Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 21, 2008

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

November 18, 2007

Pre-Congestion Notification Using Single Marking for Admission and Termination

draft-charny-pcn-single-marking-03.txt

Status of this Memo

Copyright Notice

Abstract

Pre-Congestion Notification described in [I-D.eardley-pcn-architecture] and earlier in

[I-D.briscoe-tsvwg-cl-architecture] approach proposes the use of an Admission Control mechanism to limit the amount of real-time PCN traffic to a configured level during the normal operating conditions, and the use of a Flow Termination mechanism 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 [I-D.eardley-pcn-architecture], Admission and Flow Termination use two different markings and two different metering mechanisms in the internal nodes of the PCN region. This draft proposes a mechanism using a single marking and metering for both Admission and Flow Termination, and presents an analysis of the tradeoffs. A side-effect of this proposal is that a different marking and metering Admission mechanism than that proposed in [I-D.eardley-pcn-architecture] may be also feasible, and may result in a number of benefits. In addition, this draft proposes a migration path for incremental deployment of this approach as an intermediate step to the dual-marking approach.

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

1.1. Changes from -02 version

- Added Flow Termination results (Section 7 and Section 8.3)
- Minor other edits
- Alignment with draft-charny-pcn-comparison

1.2. Terminology

This draft uses the terminology defined in [I-D.eardley-pcn-architecture]

1.3. Background and Motivation

Pre-Congestion Notification [I-D.eardley-pcn-architecture] approach proposes to use an Admission Control mechanism to limit the amount of real-time PCN traffic to a configured level during the normal operating conditions, and to use a Flow Termination mechanism 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 [I-D.eardley-pcn-architecture], Admission and Flow Termination use two different markings and two different metering mechanisms in the internal nodes of the PCN region. Admission Control algorithms for variable-rate real-time traffic such as video have traditionally been based on the observation of the queue length, and hence re-using these techniques and ideas in the context of pre-congestion notification is highly attractive, and motivated the threshold- and ramp- marking and metering techniques based on the virtual queue implementation described in [I-D.briscoe-tsvwg-cl-architecture] for Admission. On the other hand, for Flow Termination, it is desirable to know how many flows need to be terminated, and that in turn motivates excess-rate-based Flow Termination metering. This provides some motivation for employing different metering algorithm for Admission and for Flow Termination.

Furthermore, it is frequently desirable to trigger Flow Termination at a substantially higher traffic level than the level at which no new flows are to be admitted. There are multiple reasons for the requirement to enforce a different configured-admissible-rate and configured-termination-rate. These include, for example:

- End-users are typically more annoyed by their established call dying than by getting a busy tone at call establishment. Hence

decisions to terminate flows may need to be done at a higher load level than the decision to stop admitting.

- There are often very tight (possibly legal) obligations on network operators to not drop established calls.

- Voice Call Routing often has the ability to route/establish the call on another network (e.g., PSTN) if it is determined at call establishment that one network (e.g., packet network) can not accept the call. Therefore, not admitting a call on the packet network at initial establishment may not impact the end-user. In contrast, it is usually not possible to reroute an established call onto another network mid-call. This means that call Termination can not be hidden to the end-user.

- Flow Termination is typically useful in failure situations where some loads get rerouted thereby increasing the load on remaining links. Because the failure may only be temporary, the operator may be ready to tolerate a small degradation during the interim failure period. This also argues for a higher configured-termination-rate than configured-admissible-rate

- A congestion notification based Admission scheme has some inherent inaccuracies because of its reactive nature and thus may potentially over admit in some situations (such as burst of calls arrival). If the Flow Termination scheme reacted at the same rate threshold as the Admission , calls may get routinely dropped after establishment because of over admission, even under steady state conditions.

These considerations argue for metering for Admission and Flow Termination at different traffic levels and hence, implicitly, for different markings and metering schemes.

Different marking schemes require different codepoints. Thus, such separate markings consume valuable real-estate in the packet header, especially scarce in the case of MPLS Pre-Congestion Notification [I-D.davie-ecn-mpls] . Furthermore, two different metering techniques involve additional complexity in the data path of the internal routers of the PCN-domain.

To this end, [I-D.briscoe-tsvwg-cl-architecture] proposes an approach, referred to as "implicit Preemption marking" in that draft, that does not require separate termination-marking. However, it does require two separate measurement schemes: one measurement for Admission and another measurement for Flow Termination. Furthermore, this approach mandates that the configured-termination-rate be equal to a drop rate. This approach effectively uses dropping as the way

to convey information about how much traffic can "fit" under the configured-termination-rate, instead of using a separate termination marking. This is a significant restriction in that it results in flow termination only taking effect once packets actually get dropped.

This document presents an approach that allows the use of a single PCN marking and a single metering technique at the internal devices without requiring that the dropping and flow termination thresholds be the same. We argue that this approach can be used as intermediate step in implementation and deployment of a full-fledged dual-marking PCN implementation. We also quantify performance tradeoffs that are associated with the choice of the Single Marking approach.

2. The Single Marking Approach

2.1. High Level description

The proposed approach is based on several simple ideas:

- Replace virtual-queue-based threshold- or ramp-marking for Admission Control by excess-rate-marking:
  - meter traffic exceeding the configured-admissible-rate and mark *excess* traffic (e.g. using a token bucket with the rate configured with the rate equal to configured-admissible-rate)
  - at the PCN-boundary-node, stop admitting traffic when the fraction of marked traffic for a given edge-to-edge aggregate exceeds a configured threshold (e.g. stop admitting when 1% of all traffic in the edge-to-edge aggregate received at the ingress is marked)
- Impose a PCN-domain-wide constraint on the ratio U between the configured-admissible-rate on a link and level of the PCN load on the link at which Flow Termination needs to be triggered (but do not explicitly configure configured-termination-rate). For example, one might impose a policy that Flow Termination is triggered when PCN traffic exceeds 120% of the configured-admissible-rate on any link of the PCN-domain).

The remaining part of this section describes the possible operation of the system.

2.2. Operation at the PCN-interior-node

The PCN-interior-node meters the aggregate PCN traffic and marks the excess rate. A number of implementations are possible to achieve that. A token bucket implementation is particularly attractive because of its relative simplicity, and even more so because a token bucket implementation is readily available in the vast majority of existing equipment. The rate of the token bucket is configured to correspond to the configured-admissible-rate, and the depth of the token bucket can be configured by an operator based on the desired tolerance to PCN traffic burstiness.

Note that no configured-termination-rate is explicitly configured at the PCN-interior-node, and the PCN-interior-node does nothing at all to enforce it. All marking is based on the single configured rate threshold (configured-admissible-rate).

2.3. Operation at the PCN-egress-node

The PCN-egress-node measures the rate of both marked and unmarked traffic on a per-ingress basis, and reports to the PCN-ingress-node two values: the rate of unmarked traffic from this ingress node, which we deem Sustainable Admission Rate (SAR) and the Congestion Level Estimate (CLE), which is the fraction of the marked traffic received from this ingress node. Note that Sustainable Admission Rate is analogous to the sustainable termination rate of CL, except in this case it is based on the configured-admissible- rather than termination threshold, while the CLE is exactly the same as that of CL. The details of the rate measurement are outside the scope of this draft.

2.4. Operation at the PCN-ingress-node

2.4.1. Admission Decision

Just as in CL, the admission decision is based on the CLE. The ingress node stops admission of new flows if the CLE is above a pre-defined threshold (e.g. 1%). Note that although the logic of the decision is exactly the same as in the case of CL, the detailed semantics of the marking is different. This is because the marking used for admission in this proposal reflects the excess rate over the configured-admissible-rate, while in CL, the marking is based on exceeding a virtual queue threshold. Notably, in the current proposal, if the average sustained rate of admitted traffic is 5% over the admission threshold, then 5% of the traffic is expected to be marked, whereas in the context of CL a steady 5% overload should eventually result in 100% of all traffic being admission marked. A consequence of this is that for "smooth" constant-rate traffic, the

approach presented here will not mark any traffic at all until the rate of the traffic exceeds the configured admission threshold by the amount corresponding to the chosen CLE threshold.

At first glance this may seem to result in a violation of the pre-congestion notification premise that attempts to stop admission before the desired traffic level is reached. However, in reality one can simply embed the CLE level into the desired configuration of the admission threshold. That is, if a certain rate X is the actual target admission threshold, then one should configure the rate of the metering device (e.g. the rate of the token bucket) to X-y where y corresponds to the level of CLE that would trigger admission blocking decision.

A more important distinction is that the ramp- version of the virtual-queue based marking reacts to short-term burstiness of traffic, while the excess-rate based marking is only capable of reacting to rate violations at the timescale chosen for rate measurement. Based on our investigation, it seems that this distinction is not crucial in the context of PCN when no actual queuing is expected even if the virtual queue is full. More discussion on this is presented later in the draft.

2.4.2. Flow Termination Decision

When the ingress observes a non-zero CLE and Sustainable Admission Rate (SAR), it first computes the Sustainable Termination Rate (STR) by simply multiplying SAR by the system-wide constant U where U is the system-wide ratio between (implicit) termination and admission thresholds on all links in the PCN domain: STR = SAR*U. The PCN-ingress-node then performs exactly the same operation as in CL with respect to STR: it terminates the appropriate number of flows to ensure that the rate of traffic it sends to the corresponding egress node does not exceed STR.

Note: In certain cases where ingress-egress aggregations are not sufficient, additional mechanism may be needed to improve the accuracy of algorithm. One possibility is to guard/activate the termination control with a trigger computed from EWMA smoothed egress measurements (e.g. the termination should be triggered when the ratio of smoothed marked and smoothed unmarked traffic is greater than U-1). Sections 7 and 8.3 provide additional discussion on this issue. For sufficient levels of aggregation of IEA traffic, no smoothing of the termination trigger is required.

Just as in the case of CL, an implementation may decide to slow down the termination process by preempting fewer flows than is necessary to cap its traffic to STR by employing a variety of techniques such

as safety factors or hysteresis. In summary, the operation of Termination at the ingress node is mostly identical to that of CL, with the only exception that the sustainable Termination rate is computed from the sustainable admission rate rather than derived from a separate marking. As discussed earlier, this is enabled by imposing a system-wide restriction on the termination-to-admission thresholds ratio and changing the semantics of the admission marking from ramp- or threshold - to excess-rate-marking.

3. Benefits of Allowing the Single Marking Approach

The following is a summary of benefits associated with enabling the Single Marking (SM) approach. Some tradeoffs will be discussed in section 7 below.

- Reduced implementation requirements on core routers due to a single metering implementation instead of two different ones.
- Ease of use on existing hardware: given that the proposed approach is particularly amenable to a token bucket implementation, the availability of token buckets on virtually all commercially available routers makes this approach especially attractive.
- Enabling incremental implementation and deployment of PCN (see section 4).
- Reduced number of codepoints which need to be conveyed in the packet header. If the PCN-bits used in the packets header to convey the congestion notification information are the ECN-bits in an IP core and the EXP-bits in an MPLS core, those are very expensive real-estate. The current proposals need 5 codepoints, which is especially important in the context of MPLS where there is only a total of 8 EXP codepoints which must also be shared with DiffServ. Eliminating one codepoint considerably helps.
- A possibility of using a token-bucket-based, excess-rate-based implementation for admission provides extra flexibility for the choice of an admission mechanism, even if two separate markings and thresholds are used.

Subsequent sections argue that these benefits can be achieved with a relatively minor enhancements to the proposed PCN architecture as defined in [I-D.eardley-pcn-architecture], allow simpler implementations at the PCN-interior nodes, and trivial modifications at the PCN- boundary nodes. However, a number of tradeoffs need to be also considered, as discussed in section 7.

4. Impact on PCN Architectural Framework

The goal of this section is to propose several minor changes to the PCN architecture framework as currently described in [I-D.eardley-pcn-architecture] in order to enable the single marking approach.

4.1. Impact on the PCN-Internal-Node

No changes are required to the PCN-internal-node in architectural framework in [I-D.eardley-pcn-architecture] in order to support the Single Marking Proposal. The current architecture [I-D.eardley-pcn-architecture] already allows only one marking and metering scheme rather than two by supporting either "admission only" or "termination only" functionality. To support the SM proposal a single threshold (i.e. Configured-termination-rate) must be configured at the PCN-internal-node, and excess-rate marking as described in should be used to mark packets as described in [I-D.briscoe-tsvwg-cl-architecture].

The configuration parameter(s) at the PCN-ingress-nodes and PCN-egress-node (described in section 4.2) will determine how the marking should be interpreted by the PCN-boundary-nodes.

4.2. Impact on the PCN-boundary nodes

We propose an addition of one global configuration parameter MARKING_MODE to be used at all PCN boundary nodes. If MARKING_MODE = DUAL_MARKING, the behavior of the appropriate PCN-boundary-node as described in the current version of [I-D.eardley-pcn-architecture]. If MARKING_MODE = SINGLE_MARKING, the behavior of the appropriate boundary nodes is as described in the subsequent subsections.

4.2.1. Impact on PCN-Egress-Node

The exact operation of the PCN-Egress-node depends on whether it is admission marking (AM-marking) or termination-marking (TM-marking) that is used for SM. An assumption made in draft-charny-pcn-comparison-00 is to use AM-marked packets for SM instead of TM-marked packets. In that case the MARKING_MODE will signal that Sustainable-Rate must be measured against the AM-marked packets, while Congestion-Level-Estimate (CLE) will be measured against AM-marked packets just as in the case of CL. If, however, TM-marking is used for SM, then CLE in SM will need to be measured against the TM-marked packets.

In more detail, if the encoding used for SM is that of TM-marking, then the setting MARKING_MODE=SINGLE_MARKING indicates that the CLE

is measured against termination-marked packets, while if MARKING_MODE=DUAL_MARKING, the CLE is measured against admission-marked packets. The method of measurement of CLE does not depend on the choice of the marking against which the measurement is performed.

If, however, the encoding used for SM is that of AM-marking, then the setting MARKING_MODE=SINGLE_MARKING indicates that the Sustainable-Rate is measured against AM-marked packets, while the setting of MARKING_MODE=DUAL_MARKING indicates that Sustainable-Rate should be measured against TM-marked packets

We note that from the implementation point of view, the same two functions (measuring the CLE and measuring the Sustainable-Aggregate-Rate are required by both the SM approach and the approach in CL, so the difference in the implementation complexity of the PCN-egress-node is quite negligible and amounts to checking which encoding is used for which function based on the setting of a global parameter. If this checking is implemented, then switching the egress nodes from supporting SM to supporting CL amounts to changing the setting of the global parameter.

4.2.2. Impact on the PCN-Ingress-Node

If MARKING_MODE=DUAL_MARKING, the PCN-ingress-node behaves exactly as described in [I-D.eardley-pcn-architecture]. If MARKING_MODE = SINGLE_MARKING, then an additional global parameter U is defined. U must be configured at all PCN-ingress-nodes and has the meaning of the desired ratio between the traffic level at which termination should occur and the desired admission threshold, as described in section 2.4 above. The value of U must be greater than or equal to 1. The value of this constant U is used to multiply the Sustainable Aggregate Rate received from a given PCN-egress-node to compute the rate threshold used for flow termination decisions.

In more detail, if MARKING_MODE=SINGLE_MARKING, then

- A PCN-ingress-node receives CLE and/or Sustainable Aggregate Rate from each PCN-egress-node it has traffic to. This is fully compatible with PCN architecture as described in [I-D.eardley-pcn-architecture].

- A PCN-ingress-node bases its admission decisions on the value of CLE. Specifically, once the value of CLE exceeds a configured threshold, the PCN-ingress-node stops admitting new flows. It restarts admitting when the CLE value goes down below the specified threshold. This is fully compatible with PCN architecture as described in [I-D.eardley-pcn-architecture].

- A PCN-ingress node receiving a Sustainable Rate from a particular PCN-egress node measures its traffic to that egress node. This again is fully compatible with PCN architecture as described in draft-earley-pcn-architecture-00.

- The PCN-ingress-node computes the desired Termination Rate to a particular PCN-egress-node by multiplying the Sustainable Aggregate Rate from a given PCN-egress-node by the value of the configuration parameter U. This computation step represents a proposed change to the current version of [I-D.eardley-pcn-architecture].

- Once the Termination Rate is computed, it is used for the flow termination decision in a manner fully compatible with [I-D.eardley-pcn-architecture]. Namely the PCN-ingress-node compares the measured traffic rate destined to the given PCN-egress-node with the computed Termination rate for that egress node, and terminates a set of traffic flows to reduce the rate exceeding that Termination rate. This is fully compatible with [I-D.eardley-pcn-architecture].

We note that as in the case of the PCN-egress-node, the change in the implementation of the PCN-ingress-node to support SM is quite negligible (a single multiplication per ingress rate measurement interval for each egress node). [Note: If additional smoothing of the termination signal is required to deal with low IE aggregation as mentioned in section 2.4.2, this smoothing constitutes an additional requirement on the PCN-ingress-node.]

4.3. Summary of Proposed Enhancements Required for Support of Single Marking Options

The enhancements to the PCN architecture as defined in [I-D.eardley-pcn-architecture], in summary, amount to:

- defining a global (within the PCN domain) configuration parameter MARKING_MODE at PCN-boundary nodes

- Defining a global (within the PCN domain) configuration parameter U at the PCN-ingress-nodes. This parameter signifies the implicit ratio between the termination and admission thresholds at all links

- Multiplication of Sustainable-Aggregate-Rate by the constant U at the PCN-ingress-nodes if MARKING_MODE=SINGLE_MARKING

- Using the MARKING_MODE parameter to guide which marking is used to measure the CLE (but the measurement functionality is unchanged)

4.4. Proposed Optional Renaming of the Marking and Marking Thresholds

Previous work on example mechanisms [I-D.briscoe-tsvwg-cl-architecture] implementing the architecture of [I-D.eardley-pcn-architecture] assumed that the semantics of admission control marking and termination marking differ. Specifically, it was assumed that for termination purposes the semantics of the marking is related to the excess rate over the configured (termination) rate, or even more precisely, the amount of traffic that remains unmarked (sustainable rate) after the excess traffic is marked. Some of the recent proposals assume yet different marking semantics [I-D.babiarz-pcn-3sm], [I-D.westberg-pcn-load-control].

Even though specific association with marking semantics and function (admission vs termination) has been assumed in prior work, it is important to note that in the current architecture draft [I-D.eardley-pcn-architecture], the associations of specific marking semantics (virtual queue vs excess rate) with specific functions (admission vs termination) are actually *not* directly assumed. In fact , the architecture document does not explicitly define the marking mechanism, but rather states the existence of two different marking mechanisms, and also allows implementation of either one or both of these mechanisms in a PCN- domain.

We argue that this separation of the marking semantics from the functional use of the marking is important to make sure that devices supporting the same marking can interoperate in delivering the function which is based on specific supported marking semantics.

To divorce the function (admission vs termination) and the semantics (excess rate marking, virtual queue marking), it may be beneficial to rename the marking to be associated with the semantics rather than the function to explicitly disassociate the two functions. Specifically, it may be beneficial to change the "admission-marking" and "termination-marking" currently defined in the architecture as "Type Q" or "virtual-queue-based" marking, and "Type R" or "excess-rate-based" marking. Of course, other choices of the naming are possible (including keeping the ones currently used in [I-D.eardley-pcn-architecture]).

With this renaming, the dual marking approach in [I-D.briscoe-tsvwg-cl-architecture] would require PCN-internal-nodes to support both Type R and Type Q marking, while SM would require support of Type-R marking only.

We conclude by emphasizing that the changes proposed here amount to merely a renaming rather than a change to the proposed architecture,

and are therefore entirely optional.

4.5. An Optimization Using a Single Configuration Parameter for Single Marking

We note finally that it is possible to use a single configuration constant U instead of two constants (U and MARKING_TYPE). Specifically, one can simply interpret the value of U=1 as the dual-marking approach (equivalent to MARKING_TYPE=DUAL_MARKING) and use U>1 to indicate SM. This is discussed in detail in Section 9.

5. Incremental Deployment Considerations

As most of today's routers already implement a token bucket, implementing token-bucket based excess-rate marking at PCN-ingress nodes is a relatively small incremental step for most of today's implementations. Implementing an additional metering and marking scheme in the datapath required by the dual-marking approach without encountering performance degradation is a larger step. The SM approach may be used as an intermediate step towards the deployment of a dual-marking approach in the sense that routers implementing single-marking functionality only may be deployed first and then incrementally upgraded to CL.

The deployment steps might be as follows:

- Initially all PCN-ingress-nodes might implement Excess-rate (Type R) type marking and metering only

- All PCN-boundary nodes implement the full functionality as described in this document (including the configuration parameters MARKING_TYPE and U) from the start. Since the PCN-boundary-node behavior is enabled by simply changing the values of the configuration parameters, all boundary nodes become immediately compatible with both dual-marking (CL) and single-marking.

- Initially all boundary nodes are configured parameter settings indicating SM option.

- When a PCN-internal node with dual-marking functionality replaces a subset of PCN-internal-nodes, the virtual-queue-based (Type Q) marking is simply ignored by the boundary nodes until all PCN-internal-nodes in the PCN-domain implement the dual-marking metering and marking. At that time the value of the configuration parameters may be reset to at all boundary nodes to indicate the Dual Marking configuration.

- Note that if a subset of PCN-boundary-nodes communicates only with each other, and all PCN-internal-nodes their traffic traverses have been upgraded, this subset of nodes can be upgraded to two dual-marking behavior while the rest of the PCN-domain can still run the SM case. This would entail configuring two thresholds at the PCN-internal-nodes, and setting the value of the configuration parameters appropriately in this subset.

- Finally note that if the configuration parameter U is configured per ingress-egress-pair rather than per boundary node, then each ingress-egress pair can be upgraded to the dual marking simultaneously. While we do not recommend that U is defined on a per-ingress-egress pair, such possibility should be noted and considered.

6. Tradeoffs, Issues and Limitations of Single Marking Approach

6.1. Global Configuration Requirements

An obvious restriction necessary for the single-marking approach is that the ratio of (implicit) termination and admission thresholds remains the same on all links in the PCN region. While clearly a limitation, this does not appear to be particularly crippling, and does not appear to outweigh the benefits of reducing the overhead in the router implementation and savings in codepoints in the case of a single PCN domain, or in the case of multiple concatenated PCN regions. The case when this limitation becomes more inconvenient is when an operator wants to merge two previously separate PCN regions (which may have different admission-to-termination ratios) into a single PCN region. In this case it becomes necessary to do a network-wide reconfiguration to align the settings.

The fixed ratio between the implicit termination rate and the configured-admissible-rate also has an implications on traffic engineering considerations. Those are discussed in section 7.7 below.

SM also requires that all PCN-boundary-nodes use the same setting of the global parameters U and MARKING_MODE.

6.2. Assumptions on Loss

Just as in the case of [I-D.briscoe-tsvwg-cl-architecture], the approach presented in this draft assumes that the configured-admissible-rate is configured at each link below the service rate of the traffic using PCN. This assumption is significant because the algorithm relies on the fact that if admission threshold is exceeded,

enough marked traffic reaches the pcn-egress-node to reach the configured CLE level. If this condition does not hold, then traffic may get dropped without ever triggering admission decision.

6.3. Effect of Reaction Timescale of Admission Mechanism

As mentioned earlier in this draft, there is a potential concern that slower reaction time of admissions mechanism presented in this draft compared to [I-D.briscoe-tsvwg-cl-architecture] may result in overshoot when the load grows rapidly, and undershoot when the load drops rapidly. While this is a valid concern theoretically, it should be noted that at least for the traffic and parameters used in the simulation study reported here, there was no indication that this was a problem.

6.4. Performance Implications and Tradeoffs

Replacement of a relatively well-studied queue-based measurement-based admission control approach by a cruder excess-rate measurement technique raises a number of algorithmic and performance concerns that need to be carefully evaluated. For example, a token-bucket excess rate measurement is expected to be substantially more sensitive to traffic burstiness and parameter setting, which may have a significant effect in the case of lower levels of traffic aggregation, especially for variable-rate traffic such as video. In addition, the appropriate timescale of rate measurement needs to be carefully evaluated, and in general it depends on the degree of expected traffic variability which is frequently unknown.

In view of that, an initial performance comparison of the use token-bucket based excess-rate metering is presented in the following section. Within the constraints of this study, the performance tradeoffs observed between the queue-based technique for admission control suggested in [I-D.briscoe-tsvwg-cl-architecture] and a simpler token-bucket-based excess rate measurement for admission control do not appear to be a cause of substantial concern for cases when traffic aggregation is reasonably high at the bottleneck links as well as on a per ingress-egress pair basis. Details of the simulation study, as well as additional discussion of its implications are presented in section 7.

Also, one mitigating consideration in favor of the simpler mechanism is that in a typical DiffServ environment, the real-time traffic is expected to be served at a higher priority and/or the target admission rate is expected to be substantially below the speed at which the real-time queue is actually served. If these assumptions hold, then there is some margin of safety for an admission control algorithm, making the requirements for admission control more

forgiving to bounded errors - see additional discussion in section 7.

Flow Termination mechanisms of Single Marking and CL are both based on excess-rate metering and marking, as so it may be inferred that their performance is similar. However, there is a subtle difference between the two mechanisms stemming from the fact that in SM, packets continue to be marked when traffic has reduced between the (implicit) termination threshold and the (explicit) admission threshold. This "extra" marking may result in over-termination compared to CL, especially in multi-bottleneck topologies. We quantify this over-termination in Sections 7 and 8. While we believe that the extent of this over-termination is tolerable for practical purposes, it needs to be taken into account when considering performance tradeoffs of the two mechanisms.

6.5. Effect on Proposed Anti-Cheating Mechanisms

Replacement of the queue-based admission control mechanism of [I-D.briscoe-tsvwg-cl-architecture] by an excess-rate based admission marking changing the semantics of the pre-congestion marking, and consequently interferes with mechanisms for cheating detection discussed in [I-D.briscoe-tsvwg-re-ecn-border-cheat]. Implications of excess-rate based marking on the anti-cheating mechanisms need to be considered.

6.6. ECMP Handling

An issue not directly addressed by neither the dual-marking approach described in [I-D.briscoe-tsvwg-cl-architecture] nor the single-marking approach described in this draft is that if ECMP is enabled in the PCN-domain, then the PCN-edge nodes do not have a way of knowing whether specific flows in the ingress-egress aggregate (IEA) followed the same path or not. If multiple paths are followed, then some of those paths may be experiencing pre-congestion marking, and some are not. Hence, for example, an ingress node may choose to terminate a flow which takes an entirely un-congested path. This will not only unnecessarily terminate some flows, but also will not eliminate congestion on the actually congested path. While eventually, after several iterations, the correct number of flows might be terminated on the congestion path, this is clearly suboptimal, as the termination takes longer, and many flows are potentially terminated unnecessarily.

Two approaches for solving this problem were proposed in draft-babiarz-pcn-explicit-marking and draft-westberg-pcn-load-control. The former handles ECMP by terminating those flows that are termination-marked as soon as the termination marking is seen. The latter uses an additional DiffServ

marking/codepoint to mark all packets of the flows passing through a congestion point, with the PCN-boundary-nodes terminating only those flows which are marked with this additional marks. Both of these approaches also differ in the termination-marking semantics, but we omit the discussion of these differences as they can be considered largely independent of the ECMP issue.

It should be noted that although not proposed in this draft, either of these ideas can be used with dual- and single- marking approaches discussed here. Specifically, in CL, when a PCN-ingress-node decides which flows to terminate, it can choose for termination only those flows that are termination-marked. Likewise, at the cost of an additional (DiffServ) codepoint, a PCN-internal-node can mark all packets of all flows using this additional marking, and then the PCN-boundary-nodes can use this additional marking to guide their flow termination decisions. In SM, since only one codepoint is used, this approach will result in choosing only those flows for termination which traverse at least one link where the traffic level is above the admission threshold. This may result in termination of the some flows erroneously.

Either of these approaches appears to imply changes to the PCN architecture as proposed in draft-eardley-pcn-architecture-00. Such changes have not been considered in this draft at this point.

6.7. Traffic Engineering Considerations

Dual-marking PCN can be viewed as a replacement for Resilient Network Provisioning (RNP). It is reasonable to expect that an operator currently using DiffServ provisioning for real-time traffic might consider a move to PCN. For such a move it is necessary to understand how to set the PCN rate thresholds to make sure that the move to PCN does not detrimentally affect the guarantees currently offered to the operator.

The key question addressed in this section is how to set PCN admission and termination thresholds in the dual marking approach or the single admission threshold and the scaling factor U reflecting the implicit termination threshold in the single-marking approach so that the result is "not worse" than provisioning in the amount of traffic that can be admitted. Even more specifically we will address what if any are the tradeoffs between the dual-marking and the single-approach arise when answering this question. This question was first raised in [Menth] and is further addressed below.

Typically, RNP would size the network (in this specific case traffic that is expected to use PCN) by making sure that capacity available for this (PCN) type of traffic is sufficient for PCN traffic under

"normal" circumstances (that is, under no failure condition, for a given traffic matrix), and under a specific set of single failure scenarios (e.g. failure of each individual single link). Some of the obvious limitations of such provisioning is that

- the traffic matrix is often not known well, and at times, especially during flash-crowds, the actual traffic matrix can differ substantially from the one assumed by provisioning

- unpredicted, non-planned failures can occur (e.g. multiple links, nodes, etc), causing overload.

It is specifically such unplanned cases that serve as the motivation for PCN. Yet, one may want to make sure that for cases that RNP can (and does today) plan for, PCN does no worse when an operator makes the decision to implement PCN on a currently provisioned network. This question directly relates to the choice of the PCN configured admission and termination thresholds.

For the dual-marking approach, where the termination and admission thresholds are set independently on any link, one can address this issue as follows [Menth]. If a provisioning tool is available, for a given traffic matrix, one can determine the utilization of any link used by traffic expected to use PCN under the no-failure condition, and simply set the configured-admissible-rate to that "no-failure utilization". Then a network using PCN will be able to admit as much traffic as the RNP, and will reject any traffic that exceeds the expected traffic matrix. To address resiliency against a set of planned failures, one can use RNP to find the worst-case utilization of any link under the set of all provisioned failures, and then set the configured-termination-rate to that worst case utilization.

Clearly, such setting of PCN thresholds with the dual-marking approach will achieve the following goals:

- PCN will admit the same traffic matrix as used by RNP and will protect it against all planned failures without terminating any traffic

- When traffic deviates from the planned traffic matrix, PCN will admit such traffic as long as the total usage of any link (without failure) does not exceed the configured-admission threshold, and all admitted traffic will be protected against all planned failures

- Additional traffic will not be admitted under the no-failure conditions, and traffic exceeding configure-termination threshold during non-planned failures will be terminated.

- Under non-planned failures, some of the planned traffic matrix may be terminated, but the remaining traffic will be able to receive its QoS treatment.

The above argues that an operator moving from a purely provisioned network to a PCN network can find the settings of the PCN threshold with dual marking in such a way that all admitted traffic is protected against all planned failures.

It is easy to see that with the single-marking scheme, the above approach does not work directly [Menth]. Indeed, the ratio between the configured-termination thresholds and the configured-admissible-rate may not be constant on all links. Since the single-marking approach requires the (implicit) termination rate to be within a fixed factor of the configured admission rate, it can be argued (as was argued in [Menth].) that one needs to set the system-wide ratio U between the (implicit) termination threshold and the configured admission threshold to correspond to the largest ratio between the worst case resilient utilization and the no-failure utilization of RNP, and set the admission threshold on each link to the worst case resilient utilization divided by that system wide ratio. Such approach would result in lower admission thresholds on some links than that of the dual-marking setting of the admission threshold proposed above. It can therefore be argued that PCN with SM will be able to admit *less* traffic that can be fully protected under the planned set of failures than both RNP and the dual-marking approach.

However, the settings of the single-marking threshold proposed above are not the only one possible, and in fact we propose here that the settings are chosen differently. Such different settings (described below) will result in the following properties of the PCN network:

- PCN will admit the same traffic matrix as used by RNP *or more*

- The traffic matrix assumed by RNP will be fully protected against all planned failures without terminating any admitted traffic

- When traffic deviates from the planned traffic matrix, PCN will admit such traffic as long as the total usage of any link (without failure) does not exceed the configured-admission threshold, However, not all admitted traffic will be protected against all planned failures (i.e. even under planned failures, traffic exceeding the planned traffic matrix may be preempted)

- Under non-planned failures, some of the planned traffic matrix may be terminated, but the remaining traffic will be able to receive its QoS treatment.

It is easy to see that all of these properties can be achieved if instead of using the largest ratio between worst case resilient utilization to the no-failure utilization of RNP across all links for setting the system wide constant U in the single-marking approach as proposed in [Menth], one uses the *smallest* ratio, and set the configured-admissible-rate to the worst case resilient utilization divided by that ratio. With such setting, the configured-admissions threshold on each link is at least as large as the non-failure RNP utilization (and hence the planned traffic matrix is always admitted), and the implicit termination threshold is at the worst case planned resilient utilization of RNP on each link (and hence the planned traffic matrix will be fully protected against the planned failures). Therefore, with such settings, the single-marking draft does as well as RNP or dual-marking with respect to the planned matrix and planned failures. In fact, unlike the dual marking approach, it can admit more traffic on some links than the planned traffic matrix would allow, but it is only guaranteed to protect up to the planned traffic matrix under planned failures.

In summary, we have argued that both the single-marking approach and the dual-marking approach can be configured to ensure that PCN "does no worse" than RNP for the planned matrix and the planned failure conditions, (and both can do better than RNP under non-planned conditions). The tradeoff between the two is that although the planned traffic matrix can be admitted with protection guarantees against planned failures with both approaches, the nature of the guarantee for the admitted traffic is different. Dual marking (with the settings proposed) would protect all admitted traffic but would not admit more than planned), while SM (with the settings proposed) will admit more traffic than planned, but will not guarantee protection against planned failures for traffic exceeding planned utilization.

7. Performance Evaluation Comparison

7.1. Relationship to other drafts

Initial simulation results of admission and termination mechanisms of [I-D.briscoe-tsvwg-cl-architecture] were reported in [I-D.briscoe-tsvwg-cl-phb]. A follow-up study of these mechanisms is presented in a companion draft draft-zhang-cl-performance-evaluation-02.txt. The previous versions of this draft concentrated on a performance comparison of the virtual-queue-based admission control mechanism of [I-D.briscoe-tsvwg-cl-phb] and the token-bucket-based admission control described in section 2 of this draft. In this version, we added performance evaluation of the Flow Termination function of SM.

The Flow Termination results are discussed in section 7.3

7.2. Admission Control: High Level Conclusions

The results of this study indicate that there is a potential that a reasonable complexity/performance tradeoff may be viable for the choice of admission control algorithm. In turn, this suggests that using a single codepoint and metering technique for admission and termination may be a viable option.

The key high-level conclusions of the simulation study comparing the performance of queue-based and token-based admission control algorithms are summarized below:

1. At reasonable level of aggregation at the bottleneck and per ingress-egress pair traffic, both algorithms perform reasonably well for the range of traffic models considered.

2. Both schemes are stressed for small levels of ingress-egress pair aggregation levels of bursty traffic (e.g. a single video-like bursty SVD flow per ingress-egress pair). However, while the queue-based scheme results in tolerable performance even at low levels of per ingress-egress aggregation, the token-bucket-based scheme is substantially more sensitive to parameter setting than the queue-based scheme, and its performance for the high rate bursty SVD traffic with low levels of ingress-egress aggregation is quite poor unless parameters are chosen carefully to curb the error. It should be noted that the SVD traffic model used in this study is expected to be substantially more challenging for both admission and termination mechanisms that the actual video traffic, as the latter is expected to be much smoother than the bursty on-off model with high peak-to-mean ratio we used. This expectation is confirmed by the fact that simulations with actual video traces reported in this version of the draft reveal that the performance of the video traces is much closer to that of VBR voice than of our crude SVD on-off model.

3. Even for small per ingress-egress pair aggregation, reasonable performance across a range of traffic models can be obtained for both algorithms (with a narrower range of parameter setting for the token-bucket based approach) . However, at very low ingress-egress aggregation, the token bucket scheme is substantially more sensitive to parameter variations than the virtual-queue scheme. In general, the token-bucket scheme performance is quite brittle at very low aggregations, and displays substantial performance degradation with BATCH traffic, as well synchronization effects resulting in substantial over-admission (see section 8.4.2)

4. The absolute value of round-trip time (RTT) or the RTT difference between different ingress-egress pair within the range of continental propagation delays does not appear to have a visible effect on the performance of both algorithms.

5. There is no substantial effect on the bottleneck utilization of multi-bottleneck topologies for both schemes. Both schemes suffer substantial unfairness (and possibly complete starvation) of the long-haul aggregates traversing multiple bottlenecks compared to short-haul flows (a property shared by other MBAC algorithms as well). Token-bucket scheme displayed somewhat larger unfairness than the virtual-queue scheme.

7.3. Flow Termination Results

A consequence of using just a single metering and marking and a single marking encoding in SM is that when the traffic level is between admission and (implicit) termination threshold, traffic continues to be marked in SM (because it exceeds the admission threshold at which the metering occurs). This is in contrast to CL when termination marking stops as soon as the traffic falls below the termination threshold. This subtle difference results in a visible performance impact on the Termination algorithm of SM, as discussed in the next subsections. Specifically:

- SM requires more ingress-egress aggregation than CL (and the amount of aggregation needed for the termination function is higher than that of admission - see sections 7.3.1 and 8.3).

- In the multiple bottleneck scenario, where PCN traffic exceeds the configured (admission) rate on multiple links, additional over-termination may occur over that already reported for CL (see sections 7.3.2 and 8.3 for more detail).

7.3.1. Sensitivity to Low Ingress-Egress aggregation levels

In SM, the sustainable termination rate is inferred to from the Sustainable (Admission) Rate, by multiplying it by a system-wide constant U. In the case of a single bottleneck, a fluid model in which marking is uniformly distributed among the contending IEAs, the Termination Function of CL and SM would be identical. However, in reality, as shown in draft-zhang-performance-evaluation, excess-rate marking does not get distributed among contending IEAs completely uniformly, and at low ingress-egress aggregations, some IEAs get marked more than others. As a result, when traffic is close below the (implicit) termination threshold at the bottleneck , some IEAs get excessively marked, while some get less than their "fair" share of marking. This causes a false termination event at the PCN-

ingress-nodes corresponding to those IEAs which get excessively marked, even though the bottleneck load did not exceed the (implicit) termination threshold. This effect is especially pronounced for low and medium aggregates of highly bursty traffic.

We investigated how much aggregation is needed to remove this effect completely, and found that the number of flows in the IEAs necessary to reduce this error to within 4-10% ranged from about 50 to 150 for different traffic types we tested (see section 8.3 in the Appendix for more detailed results).

We also found that for lower aggregation levels, the results could be improved to be comparable with CL with respect to over-termination if the ingresses used EWMA smoothing to the ratio of marked and unmarked traffic when triggering the termination event. Such smoothing, however, would add latency to the termination decision. The exact magnitude of this additional latency depends on the value of the global parameter U, the extent of the overload (i.e. excess over (implicit) termination threshold), and the exponential weight in the EWMA smoothing. In the range of parameters we investigated that seem practically reasonable, the additional latency is bounded by 1-2 sec (see detailed results in section 8.3). Such smoothing is not necessary at larger levels of ingress-egress aggregation.

In conclusion, to avoid over-termination on a single bottleneck due to non-uniformity of packet marking distribution among contending IEAs, SM needs substantially more ingress-egress aggregation than CL, if no additional mechanism are used to smooth the termination trigger.

7.3.2. Over-termination in the Multi-bottleneck Scenarios

As we showed in draft-zhang-performance-evaluation, when long-haul flows traverse more than one bottleneck, each addition bottleneck incurs additional termination-marking, which causes long-haul terminates more than its fair-share, and the unfairness might in turn cause over-termination on the upstream bottleneck.

SM has a similar issue. However, in addition, this issue is further amplified, as discussed below. This amplification is due to the fact that in SM, metering is done at the lower (admission) threshold, and so the quantity of the additional marking received at subsequent bottleneck is amplified by factor U (ratio of termination/admission threshold). It in turn reduces the sustainable rate (i.e. the rate of unmarked packets), as seen by the PCN-egress-node. It can be shown that this additional marking generally results in SM terminating more traffic than CL under the same circumstances, when multiple bottlenecks are traversed. The degree of over-termination

strongly depends on the number of bottlenecks in the topology, and on the degree of bottleneck overload above the (implicit) termination threshold.

To understand the significance of this over-termination in practice, we randomly generated ~50,000 random traffic matrices on the 5-BTN topology (see section 8 for detail), choosing the settings of the admission threshold randomly on each link. We chose U randomly in the interval 1.0<U<3.0 for each experiment. In these experiments, the overload on the "tightest" bottleneck ranged from 1-10X, and in different experiments the actual number of links where traffic exceeded (implicit) termination threshold ranges from 0 to 5. Table 7.1 below shows the distribution of over-termination for the subset of 19689 experiments with 1.2<U<2.0. We chose this subset because we believe this is a reasonable range for the choice of U in practice. We report the full distribution for the entire range of U we experimented with in section 8.3.

(preamble)

Alg.	Distribution of Over-Termination Percentage				
	0.0-0.1	0.1-0.2	0.2-0.3	0.3-0.4	0.4-0.5
CL	0.985	0.013	0.000	0.000	0.000
SM	0.659	0.294	0.043	0.001	0.000

(Table 7.1. Distribution of over-termination percentage in 19689 experiments with 1.2 <U<2.0.)

We refer the reader to Section 8.3 of the Appendix for a more detailed discussion on this issue.

7.4. Future work

This study is but the first step in performance evaluation of the SM algorithm. Further evaluation should include a range of investigation, including the following

- effect of signaling delays/probing

- effect of loss of marked packets

8. Appendix A: Simulation Details

8.1. Simulation Setup and Environment

8.1.1. Network and Signaling Models

Network topologies used in this study are shown in the Figures below. The network is modeled as either Single Link (Fig. A.1), Multi Link Network with a single bottleneck (termed "RTT", Fig. A.2), or a range of multi-bottleneck topologies shown in Fig. A.3 (termed "Parking Lot").

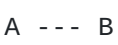


Figure A.1: Simulated Single Link Network.

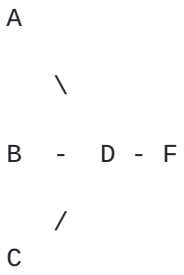
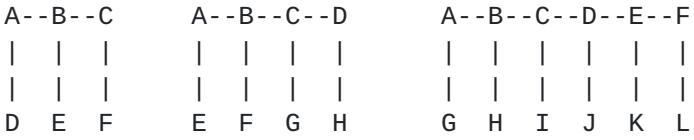


Figure A.2: Simulated Multi Link Network.



(a) (b) (c)

Figure A.3: Simulated Multiple-bottleneck (Parking Lot)Topologies.

Figure A.1 shows a single link between an ingress and an egress node, all flows enter at node A and depart at node B. This topology is used for the basic verification of the behavior of the algorithms with respect to a single IEA in isolation.

In Figure A.2, A set of ingresses (A,B,C) are connected to an interior node in the network (D). This topology is used to study the behavior of the algorithm where many IEAs share a single bottleneck link. The number of ingresses varied in different simulation experiments in the range of 2-100. All links have generally different propagation delays, in the range 1ms - 100 ms (although in some experiments all propagation delays are set the same. This node

D 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 D-F, where pre-congestion notification algorithm is enabled. The capacities of the ingress links are not limiting, and hence no PCN is enable on those. The bottleneck link D-F is modeled with a 10ms propagation delay in all simulations. Therefore the range of round-trip delays in the experiments is from 22ms to 220ms.

Another type of network of interest is multi-bottleneck (or Parking Lot, PLT for short) topology. The simplest PLT with 2 bottlenecks is illustrated in Fig A.3(a). An example traffic matrix with this network on this topology is as follows:

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

In the 2-hop PLT shown in Fig. A.3(a) the points of congestion are links A--B and B--C. Capacity of all other links is not limiting. We also experiment with larger PLT topologies with 3 bottlenecks(see Fig A.3(b)) and 5 bottlenecks (Fig A.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), and N ingress-egress pairs that carry flows traversing a single bottleneck link and exiting at the next "hop". In all cases, only the "horizontal" links in Fig. A.3 were the bottlenecks, with capacities of all "vertical" links non-limiting. Propagation delays for all links in all PLT topologies are set to 1ms.

Due to time limitations, other possible traffic matrices (e.g. some of the flows traversing a subset of several bottleneck links) 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 aggregation - above 10 Mbps for voice and 622 Mbps for SVD, up to 2.4 Gbps. But we also investigated slower 'bottleneck' links down to 512 Kbps in some experiments. Higher rate bottleneck speeds wee not considered due to the simulation time limitations. It should generally be expected that the higher link speeds will result in higher levels of aggregation, and hence generally better performance of the measurement-based

algorithms. Therefore is seems reasonable to believe that the link speeds studied do provide meaningful evaluation targets.

In the simulation model, a call requests 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. If the Congestion-Level-Estimate is below the specified CLE-threshold, the call is admitted, otherwise it is rejected. An admitted call sends packets according to one of the chosen traffic models for the duration of the call (see next section). Propagation delay from source to the ingress and from destination to the egress is assumed negligible and is not modeled.

In the simulation model of admission control, a call 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. If the Congestion Level Estimate is below the specified CLE- threshold, the call is admitted, otherwise it is rejected. For Flow Termination, once the ingress node of a PCN-domain decides to terminate a flow, that flow is preempted immediately and sends no more packets from that time on. The life of a flow outside the domain described above is not modeled. Propagation delay from source to the ingress and from destination to the egress is assumed negligible and is not modeled.

8.1.2. Traffic Models

Four types of traffic were simulated (CBR voice, on-off traffic approximating voice with silence compression, and on-off traffic with higher peak and mean rates (we termed the latter "Synthetic Video" (SVD) as the chosen peak and mean rate was similar to that of an MPEG video stream. (but for SVD no attempt was made to match any other parameters of this traffic to those of a video stream), and finally real video traces from http://www.tkn.tu-berlin.de/research/trace/trace.html (courtesy Telecommunication Networks Group of Technical University of Berlin).

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-admissible-rate in each experiment. Overloads in the range 1x to 5x and underload with 0.95x have been investigated. Note that the rationale for looking at the load 1 and below is to see

if any significant amount of "false rejects" would be seen (i.e. one would assume that all traffic should be accepted if the total demand is below the admission threshold). For on-off traffic, on and off periods were exponentially distributed with the specified mean. Traffic parameters for each type are summarized below:

8.1.2.1. Voice Traffic Models

Table A.1 below describes all voice codecs we modeled in our simulation results.

The first two rows correspond to our two basic models corresponding 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 listed 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 A.1 Simulated Voice Codices.

8.1.2.2. "Synthetic Video": High Rate ON-OFF traffic with Video-like Mean and Peak Rates ("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 real 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".

- 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 660m

8.1.2.3. 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. Each trace in that repository s 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 in our simulation is much smaller than the length of the trace, we checked whether the expected rate of flow corresponds to the trace's long term average. To do so, we simulated a number of flows starting from random locations in the trace with duration chosen to be exponentially distributed with the mean of 1min. The results show that the expected rate of flow is roughly the same as the trace's average.

In summary, our simulations use a set of segments of the 120 min trace chosen at random offset from the beginning and with mean duration of 1 min.

Since the traces provide only the frame size, we also simulated packetization of the frame as a CBR segment with packet size and

inter-arrival time corresponding to those of our SVD model. Since the frame size is not always a multiple of the chosen packet size, the last packet in a frame may be shorter than 1500 bytes chosen for the SVD encoding.

Traffic characteristics for our VTR models 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.

8.1.2.4. 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-interarrival-time of the CBR portion of the corresponding base model. To simulate a range of queuing 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

8.1.3. 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.

8.2. Admission Control

8.2.1. Parameter Settings

8.2.1.1. Queue-based settings

All the queue-based simulations were run with the following Virtual Queue thresholds:

- virtual-queue-rate: configured-admissible-rate, 1/2 link speed
- 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

At the egress, the CLE is computed as an exponential weighted moving average (EWMA) on an interval basis, with 100ms measurement interval chosen in all simulations. We simulated the EWMA weight ranging 0.1 to 0.9. The CLE threshold is chosen to be 0.05, 0.15, 0.25, and 0.5.

8.2.1.2. Token Bucket Settings

The token bucket rate is set to the configured-admissible-rate, which is half of the link speed in all experiments. Token bucket depth ranges from 64 to 512 packets. Our simulation results indicate that depth of token bucket has no significant impact on the performance of the algorithms and hence, in the rest of the section, we only present the result with 256 packets bucket depth.

The CLE is calculated using EWMA just as in the case of virtual-queue settings, with weights from 0.1 to 0.9. The CLE thresholds are chosen to be 0.0001, 0.001, 0.01, 0.05 in this case. Note that the since meaning of the CLE is different for the Token bucket and queue-based algorithms, so there is no direct correspondence between the choice of the CLE thresholds in the two cases.

8.2.2. Sensitivity to EWMA weight and CLE

Table A.2 summarized the comparison result of over-admission-percentage values from 15 experiments with different [weight, CLE threshold] settings for each type of traffic and each topology. The Ratio of the demand on the bottleneck link to the configured

admission threshold is set to 5x. (In the results for 0.95x can be found in previous draft). For parking lot topologies we report the worst case result across all bottlenecks. We present here only the extreme value over the range of resulting over-admission-percentage values.

We found that the virtual-queue admission control algorithm works reliably with the range of parameters we simulated, for all five types of traffic. In addition, except for SVD, the performance is insensitive to the parameters change under all tested topologies. For SVD, the algorithms does show certain sensitivity to the tested parameters. The high level conclusion that can be drawn is that (predictably) high peak-to-mean ratio SVD traffic is substantially more stressful to the queue-based admission control algorithm, but a set of parameters exists that keeps the over-admission within about -4% - +7% of the expected load even for the bursty SVD traffic.

The token bucket-based admission control algorithm shows higher sensitivity to the parameter settings compared to the virtual queue based algorithm. It is important to note here that for the token bucket-based admission control no traffic will be marked until the rate of traffic exceeds the configured admission rate by the chosen CLE. As a consequence, even with the ideal performance of the algorithms, the over-admission-percentage will not be 0, rather it is expected to equal to CLE threshold if the algorithm performs as expected. Therefore, a more meaningful metric for the token-based results is actually the over-admission-percentage (listed below) minus the corresponding (CLE threshold * 100). For example, for CLE = 0.01, one would expect that 1% over-admission is inherently embedded in the algorithm. When comparing the performance of token bucket (with the adjusted over-admission-percentage) to its corresponding virtual queue result, we found that token bucket performs only slightly worse for voice-like CBR VBR, and MIX traffic.

The results for SVD traffic require some additional commentary. Note from the results in Table A.2. in the Single Link topology the performance of the token-based solution is comparable to the performance of the queue-based scheme. However, for the RTT topology, the worse case performance for SVD traffic becomes very bad, with up to 23% over-admission in a high overload. We investigated two potential causes of this drastic degradation of performance by concentrating on two key differences between the Single Link and the RTT topologies: the difference in the round-trip times and the degree of aggregation in a per ingress-egress pair aggregate.

To investigate the effect of the difference in round-trip times, we also conducted a subset of the experiments described above using the

RTT topology that has the same RTT across all ingress-egress pairs rather than the range of RTTs in one experiment. We found out that neither the absolute nor the relative difference in RTT between different ingress-egress pairs appear to have any visible effect on the over-load performance or the fairness of both algorithms (we do not present these results here as their are essentially identical to those in Table A.2). In view of that and noting that in the RTT topology we used for these experiments for the SVD traffic, there is only 1 highly bursty flow per ingress, we believe that the severe degradation of performance in this topology is directly attributable to the lack of traffic aggregation on the ingress-egress pair basis.

(preamble)

Type	Topo	Over Admission Perc Stats			
		Queue-based		Bucket-Based	
		Min	Max	Min	Max
CBR	S.Link	0.224	1.105	-0.99	1.373
	RTT	0.200	1.192	6.495	9.403
	PLT	-0.93	0.990	-2.24	2.215
VBR	S.Link	-0.07	1.646	-2.94	2.760
	RTT	-0.11	1.830	-1.92	6.384
	PLT	-1.48	1.644	-4.34	3.707
MIX	S.Link	-0.14	1.961	-2.85	2.153
	RTT	-0.46	1.803	-3.18	2.445
	PLT	-1.62	1.031	-3.69	2.955
VTR	S.Link	-0.05	1.581	-2.36	2.247
	RTT	-0.57	1.313	-1.44	4.947
	PLT	-1.24	1.071	-3.05	2.828
SVD	S.Link	-2.73	6.525	-11.25	6.227
	RTT	-2.98	5.357	-4.30	23.48
	PLT	-4.84	4.294	-11.40	6.126

Table A.2 Parameter sensitivity: Queue-based v.s. Token Bucket-based. For the single bottleneck topologies (S. Link and RTT) the overload column represents the ratio of the mean demand on the bottleneck link to the configured admission threshold. For parking lot topologies we report the worst case result across all bottlenecks. We present here only the worst case value over the range of resulting over-admission-percentage values.

8.2.3. Effect of Ingress-Egress Aggregation

To investigate the effect of Ingress-Egress Aggregation, we fix a particular EWMA weight and CLE setting (in this case, weight=0.3, for virtual queue scheme CLE=0.05, and for the token bucket scheme CLE=0.0001), vary the level of ingress-egress aggregation by using RTT topologies with different number of ingresses.

Table A.3 shows the change of over-admission-percentage with respect to the increase in the number of ingress for both virtual queue and token bucket. For all traffic, the leftmost column in the represents the case with the largest aggregation (only two ingresses), while the right most column represents the lowest level of aggregation (expected number calls per ingress is just 1 in this case). In all experiments the aggregate load on the bottleneck is the same across each traffic type (with the aggregate load being evenly divided between all ingresses).

As seen from Table A.3. the virtual queue based approach is relatively insensitive to the level of ingress-egress aggregation. On the other hand, the Token Bucket based approach is performing significantly worse at lower levels of ingress-egress aggregation. For example for CBR (with expect 1-call per ingress), the over-admission-percentage can be as bad as 45%.

(preamble)

	Type	Number of Ingresses					
		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
Virtual		2	10	70	300	600	1000
Queue	MIX	1.080	1.163	1.105	1.042	1.132	1.098
Based		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

	Type	Number of Ingresses					
		2	10	100	300	600	1000
	CBR	0.725	0.753	7.666	21.16	33.69	44.58
		2	10	100	300	600	1800
	VBR	0.532	0.477	1.409	3.044	5.812	14.80
Token							
Bucket		2	10	100	300	600	1800
Based	MIX	0.736	0.649	1.960	4.652	10.31	27.69
		2	10	70	140	300	600
	VTR	0.758	0.889	1.335	1.694	4.128	13.28
		2	10	35	100	140	300
	SVD	-1.64	-0.93	0.237	4.732	7.103	8.799

(Table A.3 Synchronization effect with low Ingress-Egress Aggregation: Queue-based v.s. Token bucket-based)

Our investigation reveals that the cause of the poor performance of the token bucket scheme in our experiments is attributed directly to the same "synchronization" effect as was earlier described in the Termination (preemption) results in draft-zhang-pcn-performance-evaluation, and to which we refer the reader for a more detailed description of this effect. In short however, for CBR traffic, a periodic pattern arises where packets of a given flow see roughly the same state of the token bucket at the

bottleneck, and hence either all get marked, or all do not get marked. As a result, at low levels of aggregation a subset of ingresses always get their packets marked, while some other ingresses do not.

As reported in draft-zhang-pcn-performance-evaluation, in the case of Termination this synchronization effect is beneficial to the algorithm. In contrast, for Admission, this synchronization is detrimental to the algorithm performance at low aggregations. This can be easily explained by noting that ingresses which packets do not get marked continue admitting new traffic even if the aggregate bottleneck load has been reached or exceeded. Since most of the other traffic patterns contain large CBR segments, this effect is seen with other traffic types as well, although to a different extent.

A natural initial reaction can be to write-off this effect as purely a simulation artifact. In fact, one can expect that if some jitter is introduced into the strict CBR traffic pattern so that the packet transmission is longer strictly periodic, then the "synchronization" effect might be easily broken.

To verify whether this is indeed the case, we ran the experiment with same topologies and parameter settings, but with randomized version of the base traffic types. The results are summarized in Table A.4. Note, the column label with f (e.g. 0.0001) correspond to randomized traffic with a randomization-interval of f x packet-interarrival-time. It also means that on average, the packets are delayed by f x packet-interarrival-time / 2. In addition, the column of "No-Rand" actually correspond to the token bucket results in Table A.3). It turns out that indeed introducing enough jitter does break the synchronization effect and the performance of the algorithm much improves. However, it takes sufficient amount of the randomization before it is noticed. For instance, in the CBR graph, the only column that shows no aggregation effect is the one labeled with "0.05", which translates to expected packet deviation from its ideal CBR transmit time of 0.5ms. While 0.5ms per-hop deviation is not unreasonable to expect, in well provisioned networks with a relatively small amount of voice traffic in the priority queue one might find lower levels of network-induced jitter. In any case, these results indicates the "synchronization" effect can not be completely written off as a simulation artifact. The good news, however, that this effect is visible only at very low ingress-egress aggregation levels, and as the ingress-egress aggregation increases, the effect quickly disappears.

We observed the synchronization effect consistently across all types of traffic we tested with the exception of VTR. VTR also exhibits

some aggregation effect - however randomization of its CBR portion has almost have no effect on performance. We suspect this is because the randomization we perform is at packet level, while the synchronization that seems to be causing the performance degradation at low ingress-egress aggregation for VTR traffic occurs at frame-level. Although our investigation of this issue is not completed yet, our preliminary results show that if we calculating random deviation for our artificially induced jitter using frame inter-arrival time instead of packet-interarrival-time, we can reduce the over-admission percentage for VTR to roughly 3%. It is unclear however, whether such randomization at the frame level meaningfully reflects network-introduced jitter.

	No. Ingr	Randomization Interval					
		No-Rand	0.0001	0.001	0.005	0.01	0.05
CBR	2	0.725	0.683	0.784	0.725	0.772	0.787
	10	0.753	0.725	0.543	0.645	0.733	0.854
	100	7.666	5.593	2.706	1.454	1.226	0.692
	300	21.16	15.52	6.699	3.105	2.478	1.624
	600	33.69	25.51	11.41	6.021	4.676	2.916
	1000	44.58	36.20	17.03	7.094	5.371	3.076
VBR	2	0.532	0.645	0.670	0.555	0.237	0.740
	10	0.477	0.596	0.703	0.494	0.662	0.533
	100	1.409	1.236	1.043	0.810	1.202	1.016
	300	3.044	2.652	2.093	1.588	1.755	1.671
	600	5.812	4.913	3.539	2.963	2.803	2.277
	1800	14.80	12.59	8.039	6.587	5.694	4.733
MIX	2	0.736	0.753	0.627	0.751	0.850	0.820
	10	0.649	0.737	0.780	0.824	0.867	0.787
	100	1.960	1.705	1.428	1.160	1.149	1.034
	300	4.652	4.724	3.760	2.692	2.449	2.027
	600	10.31	9.629	7.289	5.520	4.958	3.710
	1000	17.21	15.96	11.05	8.700	7.382	5.061
	1800	27.69	23.46	16.53	12.04	10.84	8.563
VTR	2	0.758	0.756	0.872	0.894	0.825	0.849
	10	0.889	0.939	0.785	0.704	0.843	0.574
	70	1.335	1.101	1.066	1.181	0.978	0.946
	140	1.694	1.162	1.979	1.791	1.684	1.573
	300	4.128	4.191	3.545	3.307	3.964	3.465
	600	13.28	13.76	13.81	13.18	12.97	12.35
SVD	2	-1.64	-2.30	-2.14	-1.61	-1.01	-0.89
	10	-0.93	-1.65	-2.41	-2.98	-2.58	-2.27
	35	0.237	-0.31	-0.35	-1.02	-0.96	-2.16
	100	4.732	4.640	4.152	2.287	1.887	-0.03
	140	7.103	6.002	5.560	4.974	3.619	0.091
	300	8.799	10.72	9.840	7.530	6.281	4.270

(Table A.4 Ingress-Egress Aggregation: Token-based results for Randomized traffic))

Finally, we investigated the impact of call arrival assumptions at different levels of ingress-egress aggregation by comparing the results with Poisson and BATCH arrivals. We reported in draft-zhang-pcn-performance-evaluation that virtual queue -based

admission is relatively insensitive to the BATCH vs Poisson arrivals, even at lower aggregation levels. In contrast, the call arrival assumption does affect the performance of token bucket-based algorithm, and causes substantial degradation of performance at low ingress-egress aggregation level. An example result with CBR traffic is presented in table A.5. Here we use batch arrival with mean = 5. The results show that with the lowest aggregation, the batch arrival gives worse result than the normal Poisson arrival, however, as the level of aggregation become sufficient (e.g. 100 ingress, 10 call/ ingress), the difference becomes insignificant. This behavior is consistent across all types of traffic.

(preamble)

	No.	Deviation Interval					
	Ingr	No-Rand	0.0001	0.001	0.005	0.01	0.05
CBR	2	0.918	1.007	0.836	0.933	1.014	0.971
	10	1.221	0.936	0.767	0.906	0.920	0.857
	100	8.857	7.092	3.265	1.821	1.463	1.036
	300	29.39	22.59	8.596	4.979	4.550	2.165
	600	43.36	37.12	17.37	10.02	8.005	4.223
	1000	63.60	50.36	25.48	12.82	9.339	6.219

(Table A.5 In/Egress Aggregation with batch traffic: Token-based results)

8.2.4. Effect of Multiple Bottlenecks

The results in Table A.2 (Section 9.5.1, parameter sensitivity study) implied that from the bottleneck point of view, the performance on the multiple-bottleneck topology, for all types of traffic, is comparable to the ones on the SingleLink, for both queue-based and token bucket-based algorithms. However, the results in Table A.2 only show the worst case values over all bottleneck links. In this section we consider two other aspects of the Multiple Bottleneck effects: relative performance at individual bottlenecks and fairness of bandwidth usage between the short- and the long- haul IEAs.

8.2.4.1. Relative performance of different bottlenecks

In Table A.5, we show a snapshot of the behavior with 5 bottleneck topology, with the goal of studying the performance of different bottlenecks more closely. Here, the over-admission-percentage displayed is an average across all 15 experiments with different [weight, CLE] setting. (We do observe the same behavior in each of the individual experiment, hence providing a summarized statistics is meaningful).

One differences in token-bucket case vs the queue-based admissions in the PLT topology case revealed in Table A.6 is that there appears to be a consistent relationship between the position of the bottleneck link (how far downstream it is) and its over-admission-percentage. The data shows the further downstream the bottleneck is, the more it tends to over-admit, regardless the type of the traffic. The exact cause of this phenomenon is yet to be explained, but the effect of it seems to be insignificant in magnitude, at least in the experiments we ran.

(preamble)

	Traffic Type	Bottleneck LinkId 1	2	3	4	5
Queue Based	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
	Traffic Type	Bottleneck LinkId 1	2	3	4	5
Token Bucket Based	CBR	0.121	0.300	0.413	0.515	0.700
	VBR	-0.07	0.251	0.496	0.698	1.044
	MIX	0.042	0.350	0.468	0.716	0.924
	VTR	0.277	0.488	0.642	0.907	1.117
	SVD	-2.64	-2.50	-1.72	-1.57	-1.19

Table A.6 Bottleneck Performance: queue-based v.s. token bucket-based

8.2.4.2. (Un)Fairness Between Different Ingress-Egress pairs

It was reported in draft-zhang-pcn-performance-evaluation that virtual-queue-based admission control favors significantly short-haul connection over long-haul connections. As was discussed there, this property is in fact common for measurement-based admission control algorithms (see for example [Jamin] for a discussion). It is common

knowledge that in the limit of large demands, long-haul connections can be completely starved. We show in draft-zhang-performance-evaluation that in fact starvation of long-haul connections can occur even with relatively small (but constant) overloads. We identify there that the primary reason for it is a de-synchronization of the "congestion periods" at different bottlenecks, resulting in the long-haul connections almost always seeing at least one bottleneck and hence almost never being allowed to admit new flows. We refer the reader to that draft for more detail.

Here we investigate the comparative behavior of the token-bucket based scheme and virtual queue based scheme with respect to fairness.

The fairness is illustrated using the ratio between bandwidth of the long-haul aggregates and the short-haul aggregates. As is intuitively expected, (and also confirmed experimentally), the unfairness is the larger the higher the demand, and the more bottlenecks traversed by the long-haul aggregate Therefore, we report here the "worst case" results across our experiments corresponding to the 5x demand overload and the 5-PLT topology.

Table A.7 summaries, at 5x overload, with CLE=0.05 (for virtual queue), 0.0001(for token bucket), the fairness results to different weight and topology. We display the ratio as function of time, in 10 sec increments, (the reported ratios are averaged over the corresponding 10 simulation-second interval). The result presented in this section uses the aggregates that traverse the first bottleneck. The results on all other bottlenecks are extremely similar.

(preamble)

	Topo	Weight	Simulation Time (s)							
			10	20	30	40	50	60	70	80
Virtual Queue Based	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
	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
	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

	Topo	Weight	Simulation Time (s)							
			10	20	30	40	50	60	70	80
Token Bucket Based	PLT3	0.1	1.02	1.26	1.45	1.57	1.69	1.76	1.92	1.94
		0.5	1.07	1.41	1.89	2.36	2.89	3.63	3.70	3.82
		0.9	1.07	1.33	1.59	1.94	2.41	2.80	2.75	2.90
	PLT5	0.1	1.03	1.48	1.83	2.34	2.95	3.33	4.32	4.65
		0.5	1.08	1.53	1.90	2.44	3.04	3.42	4.47	4.83
		0.9	1.08	1.48	1.80	2.26	2.82	3.19	4.23	4.16
	PLT2	0.1	1.03	1.10	1.43	2.06	2.28	2.85	3.09	2.90
		0.5	1.07	1.32	1.47	1.72	1.71	1.81	1.89	1.94
		0.9	1.09	1.27	1.51	1.86	1.82	1.88	1.88	2.06

Table A.7 Fairness performance: Virtual Queue v.s. Token Bucket. The numbers in the cells represent the ratio between the bandwidth of the long- and short-haul aggregates. Each row represents the time series of these results in 10 simulation second increments.

To summarize, we observed consistent beatdown effect across all experiments for both virtual-queue and token-bucket admission algorithms, 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.

8.3. Termination Control

8.3.1. Ingress-Egress Aggregation Experiments

In this section, we investigate sensitivity of the Flow Termination Function of SM. From our admission control experiments it is clear that SM is extremely sensitive to very low IE-aggregation (on the order of 1-10 flows), limiting applicability of SM at these aggregation levels. We show here that the Termination Function of CL requires even more IE aggregation, as we quantify in this section.

The table below shows comparative accuracy of CL and SM at different aggregation levels in a single bottleneck topology with multiple IEAs sharing the bottleneck. As can be seen from this table, the actual degree of IE aggregation necessary to achieve an over-termination within 10% ranges from ~50 to about ~150 for different traffic types (note that extremely bursty high-rate SVD traffic the maximum number of flows in an IEA we ran was 69, which was not sufficient to reach a 10% over-termination error bound we targeted. We did not run higher number of SVD flows per IEA due to time limitations).

	No. Ingre	Flow per Ingre	Over-Term. Perc. CL	SM
CBR	2	285	-0.106	4.112
	10	57	0.388	6.710
	35	16	1.035	14.64
	70	8	0.727	16.39
VBR	2	849	0.912	2.808
	10	169	4.032	10.47
	35	48	2.757	22.26
	100	16	3.966	22.52
MIX	2	662	1.297	3.672
	10	132	2.698	7.809
	35	37	1.978	14.83
	100	13	4.265	17.29
VTR	2	158	3.513	3.718
	10	31	4.532	14.82
	35	9	6.842	22.95
	70	4	8.458	22.31
SVD	2	69	7.811	20.90
	10	13	10.69	27.38
	35	4	8.322	20.78

Table A.8 Over-termination comparison between CL and SM at medium/ high IE aggregation

It turns out that the reason for this higher sensitivity to low ingress-egress aggregation lies in the non-uniformity in the marking distribution across different IEAs. As a result of this non-uniformity , when traffic is close below the (implicit) termination threshold at the bottleneck , some IEAs get excessively marked, causing a false termination event at the corresponding PCN-ingress-nodes, in turn causing extra over-termination.

	No. Ingre	Flow per Ingre	Over-Term. Perc. SM	SM-SM
CBR	2	285	4.112	2.243
	10	57	6.710	3.142
	35	16	14.64	6.549
	70	8	16.39	8.496
VBR	2	849	2.808	0.951
	10	169	10.47	4.096
	35	48	22.26	6.987
	100	16	22.52	8.567
MIX	2	662	3.672	2.574
	10	132	7.809	3.822
	35	37	14.83	4.936
	100	13	17.29	6.956
VTR	2	158	3.718	3.866
	10	31	14.82	7.507
	35	9	22.95	10.29
	70	4	22.31	8.528
SVD	2	69	20.90	9.272
	10	13	27.38	12.46
	35	4	20.78	10.14

Table A.9 Over-termination comparison between SM and SM with smoothed trigger. Here EWMA weight = 0.9 (heavy history)

We investigated whether this effect can be removed by smoothing (using EWMA) the ratio between marked and unmarked traffic that we use at the ingress node to trigger the termination event. Table A.9 above presents the results for the EWMA weight of 0.9 corresponding to a long history. It can be seen that such smoothing does in fact help reduce over-termination. However, it also increases the reaction time of flow termination. This increased latency grows for larger U and decreases with the increase in the excess load over the (implicit) termination threshold.

Table A.10 quantifies this extra delay (Note: these results are for 100 ms measurement intervals at the ingress, and for negligible round-trip time. The actual extra latency is obtained by adding the RTT to the results of table 8.3.

(preamble)

U \ R	0.2	0.3	0.4	0.5
1.1	0.5	0.4	0.3	0.3
1.3	1.0	0.8	0.7	0.7
1.5	1.4	1.1	1.0	0.9
1.7	1.6	1.4	1.2	1.1
1.9	1.8	1.6	1.4	1.3
2.0	1.9	1.6	1.5	1.4

Table A.10. Additional latency due to smoothing of termination signal and the PCN-ingress-node (in sec; W=0.9)

We note that this smoothing is only necessary at the lower range of the IE aggregation levels we considered, and is not necessary as soon as the aggregation level reaches 50-150 flows (for different traffic types) in our experiments. For the lower aggregation level, the smoothing may be useful, at the expense of the additional latency.

8.3.2. Multiple Bottlenecks Experiments

As discussed in Section 7.3, the fact that SM marks traffic when the bottleneck load is below (implicit) termination threshold but above the configured admission threshold, causes additional "beat-down" effect of flows traversing multiple bottlenecks, compared to the beat-down effect already observed for CL in draft-zhang-performance-evaluation.

We start with the setup with 2- and 5-PLT topology similar to that of draft-zhang-performance-evaluation. 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 below shows the comparative over-termination on the bottlenecks (2 and 5 PLT topology) for both CL and SM. The bottleneck rows are ordered based on the flow traversal order (from upstream to downstream).

As in the results we presented in draft-zhang-performance-evaluation, we report over-termination compared to the "reference" over-termination which we compute as follows for the multi-bottleneck topology. We take each link in the topology separately and compute

the "rate-proportionally fair" rates that each IEA sharing this bottleneck will need to be reduced to (in proportion to their demands), so that the load on that bottleneck independently becomes equal to the termination threshold (this threshold being implicit for SM, explicit for CL), assuming the initial sum of rates exceeds this threshold. After this is done independently for each bottleneck, we assign each IEA the smallest of its scaled down rates across all bottlenecks. We then compute the "reference" utilization on each link by summing up the scaled down rates of each IEA sharing this link. Our over-termination is then reported in reference to this "reference" utilization. We note that this reference utilization may frequently be already below the termination threshold of a given link. This can happen easily in the case when a large number of flows sharing a given link is "bottlenecked" elsewhere.

Topo.		CBR		VBR		VTR		SVD	
2/5 PLT		CL	SM	CL	SM	CL	SM	CL	SM
2	BN1	5.93	20.93	6.49	21.31	9.07	21.88	9.28	23.18
	BN2	0.56	9.89	2.21	9.89	3.61	8.74	7.99	12.92
5	BN1	9.63	35.04	10.9	34.06	11.41	36.30	14.23	39.37
	BN2	4.54	23.51	6.19	22.83	5.66	23.53	9.67	28.45
	BN3	2.05	23.36	2.46	23.18	3.47	24.64	5.73	27.01
	BN4	0.90	23.78	1.40	23.46	3.13	24.02	3.98	27.59
	BN5	0.00	24.08	0.30	23.11	2.81	23.83	5.54	28.45

Table A.11 Over-termination comparison of CL and SM for 2 and 5 PLT topology

We note that in these experiments SM does significantly worse than CL across all traffic. The most upstream bottleneck suffers the most over-termination due to the fact that the long-haul IA gets severely beaten down, while the short-haul flows terminate their fair share. (In this experiment almost 90% of the long-haul IA is terminated).

In our PLT setup each IEA is heavily aggregated, so we do not expect smoothing of the termination trigger to have a significant effect. Table A.12 Summarizes the performance of the same setup with smoothing.

Topo.	CBR		VBR		VTR		SVD	
2/5 PLT	SM	SM-SM	SM	SM-SM	SM	SM-SM	SM	SM-SM
2 BN1	20.93	14.29	21.31	13.18	21.88	15.14	23.18	16.53
BN2	9.89	14.37	9.89	13.32	8.74	14.22	12.92	16.89
5 BN1	35.04	24.27	34.06	23.17	36.30	23.98	39.37	30.83
BN2	23.51	24.15	22.83	23.87	23.53	24.21	28.45	31.60
BN3	23.36	23.94	23.18	23.67	24.64	25.23	27.01	29.65
BN4	23.78	24.56	23.46	23.86	24.02	25.04	27.59	29.25
BN5	24.08	24.24	23.11	24.08	23.83	24.95	28.45	29.94

Table A.12. Over-termination comparison of SM and smoothed SM for 2 and 5 PLT topology

Smaller over-termination on the upstream bottlenecks, especially 1) is due to the fact that with smoothing, the short IEA on the bottleneck 1 did not terminate at all, which makes it the bottleneck 1 less over-terminated than in the case of SM. The reason for this is that in the smooth-SM, the additional markings received (due to multi-bottleneck effect) by the long IEA make its smoothing process much faster than the short IEA. Again in this particular setup, there is enough flows in the long IEA to be terminated and bring the bottleneck load way below the termination threshold, while short IEA never gets to react.

Our next task was to investigate whether particularly bad performance of SM in this case is a common occurrence. To do so, we took the 5-PLT topology and generated on the order of ~50,000 random traffic matrices, and random settings of the admission threshold and the parameter U, resulting in creation of anywhere from 0 to 5 bottlenecks on this topology in each experiment. We limited the range of U from 1.0 to 3.0, and the maximum overload on any link was at most ~10x of the (implicit) termination threshold. To enable us to run these many experiments in a reasonable time, we implemented a fluid model, and later compared its accuracy with the packet simulations on a subset of topologies to confirm reliability of the fluid model simulation results (see below).

Table A.13 gives a summary of the experimental frequency of the setups with a particular range of the termination error on the most loaded bottleneck. In addition, we checked whether the termination error in those setups is so big as to bring the load on the most loaded bottleneck below its admission threshold. This data is shown by summarizing the experimental frequency of experiments where the resulting load after termination (End-Load) is above the admission threshold . Since the frequency of these cases depends on the number

of bottlenecks in the experiment, we report this by the number of bottlenecks.

We split the results in Table A.13 into those with small U (where the (implicit) termination threshold is very close to the admission threshold), medium U (where the implicit termination threshold is between 1.2 - and 2 times the admission threshold, and large U (greater than 2 times admission threshold).

Given the accuracy results from our packet experiments, it seems that the reasonable setting of U must be at least 120% of the admission threshold to reduce the probability that the termination error will bring the load below admission threshold. Therefore, we present the results for small U for completeness only. We also believe that in practice setups with large U>2.0 should be rare, and hence we report the large U results separately as well.

At a high level the results of table A.13 imply that for small U, over-termination is small, but it is enough to frequently drive the bottleneck load below admission threshold, especially with larger number of bottlenecks. For large U, the over-termination is larger, but the bottleneck load almost never falls below admission threshold. Finally, for medium U, which, in our opinion is the case of practical importance, the over-termination for SM is below 10% in ~65% of the experiments, is within 20% for about 30% of the experiments, and between 30 an d 40% for the remaining 5%. In contrast, CL remains within 10% over-termination most of the time. For this medium U, this over-termination almost never causes the bottleneck load to drop below admission threshold for up to 3 bottlenecks, while ~10% of the 4 bottleneck cases and ~20% of the 5-bottleneck cases do drop below the admission threshold. Note that CL also occasionally drives the load below admission threshold, albeit not as often as SM - e.g. in ~3% of the simulations for the 4 bottlenecks and about 7% for the 5 bottlenecks, for medium U.

We note that the cause of the after-termination event load falling below admission threshold, as well as a partial cause for the over-termination reported below is partially due to the fact that some of the flows going through the most overloaded bottleneck are nevertheless passing another bottleneck elsewhere. Even if the overall overload on that other bottleneck may not be as high, that (other) bottleneck nevertheless may be driving some of the IEAs down to a smaller rate than the bottleneck with the largest overload we are considering. This is confirmed by the fact that the Reference termination occasionally also falls below the admission threshold as well - see REF rows of the "End-load Above Threshold" tables in Table A.13.

(preamble)
Small U Total Expr: 5185 1.0 < U <= 1.2

Alg.	Distribution of Over-Termination Percentage				
	0-10%	10-20%	20-30%	30-40%	40-50%
CL	0.986	0.013	0.000	0.000	0.000
SM	0.968	0.031	0.000	0.000	0.000

Alg.	Fract. End-load Above Admission Threshold					
	0 BN	1 BN	2 BN	3 BN	4 BN	5 BN
CL	1	0.914	0.851	0.732	0.490	0.357
SM	1	0.934	0.869	0.725	0.488	0.374
REF	1	0.980	0.982	0.977	0.978	0.968

Medium U Total Expr: 19689 1.2 < U < 2.0

Alg.	Distribution of Over-Termination Percentage				
	0-10%	10-20%	20-30%	30-40%	40-50%
CL	0.985	0.013	0.000	0.000	0.000
SM	0.659	0.294	0.043	0.001	0.000

Alg.	Fract. End-load Above Admission Threshold					
	0 BN	1 BN	2 BN	3 BN	4 BN	5 BN
CL	1	0.991	0.993	0.991	0.969	0.928
SM	1	0.990	0.991	0.977	0.909	0.831
REF	1	0.991	0.994	0.994	0.996	0.993

Large U Total Expr: 25129 2.0 <= U <= 3.0

Alg.	Distribution of Over-Termination Percentage				
	0-10%	10-20%	20-30%	30-40%	40-50%
CL	0.984	0.014	0.000	0.000	0.000
SM	0.254	0.384	0.275	0.075	0.008

Alg.	Fract. End-load Above Admission Threshold					
	0 BN	1 BN	2 BN	3 BN	4 BN	5 BN
CL	1	0.998	0.998	0.997	0.997	0.996
SM	1	0.997	0.995	0.992	0.969	0.945
REF	1	0.998	0.998	0.997	0.998	0.999

Table A.13. Distribution of over-termination percentage and frequency of the load after termination event (denoted "End-Load") remaining above admission threshold. The REF rows in the "End-load Above Admission Threshold" tables correspond to the Reference termination against which the over-termination percentage is computed

To investigate how the fluid simulation results relate to the packet simulation, we also ran a subset of approximately 2000 experiments through a packet simulator with the same parameter settings and traffic loads as the corresponding fluid simulations, and compared the results. We found that the error in most experiments is relatively small, allowing us to conjecture that statistics of the fluid experiments adequately approximate the expected packet-level results in these experiments.

The distribution of the error between the fluid and packet simulation results for these experiments is shown in the following table:

(preamble)

Alg.	Error Dist. in Over-Term. Perc. (Fluid-Packet)				
	-0.2~-0.1	-0.1~0.0	0.0~0.1	0.1~0.2	0.2-0.4
CL	0.000	0.032	0.963	0.000	0.000
SM	0.003	0.334	0.640	0.013	0.000

Table A.14 Error Between Fluid and Packet Simulations in about 2000 experiments

As can bee seen, the error is relatively contained.

Finally, we ran a similar set of fluid experiments with the smoothed trigger signal (see section 8.3.1), and found that there is no visible difference in the statistical performance of the smoothed version and non-smoothed version of the algorithm. In some cases smoothing performed better than the non-smoothed version, as in the example reported in Tables A11 and A12 , and in other cases non-smoothed version outperformed the smoothed version, with the overall distribution of over-termination errors remaining extremely similar for smoothed and non-smoothed versions. We therefore conclude that smoothing is necessary only to deal with low levels of ingress-egress aggregations, and have no effect on the over-termination in the multi-bottleneck scenario as long as the IEAs are sufficiently aggregated.

9. Appendix B. Controlling The Single Marking Configuration with a Single Parameter

9.1. Assumption

This section assumes that TM-marking is used for SM marking encoding.

9.2. Details of the Proposed Enhancements to PCN Architecture

9.2.1. PCN-Internal-Node

No substantive change is required for the PCN framework (as defined in [I-D.eardley-pcn-architecture]) to enable SM Operation in the PCN Internal Node. The architecture already allows the implementation of only one marking and metering algorithm at the PCN-internal-node.

However, we propose to rename the terms "configured-admissible-rate" and "configured-termination-rate" to "Type Q threshold" and "Type R"

threshold. The architecture should allow configuring either one of these thresholds or both at the PCN-ingress node. The type of the threshold determines the type of the marking semantics/algorithm associated with the threshold.

9.2.2. PCN-Egress-Node

The only proposed change at the PCN-egress-node is the addition of a single (globally defined) configuration constant U. The setting of this constant defines the type of marking CLE is measured against. If U=1, the system defaults to the dual-marking behavior and the CLE is measured against Type Q marked packets. If U>1, the CLE is measured against Type R marked traffic. No other change is required.

In more detail,

- If U=1, a PCN-egress-node expects to receive either Type Q marking only (the network implements virtual-queue-based admission only), or Type R marking only (the system implements excess-rate-based flow termination only), or both (the system implements dual-marking admission and termination).

- If U>1, a PCN egress node expects to receive only type-R marking (the network implements single-marking approach).

- If U=1 and Type-Q marking is received (as indicated by the encoding in the PCN packets), then the PCN-egress-node always measures the CLE (fraction of traffic carrying Type-Q marks) on a per-ingress basis against Type Q marking. This represents no change (other than renaming "admission-marked-packets" to "type Q-marked" packets) compared to the current architecture. The PCN-egress-node then signals the (type Q-based) CLE to the PCN-ingress-node - again as already enabled by the current PCN architecture.

- If U=1 and a PCN-egress-node receives "Type R" marking (as indicated in the encoding of the PCN packets), it measures sustainable rate with respect to Type-R marked traffic, (i.e. it measures the amount of traffic without the "Type-R" marks). This also is just a renaming change (with termination-marking renamed to "Type R" marking) and is fully compatible with the current PCN architecture.

- If U > 1, the PCN-egress node computes both the CLE and the Sustainable rate with respect to Type-R marking.

- Once computed, the CLE and/or the Sustainable rate are communicated to the PCN-ingress-node as described in

[I-D.eardley-pcn-architecture].

9.2.3. PCN-Ingress-Node

The only proposed change at the PCN-ingress-node is the addition of a single (globally defined) configuration constant U (in fact, this is the same constant as defined for the PCN-egress-node, so U in fact is a per PCN-boundary-node constant; its value however is assumed to be global for all PCN-boundary nodes in the PCN-domain (or at least a subset of nodes communicating with each other only)). The value of this constant is used to multiply the sustainable rate received from a given PCN-egress-node to compute the rate threshold used for flow termination decisions. The value U=1 corresponds to the dual-marking approach, and results in using the sustainable rate received from the PCN-egress-node directly. The value U>1 corresponds to the SM approach and its (globally defined) value signifies the desired system-wide implicit ratio between flow termination and flow admission thresholds as described in Section 2.

Note that constant U is assumed to be defined per PCN-boundary node (i.e. the ingress and the egress functions of the PCN-boundary-node use the same configuration constant to guide their behavior.

In more detail:

- A PCN-ingress-node receives CLE and/or Sustainable Rate from each PCN-egress-node it has traffic to. This is fully compatible with PCN architecture as described in [I-D.eardley-pcn-architecture].

- A PCN-ingress-node bases its admission decisions on the value of CLE. Specifically, once the value of CLE exceeds a configured threshold, the PCN-ingress-node stops admitting new flows. It restarts admitting when the CLE value goes down below the specified threshold. This is fully compatible with PCN architecture as described in draft-earley-pcn-architecture-00.

- A PCN-ingress node receiving a Sustainable Rate from a particular PCN-egress node measures its traffic to that egress node. This again is fully compatible with PCN architecture as described in draft-earley-pcn-architecture-00.

- The PCN-ingress-node computes the desired Termination Rate to a particular PCN-egress-node by multiplying the sustainable rate from a given PCN-egress-node by the value of the configuration parameter U. This computation step represents a proposed change to the current version of [I-D.eardley-pcn-architecture].

- Once the Termination Rate is computed, it is used for the flow termination decision in a manner fully compatible with [I-D.eardley-pcn-architecture]. Namely the PCN-ingress-node compares the measured traffic rate destined to the given PCN-egress-node with the computed Termination rate for that egress node, and terminates a set of traffic flows to reduce the rate exceeding that Termination rate. This is fully compatible with [I-D.eardley-pcn-architecture].

10. Security Considerations

TBD

11. References

11.1. Normative References

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

11.2. Informative References

[I-D.babiarz-pcn-3sm]
Babiarz, J., "Three State PCN Marking", draft-babiarz-pcn-3sm-00 (work in progress), July 2007.

[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-04 (work in progress), July 2007.

[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.

[I-D.westberg-pcn-load-control]
Westberg, L., "LC-PCN: The Load Control PCN Solution", draft-westberg-pcn-load-control-02 (work in progress), November 2007.

[I-D.zhang-pcn-performance-evaluation]
Zhang, X., "Performance Evaluation of CL-PHB Admission and Termination Algorithms", draft-zhang-pcn-performance-evaluation-02 (work in progress), July 2007.

11.3. References

[Jamin] "A Measurement-based Admission Control Algorithm for Integrated Services Packet Networks", 1997.

[Menth] "PCN-Based Resilient Network Admission Control: The Impact of a Single Bit", 2007.

Authors' Addresses

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

Email: acharny@cisco.com

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

Email: joyzhang@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

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

Email: vliatsos@cisco.com

Full Copyright Statement

Intellectual Property

Acknowledgment

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