta_0$. We are interested in the spacing Δ_1 between the end of transmissions of the first and second packets on the link of speed C, as it corresponds to the amount of time separating their arrivals at the egress TBF. This in turn, determines the egress conformance[6] of the second packet.

Given the absence of interfering streams, the egress spacing Δ_1 between the first and second packets is easily found to be given by:

$$\Delta_1 = \begin{cases} S_2/C, & \text{if } (S_2 - b_1)/r < S_1/C \\ \frac{(S_2 - b_1)}{r} - \frac{S_1 - S_2}{C}, & \text{if } (S_2 - b_1)/r \geq S_1/C \end{cases} \quad (2)$$

From Equation 2, we identify a number of cases where the second packet will be deemed non-conformant on egress. In particular, the second packet will be non-conformant whenever $r\Delta_1 < S_2 - b_1$, since b_1 is the number of tokens left behind by the first packet. Note that this will never occur if $S_2 < b_1$, as this means that there are enough tokens left in the bucket for the second packet after the first packet.

[6]We assume that upon its arrival, the first packet finds the egress TBF in the same *state*, i.e., same number of tokens, as it found the ingress TBF when it first arrived to the network.

In the first case where $(S_2 - b_1)/r < S_1/C$, the condition $r\Delta_1 < S_2 - b_1$ is verified whenever the link speed C satisfies

$$\frac{rS_2}{(S_2 - b_1)} < C < \frac{rS_1}{(S_2 - b_1)}$$

Note that this implies $S_2 > b_1$, as expected, as well as $S_2 < S_1$, i.e., that the second packet is smaller than the first one.

In the second case where $(S_2 - b_1)/r \geq S_1/C$, the condition $r\Delta_1 < S_2 - b_1$ is always verified for $S_2 < S_1$, i.e., whenever the second packet is smaller than the second one. Note that the condition $(S_2 - b_1)/r \geq S_1/C$ again implies $S_2 > b_1$. As a result, it is possible to simplify the set of conditions under which the second packet is non-conformant on egress to be:

$$\begin{array}{rcll} S_1 & > & S_2 > b_1 & \text{and} \\ C & > & \dfrac{rS_2}{(S_2 - b_1)} & \end{array} \quad (3)$$

In other words, the second packet will be non-conformant if 1) it is smaller than the first one $(S_2 < S_1)$; 2) it needs to wait for tokens $(S_2 > b_1)$; and 3) it crosses a link of speed $C > \frac{rS_2}{(S_2 - b_1)}$. Note that this latter condition means that in some cases, increasing the speed of network links, i.e., lowering the load, can abruptly worsen egress non-conformance. Note also that for sufficiently high link speeds, the amount of *time* by which the second packet is non-conformant is $\frac{S_1 - S_2}{C}$, which goes down to zero as $C \to \infty$, but nonetheless remains positive, i.e., lowering the load reduces but does not eliminate non-conformance.

The above simple example has demonstrated that in the presence of variable size packets, an initially conformant stream of packets can be deemed non-conformant on egress, even without *any* network interferences. This phenomenon does not exist when packets are of fixed sizes, and points to potential difficulties if conformance rules are strictly enforced, i.e., reshaping is not available, when crossing domain boundaries. In the rest of this section, we explore this issue further, and in order to isolate the respective impact of internal and external variations in packet sizes, we consider three possible combinations: 1. Cross streams with variable size packets and a tagged stream with fixed size packets; 2. Cross streams with fixed size packets and a tagged stream with variable size packets; and 3. Cross and tagged streams both with variable size packets.

Across all three scenarios, packet sizes for streams with fixed size packets are taken equal to 500 byte, and in the case of streams with variable size packets, packet sizes are distributed according to a truncated exponential distribution, with a mean packet size of 500 bytes and a maximum packet size of 1000 bytes. As before, a single tagged stream is assumed, and the number of cross streams is taken equal to 20 ($n_{cs} = 20$), while a path length of 10 hops ($N = 10$) is assumed. Results of experiments for the three above scenarios are found in Figures 9 and 10, which consider token buckets of size 1000 bytes (two average size packets) and 2000 bytes (four average size packets), respectively. The main conclusion from these two figures is that the dominant effect in terms of the egress conformance of a stream is its *internal* packet variability. In other words, variations of packet sizes within a stream have a more pronounced effect than the potentially larger network perturbations caused by variable packet sizes in cross streams.

This result was confirmed across a number of other configurations, e.g., by increasing the number of cross streams and varying the number of network hops crossed by the tagged stream. The dominance of internal variability was not unexpected, as illustrated in the previous example, given the potential for non-conformance

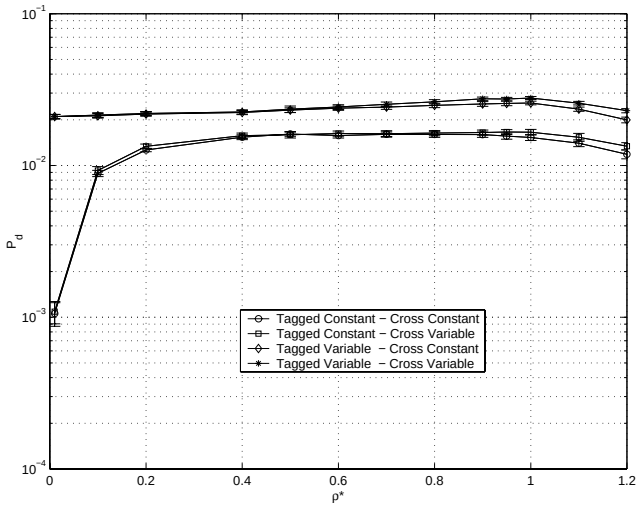

Figure 10: Impact of packet size variability in the tagged stream. ($b = 2$kbytes).

that variable size packets introduce even in the absence of any network perturbations. However, it is also caused by the fact that variability in packet sizes allows the generation of packets with a size close to the bucket size itself ($1,000$ bytes). Such packets require close to a full token bucket[7] on egress in order to be conformant. As a result, they are susceptible to network perturbations that make them (the previous packet) arrive too early (late) at the egress. Conversely, whenever large packets are successfully accepted as conformant on egress, they deplete the entire token bucket, and this can increase the likelihood that the next packet is found non-conformant.

Exploring this issue further motivated the scenario of Figure 10, which assumes a larger token bucket size of 2000 bytes or twice the maximum packet size. As can be seen from Figure 10, the gap between the two sets of curves, i.e., fixed and variable size packets in the tagged stream, has substantially narrowed, except at very low link loads (of the order of 10%). The reason for the persistence of a relatively large difference at low loads is that the resulting network perturbations are now small enough to be absorbed by the added "margin" of the larger token bucket, when packet sizes are not too large. However, for large packets, this added margin remains insufficient and they continue to experience a large level of non-conformance.

On the positive side, the relatively minor impact of external variability on conformance seems to indicate that variable size packets do not introduce significant differences, when compared to networks operating with fixed size packets, e.g., ATM networks. This is of potential interest, given the extensive investigations that have been carried out in the context of such networks.

In the next section, we revisit many of the previous issues, but in the context of a different service model. Specifically, we consider a service model where individual service contracts are mapped onto aggregate provider level contracts when crossing domain boundaries. This affects the determination of egress conformance as multiple streams are now mapped onto a common egress TBF associated with the aggregate contract.

3.3 Aggregate Contracts

[7] This assumes a token bucket size of two average size packet as used in Figure 9.

Table 1: Non-conformance (in percentage) for aggregate homogeneous contracts.

			N=2	N=5	N=10
n_{ts}=5	ρ*=0.1	n_{cs}=5	15.10	15.07	15.03
		n_{cs}=10	15.13	15.09	15.04
		n_{cs}=20	15.10	15.03	15.11
		n_{cs}=40	15.10	15.12	15.00
		n_{cs}=100	15.21	14.91	15.01
	ρ*=0.625	n_{cs}=5	13.44	11.22	9.21
		n_{cs}=10	14.10	12.59	11.08
		n_{cs}=20	14.70	14.05	13.24
		n_{cs}=40	15.00	14.91	14.48
		n_{cs}=100	15.21	14.93	14.97
	ρ*=1	n_{cs}=5	6.16	3.78	3.25
		n_{cs}=10	10.31	6.64	5.57
		n_{cs}=20	12.58	9.48	7.68
		n_{cs}=40	14.26	12.36	10.24
		n_{cs}=100	14.97	14.25	13.50
n_{ts}=20	ρ*=0.1	n_{cs}=5	18.72	18.78	18.78
		n_{cs}=10	18.71	18.86	18.81
		n_{cs}=20	18.83	18.78	18.86
		n_{cs}=40	18.90	18.81	18.81
		n_{cs}=100	18.75	18.87	18.84
	ρ*=0.625	n_{cs}=5	12.82	10.81	9.14
		n_{cs}=10	15.47	13.04	10.94
		n_{cs}=20	17.03	14.76	12.68
		n_{cs}=40	17.86	16.28	14.79
		n_{cs}=100	18.51	18.11	17.48
	ρ*=1	n_{cs}=5	4.21	2.51	2.06
		n_{cs}=10	7.46	5.32	4.60
		n_{cs}=20	9.49	5.57	4.61
		n_{cs}=40	14.58	9.22	7.49
		n_{cs}=100	17.35	14.03	11.53

The service model of this section assumes that while EF streams are individually policed on ingress, i.e., the traffic they inject into the network must conform to their own TBF, multiple streams are aggregated onto a common TBF, and hence jointly policed on egress. A key factor in determining the effect of this aggregate policing on egress conformance, is the relation that exists between ingress TBF's and the egress TBF. This relation is determined by parameters such as the number of individual streams mapped onto the egress TBF, and how the egress token rate and bucket size are computed from the corresponding ingress quantities.

The egress token rate is a function of the individual ingress token rates, and should normally be larger than or equal to their sum. On the other hand, the egress bucket size is likely to depend less on individual stream parameters, and instead be primarily dictated by service definition. For example, the virtual leased line model of [10], typically calls for an egress bucket size of one or two MTUs, independent of whether the egress TBF is for a single stream or multiple streams. As a result, the model we use in this section assumes an egress token rate equal to at least the sum of the ingress token rates, but a bucket size set to the maximum of the ingress bucket sizes. In other words, an egress TBF aggregating n_{ts} EF streams each with ingress TBF parameters (R, b), would be configured with parameters $(n_{ts}R, b)$. It should be noted that this selection of egress TBF parameters is bound to result in substantial egress non-conformance simply because of the possibility of simultaneous packet arrivals from multiple streams, i.e., the *intrinsic* burstiness of aggregated streams. In general, egress confor-

mance will be affected by both the potential for simultaneous arrivals and by the ability of network induced perturbations to create larger bursts. Each depends on network load, which is, therefore, expected to also play a role. On one and, higher loads increase the magnitude of network perturbations, which can contribute to the formation of larger bursts. On the other hand, higher loads mean larger queues that prevent large intrinsic bursts from propagating undisturbed through the network. In other words, at high loads the network can have a "smoothing" effect that limits the impact of aggregating multiple streams. In this section, we explore this issue and how egress non-conformance is in general affected by the use of an aggregate service model.

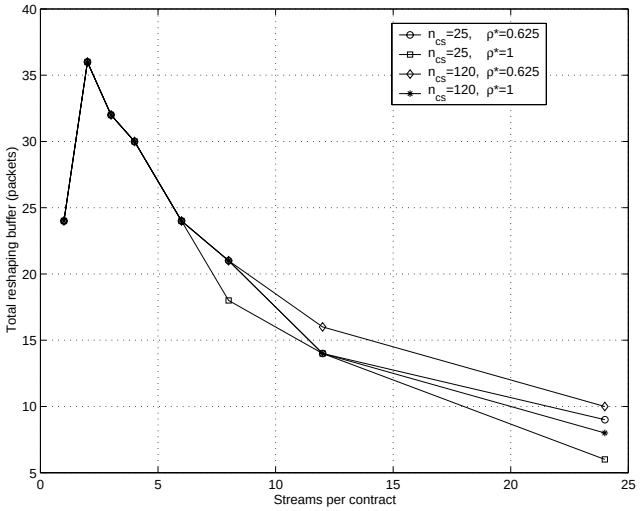

Figure 11: Reshaping buffer size for different levels of aggregation. ($N = 10$, $n_{ts} = 24$).

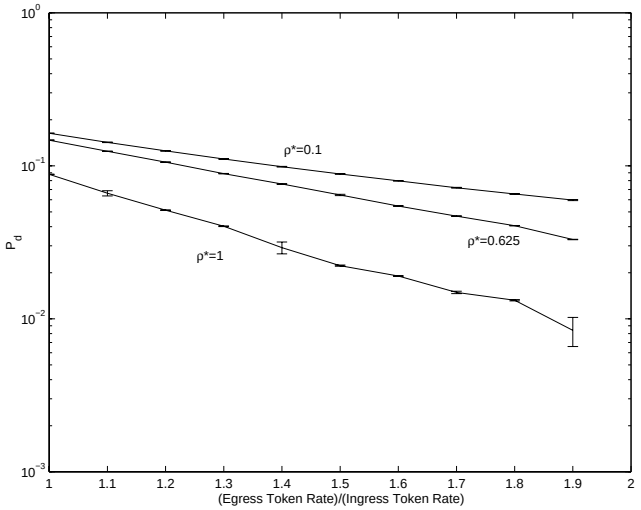

Figure 12: Impact of increasing egress token rate. ($N = 10$, $n_{ts} = 20$, $n_{cs} = 100$).

For that purpose, egress non-conformance is measured across a range of representative scenarios that involve varying link loads, the number of hops crossed, the number of cross streams, and the number of (tagged) streams being aggregated onto a common egress TBF. Several conclusions can be drawn from the data that was gathered, which point to the complex interactions at play even in the relatively simple configurations that were considered, i.e., homogeneous streams only. As mentioned above, there are two main factors that influence egress conformance: 1) the size of the intrinsic bursts created by aggregating multiple (conformant) streams onto the same egress TBF; and 2) the network induced interferences that perturb initially conformant streams. In parallel to that, there are several parameters that affect the relative impact of each one of these two factors. In particular, the number of streams being aggregated, the number of cross streams, the number of network hops being crossed, and the load on network links, all interact with each other in determining the respective weight of each factor. For example, high link loads can help smooth out large bursts, but also contribute to higher network interferences. The balance between these two effects depends on the magnitude of the burst contributed by stream aggregation. Similarly, increasing the number of cross streams can improve things by ensuring that large intrinsic aggregate bursts are broken up, but this only holds when the increase does not negatively affect the smoothing effect of the network or contribute to substantially larger levels of network interferences. Finally, a similar trade-off is present when increasing the hop count, as it can affect the relative impact of other parameters in various ways. For example, an increase in the number of cross traffic streams can either improve or worsen conformance depending on the hop count value. This is because, as mentioned before, the impact of increasing the number of cross streams can be either beneficial or detrimental depending of which of its two effects dominate, i.e., the positive impact of breaking large intrinsic bursts versus the negative effect of larger levels of network interferences. And hop count affects the outcome of this trade-off, i.e., higher hop counts increase the impact of network interferences.

The data generated from several experiments is reported in Table 1, and provide some indications on how these different factors interact with each other and influence egress conformance. The first and expected conclusion one can draw from the data, is that non-conformance is higher than when individual streams are mapped onto their own egress TBF, and the difference increases with the number of streams being aggregated. This can be seen by comparing the results of Figure 4 for $\rho^* = 1$, $\rho^* = 0.625$, and $\rho^* = 0.1$ to the corresponding values for $n_{cs} = 20$ in the second and third columns of Table 1. The differences are quite noticeable, especially for $n_{ts} = 20$, as expected, and interestingly are higher for lower link loads. In general, we see from the data that higher link load configurations result in overall better performance. As discussed earlier, this is because the smoothing effect of the network at high loads helps limit the intrinsic burstiness of aggregated streams. This is particularly visible in configurations with large potential intrinsic burstiness ($n_{ts} = 20$) and high network smoothing opportunities ($n_{cs} = 5$), where we can see improvements of close to an order of magnitude in the best scenarios.

The other two parameters that also determine how the intrinsic burstiness of aggregated streams and network interferences affect egress conformance, are the hop count N and the number of cross streams n_{cs}. From the table, we see that, at least for the scenarios considered here, increases in hop count typically improve performance, while increases in the number of cross streams usually degrade performance. This exemplifies the trade-off that exists between the greater level of network perturbations that higher hop counts or number of cross streams induces, and the greater likelihood that aggregate bursts will be broken-up as they traverse the network. For example, at low link loads ($\rho^* = 0.1$), changes in either parameter have little or no effect, while differences emerge

as link load increases due to the greater impact of traffic interactions in the network. In the context of homogeneous streams, it turns out that the break-up of aggregate streams caused by crossing a larger number of hops is the dominant effect, hence the improvement in egress conformance, while the negative impact of the perturbations created by larger numbers of cross streams is the more influential effect. However, it should be noted that, as we will see in the next section where we explore more realistic (heterogeneous) traffic mixes, this outcome is sensitive to the selected configuration. It should, therefore, only be interpreted as representative of the interactions that take place, and not necessarily as a conclusion that applies consistently across all scenarios.

After exploring some of the interactions that affect egress conformance, we investigate next what it takes to *absorb* non-conformance in the context of an aggregate service model. As before, we consider two mechanisms, namely, the use of reshaping buffers and increasing the egress token rate. When a reshaping buffer is used, we compare, for a given level of egress conformance, the size of the reshaping buffer needed with an aggregate service model, to the sum of the reshaping buffers needed in the case of individual service contracts. The results are shown in Figure 11, which plots the total amount of reshaping buffers needed to achieve a given level of conformance ($P_d \leq 10^{-5}$), as a function of the number of streams being aggregated onto the same egress TBF. The figure considers a total of 24 streams, so that the number of streams being aggregated ranges from 1 (every stream has its own service contract and, therefore, reshaping buffer) to 24 (all the streams are aggregated onto a common contract and, therefore, share the same reshaping buffer). The main, although expected, conclusion is that aggregation ultimately reduces the total amount of reshaping buffering needed. This highlight the benefits of large aggregate contract (when reshaping is used) in terms of buffer requirements. Furthermore, reshaping delays should also be lower because of the smaller ratio of total buffer size to aggregate rate. The effect of increasing the egress token rate is investigated next using a scenario where 20 streams are aggregated onto the same egress TBF ($n_{ts} = 20$), the number of cross streams is 100 ($n_{cs} = 100$), and the streams cross a total of $N = 10$ network hops. The results are displayed in Figure 12, which confirms the earlier finding that while increasing egress token rates improves things, it is relatively inefficient, i.e., a takes more than doubling the egress rate to achieve a reduction of an order of magnitude in non-conformance.

In the next section, we extend the basic scenarios we have explored so far, and investigate the effect of heterogeneity such as differences in (average) packet sizes and token rates.

4. HETEROGENEOUS SOURCES

The use of homogeneous streams was motivated by the need to better isolate the impact of various parameters, but this is clearly not representative of the environment one is likely to encounter in practice. In particular, rates and packet distributions are expected to vary across EF streams, and the impact of such differences is a dimension we explore in this section. We do so by constructing an heterogeneous traffic mix consisting of streams with variable size packets, different average rates, and average packets sizes. The investigation is carried out in the context of both individual and aggregate service models. The next two sub-sections are devoted to scenarios with individual service contracts, for which the impact of different rates and packet sizes are respectively considered. The last sub-section considers the case of aggregate contracts.

4.1 Heterogeneity In Rates

This section focuses on the impact of rate differences, and distinguishes between three types of flows with rates $R_1 = 0.1$ Mbps, $R_2 = 1$ Mbps and $R_3 = 10$ Mbps [8]. We use three tagged streams, one for each rate value, and measure their egress conformance after crossing a number of network hops. At each network hop, the three tagged streams interact with 30 cross-traffic streams, 10 from each rate group. All streams are fed by Poisson traffic sources. In addition, in order to limit the number of parameters being varied, all streams transmit fixed-size packets of size 500 bytes. The impact of different packet sizes is explored in the next sub-section.

The results for this set of experiments are shown in Table 2, where both the egress policer rate (P_R) and the load (ρ^*) on the network links are varied. The main conclusion is that the only configurations that exhibit noticeable differences are at high link loads (ρ^*). At high loads, the efficiency of increasing the egress token rate to minimize non-conformance (without requiring reshaping) is lower for high rate streams than for low rate ones. The reason is again that higher rates translate into shorter time scales, which make streams more susceptible to network perturbations. This is best understood from examples based on Table 2.

Specifically, consider the case where the egress token rate is 10% higher than the corresponding ingress rate. In this case, the time needed to generate enough tokens for a packet on egress is 91% of what it takes on ingress. For the three types of streams of Table 2 with egress token rates of $P_{R1} = 0.125$ Mbps, $P_{R2} = 1.25$ Mbps, and $P_{R3} = 12.5$ Mbps, the corresponding amounts of time by which a packet can arrive early on egress are $2900\mu s$, $290\mu s$, $29\mu s$, respectively. In comparison, the transmission time of a single 500 bytes packet on a network link is $26.2\mu s$ (for ρ^*=1). This means that for high rate streams, a 10% higher egress token generation rate is barely capable of absorbing network perturbations of about one packet. In contrast, for streams with a rate of 0.125 Mbps, a 10% higher egress token generation rate translates into an ability to absorb network perturbations of about 100 packets. More than enough to eliminate any egress non-conformance.

On the other hand, as we have seen in previous sections, although decreasing the network load does not (initially) reduce the number of non-conformant packets, it does reduce the amount (of time) by which they are non-conformant. As a result, we expect the above difference to diminish and eventually disappear for lower link loads. This can be verified from Table 2, where for $\rho^* = 0.4$ the differences between high and low bandwidth streams has all but disappeared. The network perturbations are now minimum, and even the small margin afforded to high bandwidth streams by higher egress token generation rates is sufficient to accommodate them.

Table 2: Percentage of non-conformant packets. S_R source rate; P_R egress policer rate

S_R (Mbps)	P_R (Mbps)	$\rho^* = 1$	$\rho^* = 0.4$	$\rho^* = 0.1$
0.1	0.125	4.51 ± 0.31	3.91 ± 0.24	1.60 ± 0.16
	0.138	0	0	0
1	1.25	4.63 ± 0.06	3.79 ± 0.07	1.37 ± 0.03
	1.375	≤ 0.0087	0	0
10	12.5	4.64 ± 0.03	3.43 ± 0.02	1.12 ± 0.006
	13.75	1.64 ± 0.017	0.024 ± 0.0023	0
	15	0.71 ± 0.009	≤ 0.0005	0
	16.25	0.35 ± 0.006	0	0

One of the conclusions from the results of this section is that when reshaping is **not** available at domain boundaries, it appears best to negotiate small contracts with an egress rate higher than the

[8]For all three, the ingress token generation rate is again set to be 25% higher than the traffic rate

ingress rate. In addition, priority based schemes (low load) should also be used inside the network in order to minimize the magnitude of perturbations.

Table 3: Percentage of non-conformant packets. S packet size

$S(bytes)$	$\rho^* = 1$	$\rho^* = 0.4$	$\rho^* = 0.1$
40	0	0	0
200	0.11 ± 0.006	0.089 ± 0.006	0.037 ± 0.006
1000	4.61 ± 0.11	3.85 ± 0.065	1.41 ± 0.064

4.2 Heterogeneity in Packet Sizes

In this section, we investigate the effect of heterogeneity in packet sizes rather than stream bandwidths, which are kept identical across streams. As before, we consider three types of streams, this time with different packet sizes equal to $S_1 = 40$ bytes, $S_2 = 200$ bytes, and $S_3 = 1000$ bytes. Packet sizes are kept fixed within each stream. Varying packet sizes within a stream, as done in Section 3.2, contributed only minor differences and was, therefore, omitted to simplify the discussion. All streams have identical ingress and egress TBFs, with token rates set 25% higher than the source data rate, and bucket sizes equal to $2,000$ bytes.

The results from the experiments are displayed in Table 3, and do not add much to the findings of Section 3.2. As expected, the streams with the bigger packets experienced the worst level of non-conformance. This is intuitive as the transmission (reception) of a 1000 bytes packet requires the availability of half a full token bucket. As a result, the margin of tokens left behind by each packet will typically be smaller for streams that generate large packets than for streams with small packet. This in turn makes the former more susceptible to network perturbations.

4.3 Heterogeneity and Aggregate Contracts

We consider next what is likely to be the more common scenario in practice, namely, a mixture of streams with different average packets sizes and rates, combined with the use of an aggregate service model. In addition, we also assume that packet sizes can vary within a stream, again according to a truncated exponential distribution. Streams are selected from the following combinations of average rates and average packet sizes: (0.1 Mbps,40 bytes), (1 Mbps,40 bytes), (1 Mbps,200 bytes),(1 Mbps,500 bytes) and (10 Mbps,500 bytes). As before, the ingress bucket rate was chosen to be 25% higher than the average source data rate, and both the ingress and egress bucket sizes were set to $1,000$ bytes, the maximum packet size. In our first experiment, the token rate of the aggregate egress TBF's was configured to be the sum of the corresponding ingress token rates. It was varied in subsequent experiments.

The data generated from these experiments is found in Table 4, and confirm many of the findings obtained with the simple homogeneous scenarios. However, because of the more complex nature of the interactions generated by the richer mix of traffic, we also observe some differences. They reflect the fact that the boundaries of the regions where the influence of different individual parameters dominates, shift as a function of the traffic mix. Nevertheless, there are a number of basic results that remain unchanged.

First and foremost, as can be seen from columns $N = 5$ and $N = 10$ of Table 4, the level of egress non-conformance in the aggregate service model remains higher than in the individual service model. Interestingly though, the difference is less than what was observed in the earlier homogeneous scenario. This is in part due to the presence of streams with smaller packets, but also reflects the

Table 4: Non-conformance (in percentage) for aggregate heterogeneous contracts.

			N=2	N=5	N=10
n_{ts}=5	ρ*=0.1	n_{cs}=5	4.41	5.84	8.68
		n_{cs}=10	4.28	5.16	7
		n_{cs}=20	4.22	4.67	5.67
		n_{cs}=40	4.1	4.35	4.86
		n_{cs}=100	4.15	4.22	4.36
	ρ*=0.625	n_{cs}=5	6.18	11.87	16.54
		n_{cs}=10	5.54	10.44	15.71
		n_{cs}=20	4.96	8.42	13.31
		n_{cs}=40	4.55	6.51	10.05
		n_{cs}=100	4.28	5.01	6.58
	ρ*=1	n_{cs}=5	5.82	8.18	9.31
		n_{cs}=10	5.68	8.6	10.62
		n_{cs}=20	5.28	8.06	10.94
		n_{cs}=40	4.84	6.94	9.83
		n_{cs}=100	4.43	5.55	7.43
n_{ts}=20	ρ*=0.1	n_{cs}=5	8.53	9.63	11.96
		n_{cs}=10	8.48	9.39	11.27
		n_{cs}=20	8.44	9.08	10.40
		n_{cs}=40	8.4	8.79	9.61
		n_{cs}=100	8.35	8.54	8.85
	ρ*=0.625	n_{cs}=5	8.44	12.35	16.09
		n_{cs}=10	8.54	12.27	16.51
		n_{cs}=20	8.61	11.79	16.42
		n_{cs}=40	8.6	10.89	15.2
		n_{cs}=100	8.51	9.67	12.23
	ρ*=1	n_{cs}=5	5.82	6.95	7.28
		n_{cs}=10	6.76	8.3	8.91
		n_{cs}=20	7.48	9.26	10.54
		n_{cs}=40	7.94	9.47	11.66
		n_{cs}=100	8.23	8.98	11.09

fact that the wider range of traffic interactions in an heterogeneous setting helps break-up the intrinsic bursts formed by aggregating multiple streams. This wider range of interactions also leads to a number of differences with an homogeneous environment. They are mostly representative of how different trade-offs play-out under the more diverse conditions of heterogeneous traffic mixes. For example, while higher link loads are still often beneficial because of the greater network smoothing effect they afford, this benefit is not as consistent as in the homogeneous case. In particular, larger intrinsic bursts, i.e., larger values of n_{ts}, and higher load levels, are typically required before the payoff becomes apparent. For example, the benefits of higher link loads are not seen in the $n_{ts} = 5$ or $\rho^* = 0.625$ scenarios. Instead, they require the larger intrinsic bursts of $n_{ts} = 20$ and load levels of $\rho^* = 1$.

As discussed in the preceding section, increasing the number of streams being aggregated onto a common egress TBF has two competing effects: potentially larger intrinsic bursts together with the opportunity for higher network smoothing because the aggregate traffic represents a greater fraction of link capacity. The latter effect diminishes when network load decreases, so that the negative effect of larger bursts ultimately dominates. Similarly, while increasing the number of network hops traversed by streams typically improved performance in the homogeneous case, this is not so when streams are heterogeneous. Instead, increasing N consistently degrades performance. This is because the greater network perturbations implied by higher hop counts are now dominating, in part due to the fact that the benefit of breaking up intrinsic bursts is often already achieved simply through interactions between het-

erogeneous streams.

Finally, another area of difference is in terms of the impact of the number of cross streams. In the homogeneous case, a larger number of cross streams consistently worsened performance, while this does not hold in heterogeneous scenarios. Increasing the number of cross-traffic streams actually improves performance in some scenarios, i.e., for small number of aggregated streams ($n_{ts} = 5$) and especially for large number of network hops being crossed ($N = 10$). The main reason behind this behavior is that while aggregating only a small number of streams ensures a lower initial burstiness, that burstiness can still be increased by interactions with other streams in the network, especially when the number of network hops crossed is high. However, as the number of cross-traffic streams increases, the large number of independent interferences combined with the higher speed of the links, and hence the lower relative magnitude of those interferences, ultimately improves performance. This phenomenon was not observed in the homogeneous scenario because of its lower initial burstiness. It is also not observed at high network loads, because the decrease in network smoothing due to the larger number of cross-traffic streams is the dominant effect.

In general, the above discussion points to the fact that in a more realistic heterogeneous environment, careful consideration must be given to the relative weights of the different parameters. For example, large aggregate contracts may or may not be beneficial depending on the type of queueing mechanism used in the network (high vs low load), and the intrinsic burstiness of the streams being aggregated. This means that in practice it will be difficult to accurately predict the level of egress non-conformance that can be expected. However, this may not be as severe an issue as it appears, as the sheer magnitude of the levels of non-conformance being observed even in the best scenarios, i.e., small contracts and low loads, mandates the use of additional mechanisms to absorb non-conformance at network boundaries. As discussed earlier, the two possible approaches available are either reshaping buffers or higher egress token rates. Fortunately, both yield results that are similar to what was observed in the homogeneous case, even if the relative inefficiency of relying on higher egress token rates is now more pronounced.

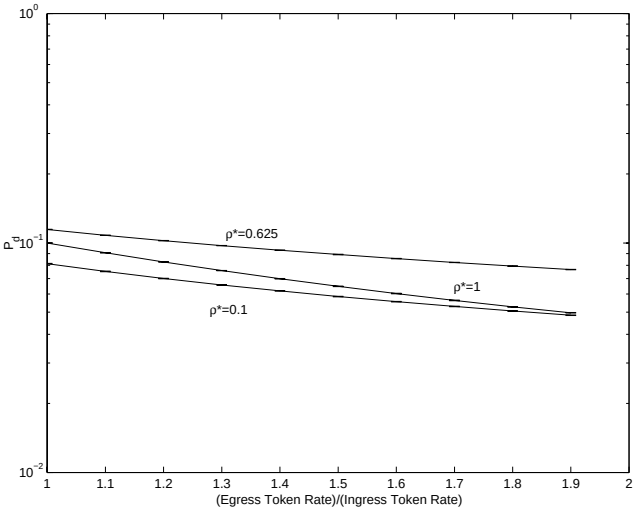

Figure 13: Impact of increasing egress token rate. ($N = 10$, $n_{ts} = 20$, $n_{cs} = 100$).

In particular, the size of the reshaping buffers needed to reduce egress non-conformance to acceptable levels, remains relatively small, and the efficiency of using buffers still improves as the number of streams being aggregated increases. This is illustrated in Table 5, which shows that a maximum buffer size of 9 packets, i.e., $4,500$ bytes, is sufficient to ensure a non-conformance probability P_d of less than 10^{-5} when aggregating 20 streams. This is similar to what was required in the case of homogeneous streams, and is to be compared with a buffer size of about one packet per stream for individual contracts, i.e., a total buffer size of 20 packets (see Figure 11). Furthermore, efficiency still improves as the number of streams being aggregated increases. This can be seen by comparing for all configurations, the total buffer size needed for one aggregate contract of 20 streams, to what four contracts of 5 streams each require. This means that the earlier conclusion that egress non-conformance can be absorbed through small reshaping buffers, still holds even in the presence of heterogeneous streams. The data of Table 5 also confirms other earlier findings regarding aggregate contracts. For example, we see that increasing the number of cross streams is usually beneficial, i.e., results in smaller reshaping buffers, except in configurations where network shaping has the potential to substantially reduce the intrinsic burstiness of stream aggregates, i.e., when both the load and the number of aggregated streams are high. This is similar to what was observed in Table 4.

Table 5: Required egress buffer size (in packets) to achieve a non-conformance probability $\leq 10^{-5}$ for aggregate heterogeneous contracts.

			N=2	N=5	N=10
n_{ts}=5	ρ*=0.1	n_{cs}=5	4	4	5
		n_{cs}=10	4	4	5
		n_{cs}=20	4	4	4
		n_{cs}=40	4	4	4
		n_{cs}=100	4	4	4
	ρ*=0.625	n_{cs}=5	4	5	7
		n_{cs}=10	4	5	6
		n_{cs}=20	4	5	5
		n_{cs}=40	4	5	5
		n_{cs}=100	4	4	5
	ρ*=1	n_{cs}=5	4	5	5
		n_{cs}=10	5	5	6
		n_{cs}=20	4	5	6
		n_{cs}=40	4	5	5
		n_{cs}=100	4	4	5
n_{ts}=20	ρ*=0.1	n_{cs}=5	8	8	9
		n_{cs}=10	8	8	8
		n_{cs}=20	8	8	8
		n_{cs}=40	8	8	8
		n_{cs}=100	8	8	8
	ρ*=0.625	n_{cs}=5	8	8	9
		n_{cs}=10	8	8	9
		n_{cs}=20	8	8	9
		n_{cs}=40	8	8	9
		n_{cs}=100	8	8	9
	ρ*=1	n_{cs}=5	5	4	4
		n_{cs}=10	6	5	5
		n_{cs}=20	7	6	6
		n_{cs}=40	7	7	8
		n_{cs}=100	8	8	8

The other mechanism available to absorb egress non-conformance, is to increase the egress token rate, and its effect is shown in Figure 13. The results are similar to those of Figure 12, except for

the previously mentioned decrease in efficiency of such a mechanism, as well as some minor differences due to the more complex interactions created by heterogeneous traffic. For example, while increasing load uniformly improved performance in the homogeneous scenario of Figure 12, i.e., it both lowered non-conformance and also increased the rate of improvements achievable through higher egress rates, Figure 13 shows a slightly different behavior. In particular, while higher loads still imply greater efficiency (the slope of decrease of non-conformance as a function of the egress rate is higher for $\rho^* = 1$), the relative ordering of non-conformance levels is different. This difference is consistent with the data of Table 4, and again reflects the impact of more varied interactions in the presence of heterogeneity.

5. CONCLUSION

The goal of this paper was to better understand the impact of traffic aggregation on conformance, in the context of a service built on top of the Differentiated Services EF PHB. The focus was on identifying the level of non-conformance that crossing a Diff-Serv domain introduces into initially conformant streams. This was investigated for two possible models of service contracts: service contracts that extend individual service agreements across domains, and service contracts that map individual agreements onto aggregate provider level service contracts. The paper also explored two different approaches, i.e., egress reshaping and higher egress token rates, for absorbing non-conformance on egress.

The findings of the paper confirm that reshaping is, if not mandatory, by far the most efficient way to eliminate egress non-conformance. The amount of reshaping buffers required is typically small, i.e., of the order of a few packets, and when multiple streams are mapped onto aggregate contracts, the relative amount of buffering required for each stream decreases, i.e., efficiency improves. However, reshaping may not always be available, especially on very high speed links, and the use of higher token egress rates combined with low network loads (priority based support of EF traffic) may provide an alternative, at least in the context of individual service contracts. For example, Figure 8 shows that for link loads of 0.4, an increase of 20% in the egress token rate brings the probability of non-conformance down to 10^{-5}. Such a solution is unfortunately not effective in the case of aggregate contracts for which increasing the egress rate yields only much smaller improvements. As a result, the use of reshaping appears unavoidable in those cases. In general, while the paper clearly did not cover all possible scenarios and parameter settings, it should provide useful information on how to offer and dimension services based on the EF PHB.

6. REFERENCES

[1] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An architecture for differentiated services. Request For Comments (Proposed Standard) RFC 2475, IETF, December 1998.

[2] M. Grossglauser and S. Keshav. On CBR service. In *Proceedings of INFOCOM'96*, pages 129–137, San Francisco, CA, March 1996.

[3] R. Grünenfelder. A correlation based end-to-end cell queueing delay characterization in an ATM network. In *Proceedings of ITC-13 Workshops: Queueing, Performance and Control in ATM*, pages 59–64, Copenhagen, Denmark, June 1991.

[4] J. Heinanen, F. Baker, W. Weiss, and J. Wroclawski. Assured Forwarding PHB group. Request For Comments (Proposed Standard) RFC 2597, IETF, June 1999.

[5] J. Heinanen and R. Guerin. A single rate three color marker. Request For Comments (Informational) RFC 2697, IETF, September 1999.

[6] J. Heinanen and R. Guerin. A two rate three color marker. Request For Comments (Informational) RFC 2698, IETF, September 1999.

[7] V. Jacobson, K. Nichols, and K. Poduri. An Expedited Forwarding PHB. Request For Comments (Proposed Standard) RFC 2598, IETF, June 1999.

[8] L. Massoulié. Large deviations orderings of point processes in some queueing networks. *Queueing Systems*, 28(4):317–335, 1998.

[9] W. Matragi, K. Sohraby, and C. Bisdikian. Jitter calculus in ATM networks: multiple nodes. *IEEE/ACM Trans. Netw.*, 5(1):122–133, February 1997.

[10] K. Nichols, V. Jacobson, and L. Zhang. A two-bit differentiated services architecture for the internet. Request For Comments (Informational) RFC 2638, IETF, July 1999.

[11] Network Simulator - NS (version 2). `http://www-mash.CS.Berkeley.EDU/ns`.

[12] J. W. Roberts, U. Mocci, and J. Virtamo, editors. *Broadband Network teletraffic - Final Report of Action COST 242*, volume 1155. Springer-Verlag, 1996.

[13] J. W. Roberts and J. Virtamo. The superposition of periodic cell arrival streams in an ATM multiplexer. *IEEE. Trans. Commun.*, 39(2):298–303, February 1991.

[14] J. Sahni, P. Goyal, and H. M. Vin. Scheduling CBR flows: FIFO or per-flow queueing? In *Proceedings of NOSSDAV'99*, AT&T Learning Center, Basking Ridge, NJ, June 1999.

[15] S. Sathaye. ATM Forum traffic management specification Version 4.0. ATM Forum 95-0013, December 1995.