University of Würzburg
Institute of Computer Science
Research Report Series

Applicability of PCN-Based Admission Control

Michael Menth and Frank Lehrieder

Report No. 468 March 2010

University of Wuerzburg, Germany
Institute of Computer Science
Chair of Communication Networks
Am Hubland, D-97074 Würzburg, Germany
phone: (+49) 931-3186644, fax: (+49) 931-3186632
{menth|lehrieder}@informatik.uni-wuerzburg.de

Applicability of PCN-Based Admission Control

Michael Menth and Frank Lehrieder
University of Würzburg, Institute of Computer Science, Germany

***Abstract*—Pre-congestion notification (PCN) marks packets when the PCN traffic rate exceeds an admissible link rate and this marking information is used as feedback from the network to take admission decisions for new flows. This idea is currently under standardization in the IETF. Different marking algorithms are discussed and various admission control (AC) algorithms are proposed that decide based on the packet markings whether further flows should be accepted or blocked. In this paper, we investigate the applicability of PCN-based AC under challenging conditions and show the limitations of different marking and AC algorithms.**

***Index Terms*—Admission control, QoS, feedback systems, packet marking.**

I. Introduction

Pre-congestion notification (PCN) is a new mechanism currently developed by the IETF to facilitate PCN-based admission control (AC) and flow termination (FT) primarily for wired networks and inelastic realtime flows [1]. Traffic belonging to the PCN service class is prioritized over non-PCN traffic such that PCN traffic does not suffer from packet loss or delay when overload occurs in a network. In addition, the rate of admitted PCN traffic is controlled such that overload cannot evolve within the PCN traffic class under normal operation. If the rate of PCN traffic becomes too large in case of a failure, FT can remove some of the admitted traffic to restore a controlled load condition [2] on the overloaded link.

The idea of PCN is that routers mark PCN packets on a specific outgoing link when its PCN traffic rate exceeds its configurable admissible or supportable rate. Currently, PCN is developed for a domain concept. That means egress nodes evaluate the packet markings. They communicate the information about marked PCN packets, i.e. PCN feedback, to ingress nodes which block admission requests for new PCN flows if required. Methods for PCN-based AC consist of two components: the packet metering and marking algorithm and the actual AC algorithm that turns the obtained packet markings into AC decisions. Quite many PCN algorithms require the notion of ingress-egress aggregates (IEAs) which is the ensemble of all PCN flows between a specific pair of ingress and egress nodes [3].

This paper studies PCN-based AC under challenging conditions. In particular we investigate whether PCN-based AC can limit the rate of admitted PCN traffic at all and if so what the level of overadmission is in case the admitted PCN traffic rate exceeds the desired admissible rate. Simple probe-based AC (PBAC) requires that markers mark all PCN packets if pre-congestion occurs on a bottleneck link, but some tweaks adapt PBAC to marking schemes that mark only a small fraction of packets. It is not clear whether or how well this adaptation works. Some AC algorithms require a sufficiently high packet frequency per IEA or at least one admitted flow per IEA to work properly. Multipath routing can lead to difficulties for IEA-based AC methods. PCN captures and reports the feedback of all active flows and an implicit requirement is that flows start transmission immediately after admission. It is not clear what happens when these conditions are not met. Our results show that AC mechanisms can break due to unlimited overadmission, they can become inaccurate due to limited and predictable overadmission, or they can become inefficient due to early blocking and waste of resources. They are a contribution to the standardization process and help network providers to choose the right AC algorithms for their application scenario.

The paper is structured as follows. Section II explains PCN, metering and marking algorithms as well as various AC algorithms. Section III reviews related work. Section IV studies AC methods under challenging conditions. Finally, Section V summarizes this work and draws conclusions.

II. Admission Control Based on Pre-Congestion Notification (PCN)

In this section we explain the general idea of PCN-based admission control (AC) and flow termination (FT) and illustrate their application in a domain context in the Internet. We explain the metering and marking algorithms briefly and the AC algorithms in more detail.

A. Pre-Congestion Notification (PCN)

PCN defines a new traffic class that receives preferred treatment by PCN nodes. It provides information to support AC and FT for this traffic type. PCN introduces an admissible and a supportable rate threshold ($AR(l)$, $SR(l)$) for each link l of the network which imply three different load regimes as illustrated in Figure 1. If the PCN traffic rate $r(l)$ is below $AR(l)$, there is no pre-congestion and further flows may be admitted. If the PCN traffic rate $r(l)$ is above $AR(l)$, the link is AR-pre-congested and the rate above $AR(l)$ is AR-overload. In this state, no further flows should be admitted. If the PCN traffic rate $r(l)$ is above $SR(l)$, the link is SR-pre-congested and the rate above $SR(l)$ is SR-overload. In this state, some already admitted flows should be terminated to reduce the PCN rate $r(l)$ below $SR(l)$.

This work was funded by Deutsche Forschungsgemeinschaft (DFG) under grant TR257/18-2. The authors alone are responsible for the content of the paper.

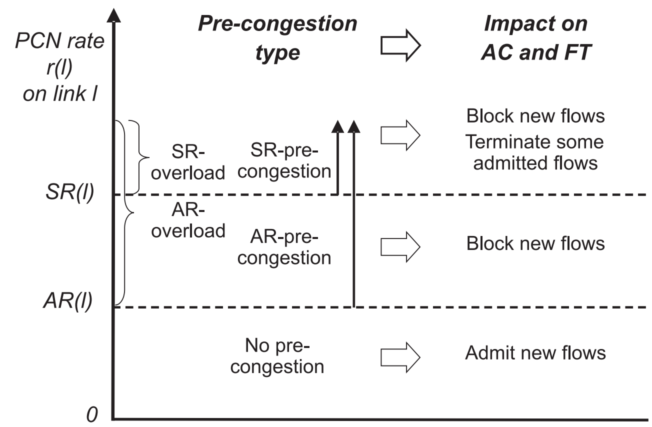


Fig. 1. The admissible and the supportable rate ($AR(l), SR(l)$) define three types of pre-congestion.

B. Edge-to-Edge PCN

Edge-to-edge PCN assumes that some end-to-end signalling protocol (e.g. SIP or RSVP) or a similar mechanism requests admission for a new flow to cross a so-called PCN domain similar to the IntServ-over-DiffServ concept [4]. Thus, edge-to-edge PCN is a per-domain QoS mechanism and an alternative to RSVP clouds or extreme capacity overprovisioning. This is illustrated in Figure 2. Traffic enters the PCN domain only through PCN ingress nodes and leaves it only through PCN egress nodes. Ingress nodes set a special header codepoint to make the packets distinguishable from other traffic and the egress nodes clear the codepoint. The nodes within a PCN domain are PCN nodes. They monitor the PCN traffic rate on their links and possibly remark the traffic in case of *AR*- or *SR*-pre-congestion. PCN egress nodes evaluate the markings of the traffic and send a digest to the AC and FT entities of the PCN domain.

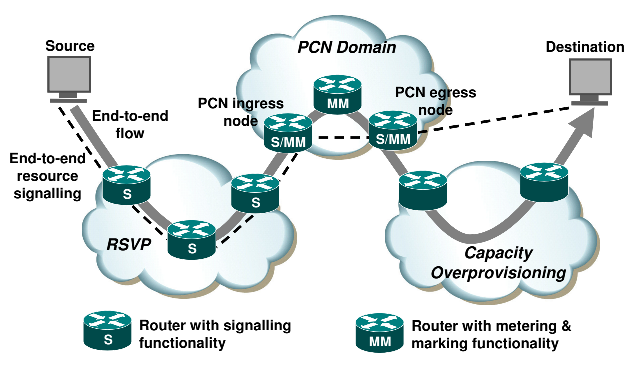


Fig. 2. Edge-to-edge PCN is triggered by admission requests from external signalling protocols and guarantees QoS within a single PCN domain.

C. PCN Feedback

PCN nodes re-mark PCN packets depending on the load regime. At the ingress node, all packets are marked with "no-pre-congestion" (NP). In case of *AR*-pre-congestion, some or all NP-marked packets are re-marked to "admission-stop" (AS). In case of *SR*-pre-congestion, some NP- and AS-marked packets are re-marked to "excess traffic" (ET). The packet markings are analyzed by the egress routers and a digest thereof is communicated to ingress routers based on which they admit or block admission requests or even terminate already admitted flows. As we focus only on the admission control part, we abandon now the idea of *SR* threshold, *SR*-pre-congestion, and ET-marking for the rest of the paper.

D. PCN Metering and Marking

There are two basic marking strategies: excess and exhaustive marking. A token bucket based meter tracks whether a certain reference rate is exceeded. Exhaustive marking marks all packets when the PCN traffic rate exceeds the reference rate. Excess marking marks only those packets that exceed the reference rate. When the reference rate is set to the admissible rate, exhaustive marking marks all packets in case of *AR*-pre-congestion and yields a very clear signal while excess marking marks only a subset of the packets and makes AC decisions more difficult. Nevertheless, excess marking is attractive because it can be implemented with only few modifications of existing hardware and a single excess marker can support both AC and FT [3], [5] while this is not possible with exhaustive marking.

E. Algorithms for PCN-Based Admission Control

We review various methods that detect *AR*-pre-congestion and stop the admission of further flows. They can be classified into ingress-egress aggregate (IEA) based AC and probe-based AC. All presented algorithms basically work with exhaustive and excess marking to signal *AR*-pre-congestion.

1) IEA-Based AC (IEABAC): One class of AC algorithms relies on PCN feedback from ingress-egress aggregates (IEAs), i.e. all flows between a specific PCN ingress and egress node, and we call it IEA-based AC (IEABAC). Each IEA is associated with an AC state K which is turned to *admit* or *block* depending on the PCN feedback. When a new flow requests admission, the AC entity needs to find out which IEA the new flow belongs to and then it admits or blocks it depending on the AC state K of that IEA. In the following, we present two different AC algorithms that control the AC state K of an IEA based on aggregated PCN feedback. Pseudocode for both algorithms is provided in [6].

a) CLE-Based AC (CLEBAC): The PCN egress node measures the rates of AS-marked and non-AS-marked traffic (ASR, $nASR$) per IEA [5], [7]. This is done based on measurement intervals of duration D_{MI}. Then, the congestion level estimates (CLEs) $CLE = \frac{ASR}{ASR+nASR}$ are calculated. If the CLE value exceeds an admission-stop threshold T_{CLE}^{AStop}, the AC state K is turned to *block*, if it falls below an admission-continue threshold T_{CLE}^{ACont}, the AC state K is turned to *admit*; otherwise, the AC state K is not changed. This hysteresis avoids state oscillation. Thus, the method depends on 3 parameters: D_{MI}, T_{CLE}^{AStop}, and T_{CLE}^{ACont}.

b) Observation-Based AC (OBAC): The PCN egress node observes the packet streams per IEA and turns the AC state K of an IEA to *block* when it observes an AS-marked packet [8]. It turns the state back to *admit* when it has not observed an AS-marked packet for D_{block}^{min} time. D_{block}^{min} is the only configuration parameter for OBAC.

2) Probe-Based AC (PBAC): With probing [8], one or more unmarked probe packets are sent by the PCN ingress node upon an admission request. They must have the same source and destination address and port as future data packets to guarantee that they are routed on the same path as future data packets also in case of multipath routing. The probe packets are intercepted by the PCN egress node. If they are all unmarked, the new flow is admitted, otherwise it is blocked.

3) Implicit Probing: An obvious disadvantage of simple PBAC is that a PCN ingress node must trigger probe packets when it receives an admission request. The required admission decision is delayed until the feedback of the probe packets returns. As a consequence, PCN ingress nodes must buffer pending admission requests. This increases their complexity and delays call setups. To avoid that, probing can be done implicitly by reusing messages of the end-to-end signalling protocol [8]. For instance, the resource reservation protocol (RSVP) [9] can be modified. The first PATH message of a call setup is exploited for probing purposes. If the PCN egress node receives an AS-marked PATH message for a new connection, it simply blocks the call by returning a PATH-TEAR message. Otherwise, the PATH message is forwarded downstream and the new call will immediately be accepted by the PCN ingress node when the corresponding RESV message returns upstream to ask for admission. Hence, implicit probing does not introduce additional probing delay, but its applicability is limited to PBAC using only a single probe packet and to scenarios with extensible end-to-end signalling protocols.

III. Related Work

We first review related work regarding other marking mechanisms and stateless core concepts for AC because they can be viewed as historic roots of PCN. Then, we give a short summary of related PCN studies.

A. Related Marking Mechanisms

We present RED and ECN because they can be seen as precursors of PCN marking.

1) Random Early Detection (RED): RED was originally presented in [10], and in [11] it was recommended for deployment in the Internet. It was designed to detect incipient congestion by measuring a time-dependent average buffer occupation *avg* in routers and to take appropriate countermeasures. That means, packets are dropped or marked to indicate congestion to TCP senders and the probability for that action increases linearly with the average queue length *avg*. The value of *avg* relates to the physical queue size which is unlike PCN metering that relates to the configured admissible or supportable rate.

2) Explicit Congestion Notification: Explicit congestion notification (ECN) is built on the idea of RED to signal incipient congestion to TCP senders in order to reduce their sending window [12]. Packets of non-ECN-capable flows can be differentiated by a "not-ECN-capable transport" (not-ECT, '00') codepoint from packets of a ECN-capable flow which have an "ECN-capable transport" (ECT) codepoint. In case of incipient congestion, RED gateways possibly drop not-ECT packets while they just switch the codepoint of ECT packets to "congestion experienced" (CE, '11') instead of discarding them. This improves the TCP throughput since packet retransmission is no longer needed. Both the ECN encoding in the packet header and the behavior of ECN-capable senders and receivers after the reception of a marked packet is defined in [12]. ECN comes with two different codepoints for ECT: ECT(0) ('10') and ECT(1) ('01'). They serve as nonces to detect cheating network equipment or receivers [13] that do not conform to the ECN semantics. The four codepoints are encoded in the (currently unused) bits of the differentiated services codepoint (DSCP) in the IP header which is a redefinition of the type of service octet [14]. The ECN bits can be redefined by other protocols and [15] gives guidelines for that. This may be useful for the encoding of PCN codepoints, but this aspect is not the focus of this paper.

B. Admission Control

We briefly review some specific AC methods that can be seen as forerunners of the PCN principle.

1) Admission Control Based on Reservation Tickets: To keep a reservation for a flow across a network alive, ingress routers send reservation tickets in regular intervals to the egress routers. Intermediate routers estimate the rate of the tickets and can thereby estimate the expected load. If a new reservation sends probe tickets, intermediate routers forward them to the egress router if they have still enough capacity to support the new flow and the egress router bounces them back to the ingress router indicating a successful reservation; otherwise, the intermediate routers discard the probe tickets and the reservation request is denied. The tickets can also be marked by a packet state. Several stateless core mechanisms work according to this idea [16]–[18].

2) Admission Control Based on Packet Marking: Gibbens and Kelly [19], [20] theoretically investigated AC based on the feedback of marked packets whereby packets are marked by routers based on a virtual queue with configurable bandwidth. This core idea is adopted by PCN. Marking based on a virtual instead of a physical queue also allows to limit the utilization of the link bandwidth by premium traffic to arbitrary values between 0 and 100%. Karsten and Schmitt [21], [22] integrated these ideas into the IntServ framework and implemented a prototype. They point out that the marking can also be based on the CPU usage of the routers instead of the link utilization if this turns out to be the limiting resource for packet forwarding.

3) Resilient Admission Control: Resilient admission control admits only so much traffic that it still can be carried after rerouting in a protected failure scenario [23]. It is necessary since overload in wide area networks mostly occurs due to link failures and not due to increased user activity [24]. It can be implemented with PCN by setting the admissible rate thresholds $AR(l)$ low enough such that the PCN rate $r(l)$ on a link l is lower than the supportable rate threshold $SR(l)$ after rerouting.

C. Related Studies in PCN

An overview of PCN including a multitude of AC and FT mechanisms is given in [3]. In [25], a high level summary is provided about a large set of simulation results regarding PCN-based AC and FT and shows that these methods work well in most studied cases.

Ramp marking and threshold marking are two different implementation options for exhaustive marking. Their impact on packet marking probabilities has been investigated in [26]. It turned out that threshold marking is as good as ramp marking which excluded ramp marking from further consideration because it is more complex than threshold marking.

A two-layer architecture for PCN-based AC and FT was presented in [6] and flow blocking probabilities have been studied for single aggregates and static load conditions. In this work, we consider the evolution of the admitted PCN traffic rate on a bottleneck link that is possibly composed by the traffic of many aggregates and provide results about potential overadmission. In contrast to [25], we focus in this work on the behavior of PCN-based AC under challenging conditions, i.e., on scenarios where the proposed mechanisms might not work as desired. We provide an understanding of these problems which helps to discern whether these methods are applicable in specific application scenarios.

The work presented in [27] proposes various algorithms for PCN-based marked flow termination (MFT) and gives recommendations for their configuration. In [28], MFT is adapted to PCN marking based on *AR*-overload and its performance is evaluated. Overtermination due to multiple bottlenecks is investigated in [29].

The efficiency of resilient PCN-based AC with flow termination and other resilient AC methods without flow termination in optimally dimensioned networks is evaluated in [30]. An additional investigation about how *AR* and *SR* thresholds should be set in PCN domains with resilience requirements is contained in [31]. Furthermore, it studies how link weights should be set in IP networks in order to maximize the admissible traffic rates. The authors of [32] investigate the impact of admissible and supportable rate thresholds on the admission and termination of on/off traffic.

IV. PCN-Based AC under Challenging Conditions

In this section we investigate how well PCN-based AC can block additional traffic under heavy load conditions, i.e., when AC should become active. We first look at PBAC, then at CLEBAC with exhaustive and excess marking and report our experience with OBAC. Thereby, we identify scenarios where the respective AC algorithms require special care for parameter settings or do not work as desired. Furthermore, we study CLEBAC and OBAC in case of multipath routing and investigate the impact of initial media delay on the behavior of general PCN-based AC.

A. Simulation Setup

We simulate the time-dependent traffic rate on a PCN-based admission-controlled bottleneck link in different networking scenarios using a packet-based custom-made simulation written in Java. The bottleneck link carries PCN traffic from n_{IEA} different ingress-egress aggregates (IEAs), each of which has on average n_{IEA}^{flows} homogeneous PCN flows. We use the following default settings. Flows of a single IEA arrive according to a Poisson process with an expected rate λ_{IEA}, thus, the expected flow arrival rate on the bottleneck link is $n_{IEA} \cdot \lambda_{IEA}$. The flow holding time is exponentially distributed with an average value of $\frac{1}{\mu} = 90$ s. Hence, the expected average number of flows per IEA is $n_{IEA}^{flows} = \frac{\lambda_{IEA}}{\mu}$ provided that no flow blocking occurs. The expected number of flows on the bottleneck link is $n_{AR} = n_{IEA} \cdot n_{IEA}^{flows}$. We use this number of flows to dimension the admissible rate *AR* of the bottleneck link. We study PCN-based AC under challenging conditions, i.e. when the flow arrival rate is f_{crowd}^{flash} larger than expected. We call f_{crowd}^{flash} the flash crowd factor as a flash crowd commonly denotes an unexpectedly high request rate.

We assume periodic traffic with constant packet inter-arrival times $IAT = 20$ ms and constant packet sizes $B = 200$ bytes which are typical values for constant bitrate voice traffic in IP networks [33]. Hence, the flow rate is 80 kbit/s. To avoid simulation artifacts due to overly exact arrival times we add some uniformly distributed jitter to the packet transmission times of at most $D_{pkt}^{max} = 1$ ms. The excess and exhaustive marker on the bottleneck link is configured with a bucket size of $0.05\ \text{s} \cdot AR$ and the marking threshold of the exhaustive marker is set to $0.25\ \text{s} \cdot AR$. We simulate the time-dependent PCN traffic rate $r(t)$ on the bottleneck link. To make things simple, we consider the number of admitted flows n in accompanying analyses. The experiments start with an empty system and flows arriving with a rate $f_{crowd}^{flash} \cdot \lambda_{IEA}$ per IEA are continuously admitted until blocking occurs. *AR*-overload is the rate of admitted PCN traffic above the admissible rate on the bottleneck link, i.e. $\max(0, r(t) - AR)$ or $\max(0, n - n_{AR})$ in terms of flows. We define the *level of overadmission* as the fraction of the *AR*-overload and the admissible rate, i.e., it is essentially the relative *AR*-overload, and it is the main performance measure in this study.

We simulate the time-dependent PCN traffic rates in our experiments multiple times and produce time-dependent averages. We perform so many runs that confidence intervals for the obtained mean values are small, but omit them in the figures for the sake of clarity.

B. PBAC with Exhaustive Marking

PBAC with exhaustive marking needs only a single packet for probing. PBAC does not require the notion of IEAs. To be conform with our notation, we set $n_{IEA} = 1$ and n_{IEA}^{flows} is then the number of simulated flows in the context of PBAC. We use $n_{IEA}^{flows} = 100$ flows and, hence, we configure the admissible rate on the bottleneck link with $AR = 8$ Mbit/s.

1) Flows with Finite Holding Times: Figure 3(a) shows the time-dependent PCN traffic rate for various flash crowd factors. The initial growth rate of the admitted PCN traffic scales with the flash crowd factor. PBAC with exhaustive marking blocks quite reliably when n_{AR} flows are admitted on the bottleneck link and, therefore, we observe hardly any overadmission.

2) Flows with Infinite Holding Times: Figure 3(b) shows the time-dependent PCN traffic rate for various flash crowd factors in case of infinite flow holding times. Infinite flow holding times are an approximation of flows that exceed the expected holding time values by far. Now we observe little overadmission that is proportional to the flash crowd factor f_{crowd}^{flash}. Surplus flows are admitted during the interval when the number of admitted flows n has already recently reached n_{AR}, but the exhaustive marker of the bottleneck link has not yet recognized it as it needs some time to empty its token bucket. The same happens for finite flow holding times in Figure 3(a) but it is only rudimentary visible because temporary overadmission is corrected by finishing flows. With infinite holding times the number of admitted flows remains constant, too, although this correction factor is removed.

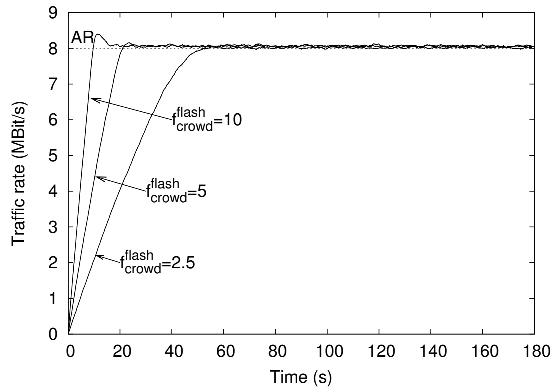
3) Conclusion: PBAC works fine with exhaustive marking even in case of very high flow arrival rates.

C. PBAC with Excess Marking

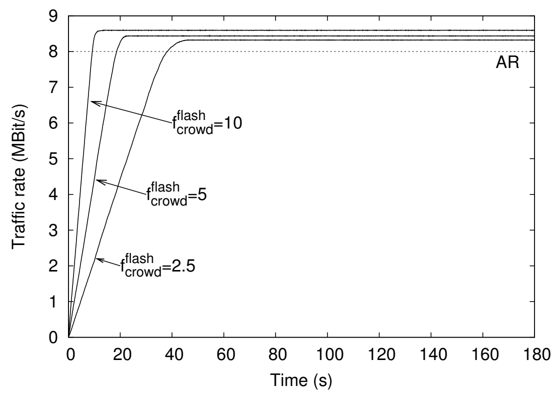
PBAC can also be combined with excess marking which possibly makes sense when exhaustive marking is not implemented by router vendors. However, when excess marking is used to detect *AR*-overload, only the excess traffic above the admissible rate on the bottleneck link is marked. On the one hand, the number of probe packets per admission request n_p should be large to admit new requests with sufficient confidence [6]. On the other hand, n_p should be small because many probe packets delay admission decisions. We implement the probing process as follows. Upon an admission request, the ingress node sends n_p probe packets using the characteristic packet sizes and inter-arrival times of the prospective data flow, i.e. $B = 200$ bytes and $IAT = 20$ ms in our experiment. However, we choose exponentially distributed inter-arrival times to get a reliable estimate of the load condition on the bottleneck link [34].

1) Flows with Finite Holding Times: We investigate for PBAC with excess marking the impact of the number of probe packets n_p on overadmission. To get a deeper understanding of the results, we analytically explain our findings and derive formulae that predict the expected overadmission. Figure 3(c) reports the time-dependent admitted PCN traffic rate for PBAC with excess marking, flows with an average holding time of 90 s, a flash crowd factor of $f_{crowd}^{flash} = 5$, and different numbers of probe packets n_p. The figure shows that PBAC cannot limit the PCN traffic on the bottleneck link to the desired value of $AR = 8$ Mbit/s. The level of overadmission obviously decreases with an increasing number of probe packets. The overadmission also depends on the flash crowd factor f_{crowd}^{flash}, but we do not illustrate this obvious connection. We derive the level of overadmission analytically. The admitted PCN traffic rate achieves an equilibrium when the rate of admitted flows – in terms of number of admitted flows per second – equals the rate of finishing flows:

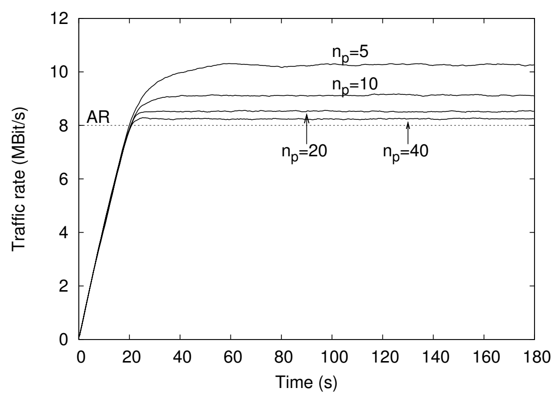
$$p(n) \cdot f_{crowd}^{flash} \cdot \lambda_{IEA} = n \cdot \mu. \qquad (1)$$



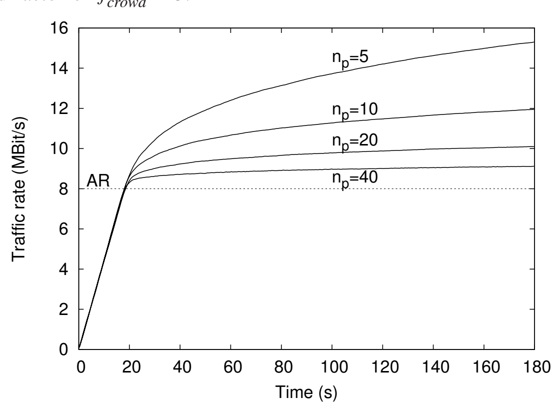
(a) Exhaustive marking and flows with a holding time of $\frac{1}{\mu} = 90$ s.



(b) Exhaustive marking and flows with infinite holding times.



(c) Excess marking, flows with a holding time of $\frac{1}{\mu} = 90$ s, and a flash crowd factor of $f_{crowd}^{flash} = 5$.



(d) Excess marking, flows with infinite holding times, and a flash crowd factor of $f_{crowd}^{flash} = 5$.

Fig. 3. Probe-based admission control (PBAC).

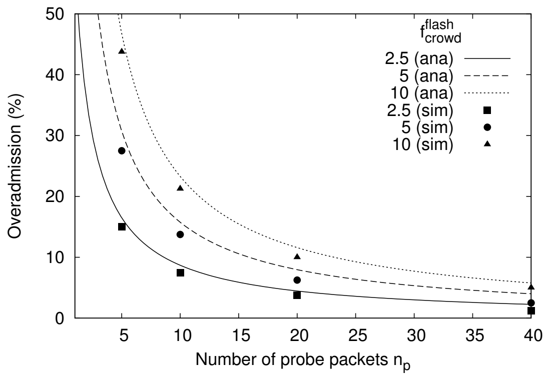


Fig. 4. Analytical overadmission depending on the number of probe packets n_p and the flash crowd factor f_{crowd}^{flash}. The lines are validated with simulation results.

Thereby, $\frac{n_{AR}}{n}$ is the probability that a packet is not AS-marked in case of AR-pre-congestion ($n \geq n_{AR}$) and $p(n) = \left(\frac{n_{AR}}{n}\right)^{n_p}$ is the probability for false admission. Equation (1) is true for $\frac{n}{n_{AR}} = \sqrt[n_p+1]{f_{crowd}^{flash}}$. Thus, $\sqrt[n_p+1]{f_{crowd}^{flash}} - 1$ is the expected level of overadmission and Figure 4 shows it depending on the number of probes n_p and for different flash crowd factors f_{crowd}^{flash}. The points in the figure correspond to the overadmission obtained from simulations and validate our analytical model. The analysis slightly overestimates the simulated values. With $n_p = 20$ probes per admission request, the overadmission is bounded by 12.5% for flash crowd factors $f_{crowd}^{flash} \in \{2.5, 5, 10\}$. As the analysis is based on mean values, it helps to understand the observations, but it does not give evidence about extreme values. The strongest fluctuations are obtained for the smallest number of probes $n_p = 5$. The 10% and 90% quantiles of the admitted PCN traffic rate – which are not reported in Figure 3(c) for the sake of clarity – show that the overadmission varies only moderately.

2) Flows with Infinite Holding Times: Figure 3(d) shows the time-dependent PCN traffic rate on the bottleneck link for the same settings as above but now for infinite instead of finite flow holding times. The PCN rate continuously increases because PBAC sometimes falsely admits flows in spite of AR-pre-congestion. This cannot be corrected by finishing flows because the flows in the experiment have an infinite holding time. Thus, PBAC cannot effectively block new admission request in this experiment which is obvious for $n_p = 10$ or fewer probes per admission request. However, the increase of admitted PCN traffic slows down over time because false admissions become less likely since the packet marking probability increases with increasing AR-overload. As a result, we see only little overadmission for $n_p = 20$ or more probes per admission request.

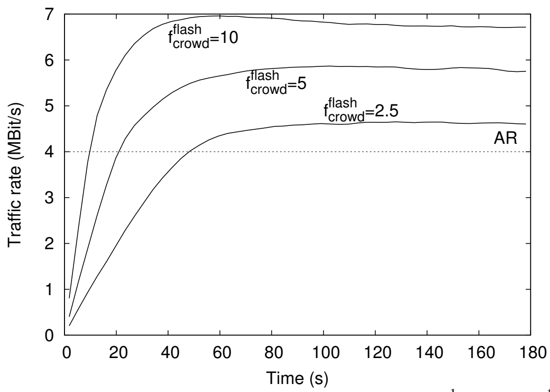
3) Conclusion: PBAC with excess marking works when the number of probe packets per admission request is large enough. However, this cannot support implicit probing and is possibly not acceptable for some applications.

D. CLEBAC with Exhaustive Marking

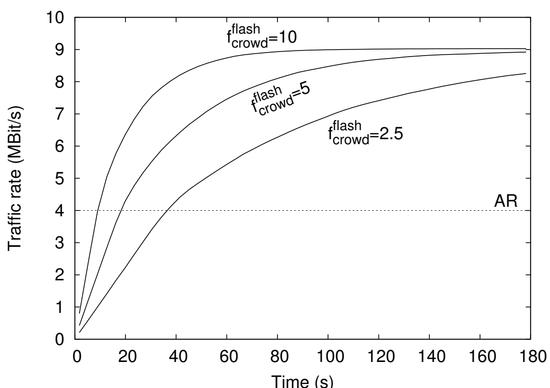
CLEBAC can block a new admission request for a specific IEA only if it has feedback information from that IEA. When the IEA carries no flows, its AC state K is set to *admit* to avoid starvation. Thus, empty IEAs always admit new PCN flows. This is problematic when the expected average number of flows per IEA n_{IEA}^{flows} is very low, and in particular smaller than 1, which is not an unrealistic assumption [35]. In such a situation overadmission already occurs when every IEA has admitted one flow. This problem is common to all aggregate-based AC methods, i.e. also to OBAC. In contrast to Section IV-B and Section IV-C, we now simulate $n_{IEA} = 100$ IEAs sharing a single bottleneck link.

1) Flows with Finite Holding Times: We simulate the time-dependent admitted PCN traffic rate for flows with average holding times of $\frac{1}{\mu} = 90$ s in the presence of different flash crowd factors. Figure 5(a) shows the results for an aggregation level of $n_{IEA}^{flows} = 0.5$ flows per IEA. For a moderate flash crowd factor of $f_{crowd}^{flash} = 2.5$ the PCN traffic rate approaches almost monotonously its typical long-term value. This is different for large flash crowd factors $f_{crowd}^{flash} = 10$. The PCN traffic rate initially overshoots its typical long-term value and then decreases. The typical long-term value exhibits significant overadmission for $n_{IEA}^{flows} = 0.5$ which depends on f_{crowd}^{flash}. Simulation results for $n_{IEA}^{flows} = 1.0$ (no figure) reveal hardly any long-term overadmission but a significant initial overshoot when the PCN traffic rate on the bottleneck link reaches its AR some time after simulation start. The initial overshooting for large flash crowd factors can be explained as follows. As long as the number of admitted flows n on the bottleneck link is smaller than the number of admissible flows n_{AR}, all aggregates can admit flows. As soon as n surpasses n_{AR}, only empty aggregates can admit new traffic. When flows of aggregates with several admitted flows finish, their IEAs cannot admit new flows. As a consequence, the admitted traffic rate decreases. Additional results have shown that this effect is avoided if the simulation starts with one admitted flow per IEA.

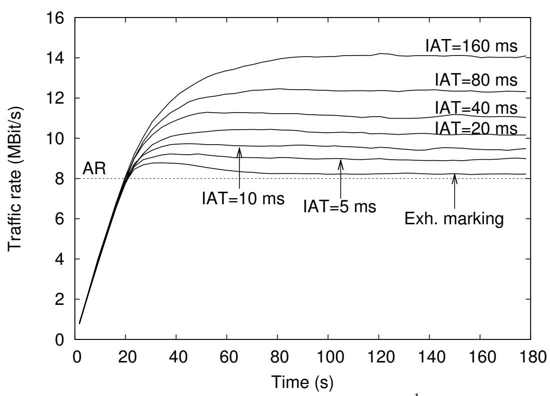
We analytically derive a rough approximation of the expected long-term overadmission. When the AR of the bottleneck link is configured for only $n_{IEA}^{flows} < 1$ flows per IEA, most aggregates carry at most one admitted flow. Therefore, we assume for our analysis that an aggregate carries either one flow or no flows. This is the approximative part of the analysis since aggregates can admit more than one flow. The approximation is good for small aggregation levels n_{IEA}^{flows} and large flash crowd factors f_{crowd}^{flash}. Thanks to this approximation, the number of admitted flows per IEA alternates between one and zero. Empty IEAs wait on average $\frac{1}{\lambda_{IEA} \cdot f_{crowd}^{flash}}$ time until the next admission request arrives and is admitted. It takes about $\frac{1}{\mu}$ time for IEAs with one admitted flow until they are empty. Hence, the average number of admitted flows per IEA over time can be calculated by $\frac{1/\mu}{1/\mu + 1/(\lambda_{IEA} \cdot f_{crowd}^{flash})} = \frac{\lambda_{IEA} \cdot f_{crowd}^{flash}}{\lambda_{IEA} \cdot f_{crowd}^{flash} + \mu}$. Thus, the average number of admitted flows on the bottleneck link is $n_{IEA} \cdot \frac{\lambda_{IEA} \cdot f_{crowd}^{flash}}{\lambda_{IEA} \cdot f_{crowd}^{flash} + \mu}$ while the AR of the bottleneck



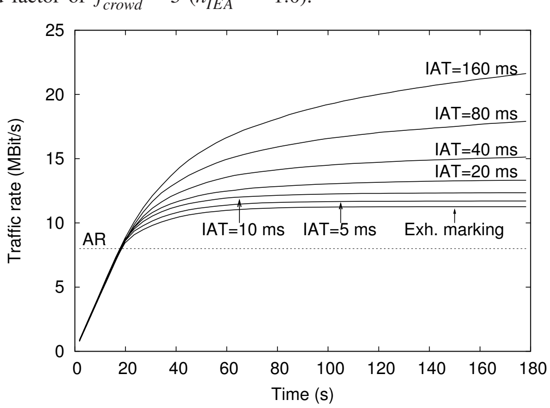
(a) Exhaustive marking and flows with a holding time of $\frac{1}{\mu} = 90$ s ($n_{IEA}^{flows} = 0.5$).



(b) Exhaustive marking and flows with infinite holding times ($n_{IEA}^{flows} = 0.5$).



(c) Excess marking, flows with a holding time of $\frac{1}{\mu} = 90$ s, and a flash crowd factor of $f_{crowd}^{flash} = 5$ ($n_{IEA}^{flows} = 1.0$).



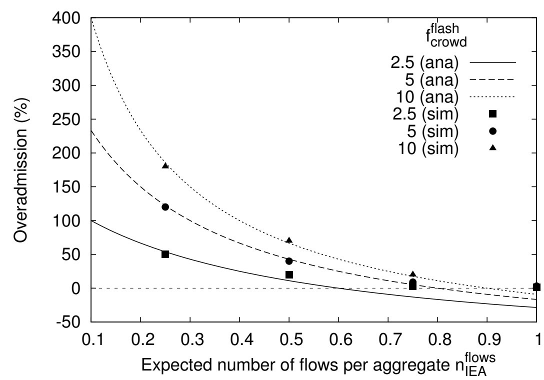
(d) Excess marking, flows with infinite holding times, and a flash crowd factor of $f_{crowd}^{flash} = 5$ ($n_{IEA}^{flows} = 1.0$).

Fig. 5. Congestion level estimate based admission control (CLEBAC) with $D_{MI} = 200$ ms, $T_{CLE}^{AStop} = 0.025$, and $T_{CLE}^{ACont} = 0.0$ ($n_{IEA=100}$).

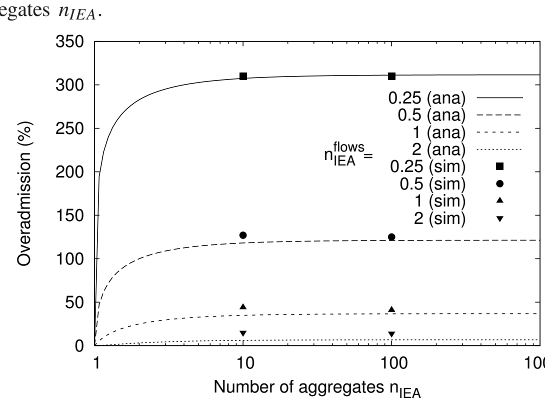
link is configured for $n_{IEA} \cdot \frac{\lambda_{IEA}}{\mu} = n_{IEA} \cdot n_{IEA}^{flows}$ per aggregate. Therefore, the level of overadmission can be approximated by

$$OA = \frac{n}{n_{AR}} - 1 = \frac{\frac{\lambda_{IEA}\cdot f_{crowd}^{flash}}{\lambda_{IEA}\cdot f_{crowd}^{flash}+\mu}}{\frac{\lambda_{IEA}}{\mu}} - 1 = \frac{f_{crowd}^{flash}}{n_{IEA}^{flows}\cdot f_{crowd}^{flash}+1} - 1. \tag{2}$$

Thus, the overadmission is independent of the number of IEAs n_{IEA} on the bottleneck link. Figure 6(a) illustrates the overadmission depending on the expected number of flows per IEA n_{IEA}^{flows} for different flash crowd factors f_{crowd}^{flash}. The validation of the analytical values (ana) by simulation results (sim) shows that the analysis is accurate for low n_{IEA}^{flows} and large f_{crowd}^{flash}. For large n_{IEA}^{flows} and small f_{crowd}^{flash} it produces obviously wrong results as overadmission cannot be negative.



(a) Flows with an average holding time of $\frac{1}{\mu} = 90$ s. Overadmission depends on n_{IEA}^{flows} and the flash crowd factor f_{crowd}^{flash} but not on the number of aggregates n_{IEA}.



(b) Flows with infinite holding times. Overadmission depends on n_{IEA}^{flows}, only little on the number of aggregates n_{IEA} and not at all on the flash crowd factor f_{crowd}^{flash}.

Fig. 6. Analytical overadmission for a small number of flows n_{IEA}^{flows} per IEA and partial validation with simulation results.

2) Flows with Infinite Holding Times: We perform the same experiment like above with flows having infinite holding times. Figure 5(b) shows the results for aggregation level of $n_{IEA}^{flows} = 0.5$ and corresponds to Figure 5(a). With infinite holding times, the rate of admitted PCN traffic is significantly larger than the one with finite holding times but also converges to a limited level of overadmission. In contrast to Figure 5(a), the flash crowd factor f_{crowd}^{flash} in Figure 5(b) has hardly any influence on the overadmission, it only controls how fast saturation is reached. Additional experiments have shown that with infinite flow holding times overadmission also occurs when the expected number of flows per aggregate is larger than 1 which is in contrast to flows with finite holding times.

We analytically derive the level of overadmission to which the average PCN traffic rate converges in case of CLEBAC with exhaustive marking and infinite flows. As long as the admitted PCN traffic rate on the bottleneck link is below its AR, any aggregate can admit new flows. When the admitted PCN traffic rate has reached the AR on the bottleneck link, at least $n_{AR} = n_{IEA} \cdot n_{IEA}^{flows}$ flows are already admitted, and empty aggregates can still admit further flows. The fraction of empty aggregates at this stage is $p(n_{IEA}, n_{IEA}^{flows}) = \left(1 - \frac{1}{n_{IEA}}\right)^{n_{IEA} \cdot n_{IEA}^{flows}}$ and the average of the absolute number of empty aggregates is $n_{IEA} \cdot p\left(n_{IEA}, n_{IEA}^{flows}\right)$. After the empty aggregates have admitted another flow, the average number of admitted flows on the bottleneck link is $n_{IEA} \cdot n_{IEA}^{flows} + n_{IEA} \cdot p\left(n_{IEA}, n_{IEA}^{flows}\right)$. Thus, the level of overadmission is

$$OA = \frac{n}{n_{AR}} - 1 = \frac{n_{IEA} \cdot p\left(n_{IEA}, n_{IEA}^{flows}\right)}{n_{AR}} = \frac{p\left(n_{IEA}, n_{IEA}^{flows}\right)}{n_{IEA}^{flows}}. \tag{3}$$

We have a static scenario when the admitted PCN traffic rate has stabilized. It is not an equilibrium as admitted flows do not finish and new flows cannot be admitted. Figure 6(b) shows the level of overadmission depending on the number of aggregates n_{IEA} and the expected number of flows n_{IEA}^{flows} per IEA. The aggregation level n_{IEA}^{flows} has a tremendous impact on overadmission while the number of aggregates n_{IEA} has hardly any influence. The figure confirms that overadmission for a low aggregation level of $n_{IEA}^{flows} = 0.5$ is slightly above 2.0 which is tremendous. The effect vanishes for larger aggregation levels $n_{IEA}^{flows} \geq 2.0$.

3) Conclusion: IEA-based AC methods like CLEBAC or OBAC lead to large overadmission in case of flash crowds when the AR of the bottleneck link is dimensioned for a low number of expected flows. Furthermore, this is also the case if excess marking is used because the marking scheme has no impact on the admission state of an empty IEA. This limits the applicability of IEA-based AC methods to scenarios which sufficiently many expected flows per IEA. PBAC does not rely on IEAs and does, therefore, not suffer from this problem.

E. CLEBAC with Excess Marking

CLEBAC performs AC decisions based on the AC state K of the IEA the requesting flow will belong to. The AC state K is controlled by the aggregated feedback of the respective IEA. As excess marking marks only the traffic that exceeds the AR of the bottleneck link, only a small fraction of the overall traffic is marked in case of moderate AR-pre-congestion and hence the packet marking probability is small. CLEBAC does not block in case of AR-pre-congestion when it does not receive any marked packets within its measurement interval which is quite likely if the packet frequency per IEA is low. We illustrate and quantify the impact of the packet frequency in the following. The experiment setup is like in Section IV-D, but we substitute exhaustive marking by excess marking. Furthermore, we set $n_{IEA}^{flows} = 1$ instead of $n_{IEA}^{flows} = 0.5$ to show that that the overadmission in this case results from the low packet frequency and not from the low aggregation level n_{IEA}^{flows} (cf. Figure 6(a)).

1) Flows with Finite Holding Times: Figure 5(c) illustrates the time-dependent PCN traffic rate for CLEBAC with excess marking and a flash crowd factor of $f_{crowd}^{flash} = 5$. The figure shows curves for flows with different packet inter-arrival times IAT and packet sizes sizes B which are chosen such that flows have a rate of 80 kbit/s in all cases. For all curves we observe a certain level of overadmission which depends on the packet inter-arrival time IAT. In case of AR-pre-congestion new flows are still accepted with some probability because some measurement intervals do not see any marked packets. The probability for that is $p(n) = \left(\frac{n_{AR}}{n}\right)^{\frac{D_{MI}}{IAT}}$. This holds under the assumption that the packets of a single IEA are marked independently of each other with some probability $\frac{n_{AR}}{n}$ which is approximatively true for a large number of IEAs ($n_{IEA} \geq 10$). The probability $p(n)$ for false acceptance becomes smaller with increasing number of packets per measurement interval which is $\frac{D_{MI}}{IAT}$ and, thus, overtermination decreases, too. Figure 5(c) also shows the equivalent curve for CLEBAC with exhaustive marking ($IAT = 20$ ms). The overtermination observed for that curve is only due to empty aggregates but not to low packet frequency per aggregate. Therefore, it is a lower bound for the other curves.

2) Flows with Infinite Holding Times: We perform the same experiment with infinite holding times analogously to Section IV-C. Hence, potential overadmission cannot be corrected by finishing flows. Figure 5(d) shows that CLEBAC cannot effectively block new admission requests and the PCN traffic rate on the bottleneck link continuously increases. This is evident at least for 5 or fewer packets per measurement interval ($IAT \geq 40$ ms or larger) which is, however, probably not a very realistic assumption.

3) Conclusion: Very low packet frequencies per IEA can lead to increased overadmission. Though, this effect might be small in most scenarios of practical relevance. In any case, the observed overadmission is again significantly larger than the comparable curve for for exhaustive marking.

F. OBAC with Exhaustive or Excess Marking

Observation-based AC (OBAC) is similar to congestion level estimate based AC (CLEBAC) because both AC methods use feedback from IEAs to control their AC states K based on which AC is performed for admission requests of flows falling into the same IEA. As a consequence, all findings for CLEBAC with exhaustive marking and low aggregation levels n_{IEA}^{flows} in Section IV-D also apply to OBAC.

CLEBAC and OBAC just differ in how they control the AC state K. In our experiments we have set the admission-stop threshold T_{CLE}^{AStop} for CLEBAC in such a way that a single packet suffices to stop admission of further flows at the end of the current measurement interval. This is very similar to OBAC that blocks as soon as it receives a marked packet. CLEBAC resumes admission as soon as no marked packets are received in future measurement intervals, but OBAC has a minimum block time T_{block}^{min} which we set to 500 ms. Thus, OBAC blocks earlier and longer than CLEBAC. Therefore, OBAC with excess marking leads to the same qualitative results as in Section IV-E and additional results show that the observed overadmission in absolute numbers is up to 50% smaller.

G. Multipath Routing

IEA-based AC methods such as CLEBAC or OBAC lead to underadmission when the network uses multipath routing. They rely on PCN feedback that possibly stems from different partial paths. As soon as one of them produces a significant amount of marked packets, admission requests of new flows are blocked disregarding the partial path of the multipath on which they will be actually carried. Therefore, it is not possible to utilize the entire capacity of all parallel paths. This is different with PBAC. PBAC probes the path on which further data packets will be carried and yields path-specific admission decisions.

For the sake of simplicity of our analysis we assume equal flow rates. To quantify the amount of non-utilized capacity by IEA-based AC methods, we consider an empty IEA using a multipath consisting of k partial paths with admissible rate AR_i ($0 \leq i < k$) in terms of number of flows. Usually, only links have an AR, but the AR of the path we refer to is the AR of its bottleneck link. The state of the IEA $s = (s_0, ..., s_{k-1})$ indicates the number of current flows s_i on the partial path i for $0 \leq i < k$. We neglect the time component and start with an empty IEA, i.e. $s_i = 0$ for $0 \leq i < k$. Then, we admit new flows sequentially to the considered IEA. When a flow is admitted, it is randomly assigned to one of the partial paths i with probability $p(i) = \frac{1}{k}$ and the number of admitted flows s_i ($0 \leq i < k$) is incremented by one. The corresponding markov chain is a simple birth process according to

$$(s_0, ..., s_i, ..., s_{k-1}) \xrightarrow{p(i)} (s_0, ..., s_i + 1, ..., s_{k-1}). \tag{4}$$

The IEA blocks as soon as the number of flows s_i of one of the partial paths i has reached AR_i. Therefore, the birth process stops whenever state s meets the stopping condition $s_i = AR_i$ for one partial path i. We collect these states s in the set $\mathcal{T}$ of terminating states. For all states $s \in \mathcal{T}$ we compute the probability $p(s)$ by an algorithm which iteratively applies Equation (4) and sums up the probabilities of all transition paths leading to any terminating state $s \in \mathcal{T}$. These probabilities sum up to $\sum_{s \in \mathcal{T}} p(s) = 1$ and yield a probability distribution of the states in which the IEA blocks further admission requests. The utilization of the multipath capacity in state s is given by

$$U(s) = \frac{\sum_{0 \leq i < k} s_i}{\sum_{0 \leq i < k} AR_i}.$$

Thus, we calculate the mean utilization of the multipath when the system blocks by $U = \sum_{s \in \mathcal{T}} U(s) \cdot p(s)$.

Figure 7 illustrates the average utilization of the multipath capacities when the IEA starts blocking. The utilization is large when partial paths have equal capacity while it is low if one of the paths has only half the capacity of the others. The utilization increases with increasing multipath capacity and decreases with increasing number of partial paths. With equal-capacity paths a utilization close to 100% is possible while the utilization is limited to lower values in case of unequal-capacity paths.

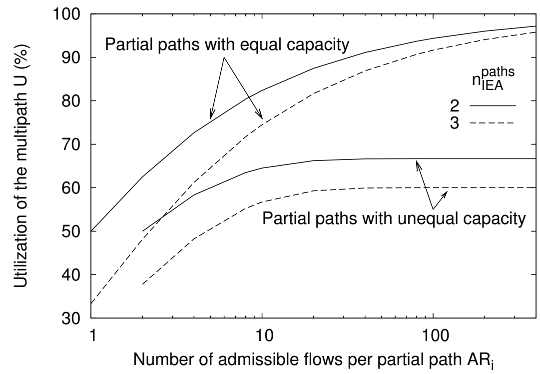


Fig. 7. Multipath routing prohibits full utilization of partial path capacities for IEA-based AC.

H. Delayed Media

In case of delayed media, the media stream starts sending traffic only D_{delay}^{media} time after its admission. Delayed media is challenging for all measurement-based AC methods because the feedback mechanism captures only transmitting flows instead of all admitted flows. We use PBAC with exhaustive marking to visualize the impact of delayed media on potential overadmission because it does not lead to any significant overadmission in the other experiments. Only a single aggregate ($n_{IEA} = 1$) is carried over the bottleneck link with an aggregation level of $n_{IEA}^{flows} = 100$ flows. Our experiments start with an empty aggregate. Figure 8 shows that the admitted PCN traffic rate strongly oscillates around the admissible rate and leads to significant temporary overadmission. In contrast to previous results, Figure 8 reports the admitted PCN traffic rate of individual traces because averaging multiple runs would smooth out the oscillation effects. Furthermore, the interval between 500 s and 650 s after simulation start is illustrated to exclude the warmup phase of the simulation as the reason for the oscillations. The strength of the temporary overload obviously depends on the initial media delay D_{delay}^{media}. Additional results for exponentially distributed initial media delay reveal less jerky curves and slightly less overadmission.

We explain and approximatively characterize the rate oscillations. When n_{AR} flows are admitted, it takes another D_{delay}^{media} interval until the last admitted flow transmits and can be reflected in the PCN feedback information. During this time admitted flows finish with an approximative rate of $n_{AR} \cdot \mu = n_{IEA}^{flows} \cdot \mu = \frac{\lambda_{IEA}}{\mu} \cdot \mu = \lambda_{IEA}$. Due to the finishing flows, the admitted PCN rate has not yet exceeded the AR of the bottleneck link after D_{delay}^{media} time although some of the recently admitted flows start transmission . This happens only after some time d later during which as many new flows have been accepted

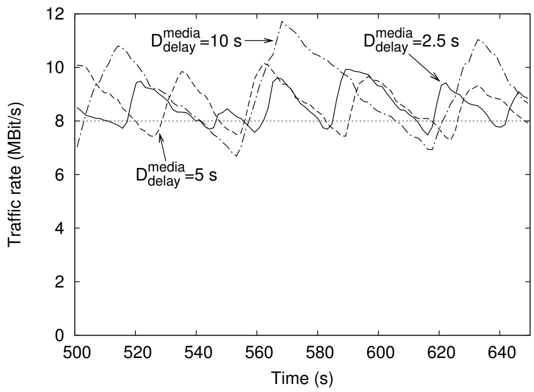


Fig. 8. Initial constant media delay and a flash crowd factor of $f_{crowd}^{flash} = 5$ lead to oscillating admitted PCN rates and temporary overadmission.

with rate $\lambda_{IEA} \cdot f_{crowd}^{flash}$ as admitted flows have quit within $d + D_{delay}^{media}$ time with rate $n_{IEA}^{flows} \cdot \mu = \lambda_{IEA}$. Thus, we have $d \cdot \lambda_{IEA} \cdot f_{crowd}^{flash} = (D_{delay}^{media} + d) \cdot \lambda_{IEA}$, i.e. $d = \frac{D_{delay}^{media}}{f_{crowd}^{flash} - 1}$. After this admission period of duration $D_{delay}^{media} + d$ the bottleneck link is again *AR*-pre-congested and blocks. D_{delay}^{media} later, all recently admitted flows have started transmission and the admitted PCN traffic rate on the bottleneck link reaches its maximum which corresponds to $n_{AR} + (D_{delay}^{media} + d) \cdot \lambda_{IEA} \cdot f_{crowd}^{flash} - (D_{delay}^{media} + d + D_{delay}^{media}) \cdot n_{AR} \cdot \mu = D_{delay}^{media} \cdot \lambda_{IEA} \cdot (f_{crowd}^{flash} - 1)$. This corresponds to a relative overadmission of

$$OA = \frac{n}{n_{AR}} - 1 = D_{delay}^{media} \cdot \mu \cdot (f_{crowd}^{flash} - 1). \tag{5}$$

AR-pre-congestion on the bottleneck disappears after the over-admitted $D_{delay}^{media} \cdot \lambda_{IEA} \cdot (f_{crowd}^{flash} - 1)$ flows have finished, i.e. after $\frac{D_{delay}^{media} \cdot \lambda_{IEA} \cdot (f_{crowd}^{flash} - 1)}{n_{AR} \cdot \mu} = D_{delay}^{media} \cdot (f_{crowd}^{flash} - 1)$ time. Thus, $(D_{delay}^{media} + d) \cdot f_{crowd}^{flash} \cdot \lambda_{IEA}$ flows are admitted within $D_{delay}^{media} + d$ time and then new flows are blocked for the remaining $D_{delay}^{media} + D_{delay}^{media} \cdot (f_{crowd}^{flash} - 1)$ time. Hence, the period of the oscillations is about

$$\begin{aligned} period &= D_{delay}^{media} + d + D_{delay}^{media} + D_{delay}^{media} \cdot (f_{crowd}^{flash} - 1) \\ &= D_{delay}^{media} \cdot \frac{\left(f_{crowd}^{flash}\right)^2}{f_{crowd}^{flash} - 1}. \end{aligned} \tag{6}$$

We validated these equations by additional simulations. They revealed the approximative nature of our findings but confirmed the trends suggested by the formulae: the media delay D_{delay}^{media} and the flash crowd factor f_{crowd}^{flash} have a significant impact on the period of the oscillation and the overadmission, the arrival rate λ_{IEA} influences only the maximum number of overadmitted flows, the average flow holding time $\frac{1}{\mu}$ influences only the relative overadmission, and the observed effects are independent of the aggregation level n_{IEA}^{flow}.

V. Conclusion

In this paper we investigated various pre-congestion notification (PCN) based admission control (AC) methods in combination with excess or exhaustive marking under challenging conditions such as increased admission request rates or a small number of flows per ingress-egress aggregate (IEA). We used either simulation or mathematical analysis for our study to produce the reported effects and to provide a rule of thumb to predict potential overadmission.

Probe-based AC (PBAC) works well with exhaustive marking. It also works with excess marking but leads to significant overadmission. Congestion level estimate based AC (CLEBAC) and observation-based AC (OBAC) rely on feedback from IEAs. In case of exhaustive marking, they cannot effectively block traffic in case of a small number of expected flows n_{IEA}^{flows} and in case of excess marking they lead to overadmission if the number of data packets per measurement interval is not sufficiently high. CLEBAC and OBAC do not work efficiently with multipath routing as they stop admission of further flows for the entire multipath of an IEA when one of its partial paths is pre-congested. Finally, we showed that initial media delay leads to oscillations of the admitted PCN traffic rate and temporary overadmission.

Future networks are expected to carry a large number of flows per link, but a small number of flows per IEA [35]. In addition, multipath routing is reality in today's networks. Given the fact that CLEBAC and OBAC do not work well under these conditions, we recommend the standardization and implementation of PBAC with exhaustive marking. It requires only a single probe packet and can possibly reuse other per-flow signalling messages for that purpose [3].

Acknowledgements

The authors would like to thank Joe Babiarz, Anna Charny, and Xinyang Zhang for their fruitful discussions.

References

[1] P. Eardley (ed.), "Pre-Congestion Notification Architecture," http://tools.ietf.org/id/draft-ietf-pcn-architecture-05.txt, Aug. 2008.

[2] J. Wroclawski, "RFC2211: Specification of the Controlled-Load Network Element Service," Sep. 1997.

[3] M. Menth, F. Lehrieder, B. Briscoe, P. Eardley, T. Moncaster, J. Babiarz, K.-H. Chan, A. Charny, G. Karagiannis, and X. J. Zhang, "PCN-Based Admission Control and Flow Termination," in *currently under submission, http://www3.informatik.uni-wuerzburg.de/staff/menth/Publications/Menth08-PCN-Comparison.pdf*, 2008.

[4] Y. Bernet, P. Ford, R. Yavatkar, F. Baker, L. Zhang, M. Speer, R. Braden, B. Davie, J. Wroclawski, and E. Felstaine, "RFC2998: A Framework for Integrated Services Operation over Diffserv Networks," Nov. 2000.

[5] A. Charny, F. L. Faucheur, V. Liatsos, and J. Zhang, "Pre-Congestion Notification Using Single Marking for Admission and Pre-emption," http://tools.ietf.org/id/draft-charny-pcn-single-marking-03.txt, Nov. 2007.

[6] M. Menth and F. Lehrieder, "Performance Evaluation of PCN-Based Admission Control," in *International Workshop on Quality of Service (IWQoS)*, Enschede, The Netherlands, Jun. 2008.

[7] B. Briscoe et al., "An Edge-to-Edge Deployment Model for Pre-Congestion Notification: Admission Control over a DiffServ Region," http://tools.ietf.org/id/draft-briscoe-tsvwg-cl-architecture-04.txt, Oct. 2006.

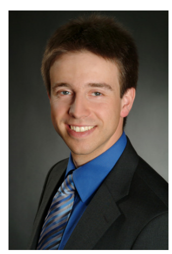
[8] J. Babiarz, X.-G. Liu, K. Chan, and M. Menth, "Three State PCN Marking," http://tools.ietf.org/id/draft-babiarz-pcn-3sm-01.txt, Nov. 2007.

[9] B. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin, "RFC2205: Resource ReSerVation Protocol (RSVP) - Version 1 Functional Specification," Sep. 1997.

[10] S. Floyd and V. Jacobson, "Random Early Detection Gateways for Congestion Avoidance," *IEEE/ACM Transactions on Networking*, vol. 1, no. 4, pp. 397–413, Aug. 1993.

[11] B. Braden et al., "RFC2309: Recommendations on Queue Management and Congestion Avoidance in the Internet," Apr. 1998.
[12] K. Ramakrishnan, S. Floyd, and D. Black, "RFC3168: The Addition of Explicit Congestion Notification (ECN) to IP," Sep. 2001.
[13] N. Spring, D. Wetherall, and D. Ely, "RFC3540: Robust Explicit Congestion Notification (ECN)," Jun. 2003.
[14] K. Nichols, S. Blake, F. Baker, and D. L. Black, "RFC2474: Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers," Dec. 1998.
[15] S. Floyd, "RFC4774: Specifying Alternate Semantics for the Explicit Congestion Notification (ECN) Field," Feb. 2007.
[16] W. Almesberger, T. Ferrari, and J.-Y. Le Boudec, "SRP: A Scalable Resource Reservation for the Internet," *Computer Communications*, vol. 21, no. 14, pp. 1200–1211, Nov. 1998.
[17] I. Stoica and H. Zhang, "Providing Guaranteed Services without per Flow Management," in *ACM SIGCOMM*, Boston, MA, Sep. 1999.
[18] R. Szábó, T. Henk, V. Rexhepi, and G. Karagiannis, "Resource Management in Differentiated Services (RMD) IP Networks," in *International Conference on Emerging Telecommunications Technologies and Applications (ICETA 2001)*, Kosice, Slovak Republic, Oct. 2001.
[19] R. J. Gibbens and F. P. Kelly, "Distributed Connection Acceptance Control for a Connectionless Network," in 16^{th} *International Teletraffic Congress (ITC)*, Edinburgh, UK, Jun. 1999, pp. 941 – 952.
[20] F. Kelly, P. Key, and S. Zachary, "Distributed Admission Control," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 12, pp. 2617–2628, 2000.
[21] M. Karsten and J. Schmitt, "Admission Control based on Packet Marking and Feedback Signalling – Mechanisms, Implementation and Experiments," Darmstadt University of Technology, Technical Report 03/2002, 2002.
[22] ——, "Packet Marking for Integrated Load Control," in *IFIP/IEEE Symposium on Integrated Management (IM)*, 2005.
[23] M. Menth, R. Martin, and J. Charzinski, "Capacity Overprovisioning for Networks with Resilience Requirements," in *ACM SIGCOMM*, Pisa, Italy, Sep. 2006.
[24] S. Iyer, S. Bhattacharyya, N. Taft, and C. Diot, "An Approach to Alleviate Link Overload as Observed on an IP Backbone," in *IEEE Infocom*, San Francisco, CA, April 2003.
[25] X. Zhang and A. Charny, "Performance Evaluation of Pre-Congestion Notification," in *International Workshop on Quality of Service (IWQoS)*, Enschede, The Netherlands, Jun. 2008.
[26] M. Menth and F. Lehrieder, "Comparison of Marking Algorithms for PCN-Based Admission Control," in 14^{th} *GI/ITG Conference on Measuring, Modelling and Evaluation of Computer and Communication Systems (MMB)*, Dortmund, Germany, Mar. 2008, pp. 77–91.
[27] ——, "PCN-Based Marked Flow Termination," in *currently under submission*, 2008.
[28] F. Lehrieder and M. Menth, "Marking Conversion for Pre-Congestion Notification," in *IEEE International Conference on Communications (ICC)*, Dresden, Germany, Jun. 2009.
[29] ——, "PCN-Based Flow Termination with Multiple Bottleneck Links," in *IEEE International Conference on Communications (ICC)*, Dresden, Germany, Jun. 2009.
[30] M. Menth, "Efficiency of PCN-Based Network Admission Control with Flow Termination," *Praxis der Informationsverarbeitung und Kommunikation (PIK)*, vol. 30, no. 2, pp. 82 – 87, Apr. 2007.
[31] M. Menth and M. Hartmann, "Threshold Configuration and Routing Optimization for PCN-Based Resilient Admission Control," *accepted for Computer Networks*, 2009.
[32] J. Jiang and R. Jain, "A Simple Analytical Model of Pre-Congestion Notification," in *currently under submission*, 2008.
[33] M. Menth, A. Binzenhöfer, and S. Mühleck, "Source Models for Speech Traffic Revisited," University of Würzburg, Institute of Computer Science, Technical Report, No. 426, May 2007, accepted for IEEE/ACM Transactions on Networking (2009).
[34] J. Sommers, P. Barford, N. G. Duffield, and A. Ron, "Improving Accuracy in End-to-End Packet Loss Measurement," in *ACM SIGCOMM*, 2005.
[35] P. Eardley, "Traffic Matrix Scenario," http://www.ietf.org/mail-archive/web/pcn/current/msg00831.html, Oct. 2007.

Michael Menth studied computer science and mathematics at the University of Wuerzburg/Germany and Austin/Texas. He worked at the University of Ulm/Germany and Wuerzburg and obtained his PhD in 2004. Currently, he is assistant professor and heading the research group "Next Generation Networks" at the Institute of Computer Science in Wuerzburg. His special interests are performance analysis, optimization of communication networks, resource management, resilience issues, and Future Internet. Dr. Menth holds numerous patent applications and received various scientific awards for innovative work.



Frank Lehrieder studied computer science and business administration at the Universities of Wuerzburg/Germany and Madrid/Spain. In May 2008, he received his diploma degree in computer science. Now, he is a researcher at the Institute of Computer Science in Wuerzburg and pursuing his PhD. His special interests are performance evaluation and resource management for Future Internet infrastructures.