Comparison of Marking Algorithms for PCN-Based Admission Control

Michael Menth and Frank Lehrieder

University of Würzburg
Institute of Computer Science
Germany
{menth,lehrieder}@informatik.uni-wuerzburg.de

Abstract. The PCN working group of the IETF discusses the use of pre-congestion notification (PCN) to implement flow admission control. Packet meters and markers are used on all links of a network and packet markings are recorded as congestion level estimates (CLEs) by the egress nodes. The working group currently discusses the pros and cons of possible marking algorithms that play a major role in this new architecture. This paper provides a detailed description of threshold and ramp marking based on a virtual queue formulation. We investigate the impact of the marking threshold and the virtual queue size on the marking behavior and develop different marking strategies. We test the robustness of the CLEs obtained for both marking schemes against different CLE parameters and traffic characteristics. Furthermore, we show that ramp marking can be well approximated by appropriately configured threshold marking.

1 Introduction

The Internet is on its way to a universal communication platform including realtime services such as voice over IP, video on demand, tele-control and tele-medicine. The more it is important that Internet service providers (ISPs) can support these high quality services within their IP-based data network. Admission control (AC) for high quality traffic, i.e. the limitation of the number of such flows in the network, seems one option to guarantee its forwarding without excessive loss and delay [1]. Previous efforts to deploy AC based on the integrated services model [2] have not prevailed because they were based on individual per-flow reservations in each node along the path of a flow which entails a high complexity for these nodes.

The IETF has recently started a second approach to standardize AC for the Internet. It is based on pre-congestion notification (PCN), i.e., interior nodes mark packets with an admission-stop (AS) codepoint if the high quality traffic exceeds the admissible link rate and egress nodes monitor these markings using congestion level estimators. If the fraction of marked packets exceeds a certain value for a specific ingress-egress traffic aggregate, no further flows are admitted for that aggregate. This architecture is rather simple and easy to implement because core nodes do not need to know individual flows. Therefore, it has a broad support by manufacturers and operators.

The focus of this work is exactly on these metering and marking algorithms. Currently, the IETF discusses two alternatives: threshold marking and ramp marking. They

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

are both based on a virtual queue whose service rate is the admissible link rate. Threshold marking marks packets only if the queue length exceeds a certain threshold. Ramp marking already marks packets with a certain probability if the queue length is smaller than that threshold. Threshold marking is simple to implement, but ramp marking might give more information about the currently admitted traffic which could be possibly exploited for the admission decision of further flows. The objective of this paper is to provide an understanding of the impact of the system parameters and traffic characteristics on the marking result and a comparison of threshold and ramp marking.

Section 2 gives an overview of related work. Section 3 explains PCN, PCN-based admission control and flow termination, and gives a detailed description of threshold and ramp marking as well as the congestion level estimator. Section 4 investigates the marking behavior of both approaches under various conditions. Section 5 summarizes this work and gives conclusions.

2 Related Work

We give an overview of admission control mechanisms, in particular of those being highly related to the PCN architecture.

2.1 General Overview

Admission control was early proposed for IP networks in [1]. Flows issue reservation requests that are signalled by protocols like RSVP. These requests carry traffic descriptors and the routers on the way either grant or deny a reservation for high priority transport of the data packets. Parameter-based AC records the traffic descriptors of the admitted flows and decides upon a new request, whether its resources will suffice to support the new flow without QoS degradation for all admitted flows. With measurement-based AC (MBAC) routers reject or accept a flow request based on their observed network load [3]. To remove reservation states inside the network, other MBAC approaches use probing at the network border, i.e., if probe packets do not return or if they return late, the network is congested and further admission requests are denied [4].

2.2 Stateless Core Admission Control Based on Router Feedback

Stateless core admission control keeps reservation states only at the network borders and in the following two approaches, border routers base their admission decisions on implicit feedback of intermediate routers similarly to PCN-based AC.

Admission Control Based on Reservation Tickets To keep a reservation alive, ingress routers send reservation tickets in regular interval to the egress router. 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. Several stateless core mechanisms work according to this idea [5, 6].

Admission Control Based on Packet Marking Gibbens and Kelly [7, 8] theoretically investigated AC based on the feedback of marked packets whereby packets were already marked by routers based on a virtual queue with configurable bandwidth. This enables early warning which is the core idea of pre-congestion notification. It also allows to limit the utilization of the link bandwidth by premium traffic to arbitrary values between 0 and 100%. Karsten and Schmitt [9, 10] 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 Admission Control and Flow Termination Based on Pre-Congestion Notification

In this section, we introduce the general concept of pre-congestion notification (PCN) and describe PCN-based admission control and flow termination. The PCN-based admission control requires a marking mechanism, for which threshold and ramp marking are candidates that are presented in detail. We also describe the congestion level estimator for the evaluation of the packet markings because its EWMA has a major impact on the dynamics of the system.

3.1 Congestion and Pre-Congestion Notification

Congestion occurs on a link l when its current rate $r(l)$ exceeds its capacity $c(l)$. As a consequence, packets are queued and potentially lost. Pre-congestion describes load conditions where the current rate $r(l)$ is larger than a defined pre-congestion rate $PCR(l)$. This $PCR(l)$ is lower than the link bandwidth $c(l)$ such that substantial packet loss and delay do not necessarily occur at that stage.

Explicit congestion notification (ECN) [11] proposes that active queue management disciplines like random early detection (RED) mark packets in the presence of incipient congestion before queues overflow. These marks are implicitly carried to the end systems and notify them to reduce their transmission rate.

In a similar way, PCN marks packets when the current rate $r(l)$ exceeds $PCR(l)$ and these markings are carried to the edge of the network or to end systems to notify them that pre-congestion occurred on a link of the path the packet has taken.

3.2 Flow Admission Control and Termination

The ongoing efforts of the IETF strive at an implementation of flow admission control and termination without explicit signaling messages in the core network. They use PCN to achieve that goal [12]. Each link of a network is associated with two different rate thresholds: the admissible rate $AR(l)$ and the supportable rate $SR(l)$. If the current traffic rate $r(l)$ of a link l exceeds $AR(l)$, no further flows should be admitted that are carried over this link. Although admission of flows stops at a low rate $AR(l)$, it is possible that the traffic rate $r(l)$ exceeds this rate because already admitted flows may increase their transmission rates or rerouting in case of network failures adds backup traffic to the link. If $r(l)$ exceeds $SR(l)$, some flows should be terminated to reduce $r(l)$ below $SR(l)$.

In this paper, we focus on flow admission control. Traffic meters and markers control the PCN traffic on each link and if the current traffic rate $r(l)$ of a link exceeds its admissible rate $AR(l)$, the marker marks all packets with an admission-stop (AS) codepoint. Algorithms for this purpose are discussed in the next subsection. The egress nodes of the PCN domain monitor the traffic grouped into ingress-egress aggregates.

If the AS codepoint is set for a substantial portion of the packets, it notifies the admission control entity to stop the admission of further flows that belong the corresponding ingress-egress aggregate. A congestion level estimator has been proposed for the monitoring and we present it at the end of this section.

Note that this architecture is just one proposal among others for future PCN-based admission control and flow termination. There are also other ideas, e.g., the "single marking" approach which requires only a single bit for traffic marking which supports both admission control and flow termination [13]. Single-marking requires a different marking behavior which is not covered in this study.

3.3 Marking Algorithms to Support Admission Control

Admission control requires a meter and marker that marks all packets if the PCN rate $r(l)$ on a link l exceeds its admissible rate $AR(l)$. The IETF currently discusses two marking alternatives for that purpose that we present as a virtual queue formulation. Note that a token bucket based formulation is also possible [14].

Threshold Marking Threshold marking has been presented in [14] and mentioned under the name "step marking" in [15] as a special case of ramp marking.

The virtual queue (VQ) algorithm simulates the development of the length $VQ.L$ of a queue with a rate $VQ.R$ and a size $VQ.S$. The rate and the size may be given in bytes or packets per second and in bytes or packets, respectively. We consider a VQ based on bytes. Algorithm 1 gives the pseudo-code for a VQ that marks all packets if its current queue length $VQ.L$ exceeds its marking threshold $VQ.T$. The VQ records its last update by the variable $VQ.lU$. At the beginning, the time since the last update of the queue is calculated using the current time now. The length of the queue is reduced by the number of bytes that could be served since then to obtain the length of the queue shortly before the packet arrival (now). The algorithm is called whenever a packet arrives. If the current length $VQ.L$ of the VQ is larger than its marking threshold $VQ.T$, the packet is out of profile and marked with an AS codepoint. Then, $VQ.L$ is increased by the size of the packet, but the VQ cannot exceed its maximum size $VQ.S$. Finally, the variable recording the last update $VQ.lU$ is updated.

Input: $VQ, packet, now$

$VQ.L = \max(0, VQ.L - (now - VQ.lU) \cdot VQ.R);$ {virtual queue length shortly before packet arrival}
if $(VQ.L > VQ.T)$ **then**
  $packet.mark =$ AS;
end if
$VQ.L = \min(VQ.S, VQ.L + packet.S);$ {virtual queue length shortly after packet arrival}
$VQ.lU = now;$

Algorithm 1: THRESHOLD MARKING

If the traffic rate exceeds the VQ rate $VQ.R$, the queue length $VQ.L$ increases, eventually exceeds the threshold $VQ.T$, and stays above that threshold such that all further packets are marked. If the traffic rate falls below the VQ rate $VQ.R$, the VQ length $VQ.L$ decreases, eventually falls below the threshold $VQ.T$, and stays below that threshold such that packet marking stops.

Input: $VQ, packet, now$

$VQ.L = \max(0, VQ.L - (now - VQ.lU) \cdot VQ.R)$; **{**virtual queue length shortly before packet arrival**}**
if $(VQ.L > VQ.T_{ramp})$ **then**
  if $(VQ.L < VQ.T)$ **then**
    if $(rand() < \frac{VQ.L - VQ.T_{ramp}}{VQ.T - VQ.T_{ramp}})$ **then**
      $packet.mark =$ AS;
    end if
  else
    $packet.mark =$ AS;
  end if
end if
$VQ.L = \min(VQ.S, VQ.L + packet.S)$; **{**virtual queue length shortly after packet arrival**}**
$VQ.lU = now$;

Algorithm 2: RAMP MARKING

Ramp Marking Ramp marking has been described in [15] and its pseudo-code is given by Algorithm 2. The VQ-based mechanism works essentially like threshold marking, but it has a lower marking threshold $VQ.T_{ramp}$ and an upper marking threshold $VQ.T$. If the length $VQ.L$ of the VQ is in between, packets are marked with a linearly increasing probability. If $VQ.L$ is above $VQ.T$, all packets are marked. The function $rand()$ returns a random number, which is uniformly distributed between 0 and 1, to support the probabilistic decision.

Ramp marking is clearly inspired by the RED queue [16]. However, its marking decision is based on the current VQ length $VQ.L$ instead of the average length of the physical queue. Moreover, RED algorithms are more complex as they modify the marking or dropping probability depending on the recently marked or dropped packets.

Comparison The PCN working group of the IETF currently debates whether ramp or threshold marking should be used for admission-stop marking. Figures 1 and 2 show the marking probability of both approaches depending on the current length L of the virtual queue. While threshold marking starts marking only at a certain threshold T, ramp marking starts marking already at a lower threshold T_{ramp} with a linearly increasing probability up to the same threshold T, from which on all packets are marked.

The advantage of threshold marking is its simplicity. It has only three parameters: the rate R, the marking threshold T, and the queue size S whereas ramp marking requires in addition the parameter T_{ramp} indicating the beginning of the probabilistic marking range. Thus, threshold marking is not only easier to configure but also easier to implement because its decisions are not stochastic like those of ramp marking which require random numbers.

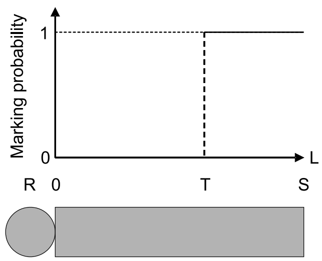


Fig. 1. The threshold marker marks packets if the length of its virtual queue exceeds its threshold T.

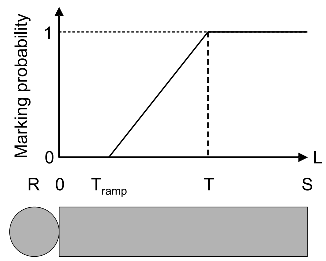


Fig. 2. The ramp marker marks all packets if the length of its virtual queue exceeds its threshold T, but it also marks packets probabilistically between T_{ramp} and T.

3.4 Congestion Level Estimator

As mentioned above, the egress nodes monitor the packet markings for each ingress-egress aggregate. This can be done by a congestion level estimator. Whenever a packet arrives, the congestion level estimator interprets a non-marked packet as 0 and a marked packet as 1. It applies an exponentially weighted moving average (EWMA) to these values to obtain time-dependent averages using

$$CLE_{n+1} = w \cdot CLE_n + (1-w) \cdot X_n \tag{1}$$

whereby X_n is a random variable which is 1 if packet n is marked and 0, otherwise. Rewriting Equation (1) shows that samples X_i contribute for longer time to the CLE but with decreasing intensity which is controlled by the weight parameter $w < 1$:

$$CLE_{n+1} = (1-w) \cdot \sum_{0 \leq i \leq n} w^i \cdot X_{n-i} \tag{2}$$

We can quantify the dynamics of the EWMA by two different approaches: the half-life time and the memory.

Half-Life Time T_H Initially, new values contribute with $(1-w)$ to the average sum; n arrivals later, they count only $(1-w) \cdot w^n$. Thus, the value counts only half after $n = \lceil \frac{-ln(2)}{ln(w)} \rceil$ arrivals. If Δ is the average time between arrivals, the half-life time of the samples X_i in the EWMA is $T_H = \lceil \frac{-ln(2)}{ln(w)} \rceil \cdot \Delta$.

Memory M The memory of the EWMA reflects how long a sample X_i contributes to the average result weighted by its strength which is explicit in Equation (2). We can calculate this memory by

$$M = \sum_{0 \leq j < \infty} (j+1) \cdot \Delta \cdot (1-w) \cdot w^j = \frac{\Delta}{1-w}. \tag{3}$$

The concepts of half-life time and memory are equivalent and help to characterize how a specific sample X_i affects the EWMA value over time using a single parameter (either T_H or M). They are more meaningful than the weight parameter w as this requires also the mean inter-update time Δ to judge the dynamics, but whose indication is mostly neglected.

A consequence from these equations is that weight parameters must be larger on faster links if past overload should be forgotten after the same time. This knowledge about the EWMA behavior is useful when we study the impact of the EWMA settings on the marking result. In practice, it is hard to control the memory rigidly because the packet rate arriving at the congestion level estimator is a priori not known and changes. Thus, the EWMA is more oblivious concerning the time if packets arrive faster than expected.

4 Sensitivity of Congestion Level Estimates to Marking Options and Traffic Characteristics

After explaining our experiment setup and performance metric, we first study the impact of the marking threshold T and the queue size S on the time average of the CLE depending on the traffic intensity. Based on these results, we develop two different marking strategies. We investigate the influence of ramp marking and provide parameters for threshold marking leading to the same CLE. We illustrate the impact of the memory of the congestion level estimator on the CLE values. We study the sensitivity of the results to different traffic characteristics and calculate the reaction speed of the markers in case of sudden overload.

4.1 Experiment Setup and Performance Metric

We use a custom-made simulator programmed in Java. The setup of our experiments is illustrated in Figure 3. Packets from n independent, homogeneous traffic sources are multiplexed onto a single link with infinite bandwidth and pass a meter and marker. The markings are evaluated by a subsequent congestion level estimator.

If not mentioned differently, we simulate around $n = 100$ homogeneous flows for sufficiently long time to obtain reliable results. However, we omit confidence intervals in all our graphs for the sake of clarity. We choose a Gamma distribution to generate the inter-arrival times A between consecutive packets within a flow with a mean of $E[A] = 20$ ms and a coefficient of variation of $c_{var}[A] = 0.1$. The packet sizes B are independent and distributed according to a deterministic phase of 50 bytes plus a negative binomial distribution. Their overall mean is $E[B] = 200$ bytes and their coefficient of variation is $c_{var}[B] = 0.5$. The values for $E[A]$ and $E[B]$ are motivated by typical voice connections that periodically send every 20 ms a packet with 160 bytes payload using a 40 bytes IP/UDP/RTP header. However, our flow model is not periodic and has variable packet sizes. We use it for two reasons. The simulation of multiplexed, strictly periodic traffic requires special care due to the non-ergodicity of the system and is very time consuming. Therefore, we relax $c_{var}[A] = 0.0$ to $c_{var}[A] = 0.1$. Furthermore, we use $c_{var}[B] = 0.5$ instead of $c_{var}[B] = 0.0$ because realtime traffic consists of packets from different applications with and without compression which leads to different packet sizes. Table 1 provides an overview of the packet sizes used in this study. However, our findings are general and do not depend on special parameter settings.

The rate of the virtual queue is $R = 8$ Mbit/s such that at most 100 flows can pass unmarked. The congestion level estimator implements an EWMA and counts packets with admission-stop marks as 1 and those without as 0. As outlined in Section 3.4, its memory M depends on the packet rate and the weight parameter w such that w needs to be adapted to the desired M and the packet frequency in the experiment for which we take the maximum packet rate that can pass unmarked. Thus, we set the weight parameter to $w = 0.998$ which corresponds to a memory of 0.1 s when 100 default flows are active. If the packet rate changes due to more bursty traffic, we adapt the weight parameter w to have the same memory.

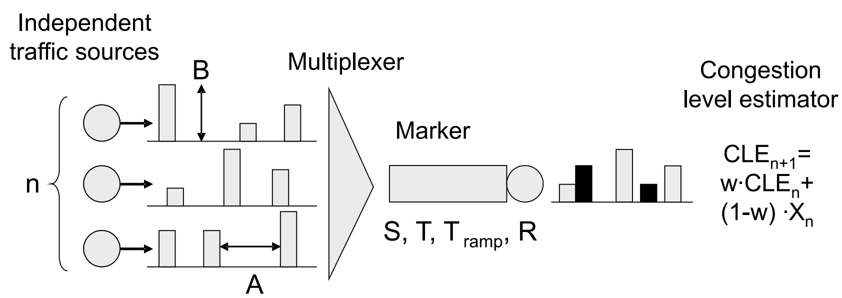


Fig. 3. Experiment setup.

Table 1. Statistical information in bytes about packet sizes B used in the simulations.

$c_{var}[B]$	$E[B]$	$\min[B]$	1% quantile	10% quantile	90% quantile	99% quantile
0.0	200	200	200	200	200	200
0.5	200	50	63	94	334	552
1.0	200	50	50	53	446	985
0.5	1000	50	219	439	1667	2521

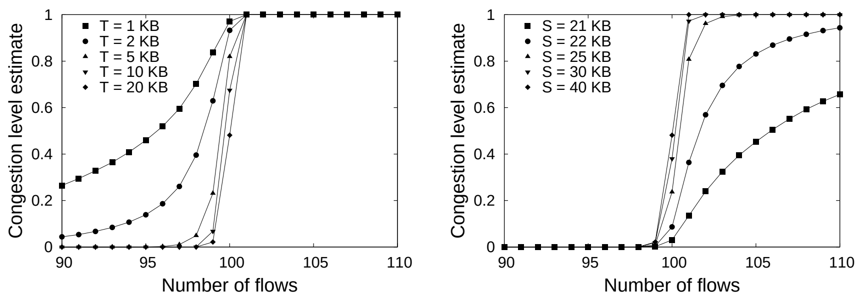
4.2 Impact of the Marking Threshold T and the Queue Size S

We first study the impact of the marking threshold T and then the one of the remaining queue size $S - T$.

We vary the marking threshold T and keep the remaining queue sizes fixed at $S - T = 20$ KB. Figure 4(a) shows the average CLE depending on the number of multiplexed flows. It increases with increasing traffic intensity. We observe that the CLE values converge for increasing traffic intensity, but they significantly differ at low load. If less than 100 flows are carried, the virtual queue is empty most of the time. However, even then their short-time rate can exceed the one of the virtual queue. As a consequence, the queue length increases and possibly goes beyond T. Packet marking starts and the CLE increases. This behavior is favored by small marking thresholds T. Thus, if the virtual queue rate is under-utilized, the probability for a large CLE decreases with increasing T.

In a similar way, we now keep the marking threshold T fixed at 20 KB and vary the remaining queue size $S - T$, i.e., we vary S. The curves in Figure 4(b) are all close to CLE=0 when the virtual queue is under-utilized. In contrast, the traffic intensity at which the curves arrive at CLE=1 depends heavily on the queue size. If more than 100 flows are carried, the virtual queue is completely filled most of the time. However, even then their short-time rate can fall below the one of the virtual queue. As a consequence, the queue length decreases and possibly falls below T. Packet marking stops and the CLE decreases. This behavior is favored if the queue size S exceeds the marking threshold T only by little, i.e., if $S-T$ is small. Thus, if the virtual queue rate is over-utilized, the probability for a large CLE increases with increasing $S-T$.

Thus, a large marking threshold T keeps the CLE small if the virtual queue is under-utilized and a large remaining queue size guarantees that the CLE is large if the virtual queue is over-utilized.



(a) Remaining queue size $S-T=20$ KB. (b) Marking threshold $T=20$ KB.

Fig. 4. CLE for threshold marking with variable queue sizes S.

4.3 Two Marking Strategies with Different Admission Control Policies

We construct threshold markers with two different CLE characteristics.

Marking with Clear Decisions (MCD) To obtain a marker with clear decisions, we need a large marking threshold T and a large remaining queue size $S-T$. Figure 5 illustrates that the corresponding CLE curve for threshold marking (TM) is close to 0.0 as long as the traffic rate is below the virtual queue rate and close to 1.0 if the traffic rate is above. As a consequence, new flows can be admitted if the current CLE is low, e.g. 0.3; otherwise, they are rejected. The point style of the curves in Figures 5–8(f) indicates the experiment and the line style indicates the mean, the 10%-, or 90%-quantiles of the CLE. The 10%- and 90%-quantiles of the CLEs for TM are very close to their averages. That means, that the obtained CLE is very reliable and the probability to falsely reject or accept flows is rather low.

Marking with Early Warning (MEW) To obtain a marker with early warning, we use a low marking threshold T and a large remaining queue size $S-T$. Figure 6 illustrates that the corresponding CLE curve for TM with $T=3$ KB and $S=40$ KB rises gently between 0.0 and 1.0 as an increasing traffic intensity approaches the virtual queue rate, and it is close to 1.0 if the traffic rate is above. As a consequence, new flows can be admitted if the current rate CLE is below 0.95; otherwise, they are rejected. The benefit of this approach is that CLE values between 0.1 and 0.95 can be interpreted as early warning of an almost fully loaded system. This information is useful to reduce the frequency of further admissions to avoid over-admission in the presence of weak flash crowds (a large number of arrivals within an exceptionally short interval). The percentiles show that the early warning information fluctuates considerably, i.e. the CLE gives only a hint regarding the current utilization but no reliable information. This makes it hard to infer the exact utilization of the virtual queue rate from the CLE values.

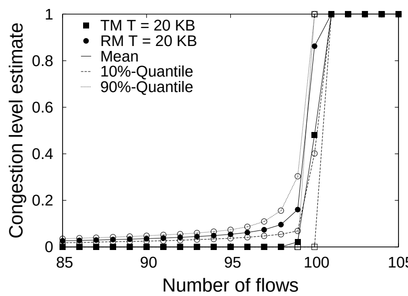


Fig. 5. Marking with clear decisions (MCD): CLEs should be 0 if traffic rate is below the virtual queue rate and 1 if it is above.

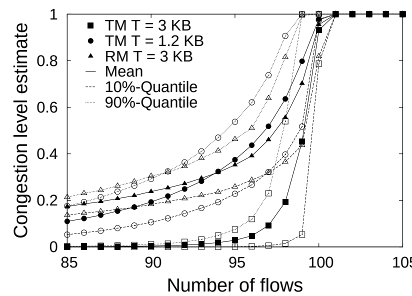


Fig. 6. Marking with early warning (MEW): CLEs should gently increase from 0 to 1 if the traffic rate is below the virtual queue rate and stay at 1 if it is above.

4.4 Impact of Ramp Marking

Ramp marking already marks packets probabilistically if the virtual queue length is below the marking threshold T (cf. Figure 2). Therefore, it marks more packets than threshold marking with the same marking threshold T and queue size S. In our study we always set the lower marking threshold to $T_{ramp} = 0$.

In Figure 5 we compare the behavior of ramp marking with the one of threshold marking for MCD using the parameters $T = 20$ KB and $S = 40$ KB. The CLEs of threshold marking exactly match the idea of MCD while those of ramp marking are clearly above 0 over the studied range. In particular, they show higher variability if the network is almost fully loaded such that some request might be falsely rejected. Thus, we do not see any advantage of ramp marking over threshold marking in case of MCD.

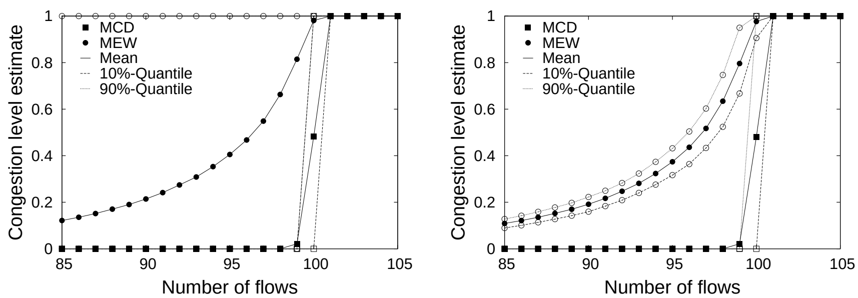
In Figure 6 we compare the behavior of ramp marking with the one of threshold marking for MEW using the parameters $T = 3$ KB and $S = 40$ KB. Ramp marking yields higher CLEs and earlier and more linear indication of an approaching saturation of the traffic load than comparable threshold marking. However, a very similar curve can be achieved with threshold marking using $T = 1.2$ KB instead of $T = 3$ KB. In addition, the shape of the curve of the modified threshold marker is even better suited for inferring the load from the CLE value as it is lower at low utilization values. Hence, there is no obvious advantage of ramp marking over threshold marking, either, at least not in this considered scenario.

We do not explicitly discuss ramp marking with values $0 < T_{ramp} < T$ because this leads to interpolations between the curves for ramp and threshold marking which are given in Figures 5 and 6 for $T = 20$ KB and $T = 3$ KB, respectively.

4.5 Impact of the Memory M of the Congestion Level Estimator

We study the impact of the memory M of the congestion level estimator (cf. Section 3.4) on the obtained CLE values. While Figures 5 and 6 present the results for $M = 0.1$ s, Figures 7(a) and 7(b) show the CLEs for threshold marking ($S = 40$ KB, $T = 20$ KB or $T = 3$ KB, respectively) for $M = 0$ s and $M = 1$ s. We observe that the memory M has hardly any influence on the average values of the CLE.

In contrast, the memory significantly impacts the percentile curves for MEW. With a memory of $M = 0$ s the CLE takes only values 0 and 1 such that the percentiles are also either 0 or 1. For $M = 0.1$ s the percentile curve in Figures 5 and 6 come closer to the average curve and even more close for $M = 1$ s in Figure 7(b). Therefore, a long memory M is good for MEW as it makes the obtained CLE values more reliable. However, the memory cannot be increased to arbitrarily high values because then the congestion level estimator reacts too late when the average traffic rate changes.



(a) Memory $M = 0$ s. (b) Memory $M = 1$ s.

Fig. 7. The congestion level estimator's memory influences the stability of the CLE (threshold marking with $S = 40$ KB and $T = 20$ KB or $T = 3$ KB, respectively).

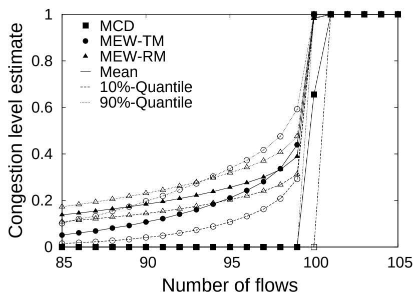
For MCD, the percentile curves almost coincide with the average curves for all three values of memory M. Thus, this marking strategy is very robust and its robustness increases with the the marking threshold T and the remaining queue size $S - T$. We do not underpin this observation by figures in this paper.

4.6 Impact of Traffic Characteristics

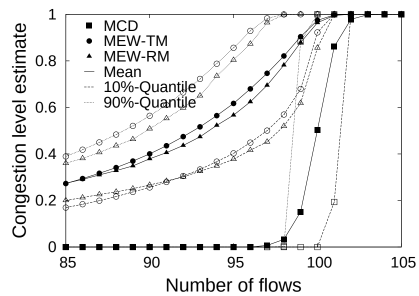
Now, we investigate how traffic characteristics influence the CLE values obtained in Figure 5 for MCD and threshold marking and in Figure 6 for MEW and both threshold and ramp marking.

Figure 8(a) shows the CLE values for equal packet sizes ($c_{var}[B] = 0.0$) which has less short-term variation compared to the default traffic. Therefore, the curves for MEW are slightly lower in the left part of the figure than in Figure 6.

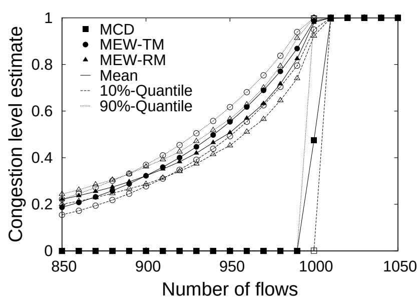
In contrast, we increase the variability of the traffic in Figures 8(b)–8(d) by increasing the coefficient of variation of the packet size to $c_{var}[B] = 1.0$, the rate of the virtual queue to accommodate $n = 1000$ flows, or the coefficient of the inter-arrival time to $c_{var}[A] = 1.0$. As a result, the figures show average CLE curves for MEW that are sightly higher than those in Figure 6. The figures mainly differ in the quantile curves.



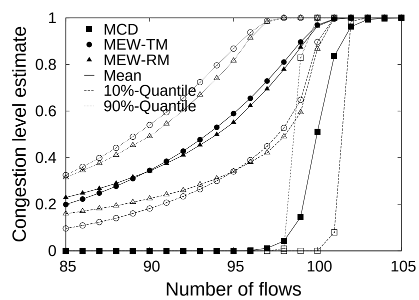
(a) Decreased packet size variation: $c_{var}[B] = 0.0$ instead of $c_{var}[B] = 0.5$.



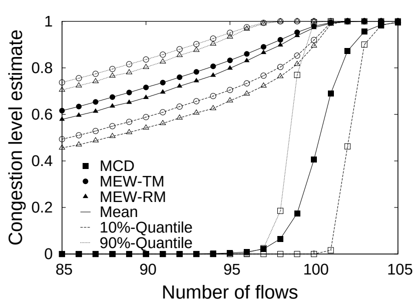
(b) Increased packet size variation: $c_{var}[B] = 1.0$ instead of $c_{var}[B] = 0.5$.



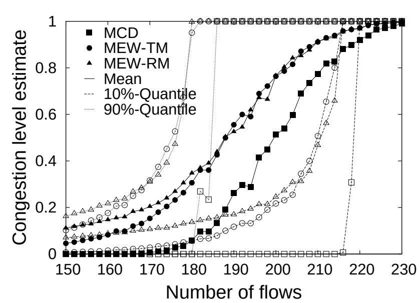
(c) Increased aggregation level: capacity for $n = 1000$ flows instead of $n = 100$ flows.



(d) Increased inter-arrival time variation: $c_{var}[A] = 1.0$ instead of $c_{var}[A] = 0.1$.



(e) Increased burstiness: $E[A] = 100$ ms and $E[B] = 1000$ bytes instead of $E[A] = 20$ ms and $E[B] = 200$ bytes.



(f) Increased long-term variation: on/off flows instead of continuous flows.

Fig. 8. CLEs for MCD based on threshold marking ($T = 20$ KB, $S = 40$ KB) and for MEW based on threshold marking ($T = 1.2$ KB, $S = 40$ KB) and on ramp marking ($T_{ramp} = 0$ KB, $T = 3$ KB, $S = 40$ KB).

Increasing the burstiness of the traffic by scaling the mean inter-arrival time and packet size by a factor 5 also adds more variability to the traffic, but its influence is dramatic: the average CLE curves for MEW in Figure 8(e) are twice as high as in Figure 6. This makes a problem of MEW obvious: a CLE value of 0.4 can signify extremely high load in the presence of very smooth traffic (cf. Figure 6) or extremely low load in the presence of very bursty traffic (cf. Figure 8(e)). Thus, mechanisms taking advantage of early warning need to know the traffic characteristics. Furthermore, the 90% quantiles of the CLE reach 0.95 quite early such that more false negatives occur for MEW than for smoother traffic, i.e., flows are rejected although the average traffic rate is still below the virtual queue rate.

The almost vertical step-up of the threshold marking curve for MCD in Figure 5 is diluted a bit through the increased variability of the traffic. In further experiments (not shown in this paper) we could show that increasing the threshold parameter T and S again leads to an abrupt jump of the curves. However, as we will show in the next section, the marking threshold T should not be set to an arbitrarily high value because large values of T slow down the reaction speed of the marking algorithm in case of sudden overload.

We now consider on/off traffic with exponentially distributed on- and off-phases with a mean of 10 s. We install the double number of flows to achieve the same aggregate rate as with continuous flows. Figure 8(f) shows that the CLE values for all marking methods rise linearly over a wide range of traffic rates. The quantile curves show that there is a 10% chance for exceeding the virtual queue rate with $n = 186$ flows as well as a 10% chance of not reaching it with $n = 214$ flows. The reason for this significantly different behavior is the fact that on/off traffic comes with medium-term traffic fluctuations. Given $n = 180$ admitted flows, on average only 90 of them are active leading to a mean rate of 7.2 Mbit/s, but there is also a good chance that 105 of them are active for a while leading to 8.4 Mbit/s. Thus, MCD marks 100% of the packets if their rate exceeds 8 Mbit/s for some time and it does not mark them if their rate is below that value. In both cases, the admission decisions are correct since on/off traffic causes not only short-term but also medium-term rate fluctuation. Hence, it is hard to avoid overload just by doing PCN-based admission control since the rate of admitted traffic can increase. Hence, the use of PCN-based admission control to limit the number of on/off flows is a different problem and requires a separate study. A solution is setting the admissible rate to a value which is low enough that no problems occur if this value is slightly exceeded by the current traffic rate.

At the end of this sensitivity study we would like to point out that ramp marking for MEW behaves very similar as threshold marking in all considered scenarios and, therefore, we do not see any advantage of ramp marking over threshold marking.

4.7 Response Time of the Marking to Sudden Overload

We consider the reaction speed of the marking in case of sudden overload as it can occur in case of reroutes. To that end, we assume an empty virtual queue and a sudden overload of k flows, each having a rate of C. Thus, the entire overload rate is $k \cdot C$ and the queue length takes $\frac{T}{k \cdot C}$ time to reach the marking threshold. If we use our default parameters $C = 80$ Kbit/s, $T = 20$ KB for MCD, and $T = 1.2$ KB for MEW, it takes

40 ms for MCD and 2.4 ms for MEW to detect an overload of 50%, i.e. $k = 50$ flows. Thus, MEW can react faster than MCD in case of sudden overload due to its smaller marking threshold.

However, the marking threshold T for MCD can be decreased as this mainly affects the accurate shape of step function for traffic rates below the virtual queue rate. This is backed by Figure 4(a) and is not a serious problem for the admission decision as long as large CLEs are not reached for traffic rates significantly lower than the virtual queue rate.

In a similar way the reaction time of the marking can be calculated when the traffic rate was above the virtual queue rate falls then below it. It takes up to $\frac{S-T}{k \cdot C}$ time until the marking stops whereby the number of flows k indicate the free capacity $k \cdot C$. Although we set the queue size $S = 40$ KB for both MCD and MEW in our experiments, MEW works fine also with a remaining queue size of $S - T = 20$ KB. If the free capacity suffices for $k = 1$ flow, it takes 2 s until the marking stops while it takes only 200 ms for $k = 10$. These values are larger than the response times to significant overload, but less important.

5 Conclusion

One option for pre-congestion notification (PCN) based admission control requires that all packets are marked if the current link rate exceeds a pre-configured admissible rate. This can be achieved by virtual queue based marking algorithms such as simple threshold marking or more complex ramp marking.

The objective of this work was to study how marking algorithms can support admission control in order to limit the utilization of the links of a network. We did not consider the use of marking algorithms to support admission control in order to limit the packet delay because we assume that PCN will be used in high-speed networks where packet delay caused by queuing is negligible as long as link utilizations are moderate.

We investigated the influence of the parameters of the marking algorithms on their marking results which are translated into a congestion level estimate (CLE) using EWMA-based averaging. We showed that two different marking strategies can be pursued: marking such that the CLE leads to clear decisions (MCD) and marking such that the CLE yields early warning (MEW) when the rate of PCN traffic on a link approaches its admissible rate. We provided recommendations for the configuration of the marking threshold T and the size S of the virtual queue in both cases. Ramp marking increases the level of early warning compared to threshold marking, but this can be approximated by smaller marking thresholds for simple threshold marking such that there is no obvious need for ramp marking.

The CLE values for MEW fluctuate, therefore, it is difficult to infer the exact, current traffic rate from the CLE values which is required to take advantage of early warning. A sensitivity study revealed that the average CLE values for MEW depend heavily on the traffic characteristics. This makes the use of early warning difficult: either the marking parameters need to be adapted to produce similar warnings for different traffic types or the mechanism taking early warning into account requires knowledge about the traffic characteristics to correctly interpret the CLE level. In contrast, CLE values for MCD show hardly any variation and are robust against different traffic types.

For the sake of simplicity, we advocate for the use of MCD for PCN-based admission control instead of MEW because the interpretation of early warning is difficult due to its high variation and dependency on traffic characteristics. Furthermore, we think that ramp marking is not needed for PCN since similar markings can be obtained by appropriately configured threshold marking and we do not see any benefit that justifies the implementation complexity of ramp marking.

Acknowledgements

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

References

1. Shenker, S.: Fundamental Design Issues for the Future Internet. IEEE Journal on Selected Areas in Communications **13**(7) (1995) 1176–1188
2. Braden, B., Clark, D., Shenker, S.: RFC1633: Integrated Services in the Internet Architecture: an Overview (1994)
3. Jamin, S., Danzig, P., Shenker, S.J., Zhang, L.: Measurement-Based Admission Control Algorithms for Controlled-Load Services Packet Networks. In: ACM SIGCOMM. (1995)
4. Breslau, L., Knightly, E.W., Shenker, S., Zhang, H.: Endpoint Admission Control: Architectural Issues and Performance. In: ACM SIGCOMM. (2000)
5. Stoica, I., Zhang, H.: Providing Guaranteed Services without per Flow Management. ACM SIGCOMM Computer Communications Review **29**(4) (1999)
6. Szábó, R., Henk, T., Rexhepi, V., Karagiannis, G.: Resource Management in Differentiated Services (RMD) IP Networks. In: International Conference on Emerging Telecommunications Technologies and Applications (ICETA 2001), Kosice, Slovak Republic (2001)
7. Gibbens, R.J., Kelly, F.P.: Distributed Connection Acceptance Control for a Connectionless Network. In: 16^{th} International Teletraffic Congress (ITC), Edinburg (1999) 941 – 952
8. Kelly, F., Key, P., Zachary, S.: Distributed Admission Control. IEEE Journal on Selected Areas in Communications **18**(12) (2000) 2617–2628
9. Karsten, M., Schmitt, J.: Admission Control based on Packet Marking and Feedback Signalling – Mechanisms, Implementation and Experiments. Technical Report 03/2002, Darmstadt University of Technology (2002)
10. Karsten, M., Schmitt, J.: Packet Marking for Integrated Load Control. In: IFIP/IEEE Symposium on Integrated Management (IM). (2005)
11. Ramakrishnan, K., Floyd, S., Black, D.: RFC3168: The Addition of Explicit Congestion Notification (ECN) to IP (2001)
12. Eardley (ed.), P.: Pre-Congestion Notification Architecture. http://www.ietf.org/internet-drafts/draft-ietf-pcn-architecture-02.txt (2007)
13. Charny, A., Faucheur, F.L., Liatsos, V., Zhang, J.: Pre-Congestion Notification Using Single Marking for Admission and Pre-emption. http://www.ietf.org/internet-drafts/draft-charny-pcn-single-marking-02.txt (2007)
14. Babiarz, J., Liu, X.G., Chan, K., Menth, M.: Three State PCN Marking. http://www.ietf.org/internet-drafts/draft-babiarz-pcn-3sm-01.txt (2007)
15. Briscoe, B., Eardley, P., Songhurst, D., Faucheur, F.L., Charny, A., Babiarz, J., Chan, K., Dudley, S., Karagiannis, G., Bader, A., Westberg, L.: Pre-Congestion Notification Marking. http://www.ietf.org/internet-drafts/draft-briscoe-tsvwg-cl-phb-03.txt (2006)
16. Floyd, S., Jacobson, V.: Random Early Detection Gateways for Congestion Avoidance. IEEE/ACM Transactions on Networking **1**(4) (1993) 397–413