Network Working Group
Internet-Draft
Intended status: Informational
Expires: January 6, 2008

J. Zhang
Cisco Systems, Inc. and Cornell University
A. Charny
V. Liatsos
F. Le Faucheur
Cisco Systems, Inc.

July 9, 2007

Performance Evaluation of CL-PHB Admission and Termination Algorithms draft-zhang-pcn-performance-evaluation-02.txt

Status of this Memo

Copyright Notice

Abstract

Pre-Congestion Notification [I-D.briscoe-tsvwg-cl-architecture] approach proposes Admission Control to limit the amount of real-time PCN traffic to a configured level during the normal operating conditions, and Flow Termination used to tear-down some of the flows

to bring the PCN traffic level down to a desirable amount during unexpected events such as network failures, with the goal of maintaining the QoS assurances to the remaining flows. Preliminary performance evaluation results on example admission and termination mechanisms were presented in [I-D.briscoe-tsvwg-cl-phb] and in earlier versions of this draft. This draft presents the results of a follow-up simulation study.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

Table of Contents

1. Introduction

Pre-Congestion Notification approach (draft-eardley-pcn-architecture, [I-D.briscoe-tsvwg-cl-architecture]) proposes Admission Control to limit the amount of real-time PCN traffic to a configured level during the normal operating conditions, and Flow Termination used to tear down some of the flows to bring the PCN traffic level down to a desirable amount during unexpected events such as network failures, with the goal of maintaining the QoS assurances to the remaining flows. In draft-eardley-pcn-architecture , Admission and Termination use two different markings and two different metering mechanisms in the internal nodes of the PCN region. Here and elsewhere in this document we will omit "Flow" and refer to Flow Termination simply as Termination.

An initial simulation study was reported in [I-D.briscoe-tsvwg-cl-phb], where it was shown that both Admission and Termination mechanisms discussed there have reasonable performance in a limited set of experiments performed there. This draft reports the next installment of the simulation results. For completeness and convenience of exposition, most of the results earlier presented in [I-D.briscoe-tsvwg-cl-phb] have been moved into this draft.

The new results presented in the current draft further confirm that Admission and Termination algorithms of [I-D.briscoe-tsvwg-cl-phb] perform well under a range of operating conditions and are relatively insensitive to parameter variations around a chosen operation range.

Perhaps the most interesting (and somewhat unexpected) conclusion that can be drawn from these results is that both Admission and Termination algorithms appear to be not as sensitive to low per ingress-egress-pair aggregation as one might fear. This result is quite encouraging: while it seems reasonable to assume sufficient bottleneck link aggregation, it is not very clear whether one can safely assume high levels of aggregation on a per ingress-egress-pair basis. Yet, low levels of ingress-egress aggregation remain a potential concern, especially for the Termination mechanism. More discussion on this is presented in section 4. Other conclusions are presented in Section 5.]

Section 2 describes simulation environment and models, Admission and termination simulation results are presented in sections 3 and 4, and section 5 summarizes the results of the simulations so far and lists areas for further study.

1.1. Changes from the previous version

- Refined the analysis of low aggregation effect on Termination
- Added batch arrivals experiments for Termination
- Added Fairness analysis for Admission
- Added experiments with different voice codecs mixes sharing the bottleneck
- Replaced the Terminology section with a pointer to draft-eardley-pcn-architecture
- Miscellaneous editorial changes and clarifications based on feedback to the previous version

1.2. Terminology

This draft uses the terminology as defined in draft-eardly-pcn-architecture-00.

2. Simulation Setup and Environment

2.1. Network Models

We use three types of topologies, described in this section. In the simplest topology shown in Fig. 2.1 the network is modelled as a single link between an ingress and an egress node, all flows sharing the same link. Figure 2.1 shows the modelled network. A is the ingress node and B is the egress node.

Fig. 2.1 Simulated Single Link Network (Referred to as Single Link Topology)

A subset of simulations uses a network structured similarly to the network shown on Figure 2.2. A set of ingresses (A,B,C) connected to an interior node in the network (D) with links of different propagation delay. This node in turn is connected to the egress (F). In this topology, different sets of flows between each ingress and the egress converge on the single link, where Pre-congestion notification algorithm is enabled. The ingress link capacity is assumed to be sufficiently large so that neither Admission nor

Termination mechanisms have any effect on them. All links are assigned a propagation delay. The point of congestion (link (D-F) connecting the interior node to the egress node) is modelled with a 1ms or 10ms propagation delay. In our simulations, the number of ingress nodes in the network range from 2 to 1800 nodes, each connected to the interior node with a range of propagation delay (1ms to 100ms). In some experiments all ingress links have the same propagation delay, and in some experiments the delay of different ingresses vary in the range from 1 to 100 ms.

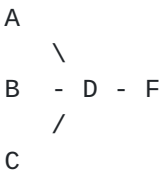

Fig. 2.2. Simulated Multi-Link Network (Referred to as RTT Topology)

Another type of network of interest is multi-bottleneck topology that we call Parking Lot (PLT). The simplest PLT with 2 bottlenecks is illustrated in Fig 2.3(a). An example traffic matrix with this network on this topology is as follows:

- o an aggregate of "2-hop" flows entering the network at A and leaving at C (via the two links A-B-C)
- o an aggregate of "1-hop" flows entering the network at D and leaving at E (via A-B)
- o an aggregate of "1-hop" flows entering the network at E and leaving at F (via B-C)

In the 2-hop PLT of Fig. 2.3(a) the points of congestion are links A--B and B--C. Capacity of all other links is not limiting. This topology and traffic matrix models the network where some flows cross multiple bottlenecks, each with substantial amount of cross-traffic.

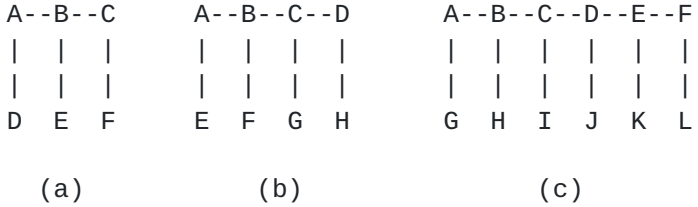

Figure 2.3: Simulated Multiple-bottleneck (Parking Lot) Topologies.

We also experiment with larger PLT topologies with 3 bottlenecks (see Fig 2.3(b)) and 5 bottlenecks (Fig 2.3 (c)). In all cases, we

simulated one ingress-egress pair that carries the aggregate of "long" flows traversing all the N bottlenecks (where N is the number of bottleneck links in the PLT topology, shown as "horizontal" links in Fig. 2.3), and N ingress-egress pairs that carry flows traversing a single bottleneck link and exiting at the next "hop". In all cases, capacities of all "vertical" links are non-limiting, so neither Termination not Admission mechanisms are never triggered on these links. Propagation delays for all links in all PLT topologies are set to 1ms.

These topologies aim to model the cross traffic and congestion that can occur in the hierarchically structured networks deployed by many network providers.

Due to time limitations, other possible traffic matrices (e.g. some of the flows traversing a subset of several bottleneck links in Fig 2.3) have not yet been considered and remain the area for future investigation.

Our simulations concentrated primarily on the range of capacities of 'bottleneck' links with sufficient level of bottleneck aggregation - above 10 Mbps for voice and 622 Mbps for "video", up to 2.4 Gbps. But we also investigated slower 'bottleneck' links down to 512 Kbps in some experiments.

2.2. Call Signaling Model

In the simulation model of Flow Admission Control, a flow request arrives at the ingress and immediately sends a message to the egress. The message arrives at the egress after the propagation time plus link processing time (but no queuing delay). When the egress receives this message, it immediately responds to the ingress with the current Congestion Level Estimate (CLE). If the CLE is below the specified CLE- threshold, the flow is admitted, otherwise it is rejected.

For Termination, once the ingress node of a PCN region decides to terminate a flow, that flow is terminated immediately and sends no more packets from that time on. The life of a flow outside the domain described above is not modelled. Propagation delay from source to the ingress and from destination to the egress is assumed negligible and is not modelled.

2.3. Traffic Models

We simulated four models of real-time traffic - two voice models and two video models. The voice models included CBR voice and on-off traffic approximating voice with silence compression. For video, we

simulated on-off traffic with peak and mean rates corresponding to an MPEG-2 video stream (we termed the latter Synthetic Video (SVD)), and a real video trace (VTR).

The distribution of flow duration was chosen to be exponentially distributed with mean 1min, regardless of the traffic type. In most of the experiments flows arrived according to a Poisson distribution with mean arrival rate chosen to achieve a desired amount of overload over the configured-admission-rate in each experiment. Overloads in the range 1x to 5x and underload with 0.95x have been investigated. For on-off traffic, on and off periods were exponentially distributed with the specified mean. Traffic parameters for each flow are summarized below .

2.3.1. Voice Traffic Models

The table below describes all voice codecs we modeled in our simulation results.

The first two rows correspond to our two basic models (they correspond to the older G.711 encoding with and without silence compression). These two models are referred simply as "CBR" and "VBR" in the reported simulation results.

We also simulated several "mixes" of the different codecs reported in the table below. The primary mix consists of equal proportion of all voice codecs list below. We have also simulated various other mix consist different proportion of the subset of all codecs. Though these result are not reported in this draft due to their similarities to the primary mix result.

Name/Codecs	Packet Size (Bytes)	Inter-Arrival Time (ms)	On/Off Period Ratio	Average Rate (kbps)
"CBR"	160	20	1	64
"VBR"	160	20	0.34	21.75
G.711 CBR	200	20	1	80
G.711 VBR	200	20	0.4	32
G.711 CBR	120	10	1	96
G.711 VBR	120	10	0.4	38.4
G.729 CBR	60	20	1	24
G.729 VBR	60	20	0.4	9.6

Table 2.1. Simulated Voice Codecs.

2.3.2. **Synthetic "Video" - High Peak-to-Mean Ratio VBR** Traffic (SVD)

This model is on-off traffic with video-like mean-to-peak ratio and mean rate approximating that of an MPEG-2 video stream. No attempt is made to simulate any other aspects of a video stream, and this model is merely that of on-off traffic. Although there is no claim that this model represents the performance of video traffic under the algorithms in question adequately, intuitively, this model should be more challenging for a measurement-based algorithm than the actual MPEG video, and as a result, 'good' or "reasonable" performance on this traffic model indicates that MPEG traffic should perform at least as well. We term this type of traffic SVD for "Synthetic Video". Parameters used for this traffic models are:

- Long term average rate 4 Mbps
- On Period mean duration 340ms; during the on-period the packets are sent at 12 Mbps
- 1500 byte packets, packet inter-arrival: 1ms
- Off Period mean duration 660ms

2.3.3. Real Video Traces (VTR)

We used a publicly available library of frame size traces of long MPEG-4 and H.263 encoded video obtained from http://www.tkn.tu-berlin.de/research/trace/trace.html (courtesy Telecommunication Networks Group of Technical University of Berlin). Each trace is roughly 60 minutes in length, consisting of a list of records in the format of <FrameArrivalTime, FrameSize>. Among the 160 available traces, we picked the two with the highest average rate (averaged over the trace length, in this case, 60 minutes. In addition, the two also have a similar average rate). The trace file used in the simulation is the concatenation of the two. Since the duration of the flow is much smaller than the length of the trace, we need to check how the expected rate of flow relates to the trace's long term average. To do so, we simulate a number of flows starting from random locations in the trace with duration chosen to be exponentially distributed with mean 1min. The results show that the expected rate of flow is roughly the same as the trace's average. Traffic characteristics are summarized below:

- Average rate 769 Kbps
- Each frame is sent with packet length 1500 bytes and packet inter-arrival time 1ms
- No traffic is sent between frames.

2.3.4. Randomization of Base Traffic Models

To emulate some degree of network-introduced jitter, in some experiments we implemented limited randomization of the base models by randomly moving the packet by a small amount of time around its transmission time in the corresponding base traffic model. More specifically, for each packet we chose a random number R, which is picked from uniform distribution in a randomization-interval, and delayed the packet by R compared to its ideal departure time. We choose randomization-interval to be a fraction of packet-inter-arrive-time of the CBR portion of the corresponding base model. To simulate a range of queueing delays, we varied this fraction from 0.0001 to 0.1. While we do not claim this to be an adequate model for network-introduced jitter, we chose it for the simplicity of implementation as a means to gain insight on any simulation artifacts of strictly CBR traffic generation. We implemented randomized versions of all 5 traffic streams (CBR, VBR, MIX, SVD and VTR) by randomizing the CBR portion of each model.

2.4. Performance Metrics

In all our experiments we use as performance metric the percent deviation of the mean rate achieved in the experiment from the expected load level. We term these "over-admission" and "over-termination" percentages, depending on the type of the experiment.

More specifically, our experiments measure the actual achieved throughput at 50 ms intervals, and then compute the average of these 50ms rate samples over the duration of the experiment (where relevant, excluding warmup/startup conditions). We then compare this experiment average to the desired traffic load.

Initially in our experiments we also computed the variance of the traffic around the mean, and found that in the vast majority of the experiments it was quite small. Therefore, in this draft we omit the variance and limit the reporting to the over-admission and over-termination percentages only.

2.5. Simulation Environment

The simulation study reported here used purpose built discrete-event simulator implemented in ECLiPSe Language (http://eclipse.crosscoreop.com/eclipse). The latter is intended for general programming tasks, and is especially suitable for rapid prototyping. Simulations were run on Enterprise Linux Red Hat, IBM eServer x335, 3.2GHz Intel Xeon, 4GB RAM.

3. Admission Control

3.1. Parameter Settings

3.1.1. Virtual queue settings

Unless otherwise specified, most of the simulations were run with the following Virtual Queue thresholds:

- min-marking-threshold: 5ms at virtual queue rate
- max-marking-threshold: 15ms at virtual queue rate
- virtual-queue-upper-limit: 20ms at virtual queue rate

The virtual-queue-upper-limit puts an upper bound on how much the virtual queue can grow. Note that the virtual queue is drained at a configured rate smaller than the link speed.

Most of the simulations were set with the configured-admissible-rate at half the link speed. Note that as long as there is no packet loss, the admission control scheme successfully keeps the load of admitted flows at the desired level regardless of the actual setting of the configured-admissible-rate. However, it is not clear if this remains true when the configured-admissible-rate is close to the link speed/actual queue service rate. Further work is necessary to quantify the performance of the scheme with smaller service rate/ virtual queue rate ratio, where packet loss may be an issue.

3.1.2. Egress measurements

The CLE is computed as an exponential weighted moving average (EWMA) with a weight of 0.01. In the simulation results presented in sections 3.2 and 3.3 the CLE is computed on a per-packet basis as it is that setting that was used in [I-D.briscoe-tsvwg-cl-phb], from which these results are taken. For those experiments the CLE value 0.5 and EWMA weight of 0.01 are used unless otherwise specified. Our subsequent study indicated that there is no significant difference between the observed performance of interval-based and per-packet egress measurements. Since interval based measurements for a large number of ingresses are substantially easier for hardware implementations, subsequent studies reported in the rest of this draft concentrated on the interval based egress measurement. The measurement interval was chosen to be 100ms, and a range of CLE values and EWMA weights was explored, as specified in specific experiment descriptions.

3.2. What Bottleneck Aggregation is Sufficient?

One of the assumptions in [I-D.briscoe-tsvwg-cl-architecture] is that there is sufficient aggregation on the "bottleneck" links. Our first set of experiments revolved around getting some preliminary intuition of what constitutes "enough bottleneck aggregation" for the traffic models we chose. To that end we fixed configured-admissible-rate at half the link speed in the range of T1 (1.5 Mbps) through 1Gbps, and examined the level of aggregation at different link speeds for different traffic models corresponding to the chosen configured admission rate at those speeds. Further, to eliminate the issue of whether ingress-egress pair aggregation has any significant effect, in the experiments performed in this section we used Single Link topology only, so that all flows shared the same ingress-egress pair.

We found that on links of capacity from 10Mbps to OC3, admission control for CBR voice and ON-OFF voice (VBR) traffic work reliably with the range of parameters we simulated, both with Poisson and Batch call arrivals. As the performance of the algorithm was quite good at these speeds, and generally becomes the better the higher the

degree of aggregation of traffic, we chose to not investigate higher link speeds for CBR and VBR voice, within the time constraints of this effort.

The performance at lower link speeds was substantially worse, and these results are not presented here. These results indicate that a rule of thumb, admission control algorithm described in [I-D.briscoe-tsvwg-cl-architecture] should not be used at aggregations substantially below 5 Mbps of aggregate rate even for voice traffic (with or without silence compression). For higher-rate on-off SVD traffic, due to time limitations we simulated 1Gbps and OC12 (622 Mbps) links and Poisson arrivals only. Note that due to the high mean and peak rates of this traffic model, slower links are unlikely to yield sufficient level of aggregation of this type of traffic to satisfy the flow aggregation assumptions of [I-D.briscoe-tsvwg-cl-architecture]. Our simulations indicated that this model also behaved quite well at these levels of aggregation, although the deviation from the configured-admissible-rate is slightly higher in this case than for the less bursty traffic models. Recalling that simulated SVD model is in fact just on-off traffic with high peak rate and video-like peak ratio, we believe that the actual video will behave only better, and hence it follows that with bottleneck aggregation of the order of 150 SVD flows the admission control algorithm is expected to perform reasonably well. Note however that this statement assumes sufficient per ingress-egress pair aggregation as well.

Due to time limitations bottleneck aggregation experiments were not performed for other traffic models.

For the chosen link speeds and traffic models, we investigated the demand overload of 2x-5x. By demand overload we mean that the sources generate traffic with the aggregate mean rate exceeding the configured-admissible-rate by the specified factor. Performance at lower levels of demand overload is expected to be only better. Higher levels of overloads have not been studied due to time limitations, especially given the expectation that the 5x demand overload is already sufficiently rare to expect in practice.

Table 3.1 below summarizes the worst case difference (in percent) between the admitted load and configured-admissible-rate (we refer to as over-admission-perc). The worst case difference was taken over all experiments with the corresponding range of link speeds and demand overloads. In general, the higher the demand, the more challenging it is for the admission control algorithm due to a larger number of near-simultaneous arrivals at higher overloads, and as a result the worst case results in Table 3.1 correspond to the 5x demand overload experiments.

Link type	traffic type	call arrival process	over-admission percent	standard deviation to conf-adm-rate ratio
T3,100Mbps,OC3	CBR	POISSON	0.5%	0.005
T3,100Mbps,OC3	VBR	POISSON	2.5%	0.025
T3,100Mbps,OC3	CBR	BATCH	1.0%	0.01
T3,100Mbps,OC3	VBR	BATCH	3.0%	0.03
1Gbps	SVD	POISSON	2.0%	0.08
OC12	SVD	POISSON	0.0%	0.1

Table 3.1. Summary of the admission control results for links above T3 speeds. Note: T3 = 45Mbps, OC3 = 155Mbps, OC12 = 622Mbps. Results correspond to 5x overload on a Single Link Topology.

3.3. Sensitivity to Call Arrival Assumptions

In the previous section we reported that at sufficient levels of aggregation Poisson call arrivals assumption was not critical in the sense that even a burstier, batch arrival process resulted in a reasonable performance for all traffic models. In this section we investigate to what extent the Poisson call arrival assumption affect the accuracy of the admission control algorithm at lower levels of bottleneck aggregation.

To that end we first investigated the comparative performance of the algorithm with Poisson and Batch call arrival processes for the CBR and VBR voice traffic. The mean call arrival rate was the same for both processes, with the demand overloads ranging from 2x to 5x. Table 3.2 below summarizes the difference between the admitted load and the configured-admissible-rate for CBR Voice in the case of Poisson and Batch arrivals. Table 3.3 provides a similar summary for on-off traffic simulating voice with silence compression. The results in the tables correspond to the worst case across all overload factors (and when multiple links speeds are listed, across all those link speeds).

Link type	arrival model	over-admission percent	standard deviation to conf-adm-rate ratio
1Mbps, T1	BATCH	30.0%	0.30
10 Mbps	BATCH	5.0%	0.08
T3,100Mbps,OC3	BATCH	1.0%	0.01
1Mbps, T1	POISSON	5.0%	0.10
10 Mbps	POISSON	1.0%	0.02
T3,100Mbps,OC3	POISSON	0.5%	0.005

Table 3.2. Comparison of Poisson and Batch call arrival models for CBR voice. Note: T1 = 1.5Mbps, T3 = 45Mbps, OC3 = 155Mbps, OC12 = 622Mbps. The results are for 5x overload on a Single Link Topology.

Link type	arrival model	over-admission percent	standard deviation to conf-adm-rate ratio
1Mbps, T1	BATCH	40.0%	0.30
10 Mbps	BATCH	8.0%	0.06
T3,100Mbps,OC3	BATCH	3.0%	0.03
1Mbps, T1	POISSON	15.0%	0.20
10 Mbps	POISSON	7.0%	0.06
T3,100Mbps,OC3	POISSON	2.5%	0.025

Table 3.3. Comparison of Poisson and Batch call arrival models for VBR voice with silence compression. Note: T1 = 1.5Mbps, T3 = 45Mbps, OC3 = 155Mbps, OC12 = 622Mbps.

As can be seen, there is substantial sensitivity to Poisson call arrivals at lower bottleneck aggregation levels, but very little performance difference is observed as long as the aggregation levels are sufficiently high.

Subsequently we also investigated sensitivity to Poisson assumption with all other traffic models and other topologies. Due to time limitations, we investigated this only at higher levels of aggregation. Specifically, all voice experiments, including various codecs mixes are run on bottleneck link with OC3 (155 Mbps) bottleneck links, VTR traces are run on 1Gbps and SVD on OC48 (2.4Gbps) links. At these levels of aggregation we have run the experiments on the entire set of topologies and parameter settings reported in this draft, and fond that the performance with BATCH arrivals is very close to that of Poisson arrivals across the entire range of these experiments.

This confirms that BATCH arrivals have little effect on the performance compared to Poisson at sufficient aggregation levels and demand overloads in the studied range.

3.4. Sensitivity to Marking Parameters at the Bottleneck

3.4.1. Ramp vs Step Marking

Draft [I-D.briscoe-tsvwg-cl-architecture] gave an option of "ramp" and "step" marking at the bottleneck. The behavior of the congestion control algorithm in all simulation experiments we performed did not substantially differ depending on whether the marking was "ramp", i.e. whether a separate min-marking-threshold and max-marking-threshold were used, with linear marking probability between these thresholds, or whether the marking was "step" with the min-marking-threshold and max-marking-threshold collapsed at the max- marking-threshold value, and marking all packets with probability 1 above this collapsed threshold. However, the difference between "ramp" and "step" may be more visible in the multiple congestion point case (evaluation of "ramp" vs "step" performance in the multi-bottleneck case remains an area for future work).

Another possible reason for this apparent lack of difference between "ramp" and "step" may relate to the choice of CLE threshold and measurement timescale. Choosing a lower CLE threshold and a faster measurement timescale may result in a better sensitivity to lower levels of marked traffic. Investigating the interaction between settings of the marking thresholds, the CLE-threshold, and the measurement parameters at the egress remains an area of future investigation.

3.4.2. Sensitivity to Virtual Queue Marking Thresholds

The limited number of simulation experiments we performed indicate that the choice of the absolute value of the min- marking-threshold, the max-marking-threshold and the virtual-queue- upper-limit can have

a visible effect on the algorithm performance. Specifically, choosing the min-marking-threshold and the max-marking- threshold too small may cause substantial under-utilization, especially on the slow links. However, at larger values of the min- marking-threshold and the max-marking-threshold, preliminary experiments suggest the algorithm's performance is insensitive to their values. The choice of the virtual-queue-upper-limit affects the amount of over-admission (above the configured-admissible-rate threshold) in some cases, although this effect is not consistent throughout the experiments. The Table 3.4 below gives a summary of the difference between the admitted load and the configured-admissible-rate as a function of the virtual queue parameters, for the SVD traffic model. The results in the table represent the worst case result among the experiments with different degree of demand overloads in the range of 2x-5x. Typically, higher deviation of admitted load from the configured-admissible-rate occurs for the higher degree of demand overload. The sensitivity of smoother CBR and VBR voice traffic models to the variation of these parameters is not as significant as that presented in Table 3.4 for SVD.

Link type	min-threshold, max-threshold, upper-limit(ms)	over-admission percent	standard deviation to conf-adm-rate ratio
1Gbps	5, 15, 20	6.0%	0.08
1Gbps	1, 5, 10	2.0%	0.07
1Gbps	5, 15, 45	2.0%	0.08
OC12	5, 15, 20	5.0%	0.11
OC12	1, 5, 10	2.0%	0.13
OC12	5, 15, 45	0.0%	0.10

Table 3.4. Sensitivity of 4 Mbps on-off SVD traffic to the virtual queue settings. Note: T1 = 1.5Mbps, T3 = 45Mbps, OC3 = 155Mbps, OC12 = 622Mbps

3.5. Sensitivity to RTT

We performed a limited amount of sensitivity analysis of the admission control algorithm used to the range of round trip propagation time (which is the dominant component of the control delay in the typical environment using Pre-congestion notification).

We considered both the case when all flows in a given experiment had the same RTT from this range, and also when RTT of different flows sharing a single bottleneck link in a single experiment had a range of round trip delays between 22 and 220 ms. The results were good for all types of traffic tested, implying that the admission control algorithm is not sensitive to the either the absolute value of the round-trip propagation time or relative value of the round-trip propagation time, at least in the range of values tested. In addition, we found no sign of unfairness to the flows with large RTT. We expect this to remain true for a wider range of round-trip propagation times.

It is important to note that these results relate to the difference in RTT of flows sharing a single bottleneck. One can expect that flows with longer RTT also traverse more bottleneck links. This effect of multiple bottlenecks is studied separately and is reported later in this draft.

3.6. Sensitivity to EWMA weight and CLE

This section represents the results of the investigation the combined effect of the EWMA weight and CLE setting at the egress in three types of settings on:

- a Single Link topology of Fig. 2.1
- RTT topology of Fig. 2.2 with 100 ingress links
- PLT topologies of Fig. 2.3

We experiment with 3 levels of CLE (0.05, 0.15, 0.25) in combination of EWMA weight ranging from 0.1 to 0.9 (in 0.2 step increase). The demand overload is taken to be 5x. For brevity, instead of listing all 15 values (for each combination of weight and CLE), we present the 4-tuple summaries across all experiments.

For PLT topology with N bottlenecks, we have N over-admission-perc. values (each corresponds to one bottleneck link). We show here only the worse case values. That is, in the overload experiments (1-5x), the maximum of the N over-admission-perc is displayed.

The results below are presented for non-randomized traffic models. Randomized versions of all traffic type were tested as well, but no meaningful difference were observed.

The simulation results reveal that for all of the traffic models tested except SVD, the admission control is rather insensitive to the EWMA weight and CLE changes. These statistics show that over-

admission-percentage values are rather similar, with the admitted load staying within -3%+2% range of the desired admission threshold, with quite limited variability.

Type	Topo	Over Admission Perc Stats			
		Min	Max	Mean	SD
CBR	S.Link	0.224	1.105	0.801	0.179
	RTT	0.200	1.192	0.851	0.198
	PLT	-0.93	0.990	0.528	0.559
VBR	S.Link	-0.07	1.646	1.272	0.396
	RTT	-0.11	1.830	1.329	0.434
	PLT	-1.48	1.644	0.798	0.958
MIX	S.Link	-0.14	1.961	1.221	0.606
	RTT	-0.46	1.803	1.171	0.693
	PLT	-1.62	1.031	0.363	0.798
VTR	S.Link	-0.05	1.581	1.055	0.441
	RTT	-0.57	1.313	0.855	0.585
	PLT	-1.24	1.071	0.508	0.739
SVD	S.Link	-2.73	6.525	3.314	3.141
	RTT	-2.98	5.357	2.541	2.618
	PLT	-4.84	4.294	1.229	2.903

Table 3.5 Summarized performance for CBR, VBR, MIX, VTR, SVD across different parameter settings and topologies.

For SVD, the algorithms does show certain sensitivity to parameters, which means that high peak-to-mean ratio SVD traffic is more stressful to the queue-based admission control algorithm, but a set of parameters exists that keeps the over-admission within about -3% - +7% of the expected load.

Note that since the configured-admissible-rate is expected to be set substantially below the actual link capacity, and PCN traffic is typically expected to be served at high priority over non-PCN traffic, 10% overload does not result in any loss as long as the configured-admissible-rate is set below 90% of the link speed. Hence, we treat 10% overload as "reasonable" for practical purposes. A negative overload indicates that less traffic is admitted than the policy threshold would allow, indicating potential underutilization.

3.7. Effect of Ingress-Egress Aggregation

We can assess the effect of Ingress-Egress aggregation on the algorithm by comparing the SingleLink results in Table 3.5 with the corresponding RTT results. As discussed earlier, the actual choice of RTT values of different ingress links does not appear to have any significant effect on the simulation results. We believe that any appreciable difference between the two topologies relates to the degree of aggregation of each ingress-egress pair. One of the outcomes of the results presented in Table 3.5 is that the admission control algorithm of [I-D.briscoe-tsvwg-cl-architecture] seems relatively insensitive to the level of ingress-egress aggregation.

(preamble)

Type	Number of Ingresses and the over-admission perc.					
	2	10	70	300	600	1000
CBR	1.003	1.024	0.976	0.354	-1.45	0.396
	2	10	70	300	600	1800
VBR	1.021	1.117	1.006	0.979	0.721	-0.85
	2	10	70	300	600	1000
MIX	1.080	1.163	1.105	1.042	1.132	1.098
	2	10	70	140	300	600
VTR	1.109	1.053	0.842	0.859	0.856	0.862
	2	10	35	70	140	300
SVD	-0.08	0.009	-0.11	-0.286	-1.56	0.914

(Table 3.6 Ingress aggregation effect. Each cell in the table shows the number of PCN-ingress-nodes generating the flows sharing the bottleneck (top number) and the corresponding over-admission percentage (bottom number). The results correspond to EWMA weight of 0.3, CLE=0.05, demand overload 5x)

Table 3.6 summarizes the effect of ingress-egress aggregation. For each traffic type, the mean number of flows sharing the bottleneck is constant in all experiments. The number of ingresses therefore is inversely proportional to the level of ingress-egress aggregation. As can be seen, the right-most column represents the lowest aggregation level (expected 1 call/ingress), indicating that algorithm is rather insensitive toward the level of ingress-egress aggregation.

These results are very encouraging: while the assumption of

reasonable aggregation of PCN traffic at an internal bottleneck seems a relatively safe one, it is much less clear that it is safe to assume that high per ingress-egress aggregation level is a safe assumption in reality. In particular, the SVD setup with only ~100 SVD flows taking up about 50% of a 1G bottleneck link bandwidth with all 100 flows coming from different ingresses seems entirely plausible. It is therefore encouraging that the algorithm seems sufficiently robust under these circumstances.

3.8. Effect of Multiple Bottlenecks

In this section we report a set of experiments on the multi-bottleneck topology.

3.8.1. Utilization of overloaded bottlenecks

Our first set of experiments (reported in Table 3.5) investigates whether multiple bottlenecks have any effect on the utilization of bottlenecks links all of which contain a mix of flows traversing multiple bottlenecks and small number of bottlenecks (in our case just one bottleneck). We term the former "long-haul" flows, and the latter "short-haul" flows.

In these experiments, we use the PLT topology where the long-haul flows traverse the entire length of the chain, and short-term flows traverse only one hop. The demands of all short- and long-haul flows are the same, and the demand overloads on each bottleneck link in the topology are also the same. We experiment with all sizes of PLT topologies from 2 to 5 and all demand overloads up to 5x, and a range of different parameter (weight and CLE) settings. For each one of them we report the utilization of all the bottleneck links. .

In Table 3.7, we show a snapshot of the behavior of all bottlenecks in a 5 bottleneck topology. Here, the over-admission-perc. displayed for each link is an average across all 15 experiments with different [weight, CLE] setting for a 5x overload. (We do observe very much the same behavior in each of the individual experiment, hence providing summarized results is meaningful). As seen from this table, there appears to be no significant difference in over-admission percentages across the different bottlenecks traversed by the "long-haul"flows in the PLT topologies. Furthermore, there is no visible performance difference in the case of multiple bottleneck topologies (PLT), compared to the case when only a single bottleneck is traversed (as in both SingleLink and RTT topologies) for the same demand overloads and parameters. We observed similiar result all experiments we run.

We ran these experiments for all traffic types, with similar results.

Traffic Type	Bottleneck LinkId 1	2	3	4	5
CBR	0.288	0.286	0.238	0.332	0.306
VBR	0.319	0.420	0.257	0.341	0.254
MIX	0.363	0.394	0.312	0.268	0.205
VTR	0.466	0.309	0.223	0.363	0.317
SVD	0.319	0.420	0.257	0.341	0.254

Table 3.7 Over-admission-percentage for PLT5 for all bottlenecks. The results are for CBR, 5x overload, averaged over all experiments with different parameter settings (there is no significant parameter sensitivity and the results for different settings are very close).

3.8.2. Fairness Between Long-haul and Short-haul flows

Our next set of experiments targeted understanding the effect of multiple bottlenecks on the fairness of sharing the bottleneck links between the long- and the short-haul flows. It is generally known [Jamin, etc] that measurement-based admission control algorithms are susceptible to the effect when long-haul flows get a much smaller share of the bottleneck than short-haul flows.

While the effect of unfairness is well known, the exact cause of it (and possibly the extent) depends on the details of the algorithm. Our first goal was to understand the extent of the unfairness that might occur. Table 3.8 shows the ratio of the bandwidth achieved by the long-haul the short-haul aggregates with respect to the simulation time. As can be seen, the long-haul flow consistently looses bandwidth as a function of time, and this effect is the more pronounced the more bottlenecks are traversed by the long-haul flow.

(preamble)

Topo	Weight	Simulation Time (s)							
		10	20	30	40	50	60	70	80
PLT5	0.1	0.99	1.04	1.14	1.14	1.23	1.23	1.35	1.46
	0.5	1.00	1.17	1.24	1.41	1.81	2.13	2.88	3.05
	0.9	1.03	1.42	1.74	2.14	2.44	2.91	3.83	4.20
PLT3	0.1	1.02	1.08	1.15	1.29	1.33	1.38	1.37	1.42
	0.5	1.02	1.04	1.07	1.19	1.24	1.30	1.34	1.33
	0.9	1.02	1.09	1.23	1.41	1.65	2.10	2.63	3.18
PLT2	0.1	1.02	0.98	1.03	1.11	1.22	1.21	1.25	1.31
	0.5	1.02	1.06	1.14	1.17	1.15	1.31	1.41	1.41
	0.9	1.02	1.04	1.11	1.30	1.56	1.61	1.62	1.67

Table 3.8. Unfairness ratio between long flow aggregate and short flow aggregate in time, for different PLT topologies and different EWMA rates. All results are for 5x overload, CBR traffic.

Table 3.8 indicates that the bandwidth of the long-haul aggregate consistently declines in time, even though its demand remains constant. This effect is frequently referred as the "beatdown". Discouraging as it is, this effect is well known for Measurement-based admission control. The intuition behind the "beatdown" is that the long-haul aggregate can admit new flows only if all the links it transverse are not in the congestion state. Hence comparing to the short-haul aggregate, the long-haul ones see congestion more often, and is in the no-admission state substantially more often as well. If the demand loads of the short-haul and long haul flows are similar, and high enough to monopolize the entire bottleneck bandwidth, the long-haul flow repeatedly loses the competition and stays in the no-admission state most of the time.

It is important to note that Table 3.8 indicates that the bandwidth of the long-haul aggregate consistently declines in time, even though its demand remains constant. In fact, in our simulation runs of about 80 simulation seconds long, we see that for all settings of the parameters and all PLT topologies we see a consistent decline of the share of the long-hauls aggregate. A question then arises on whether this effect continues (with the long-haul aggregate being eventually beaten-down to zero) or whether the long-haul aggregate eventually stabilizes at some (perhaps low) value.

Before attempting to answer this question we note that this effect is well known for Measurement-based admission control. In fact the authors of [Jamin] argue that for sufficiently large demands, in the

limit the long-haul flow is always beaten down completely. In our simulations, however, the demands at which the beatdown effect occurs is not at all infinitely large. We investigate why, even at demand overloads as small as 2X the configured-admissible-rate at the bottleneck, the long-haul aggregate is consistently beaten down. Our analysis indicates that for all parameter settings, the proportion of time at least one of the links in the topology is in the "pre-congestion state", i.e. marking enough packets to trigger no-admission is substantially higher than the percentage of time any one of the links spends in the pre-congestion state, and in many cases it is close to 100% of the time.

It seems clear that the fraction of time the long-haul aggregate on the average sees congestion on at least one of the links is a critical parameter defining whether or not the long-haul flow will be eventually starved completely or not. Clearly if the fraction of time the long-haul sees no congestion on all links along the way is close to 1, then it simply never has a chance to admit new flows, and eventually gets beaten down to zero. On the other hand, if the long-term average fraction of time when it sees no congestion simultaneously on all of the links it traverses stays above some f>0 for any long enough period of time, if the demand (or the rate of calls requesting admission) of the aggregate is constant, then the beatdown effect would never drive the long-haul flow below (call arrival rate) times f times (mean call duration). This can be easily seen by observing that if for the fraction of time f the aggregate is allowed to admit, then it effective long-term call acceptance rate is (call arrival rate) x f. When there are N flows of the aggregate in the system, the mean call departure rate is N/(mean call duration). Therefore, if the number of admitted flows in the aggregate ever reaches or goes below k=(call arrival rate) x f x (mean call duration), the mean call arrival rate will become larger than the mean departure rate, and hence the number of flows will be increasing on the average.

How realistic is it that at least one of the links traversed by the long-haul aggregate is always in congestion/marking state? While we do not know the general answer, we can argue that if congestion states of different links were independent, and each link j is in the state of congestion some fraction of time p(j), then the fraction of time p that a flow traversing n bottlenecks sees congestion on at least one link is p = 1- (1 - p(1)) *(1 - p(2)) * ... *(1 - p(n)). If n is large and/or demands on the bottlenecks are large , this p is close to 1. In our simulations, with 5 bottlenecks and 5x overload, the fraction of time when at least one of the links was marking packets was close to 100%. Of course in general we cannot assume that the "congestion" state in different links is completely independent. Yet, in all our simulations, it appears that even when

the system is tarted in a synchronized state where all the bottlenecks are "congested" at the same time, the system tends to get desynchronized in time, so that congestion periods at different bottlenecks spread in time, and in many experiments with 5-PLT almost all the time at least one of the links was in the congestion state.

We observed consistent beatdown effect across all experiments, although the exact extent of the unfairness depends on the demand overload, topology and parameters settings. To further quantify the effect of these factors remains an area of future work. We also note that the cause of the beatdown effect appears to be largely independent of the specific algorithm, and is likely to be relevant to other PCN proposals as well.

Finally we note that the for the beatdown effect to be significant, not only the demand overload amount should be substantial, but also the duration of the demand overload should be long enough. Under "normal" conditions, one should not expect prolonged substantial overloads. In the exceptional cases where high overloads do occur, they are likely to not be of very large duration. In those cases, unfairness and even starvation of some aggregates is still preferential to indiscriminately dropping packets of all flows that would occur in the absence of admission control. Hence, in practice, the effect of the beatdown effect we report here is probably limited.

4. Termination Control

4.1. Termination Model and Key Parameters

We evaluate the termination algorithm on all the topologies described in Section 2.

In the simulation, the router implementing PCN termination Marking operates as described in [I-D.briscoe-tsvwg-cl-architecture], marking all packets which find no token in the token bucket. In the case of multiple bottlenecks, only previously unmarked traffic is metered against the token bucket. When an egress gateway receives a marked packet from the ingress, it will start measuring its Sustainable-Aggregate-Rate for this ingress, if it is not already in the Termination mode. If a marked packet arrives while the egress is already in the Termination mode, the packet is ignored. The measurement is interval based, with 100ms measurement interval chosen in all simulations. At the end of the measurement interval, the egress sends the measured Sustainable-Aggregate-Rate to the ingress, and leaves the termination mode. When the ingress receives the sustainable rate from the egress, it starts its own interval immediately (unless it is already in a measurement interval), and

measures its sending rate to that egress. Then at the end of that measurement interval, it terminates the necessary amount of traffic. The ingress then leaves the termination mode until the next time it receives the sustainable rate estimate from the egress. In all our simulations the ingress used the same length of the measurement interval as the egress. Token bucket depth was set to 256 packets in all experiments presented here.

We evaluate the performance of the algorithms using a metric called "over-termination-percentage", which is defined as (actual-termination - optimal-termination) expressed in percentage of the optimal termination value. We apply this metric in two contexts: (1) the aggregate amount of terminated traffic on a given bottleneck link, and (2) the aggregate amount of terminated traffic of an ingress-egress traffic aggregate. The former relates to bottleneck utilization, and is quite straightforward: the optimal Termination would terminate all traffic above the configured-termination-rate, so "optimal" Termination is defined only by the configured-termination-rate. For the ingress-egress aggregates, the notion of optimality is closely related to the notion of fairness. In general, fairness can be defined in many different ways, and we do not attempt to argue for one being "more optimal" than the other. In this draft we call the per-ingress-egress Termination amounts optimal if the amount of terminated traffic is distributed among all ingress-egress pairs sharing a bottleneck link in proportion to their rates prior to Termination. For brevity, we omit the details of the definition for the multiple bottleneck case here as it is not central to the discussion in this draft.

4.2. Effect of RTT Difference

Our experiments indicate that absolute value of RTT within the chosen range (up to 220 ms) has no effect on the performance of the termination algorithm, as long as the RTTs of the different ingress-egress pairs are comparable. This section investigates the impact of the relative difference or RTTs of different flows sharing a single bottleneck. We show that in principle, when both short- and long-RTT ingress-egress pairs are present, the difference in RTT may cause over-termination.

To demonstrate that we consider a simple RTT topology with two ingresses, with CBR traffic. Table 4.3 shows the experiment setup and termination results. The overall traffic on the bottleneck during the event is 1761 CBR flows, which constitutes 75% of OC3 link. Ingress 2 has a RTT that around 50ms larger than Ingress 1. The actual termination (termination) and the over-termination percentage are listed for each ingress separately. The results shows that Ingress 1 over-terminates about 10% of its traffic, which

results in about 6% of the overall over-termination at the bottleneck.

Ingress	Bottleneck Eventload	RTT	Actual term	Over-term Perc
1	1178	1ms	0.405	9.59%
2	583	50ms	0.302	-0.51%

Table 4.3. Summary of the RTT difference Results.

Figure 4.3 shows a time vs. load graph that is intended to capture the effect of the flow termination algorithm in this experiment. The X-axis is the time, where a number of important time points are labeled (actual time is listed in table due to lack of space). The Y-axis is the load on the bottleneck link. The stacked graph on the right shows the behavior of each individual ingress. (The shade region is the load contributes to Ingress 1 and the clear region corresponds to Ingress 2). Finally, the dotted line represent the configured-termination-rate.

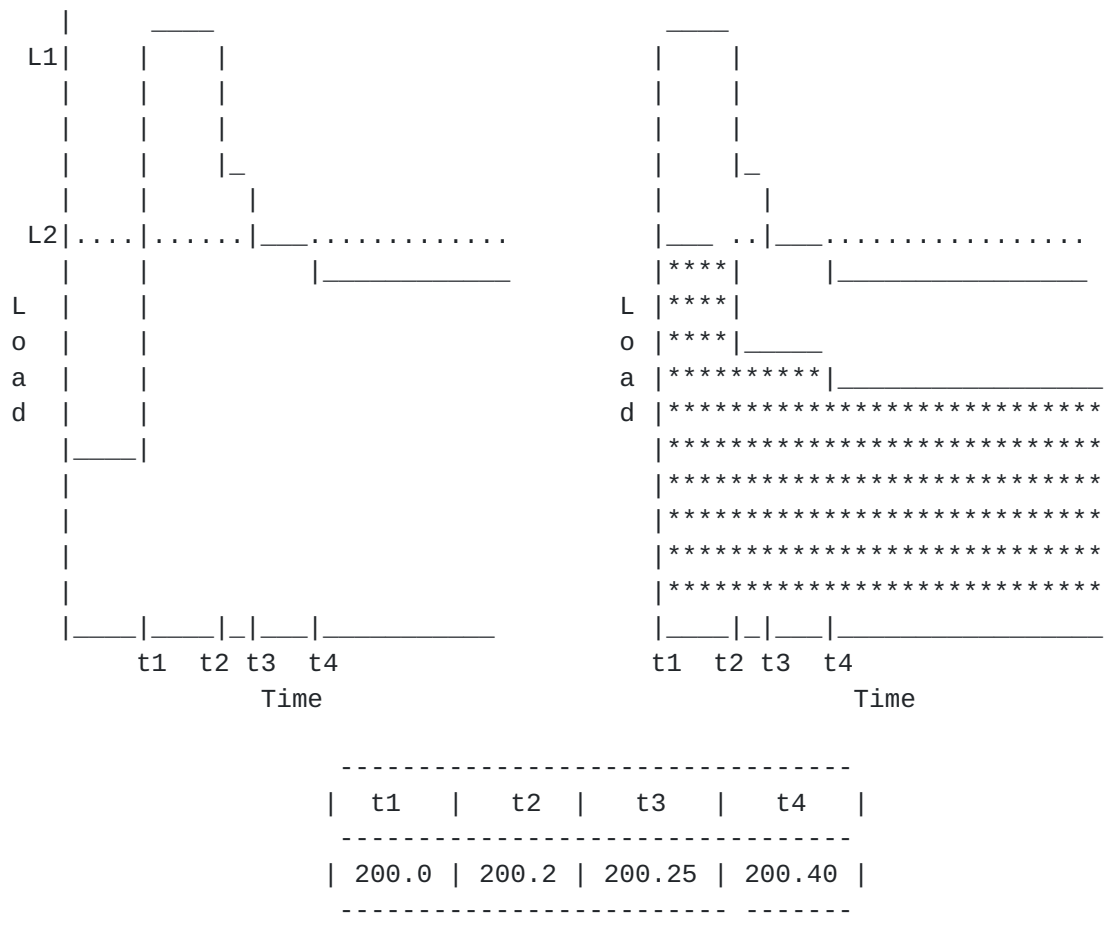

Fig 4.4. Time series of termination events in the RT Difference experiment

As the simulated failure event occur at time t1 (200s), the load on the bottleneck goes over the configured-termination-rate by 1/3, thereby activating the termination algorithm. 200ms afterward at t2, which is sum of the measurements of sustainable rate at the egress (100 ms) and the consequent ingress measurement of its current sending rate, Ingress1 with negligible RTT (1ms) start terminating its traffic. 50ms later at t3, Ingress 2 terminates its share of traffic. Note, at this point, both of ingresses had terminated the correct amount, which is why the load on bottleneck between time t3 and t4 is exactly at the configured-termination-rate. However the stacked graph shows that Ingress1 did another around of termination at t4 (200.4), which corresponds to its 10% over-termination. The reason for this effect is that during the interval between t2 and t3, when Ingress1 finishes its flow terminations, and Ingress2 has not yet started due to its longer RTT, the non-terminated traffic from Ingress2 will cause a further decrement in Ingress1's sustainable rate during the measurement interval (t2, t2+100ms). This will in turn cause Ingress1 to terminate at time t4 to compensate for that 50ms of excess traffic from Ingress2. Our follow-up results indicate

that this RTT effect exists to some degree in every experiment that has sufficient Ingress RTT difference, independent of the traffic type. Although for burstier traffic the over-termination may be worse than shown above, in our experiments we did not see over-termination that would be drastically larger. However, further investigation is needed to access whether other scenarios might lead to more substantial over-termination.

4.3. Ingress-Egress Aggregation Experiments

4.3.1. Motivation for the Investigation

While sufficiently high bottleneck aggregation is listed as one of the underlying assumptions of [I-D.briscoe-tsvwg-cl-architecture], there remains a question of whether of not sufficient degree of aggregation of traffic on a per ingress-egress pair is also necessary. We saw that in our admissions experiments, the virtual-queue-based admission algorithm performed reasonably well even with small ingress-egress aggregation levels, as long as the bottleneck aggregation level was sufficiently high. A similar investigation is performed for the case of termination.

Assuming a large degree of aggregation on a per ingress-egress pair is less attractive, as one can easily imagine that a bottleneck link in a PCN region may carry traffic from hundreds or thousands of ingresses, and there is evidence to believe that in practice cases when per-ingress-egress pair traffic is generated by a relatively small number of flows may not be uncommon. If indeed the number of flows in an ingress-egress pair is small, theoretically there exists a concern that the granularity of termination (which can operate on integer number of flows only) will result in large inaccuracies of the amount of traffic terminated in a per-ingress-egress aggregate, and consequently a large amount of over-termination. As an example of a situation creating this problem suppose that a bottleneck link is shared by 2N flows, each one of them coming from a different ingress-egress pair. Suppose that only N flows can be supported at the configured-termination-rate, so N out of 2N flows must be terminated. This means that half of the packets will get termination marked. If these marked packets are more or less uniformly distributed among the flows sharing the bottleneck, one should expect that every one of the 2N flows will have half of its packets marked. That in turn would imply that each ingress would need to terminate half of its traffic, and since it only has one flow, it would have to terminate that flow (assuming that the number of flows to terminate is rounded up to the nearest flow) or not terminate any flow at all (if the rounding down to the nearest flow is done). In either case the outcome is quite pessimistic- either all flows are terminated, or the termination will not take any effect at all. Clearly, a similar

(although perhaps less drastic) effect would be if a few flows rather than one constitute an ingress-egress pair. The effect quickly disappears when the rate of an individual flow is sufficiently small compared to the total rate of the ingress-egress aggregate.

While a number of possible changes to the ingress behavior could be considered to solve or alleviate this problem, we set out to investigate whether this problem does in fact occur in practice. The key question in that respect is whether or not the packets do indeed get marked more or less uniformly among different flows sharing a bottleneck over the timescale of the ingress and egress measurement intervals. The results of this investigation are presented in the following subsections.

4.3.2. Detailed results

To investigate the effect of small ingress-egress aggregation, we first performed the experiments with three traffic types (CBR, VBR and SVD) at different degrees of ingress aggregation. All the experiments in this section are carried out on RTT topology; the different ingress aggregation levels are obtained by varying the number of ingress links in the topology. All links' RTT are set to 1ms (to eliminate the potential RTT influence). CBR and VBR voice used an OC3 bottleneck link while SVD used an OC48 link, with configured-termination-rate set at 50% of the link bandwidth in all cases. The bottleneck aggregation was therefore quite high (with respect to the corresponding link bandwidth), but the ingress-egress aggregation was varied from 1 flow to about 1/3 of the number of flows at the bottleneck in each ingress-egress pair. The results are summarized in Table 4.1 below.

Traffic Model	BtleNeck Load	Number Ingrs.	Flows per Ingress	term Threshold	Actual term	Over-term Perc
CBR	1789	2	582		0.321	0.05%
CBR	1772	70	9	1215	0.328	1.41%
CBR	1782	600	1		0.336	1.85%
VBR	5336	2	1759		0.333	0.35%
VBR	5382	70	26	3574	0.364	2.84%
VBR	5405	1800	1		0.368	2.99%
SVD	450	2	135		0.404	8.07%
SVD	446	70	2	305	0.414	9.64%
SVD	452	140	1		0.406	8.02%

Table 4.1 Effect of ingress-egress aggregation.

In this table, bottleneck load at failure is represented as the number of flows at the bottleneck after the simulated failure event has occurred and before the termination takes place. The "Number Ingress" column shows the number of ingresses in the RTT topology.

In all cases, ideally, the algorithm should terminate roughly 1/3 of the traffic after the failure event has occurred (the exact percentage differs slightly from experiment to experiment due to some variability of load generation implementation). The second to last column shows the actual termination percentage in each experiment, and the last column shows how far it deviates from the optimal value in terms of over-termination percentage (where the optimal value is computed based on the actual traffic generated in each experiment).

The first conclusion that can be drawn from Table 4.1 is that in these experiments flow termination worked quite well for CBR and VBR, and even in the SVD case with just 1 flows per ingress the over-termination is quite bounded.

The second - far more unexpected - outcome of these results is that for all traffic types in these experiments the result show no appreciable effect of the ingress aggregation on the degree of ingress aggregation, as all the over-termination percentage do not differ significantly. Given the discussion in the previous section that predicted substantial inaccuracy of flow termination in the case of a small number of flows per ingress, this result appears both unexpected and encouraging, but does require explanation and discussion.

Further analysis of the simulation traces of CBR traffic of

experiments of Table 4.1 identified the cause of this phenomenon. It turned out that in all the simulation runs with CBR traffic, contrary to our expectation that termination marking will be more or less uniformly distributed among active flows, what actually happens is that some flows get all their packets marked, while other flows get no packets marked at all (we refer to this effect loosely as "synchronization" in the rest of this document). It is this phenomenon that, in the case of a single flow per ingress, made only the ingresses whose flows were marked terminate these flows, resulting in correct amount of termination. Further analysis showed that in fact this effect is not a simulation artifact, and is a direct consequence of periodicity of individual CBR flows in combination with incidental choice of several parameters.

As it happens, if the number of tokens arriving in the token bucket in an inter-packet interval of a single CBR flow is an integer multiple of a packet size, then if a packet of a flow is marked once, all the subsequent packets will find the same number of tokens in the token bucket and will also be marked. The proof of this fact is provided in the companion technical report. It seems clear that in general this synchronization cannot be relied upon, and we expected that for the VBR case we will see much less of it.

Again, we were in for a surprise, as trace investigation of our initial results reported in Table 4.1 revealed that even though the token bucket state encountered by the packets of the same VBR flow was not quite the same, it was close enough so that again a large number of flows were either fully marked or fully unmarked. We realized that the reason for that is that the number of flows which are in the on-period during the relevant measurement intervals is relatively stable, and hence much of the effects observed for the CBR flows approximately holds for the on-off traffic we use for our VBR model. Since the on- period had the same rate as our CBR model, and the packet size was the same for the two models, similar behavior was observed in both sets of experiments.

In our quest to further understand the unexpectedly reasonable performance at small ingress-egress aggregation we then tested the hypothesis that randomizing the packet inter-arrival time must surely break synchronization, and to that end we rerun the same set of experiments on the randomization version of all traffic. The results are summarized in Table 4.2 Note, the column label with f (e.g. 0.0001) correspond to randomized traffic with a randomization-interval of f x packet-inter-arrival-time. It also means that on average, the packets are delayed by f x packet-inter-arrival-time / 2.

	No. Ingre	No-Rand	Deviation Interval 0.0001	0.001	0.005	0.01	0.1
CBR	2	0.050	0.390	1.047	0.224	0.757	1.072
	10	0.495	0.771	0.769	1.016	0.975	0.819
	35	1.157	1.615	1.817	2.448	1.938	2.300
	70	0.841	2.428	3.693	3.098	3.710	3.528
	140	1.577	3.089	4.962	5.271	5.516	5.376
	300	1.371	2.965	6.852	9.303	9.761	9.790
	600	1.069	3.563	9.449	12.17	13.41	13.97
VBR	2	2.663	3.002	2.350	2.320	2.240	2.217
	10	1.856	3.629	3.369	1.460	2.493	3.712
	35	2.831	3.385	3.931	4.128	4.918	4.490
	100	3.952	6.257	5.018	5.732	5.099	5.568
	300	2.421	4.846	5.435	5.651	5.339	6.021
	600	2.518	1.815	3.447	4.333	4.361	4.856
	1800	2.500	0.435	2.248	2.727	1.698	2.077
MIX	2	1.173	1.863	1.185	1.341	1.736	1.164
	10	2.377	1.579	1.207	2.656	2.321	3.097
	35	2.914	3.484	2.232	2.665	2.642	3.874
	140	4.066	3.009	4.600	3.221	4.263	4.907
	300	2.806	5.436	4.088	4.113	4.895	5.602
	600	1.995	0.527	3.295	2.982	4.851	4.424
	1000	0.733	1.076	1.729	3.022	3.955	2.731
VTR	2	2.544	2.610	1.217	3.055	2.889	1.906
	10	3.741	4.329	4.112	3.936	4.328	4.348
	35	7.524	6.549	6.629	7.014	7.644	6.610
	70	7.607	7.541	8.217	7.002	7.916	8.113
	140	11.16	9.162	12.50	10.74	10.49	10.44
	300	12.15	14.41	13.57	14.32	15.62	16.35
SVD	2	8.071	10.62	8.235	10.22	10.62	8.531
	10	10.66	12.13	11.16	10.81	11.35	10.61
	35	10.69	10.14	11.86	13.52	9.306	9.900
	70	9.645	8.845	5.917	9.716	7.803	10.57
	140	8.025	9.777	9.690	8.949	6.008	10.63

Table 4.2 Effect of ingress-egress aggregation v.s. deviation. The table entries correspond to the over-termination-percentage at different aggregation levels, different randomization interval, for different traffic types.

As can be seen, the over-termination-percentage shown in Table 4.2

exhibits different trends depending on the traffic types. First for CBR, as we expected, the "randomization" indeed breaks in the synchronization, so that at low aggregation, we observe substantially more over-termination (~14%), confirming our expectation that the unexpectedly good performance cannot be expected in general for low ingress-egress aggregation levels. On the other hand, it also shows that at least a certain amount of randomization is required to break the "synchronization". For instance, with a randomization interval of the 0.0001 x packet-inter-arrival-time, no substantial increment in over-termination is observation. From this aspect, the "synchronization" effect can not be merely regarded as a simulation artifact. A final note is that given sufficient amount of aggregation (~10call/ingress or above), the difference caused by synchronization goes away.

VBR shows a different trend. It seems that given enough randomization, the effect of aggregation (over-termination) starts to emerge at the transition from medium to low aggregation level (around 100 or 300 ingress in the graph). However the effect then diminishes, so that at the lowest aggregation (expected 1 flow per ingress), we no longer observe appreciable over-termination. We believe the reason for this outcome is the following: at medium aggregation levels, even though there are a few flows per ingress, it's not enough to smooth out the burstiness in the aggregated flows. This causes each ingress-egress-pair to over-terminate a little due to occasional under-estimation of the Sustainable-Aggregate-Rate, which results the net over-termination at the bottleneck link. At the low aggregation level (with 1 or 2 flow per ingress), each VBR flow spends a large portion of its time in off-period. Once the aggregate of an ingress-egress pair is in its off period, it will send no packets, get no marking, hence will not react to the termination algorithms. Since a substantial portion of the ingress-egress aggregates can be in the off-period, only those ingresses that are in the on-period terminate their traffic.

The MIX essentially shows the added up effect of both CBR and VBR, in the sense that it shows a clear increasing of over-termination-percentage as the level of aggregation decreases, yet at the lowest aggregation, instead of having the highest over-termination (like CBR), the on-off effect of VBR dominates, hence we again don't see a significant over-termination

For TRC, a clear aggregation effect is observed, but the trends seems to be irrelevant to the degree of randomization. In fact, the result for TRC looks like a complete randomized version of CBR. We hypothesize this is indeed the case, since trace is implemented as constant-frame-rate, that's why it doesn't exhibit what appears in the VBR (namely the on-off effect), in addition, different frame

size, and packetization provide enough randomization.

The trace analysis of the SVD experiments indicates that there are a large number of of partially marked flows, which indicates that synchronization could not have been responsible for the relatively bounded over-termination of about 10% Table 4.2. We believe this performance should be traced to the burstiness of our crude SVD traffic model at the time scales commensurate with the measurement period. In addition, just as for VBR, the on-off nature of the model dominates at low aggregation, which can also be used to explain why no aggregation effect is observed. In summary, the over-termination can be expected at low aggregation for a variety of traffic, but in practice the degree of this over-termination is not as bad as the worst-case analysis might indicate. The over-termination vanishes as the level of ingress-egress aggregation becomes sufficiently large.

4.4. Multiple Bottlenecks Experiments

4.4.1. Motivation for the Investigation

In this section, we focus our analysis on the multi-bottleneck effect. That is, how would termination algorithm perform when the flows from one (or more) ingress-egress pairs traverse multiple bottleneck links. For the rest of section, we use the term "IE-aggregate" (IEA for short) to refer to the flow aggregates of a certain ingress-egress pair. In theory, we expect the IE-aggregate that travel more bottlenecks will be penalized more, which would result in over-termination on a per-ingress-egress basis. We refer to this as a "beat-down" effect. The main consequence of the beat-down effect is the excessive termination at the up-stream bottlenecks, leading to underutilization of those bottlenecks

To illustrate the beat-down effect, consider the setup with 2 bottleneck PLT in Figure 2.3(a). Recall the two bottlenecks are links A - B and B - C. Both links have the same capacity. There are two short IE-aggregates, one from Ingress D to Egress E (IEA2); the other from Ingress E to Egress F (IEA3); each traversing a single bottleneck. At the time of the failure event, each short IEA carries the traffic load that equals 1/4 of the bottleneck link size (or 1/2 of the configured-termination-rate, which in this case is set to 50% of the link bandwidth). The long IE-aggregate (IEA1), from Ingress A to Egress C, traverses both of bottlenecks and carries twice as much traffic as the short ones.

Given that we set the configured-termination-rate to be 1/2 of link size, it's easy to see that letting all IEAs terminate 1/3 of their flows will give the optimal results (which we refer to as "optimal-termination") in the sense that all bottleneck links will be fully

utilized. However, what we expect to happen is the following. When the long IE-aggregate (IEA1) traverses through the first bottleneck link, assuming uniformly random marking, 1/3 of its traffic will get termination-marked. (The short IEA2 will also get 1/3 of its traffic marked). Next, 2/3 of IEA1's unmarked traffic together with IEA3's traffic will result a load of (2/3)*(1/2) + 1/4 = 7/12 on the second bottleneck. This implies that for the aggregate IEA1, an additional (7/12-1/2) / (7/12) = 1/7 percentage of remaining unmarked traffic will be marked. And for IEA3, only 1/7 (instead of 1/3) of its traffic will be marked. To summarize, a beat-down effect in this simple setting means we should see the following termination behaviors:

- EA1 : 1/3 + 2/3 * 1/7 = 3/7 > 1/3
- IEA2 : 1/3
- IEA3 : 1/7 < 1/3
- Bottleneck1 : (3/7 * 1/2 + 1/3 * 1/4) / (3/4) = 25/63 > 1/3
- Bottleneck2 : (3/7 * 1/2 + 1/7 * 1/4) / (3/4) = 1/3

We refer to the above values as "expected-termination". In general, the more bottlenecks an IEA traverses, the more over-termination occurs at both the long IEA and the upstream bottlenecks.

The goal of our experiments was to validate to what extent the beat-down effect is visible in practice, and how much underutilization on up-stream links will actually be seen. To that end, we used 2, 3 and 5 PLT topologies with various traffic types. We are interested in whether the actual-termination exhibits the multi-bottleneck effect comparing to the optimal-termination, and also how much does the actual-termination deviate from our expected-termination. The results of this investigation are presented in the following subsections.

4.4.2. Detailed Results

For the first set of experiments, we use the similar setup as the example described in last subsection. That is, at failure event time, all bottleneck links have a load of roughly 3/4 of its link size. In addition, the long IEA constitutes 2/3 of this load, while the short one is 1/3. Table 4.7 shows the sample output for the multi-bottleneck experiments (In this case, it's with CBR traffic and 5 PLT topology). The first row (labeled IEA1) represents the long IE-Aggregate that travels multiple bottlenecks (the exact count of the bottlenecks is given in the parenthesis after the IEA's name).

The rest of IEA rows are the short IE-Aggregates that each travels only one bottleneck. The IEA rows are ordered based on the bottleneck it traverses (from upstream to downstream). The same information is shown for both IEAs and bottlenecks. The last two columns are of most interests in that they shows the how far the actual-termination deviates from the optimal, and from the expectation.

		Optimal term	Expected term	Actual term	A - O	A - E
	IEA1 (5H)	0.3090	0.4432	0.4446	13.56	0.14
	IEA1 (5H)	0.3090	0.3090	0.3231	1.42	1.42
	IEA1 (5H)	0.3034	0.1181	0.1601	-14.33	4.20
5	IEA1 (5H)	0.3048	0.0541	0.0947	-21.01	4.07
	IEA1 (5H)	0.3073	0.0293	0.0641	-24.32	3.48
B	IEA1 (5H)	0.3031	0.0049	0.0307	-27.24	2.57
R	BN1	0.3090	0.3995	0.4051	9.61	0.56
	BN2	0.3034	0.3392	0.3536	5.02	1.44
	BN3	0.3048	0.3182	0.3322	2.73	1.40
	BN4	0.3073	0.3092	0.3214	1.41	1.22
	BN5	0.3031	0.3031	0.3123	0.92	0.92

Table 4.7 Over-termination percentage with 5-PLT topology and CBR

The following Table 4.8 summarizes the main results for multi-bottleneck experiments. For each combination of the traffic type and PLT topology, it shows (actual-termination - optimal-termination)*100% (labeled as 'A-O') and (actual-termination - expect-termination)*100% (labeled as 'A-E').

		CBR		VBR		VTR		SVD	
		A-O	A-E	A-O	A-E	A-O	A-E	A-O	A-E
	IEA1(2H)	7.61	-0.71	10.36	1.06	9.19	1.26	16.07	8.55
2	IEA2(1H)	0.85	0.85	0.86	0.86	3.17	3.17	7.30	7.30
P	IEA3(1H)	-14.4	4.07	-12.39	6.42	-10.27	7.26	-1.74	13.81
L	BN1	5.45	-0.21	7.20	1.00	7.24	1.88	13.9	8.15
T	BN2	0.80	0.80	2.84	2.84	3.18	3.18	10.26	10.26
	IEA1(3H)	10.8	-0.85	13.98	1.18	11.90	0.87	19.53	9.37
3	IEA2(1H)	0.78	0.78	1.03	1.03	3.35	3.35	5.06	5.06
	IEA3(1H)	-14.09	3.98	-14.07	4.79	-10.45	6.78	-2.65	13.63
P	IEA4(1H)	-21.17	3.96	-18.94	7.38	-16.88	7.18	-6.09	16.50
L	BN1	7.9	-0.33	9.67	1.13	9.43	1.65	14.84	7.97
T	BN2	2.82	0.71	4.77	2.38	4.80	2.75	12.43	10.75
	BN3	0.9	0.69	3.23	3.23	2.87	2.87	11.65	11.65
	IEA1(5H)	13.56	0.14	16.30	0.91	14.77	1.82	23.31	11.37
	IEA2(1H)	1.42	1.42	2.17	2.17	3.20	3.20	7.26	7.26
	IEA3(1H)	-14.33	4.20	-13.65	5.35	-11.71	6.55	-8.05	8.44
	IEA4(1H)	-21.03	4.07	-21.68	5.19	-18.01	6.41	-12.31	9.68
5	IEA5(1H)	-24.32	3.48	-24.04	5.71	-21.39	5.74	-15.69	8.44
	IEA6(1H)	-27.24	2.57	-24.69	4.57	-23.20	5.20	-15.31	9.78
P	BN1	9.61	0.56	11.59	1.33	11.06	2.26	18.13	10.04
L	BN2	5.02	1.44	6.53	2.38	6.91	3.30	13.86	10.44
T	BN3	2.73	1.40	4.01	2.33	4.73	3.27	12.42	10.83
	BN4	1.41	1.22	3.12	2.50	3.54	3.06	11.08	10.43
	BN5	0.92	0.92	2.13	2.13	2.89	2.89	10.85	10.85

Table 4.8 Summary of the PLT results for 2;1 long-to-short load ratio.

It's clear from the 'A-O' results that the beat-down effect is visible across all PLT topologies and traffic types. For instance, for the long IE-aggregate (IEA1), as it travels 2, 3, 5 bottlenecks, the degree of over-termination increases (7.61, 10.85, 13.56 respectively for CBR traffic); so does the most upstream bottleneck link (BN1). Furthermore, all the downstream short IEAs (IEA3 and above) have experienced under-termination compared to their "optimal" value, while the long IEA terminated more than the "optimal" value.

Next we compare the actual-termination with the level of termination predicted by the theoretical beat-down effect based on the assumption of uniformly random marking. Our experience reported in the previous section shows that the assumption of uniform marking may not always hold in the case of bursty traffic.

As seen from Table 4.8, the results for CBR, VBR and VTR are reasonably close to those predicted by the beat-down effect (within 1% for CBR and within 3% for VBR and VTR). The larger discrepancy between the expected and the actual results for SVD are most likely the consequence of the same burstiness effect that we observed in the previous section with respect to ingress-egress aggregation experiments.

Recall that in all of above experiments, we had the long IE-aggregate carries the traffic twice as much as the short ones. Now we investigate what will happen if this load ratio changes. We can use the same method (as the one illustrated in the last subsection) to obtain the expected-termination for any given PLT topology. The expected trend is that, keeping all other conditions the same, the smaller portion the long IEA is, the more relative unfairness towards it (percentage-wise) will be displayed. In following set of experiments we chose the 1:1 as the load ratio (instead of 2:1) of the long and short aggregates, while keeping other the settings unchanged. The results, (actual-termination - optimal-termination)*100%, are summarized in Table 4.9.

		CBR		VTR	
		2:1	1:1	2:1	1:1
	IEA1(2H)	7.61	10.74	9.19	12.50
2	IEA2(1H)	0.85	0.77	3.17	2.18
P	IEA3(1H)	-14.49	-9.71	-10.27	-7.23
L	BN1	5.45	5.75	7.24	7.44
T	BN2	0.80	0.84	3.18	2.85
	IEA1(3H)	10.85	16.83	11.90	18.36
3	IEA2(1H)	0.78	0.77	3.35	2.69
	IEA3(1H)	-14.09	-10.48	-10.45	-7.24
P	IEA4(1H)	-21.17	-15.98	-16.88	-12.19
L	BN1	7.59	8.93	9.43	11.05
T	BN2	2.82	3.81	4.80	5.78
	BN3	0.69	1.10	2.87	3.34
	IEA1(5H)	13.56	23.23	14.77	24.78
	IEA2(1H)	1.42	1.06	3.20	2.06
	IEA3(1H)	-14.33	-9.98	-11.71	-6.19
	IEA4(1H)	-21.01	-16.15	-18.01	-13.35
5	IEA5(1H)	-24.32	-20.17	-21.39	-15.47
	IEA6(1H)	-27.24	-23.06	-23.30	-16.86
P	BN1	9.61	12.47	11.06	13.84
L	BN2	5.02	6.97	6.91	9.78
T	BN3	2.73	3.94	4.73	6.00
	BN4	1.41	1.91	3.54	4.65
	BN5	0.92	.70	2.89	3.77

Table 4.9 Summary of the PLT results for 1:1 long-to-short load ratio.

The results confirm our expected behavior. For instance, the row that gives the over-termination of the IEA1 that goes through 3 bottleneck links shows that in the 1:1 ratio setup, the over-termination of the long aggregate is much larger comparing to 2:1 setup. And the problem grows severely when the number of bottleneck link increases (see IEA1 (5H)). Furthermore, the increment in over-termination of the long IEA also reflects on the bottleneck link, that is, the aggregated over-termination perc. on the bottleneck link increases accordingly. The 'A-E' part of the results is very similar to the ones in Table 4.5. That is, for CBR, VBR, VTR, we have the actual-termination close to expectation.

A high-level conclusion of the results presented in this section is that the actual results confirm the predicted beat-down effect

closely with CBR, VBR and VTR traffic. For SVD, the additional over-termination at the bottleneck links is consistent with the effect of burstiness of this on-off traffic with high peak-to-mean ratio seen in other experiments.

4.5. Sensitivity to Call Arrival Assumptions

In this section we investigate to what extent the Poisson call arrival assumption affect the accuracy of the termination control algorithm. To that end we investigated the comparative performance of the algorithm with Poisson and BATCH call arrival processes for the all traffic. The mean call arrival rate was the same for both processes, with a batch mean equal to 5.

With sufficient level ingress-egress aggregation, the BATCH arrivals experiments give very similar results to the ones with Poisson arrivals. However, contrary to what we expected, in the case of low ingress-egress aggregation, the BATCH model actually performs better. This is simply because the combination of the BATCH arrival and low aggregation makes the traffic aggregate more "on/off"- like. Hence the same reason that VBR and SVD is not affected by the low aggregation (discussed in Section 4.3), can be applied here.

5. Summary of Results

The study presented here demonstrated that overall, both admission control and termination algorithms of [I-D.briscoe-tsvwg-cl-architecture] work reasonably well and are relatively insensitive to parameter variations.

We can summarize the conclusions of the study so far as follows.

5.1. Summary of Admission Control Results

- We observed no significant benefit of using "ramp" making instead of a simpler "step" marking.

- There appears to be no appreciable sensitivity of the admission algorithm to either the absolute value of the round-trip time or the relative value of the round-trip time between different flows.

- As a rule of thumb, the level of bottleneck aggregation necessary to demonstrate tolerable performance even in the simplest network topology corresponds to links of about 10 Mbps or higher for voice traffic (CBR of VBR with silence compression), assuming at least 50% of the link speed is allocated to the PCN traffic. For higher rate bursty SVD flows, 50% of the OC48 of higher appears to be a

reasonable rule of thumb. The higher the degree of bottleneck aggregation, the better the performance.

- Even though larger per ingress-egress pair aggregation results in better performance of admission control algorithm, performance remains reasonable even for really low ingress-egress aggregation levels (i.e. a single or a small number of bursty SVD flow per ingress).

- Poisson call arrival has a visible effect on performance at lower levels of aggregation (10 Mbps for voice or lower), but is of less significance at the higher levels of aggregation/link speeds

- The algorithm is relatively insensitive to variation of key parameter settings at the internal node or the ingress of the PCN domain, as long as the variations are kept within a reasonable range around "sensible" parameter settings.

- As expected, synthetic video traffic SVD was the most challenging for all topologies, and the performance of real video traces (VTR) was substantially better. Even for the SVD, however, a range of parameters exist for which performance across all experiments considered is within reasonable bounds

- The algorithms is relatively insensitive to the level of ingress-egress aggregation

- No performance degradation is observed on the bottleneck link. in a multi-bottleneck topology where some flows traverse multiple bottlenecks in the presence of cross-traffic on each of the bottleneck links. However, the algorithm suffers from the well-known phenomenon of unfairness towards flows traversing multiple bottlenecks.

5.2. Summary and Discussion of Termination Results

The simulations results presented in this installment of the simulation study further demonstrated that at least in a simple one-bottleneck topology case the termination mechanism of works reasonably well for a wide range of parameters for all traffic models we considered.

The key thrust of this study was the investigation of how much ingress-egress aggregation is needed for tolerable performance of the algorithm (assuming sufficient degree of bottleneck aggregation). We demonstrated that contrary to our expectations, it was not easy to find cases with sufficiently bad performance. We traced some of this better-than-expected performance to the effect of synchronization of

the token bucket state for certain combinations of parameter values, and we demonstrated this effect can not be simpliy regarded as simulation artifact. A question of whether this synchronization can be explored to the benefit of the general operation for voice-only PCN regions remains open, but seems of substantial interest. Further investigation with other codices and in a broader set of network conditions is warranted to address this question.

Our experiments demonstrated that the absolute value of RTT of the flows sharing the same bottleneck did not have any appreciable effect as long as the RTT of all flows were the same (or close). However, we have demonstrated that if RTTs of different flows are substantially different, longer RTT flows tend to over-terminate, resulting in overall over-termination as well.

In the multi-bottleneck case, the "beatdown" of long-haul discussed in the context of admission, cause a certain degree of over-termination. In addition, unlike the case of admission when under-admission of long-haul aggregates was compensated by the over-admission of the short-haul aggregates keeping the bottlenecks utilized, in the case of termination, any over-termination of long-haul aggregates is likely to result in under-utilization of some bottleneck links.

On the bright side, at least in the experiments we conducted, the magnitude of the over-termination was relatively small.

6. Future work

This draft is but an intermediate step in the investigation of performance of Admission and termination approaches for a PCN domain. Many of the aspects of the real networks have not been addressed due to time and resource limitations. Those are subject of on-going investigation. Some of these are listed below.

- more realistic signaling model
- Investigation of comparative ramp vs step performance for multiple bottleneck case
- Investigation of comparative ramp vs step performance at low CLE values
- More general multi-bottleneck topologies

7. IANA Considerations

This document places no requests on IANA.

8. Security Considerations

There are no new security issues or considerations introduced by this document.

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

9.2. Informative References

[I-D.briscoe-tsvwg-cl-architecture]
Briscoe, B., "An edge-to-edge Deployment Model for Pre-Congestion Notification: Admission Control over a DiffServ Region", draft-briscoe-tsvwg-cl-architecture-04 (work in progress), October 2006.

[I-D.briscoe-tsvwg-cl-phb]
Briscoe, B., "Pre-Congestion Notification marking", draft-briscoe-tsvwg-cl-phb-03 (work in progress), October 2006.

[I-D.briscoe-tsvwg-re-ecn-border-cheat]
Briscoe, B., "Emulating Border Flow Policing using Re-ECN on Bulk Data", draft-briscoe-tsvwg-re-ecn-border-cheat-01 (work in progress), June 2006.

[I-D.briscoe-tsvwg-re-ecn-tcp]
Briscoe, B., "Re-ECN: Adding Accountability for Causing Congestion to TCP/IP", draft-briscoe-tsvwg-re-ecn-tcp-03 (work in progress), October 2006.

[I-D.davie-ecn-mpls]
Davie, B., "Explicit Congestion Marking in MPLS", draft-davie-ecn-mpls-01 (work in progress), October 2006.

[I-D.eardley-pcn-architecture]
Eardley, P., "Pre-Congestion Notification Architecture", draft-eardley-pcn-architecture-00 (work in progress),

June 2007.

[I-D.lefaucheur-emergency-rsvp]
Faucheur, F., "RSVP Extensions for Emergency Services", draft-lefaucheur-emergency-rsvp-02 (work in progress), June 2006.

Authors' Addresses

Xinyang (Joy) Zhang
Cisco Systems, Inc. and Cornell University
1414 Mass. Ave.
Boxborough, MA 01719
USA

Email: joyzhang@cisco.com

Anna Charny
Cisco Systems, Inc.
1414 Mass. Ave.
Boxborough, MA 01719
USA

Email: acharny@cisco.com

Vassilis Liatsos
Cisco Systems, Inc.
1414 Mass. Ave.
Boxborough, MA 01719
USA

Email: vliatsos@cisco.com

Francois Le Faucheur
Cisco Systems, Inc.
Village d'Entreprise Green Side-Batiment T3, 400 Avenue de Roumanille
06410 Biot Sophia-Antipolis,
France

Email: flefauch@cisco.com

Full Copyright Statement

Intellectual Property

Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).