

# An Autonomic PCN based Admission Control Mechanism for Video Services in Access Networks

Steven Latré, Bart De Vleeschauwer, Wim Van de Meerssche, Simon Perrault, Filip De Turck and Piet Demeester  
Ghent University - IBBT - IBCN - Department of Information Technology  
Gaston Crommenlaan 8/201, B-9050 Gent, Belgium  
Tel: +3293314981 , Fax: +3293314899  
e-mail: steven.latre@intec.ugent.be

Koen De Schepper, Christian Hublet, Wouter Rogiest, Stefan Custers and Werner Van Leekwijck  
Alcatel-Lucent Bell Labs  
Copernicuslaan 50, B-2018 Antwerpen, Belgium

**Abstract**—The introduction of new added value services such as IPTV has introduced great challenges for today’s broadband DSL access networks as these services have stringent quality demands. In an attempt to protect the quality delivery of existing sessions, operators employ admission control mechanisms that limit the amount of sessions transmitted in the network. Current admission control mechanisms require a traffic specification of each stream, in order to know beforehand how many resources need to be reserved. For variable bit rate videos, which are bursty of nature, resources are reserved using the peak rate of the video. This leads to under-utilisation of the network as the amount of resources needed is over-dimensioned. We propose an autonomic measurement based admission control algorithm, optimised for the protection of video services in multimedia access networks. The algorithm is based on the IETF Pre Congestion Notification (PCN) mechanism and autonomously adjusts its parameters to the traffic characterisation of the video. The performance of this mechanism has been extensively evaluated in a packet based network simulation environment. Tests show that the autonomic nature of the algorithm leads to a better utilisation of the network while still avoiding any congestion in the network.

## I. INTRODUCTION

Broadband DSL access networks have undergone a significant evolution in the last years. Starting as a mere data packet forwarder, their complexity has increased considerably through the introduction of Triple Play Services, consisting of high speed internet, video services (e.g. real-time broadcast service, videophony) and telephony. Telephony and video services, often called multimedia services, have very stringent demands in terms of packet loss, delay and jitter. Even small levels of packet loss can easily deteriorate the quality of these services and introduce visual or audial artefacts in the transmitted service.

The main reason for packet loss in access networks is congestion, which is in turn caused by admitting too much sessions at the same time. The resulting packet loss impacts all sessions on the congested link, deteriorating the quality for many users at once. This situation is typically avoided by employing resource admission control techniques that limit the amount of sessions. Today, mostly centralised approaches are

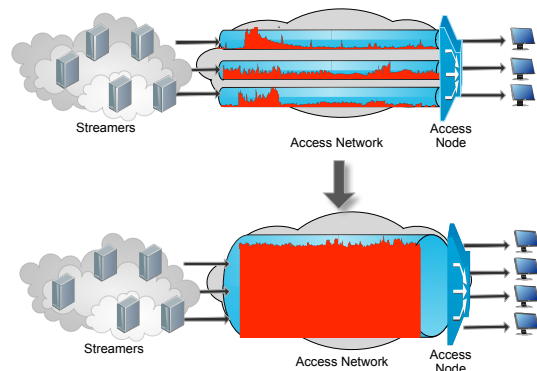


Fig. 1. Possible ways to realise an admission control system for video services. Top: the current approach where resources are reserved using the videos peak rate. Bottom: new approach where the aggregate of sessions is protected, leading to a better utilisation. Using the new approach of aggregate protection allows to carry more sessions over the same link.

used, which have an overall good performance for narrowband constant bit rate services with a traffic specification known a priori such as telephony, but are sub-optimal for video services, which exhibit bursty behaviour and have a large bandwidth variability. The traffic specification of video services heavily depends on the type of codec, video resolution (e.g. HD vs. SD) and encoder settings making it hard to know beforehand which resources need to be reserved.

As illustrated in Figure 1, operators have tried to tackle this issue by protecting each session one by one using the peak bit rate of the video as the amount of resources to reserve. While this effectively protects the video from congestion, a lot of bandwidth is wasted as this peak rate represents the worst case scenario, occurring at most a few times during the complete service, as a single VBR session is usually very variable in terms of bandwidth consumption. This results in an under-utilised access network. A more efficient way to utilise resources is to protect only the aggregate of sessions as opposed to protecting every individual session independently. This technique relies on the observation that, in practice, the relative variability of the aggregate is lower than that of the

individual sessions. In general, this gain does not necessarily occur. For instance, in the case of synchronised video sessions, for which the aggregate has a limited reduction in variability. However, these pathological cases are not typical in practical settings, and are not considered in this work. As such, we can assume that the aggregate will behave more predictably, and the peak rates of individual sessions will cancel out thanks to statistical multiplexing gain. The limited variability of the aggregate allows to carry more video sessions over the same link, enabling better resource usage than in the case of protection on a per-session basis.

Recently, the IETF suggested a measurement based approach for performing session admission, called Pre-Congestion Notification (PCN) [1]. In PCN, the admittance decision is based on previous measurement of the congestion level. In this paper, we evaluate PCN's performance for bursty video services and present a novel autonomic measurement based admission control mechanism which is able to protect the transmission of these services in multimedia access networks and maximise the link utilisation. The mechanism can cope with the bursty nature of video services and automatically adapts to the overall variability of the traffic leading to a better utilised network, while at the same time protecting the existing sessions. The autonomic algorithm can be used in the PCN architecture as an alternative to the current measurement algorithm.

The remainder of this paper is structured as follows. In Section II, an overview is given of both centralised and distributed admission control mechanisms. Section III focuses on the recently proposed PCN and discusses possible algorithmic approaches for the PCN algorithms. These approaches are evaluated in Section IV. In Section V, our novel autonomic algorithm is proposed and compared with a static variant of the same measurement algorithm. Finally, Section VI concludes this paper and discusses future work.

## II. RELATED WORK

In today's broadband DSL access networks, mostly centralised admission control mechanisms exist such as the Resource Admission Control Subsystem in TISPAN [2] or the Bandwidth Broker [3] in Diffserv. These centralised approaches require the knowledge of the network topology, the dimensioned resources and the route being followed by any flow consuming controlled resources. The main limitations of this approach is that the knowledge needed by the system can be large and difficult to keep up to date, especially when the network is reconfiguring itself as a consequence of a link or node failure.

As an alternative for centralised admission control mechanisms, several decentralised mechanisms have been proposed that allow or block sessions based on local knowledge, either through measurement [4], [5] or by deriving local rules out of centralised policies [6]. While these are valuable solutions, they have not yet been standardised or widely adopted. Of late, the IETF community is researching a measurement based admission control mechanism through the PCN Working

Group [7]. PCN [1] is a measurement based admission control mechanism that tries to protect the Quality of Service (QoS) of established inelastic flows within a Diffserv domain. In PCN, the network load is measured and signalled through the marking of packets. The inner workings of PCN are still being defined but the publication of the PCN architecture [1] document as RFC has been requested.

In recent work, different PCN based algorithms have been evaluated through simulation [8], [9]. Our work differs from this study in several ways. First, we focus on the transmission of video services, while [8] focuses on the transmission of narrowband services (e.g. VoIP). Currently, traffic which does not have a known maximum rate (e.g. video services) is out of scope in the PCN Working Group. Second, in [8] the accuracy of different algorithms is evaluated by transmitting different traffic types through the network and investigating the calculated congestion level. In the performed simulations no feedback loop, which decides to allow or block sessions, is present. In our work, we have incorporated such a feedback loop where sessions are requested given a request arrival model and sessions are allowed or blocked dynamically. Third, while we focus on PCNs measurement algorithm, [8] focuses on other aspects of the PCN architecture such as marking and encoding algorithms. In [9], the configuration of the threshold rate, one of PCNs parameters, is studied for narrowband services. Our work also focuses on this threshold rate but proposes an autonomic algorithm, optimised for bursty traffic, that dynamically adapts itself based on previous measurements.

## III. PRE CONGESTION NOTIFICATION MECHANISM

### A. PCN Architecture

As illustrated in Figure 2, three different node types are present in the PCN architecture: ingress, egress and interior nodes. At the interior node, the congestion level is measured and packets are marked once the network load exceeds a certain threshold. At the egress node, the marked packets are aggregated and the CLE is calculated using an Exponential Weighted Moving Average (EWMA). The CLE value at time  $n$  is calculated as:

$$CLE_n = X * (1 - w) + w * CLE_{n-1}, w \in [0, 1]$$

where  $w$  is the CLE weight and  $X$  is 1 if the packet is marked and 0 if the packet is not. The higher the CLE weight, the more previous measurements will contribute to the overall CLE value. Once calculated, the CLE information is signalled back to the ingress node which is responsible for allowing or blocking sessions. For more information about PCN, we refer to [1].

The work of the PCN working group is far from finalised but its applicability has been restricted by some assumptions. The two most important assumptions relate to the behaviour of the traffic. First, PCN assumes that the considered traffic is in-elastic and constrained to a known maximum rate. Second, on any potential bottleneck the number of streams must be large enough to apply stateless statistical mechanisms.

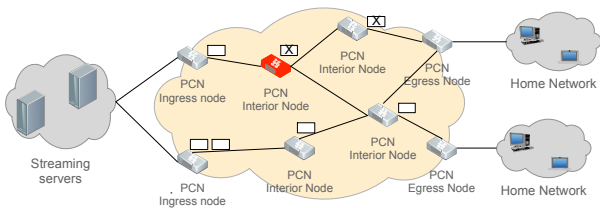


Fig. 2. Overview of the PCN architecture. Streaming servers are transmitting video to various home networks. PCN interior nodes measure the network load and start marking packets if the load becomes too high.

Through these assumptions, PCN can be applied to protect narrowband inelastic streams (e.g. VoIP in a core). However, these assumptions are violated if PCN is applied to protect video streams having a constant quality. These videos have a variable bitrate where no maximum rate is known beforehand and their average bitrate can be high when compared to the capacity of the link (e.g. close to the end user in an access network) possibly making statistical mechanisms ineffective.

### B. Interior node measurement algorithms

The original PCN mechanism, as described in [1], uses a token bucket algorithm to measure the network load. However, other approaches can also be used. In this section, we present the original token bucket approach and also propose a bandwidth metering mechanism as measurement algorithm.

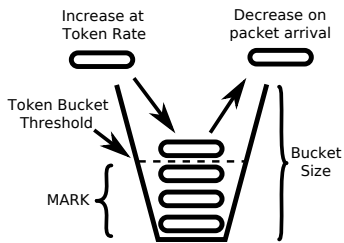


Fig. 3. Token bucket concept

1) *Token bucket*: The token bucket algorithm is typically used to limit network transmission speed but can also be used to monitor bandwidth usage. A conceptual view on a token bucket is shown in Figure 3. In essence, a token bucket is a simple bit counter with lower and upper boundaries, where the counter represents the load in the network. At a constant token rate  $R$ , tokens are added to the bucket; these tokens are removed again when a packet arrives at the interior node. As a result, the number of tokens in the bucket will increase when the bandwidth of the aggregate is lower than the given token rate  $R$ . At the same time, the number of tokens will decrease when the bandwidth of the aggregate is higher than the token rate  $R$ . As such, a token bucket provides information on the network load. In the PCN mechanism, packets are marked when the number of tokens in the bucket is lower than a predefined token bucket threshold. It is assumed that when the number of tokens drops below the token bucket threshold, the bandwidth of the aggregate is too high and new sessions need to be blocked.

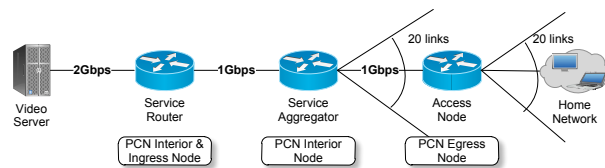


Fig. 4. Simulated network topology for all tests: the topology consists of 1 congestion point at the Service Router.

2) *Bandwidth metering*: An alternative to using a token bucket is performing a bandwidth measurement using a time window. When using this approach, an arriving packet is marked when the bandwidth measurement at that time is higher than a predefined bandwidth threshold. In our approach, we use a sliding window where the measurement interval (denoted by  $mi$ ) is fixed, and each time a packet arrives the bandwidth is measured based on the packets received during the last  $mi$  seconds. In this approach, measurement windows usually overlap. Packets sent during bandwidth peaks, introduce more measurements than packets during lower bandwidth.

The bandwidth metering requires more memory than the token bucket algorithm but has several advantages compared to the token bucket approach. First, it provides more information than just the binary value produced by the token bucket algorithm. Where a token bucket only indicates if the bandwidth aggregate is higher or lower than the token rate, the bandwidth metering algorithm provides an accurate value of the measured bandwidth. Although this exact value is not necessary for the PCN architecture it provides more information, and is hence an enabler for more intelligent actions. Second, as the sliding window based mechanism works on a time basis it is less sensitive to the bursty nature of the traffic. These advantages make the bandwidth metering algorithm a slightly more intuitive approach with the cost of an additional complexity.

## IV. PCN EVALUATION

### A. Test-setting

We studied the performance of the bandwidth metering and the original token bucket algorithm on the protection of bursty VBR traffic by investigating the impact of different configurations on the admittance process. All results were obtained using the NS-2 [10] simulator. Figure 4 illustrates the simulated network topology. This topology consists of a typical tree based access network where a video server offers a video service to the end users. Here, a 2 Gbps link between video server and Service Router is connected with a 1 Gbps link between the Service Router and Service Aggregator resulting in a congestion point at the Service Router. As no other congestion points are introduced further down the network, no more than 1 Gbps of traffic can be transmitted in the network without it being congested.

We applied PCNs architecture to our simulated network topology by deploying the measurement algorithm on the Service Router which also acts as ingress node. We emulated the transmission of a UDP based video service where different

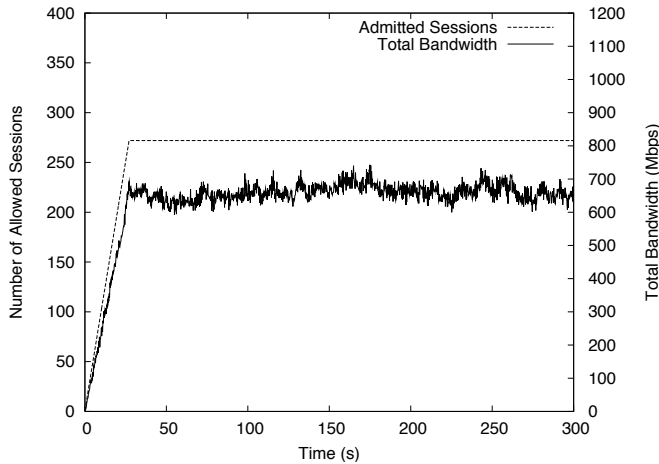


Fig. 5. Admittance process and aggregate bandwidth for the token bucket approach. The token bucket algorithm is able to timely block sessions and avoid congestion in the network. Sessions are allowed until the bandwidth aggregate is above the token rate.

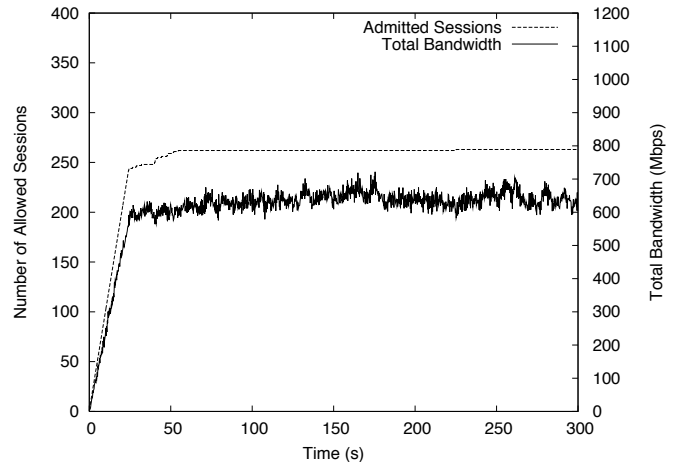


Fig. 6. Admittance process and aggregate bandwidth for the bandwidth metering approach. The bandwidth metering algorithm is able to timely block sessions and avoid congestion in the network. Sessions are allowed until the bandwidth aggregate is above the bandwidth threshold.

videos are transmitted to the 400 clients. We used different videos, representing different content classes (e.g. news scenario, action movie, sports, documentary). Each video was encoded as a constant quality H.264 video, having a frame rate of 25 fps and a resolution of 720x576 pixels. As a constant quality encoding technique was used, the encoded video is a VBR video, which exhibits bursty traffic. The bitrate of the videos averages between 2 Mbps and 3 Mbps.

During simulation, the clients request a random portion of a video. We used a random uniform process as request arrival process where we could alter the request rate. Based on observations of the measurement algorithm, the Service Router should decide whether or not to allow the new request. Once, a request has been admitted we assume that the videos do not stop. Hence, the admission process should measure an increase in the network load and block requests once it reaches a pre-congestion state.

We investigated several configurations of both measurement algorithms and varied the network conditions. We varied the request rate from 1 request per second to 10 requests per second. For each test, we performed 30 iterations where the request arrival process causes random scenarios. The CLE weight was varied from 0.9 to 0.99999. For the token bucket metering approach, we varied the token bucket depth from 80,000 bits to 25,000,000 bits. The token bucket threshold was set to 50% of its depth and the token bucket rate was set to 600 Mbps and 800 Mbps. For the bandwidth metering approach, we varied the measurement window from 20 to 300 milliseconds. Similar as in the token bucket approach, the bandwidth threshold was set to 600 Mbps and 800 Mbps.

### B. Description of results

A broad range of different configurations have been simulated, here we focus on the configurations resulting in the best results. Figure 5 illustrates the admittance process and the bandwidth aggregate over time for a token bucket depth

TABLE I  
STANDARD DEVIATION VALUES OF THE NUMBER OF ADMITTED SESSIONS AND MEASURED BANDWIDTH FOR BOTH THE TOKEN BUCKET AND BANDWIDTH METERING APPROACH.

	Token bucket	Bandwidth metering
Admitted sessions	0.31	0.29
Measured bandwidth	23.94	22.49

of 16 Mbit, a token bucket threshold of 8 Mbit and a token rate of 600 Mbps. The CLE weight was set to 0.9 leading to an almost instantaneous CLE measurement. The request rate was set to 10 requests per second. Here, we see that it is possible to configure the token bucket in a way that a stable and timely admittance decision is achieved. The aggregate bandwidth averages around the predefined token rate of 600 Mbps and no more sessions are accepted. Additionally, the admittance decision itself is also quite stable: once the first session has been blocked, only a few additional sessions are admitted until no more sessions are allowed.

Also the bandwidth metering approach can result in a stable admission control mechanism. This is illustrated in Figure 6, where a measurement window of 140 milliseconds is used. Similar to the token bucket approach, the bandwidth threshold was set to 600 Mbps, the CLE weight was set to 0.9 and the request rate was set to 10 requests per second. Figure 6 shows similar behaviour as Figure 5. The bandwidth metering is able to timely detect the pre-congestion state leading to the blocking of all sessions as soon as all bandwidth metering measurements are above the 600 Mbps threshold.

Figure 5 and 6 both represent a view of one iteration. As discussed in Section IV-A, we performed 30 iterations of each test. Table I illustrates the standard deviation values of the number of admitted sessions and measured bandwidth for both approaches. These values indicate that the admittance process is quite stable for all iterations and similar for both approaches.

### C. Discussion of results

The previous tests showed that both the token bucket and bandwidth metering approach can be used as PCNs measurement algorithm and that they have an equal performance in their admittance process. Furthermore, the tests indicated that both approaches can be configured to protect the transmission of bursty traffic, a kind of traffic which is out of scope in the original PCN work.

While we can configure the token bucket or the bandwidth metering algorithm, the bursty nature does have an impact on the behaviour of the algorithms. In both cases, the configured rate (i.e. token rate or bandwidth threshold) should be seen as a parameter in configuring the measurement algorithm. As bursty traffic is transmitted over the network, its aggregate has a certain variability and the measurement algorithms stop allowing sessions once the minimum of the bandwidth is above this threshold. This is illustrated in Figure 5 and 6 where the bandwidth aggregate is above the 600 Mbps threshold. Hence, it is not possible to configure the rate at the link capacity, which is in our network topology 1 Gbps, as this would lead to over-admittance.

The configured rate should always be lower to cope with the variability of the aggregate. This leads to an under utilisation of the network as the variability of the aggregate is not known beforehand. Therefore, it is important to find a good value for the rate as a too high value will lead to over-admission and consequently packet loss, while a too low value will lead to under-admission and consequently to more sessions being blocked. In this test, the rate should have been set to 870 Mbps which corresponds to  $1Gbps - 130Mbps$ , where 130 Mbps is the variability of the session aggregate. If it is set higher, there will be packet loss, if it is lower, sessions will get blocked when there is still enough bandwidth available. However, since the variability of the aggregate is not known the threshold can not be configured optimally.

## V. AUTONOMIC RATE ADAPTATION

### A. Algorithmic description

As discussed in the previous section, the static rate configuration leads to an under-utilisation of the network. From an operators point of view, it is crucial to both avoid congestion and maximise the link utilisation as this will lead to a higher overall quality and more revenue. However, as the session aggregate depends on the type of traffic and the request process, the bandwidth threshold can not be set statically. Therefore, we propose an autonomic algorithm where the rate is varied based on measurements in the past. The goal of this algorithm is to maximise the link utilisation while avoiding over-admission. The algorithm is autonomic in a way that it can implement the policy of link usage maximisation by automatically adapting its parameters. This algorithm should therefore be seen as an adaptive component for network optimisation and an enabler for future, more intelligent, autonomic behaviour. As illustrated in Figure 7, the algorithm measures the variability of the aggregate during a certain interval  $T_{mon}$ .

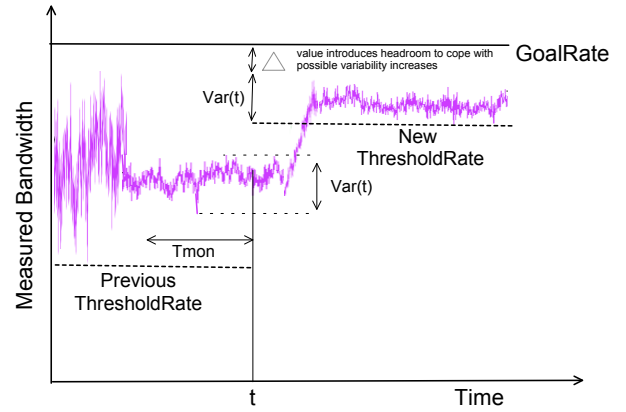


Fig. 7. Illustration of the dynamic algorithm. During a period  $T_{mon}$  the variability of the aggregate is measured and the threshold rate is set accordingly. A  $\Delta$  value is used to cope with possible increases in the variability on a larger time basis

This variability is defined as:

$$Var(t) = MaxBW_{[t-T_{mon},t]} - MinBW_{[t-T_{mon},t]}$$

Here,  $MaxBW_{[a,b]}$  and  $MinBW_{[a,b]}$  respectively denote the maximum and minimum bandwidth measurement during a given time interval  $[a,b]$ . From previous measurements we observed that the addition of one session does not lead to considerable changes in the variability. Hence, we can assume that the previous variability measurement is a good estimate for the variability in the near future. Therefore, the rate is set to:

$$ThresholdRate = GoalRate - Var(t) \times (1 + \Delta)$$

where  $GoalRate$  is the bandwidth we are targeting (e.g. the link capacity). The  $Var(t)$  measurement is only a good estimate for the near future variability: on a larger time basis, the variability can vary. However, due to the statistical multiplexing gain, we can assume that, for a fixed amount of sessions, this fluctuation will be limited. The  $\Delta$  value indicates how much we expect the variability to fluctuate in the future: a larger value will lead to a lower link utilisation, but will provide a better protection against an increase in variability. Vice versa, a lower value will provide a better link utilisation but can lead to an over admission if the variability of the aggregate increases. In practice, the algorithm will choose the threshold rate that still leaves room for the variability of the aggregate together with some headroom to cope with future fluctuations.

### B. Performance evaluation

In this section, we discuss the performance of the dynamic algorithm. Conceptually, the dynamic algorithm can be applied to both the bandwidth metering and the token bucket approach. Here, we focus on the bandwidth metering approach, as its behaviour is slightly more intuitive, and compare the behaviour of the dynamic algorithm with its static bandwidth metering variant. We employed the same test-setting as described in Section IV-A.

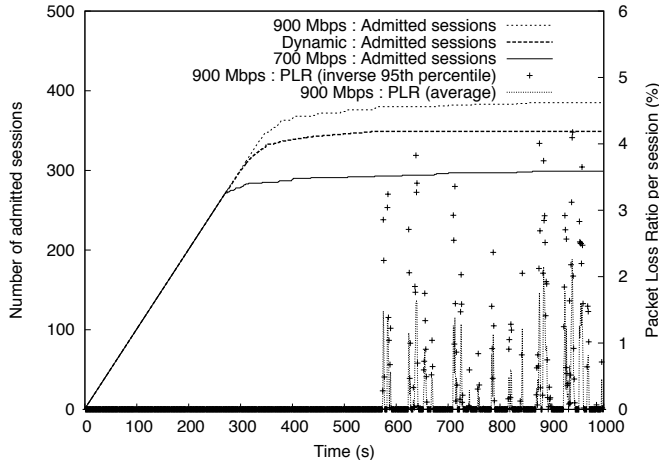


Fig. 8. Impact of the dynamic and static algorithm on the admittance process and packet loss ratio. The dynamic algorithm is able to avoid packet loss and still provide a stable admittance decision.

Figure 8 illustrates the admittance process and measured packet loss ratio for both implementations. In this test, we compared the dynamic algorithm with two configurations of the static algorithm: a bandwidth threshold of 700 Mbps and 900 Mbps. For the dynamic algorithm the  $T_{mon}$  value was set to 5 seconds, while the  $\Delta$  value was set to 0.5. The choice of these values is motivated later on in the evaluation section. We simulated 1000 seconds of network operation and set the request rate to 1 request per second. Figure 8, shows the number of admitted sessions together with the average and inverse 95th percentile packet loss ratio per session. The inverse 95th percentile is the average of the 5% highest packet loss values of all allowed sessions and provides an indication of the worst case scenario. Over time, all three configurations result in a stable and timely admittance decision. For the dynamic algorithm, the number of allowed sessions is between the number of allowed sessions for the 700 Mbps and 900 Mbps configuration. This indicates that the dynamic algorithm autonomously sets the bandwidth threshold between these two values. When looking at the packet loss ratio, both the 700 Mbps static algorithm and the dynamic algorithm were able to timely block the sessions and hence avoid congestion. The 900 Mbps static algorithm also stopped allowing sessions at a given point in time but the amount of sessions allowed was too much, resulting in packet loss. While the average bandwidth used in the last 200 seconds of the test was only 955 Mbit/s, the bursty nature of the video resulted in peaks of bandwidth usage. The static algorithm cannot cope with this variability. The average packet loss ratio increases up to 2.2 % and the inverse 95th percentile up to 4.2%. It should be noted that as we are transmitting bursty traffic, also the measured packet loss ratio has a high variation.

Similar as the tests described in Section IV-B, Figure 8 provides a view of one of 30 iterations of this test. For the number of allowed sessions, the standard deviation of all three implementations is always lower than 0.25, which is comparable to the standard deviation values obtained in

TABLE II  
NUMBER OF ADMITTED STREAMS AND PACKET LOSS RATIO (PLR) PER SESSION FOR A 5 MINUTE TIME WINDOW WHERE NO REQUESTS ARRIVE.

	Static 700 Mbps	Dynamic algorithm	Static 900 Mbps
Admitted flows	297	355	383
PLR (Average)	0.0%	0.0%	0.8%
PLR (Inverse 95th percentile)	0.0%	0.0%	2.9%
Worst case	0.0%	0.0%	3.2%

Section IV-B. For the packet loss measurements the obtained standard deviation values, in the case of the static 900 Mbps implementation, were 0.8% and 2.1% for the average and the inverse 95th percentile, respectively. For the other implementations, the standard deviation values were 0%.

The previous figure indicates that the bandwidth threshold set by the dynamic algorithm is a good value as the dynamic algorithm was able to maximise the link utilisation while still avoiding congestion. In this test, the variability of the aggregate was 130 Mbps, meaning that a bandwidth threshold of 870 Mbps is the maximum value that can be used without causing congestion. The dynamic algorithm used a bandwidth threshold of 820 Mbps, which is a bit lower as a  $\Delta$  value of 0.5 was used to cope with a possible increase in variability.

Table II illustrates the stability of both algorithms. Similar as in the previous test, we simulated 1000 seconds but stopped the request arrival process at 700 seconds. Table II shows the measured packet loss ratio per session during the 5 minute period between 700 and 1000 seconds of simulation time. Here, we see similar results as in the previous test. The dynamic and 700 Mbps algorithm are able to avoid congestion, while using the 900 Mbps algorithm results in congestion. As the 900 Mbps algorithm is still able to block sessions, the average packet loss ratio per session is limited to 0.8%. However, the inverse 95th percentile also shows that, in the 5% worst cases, the packet loss ratio had an average of 2.9%. The worst case scenario results in a packet loss ratio of 3.2%. It should be noted that video based streaming services are extremely sensitive for packet loss and that packet loss ratios of 1 % already have a great impact on the video quality. Furthermore, as the packet loss ratio is bursty over time (as indicated in Figure 8), over a small time window the packet loss ratio can be much larger, which results in a complete deterioration of the video quality over that time window.

In Figures 9 and 10, the impact of the aggregate variability on the number of admitted sessions and packet loss ratio is illustrated for the dynamic algorithm and three different configurations of the static algorithm having a bandwidth threshold of 700, 800 and 900 Mbps. In this test, we transmitted different VBR videos to vary the overall variability of the aggregate. Similar as in the previous tests, the  $T_{mon}$  value was set to 5 seconds and the  $\Delta$  value was set to 0.5. Each scenario was repeated 30 times, the standard deviation values were always lower than 6, for the number of admitted sessions, and 1%, for the measured packet loss ratio. Here, we see how an increasing aggregate variability causes all



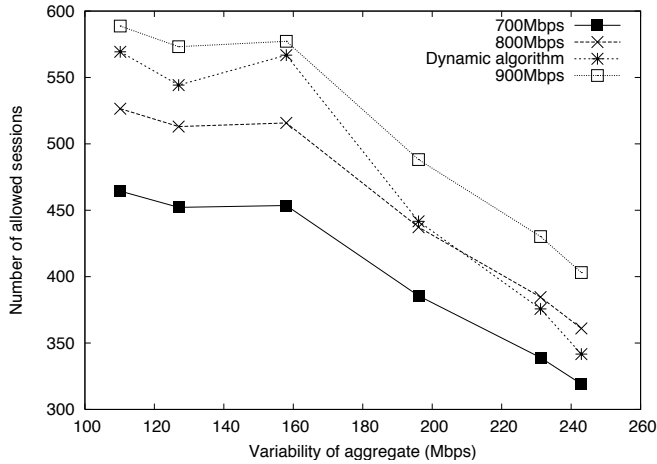


Fig. 9. Number of admitted sessions for an increasing variability. For all configurations, the number of admitted sessions decreases as the variability increases. At the 200 Mbps mark the dynamic algorithm admits less sessions than the 800 Mbps algorithm. As shown in Figure 10, this is needed as packet loss occurs, for the static 800 Mbps algorithm, starting at that value.

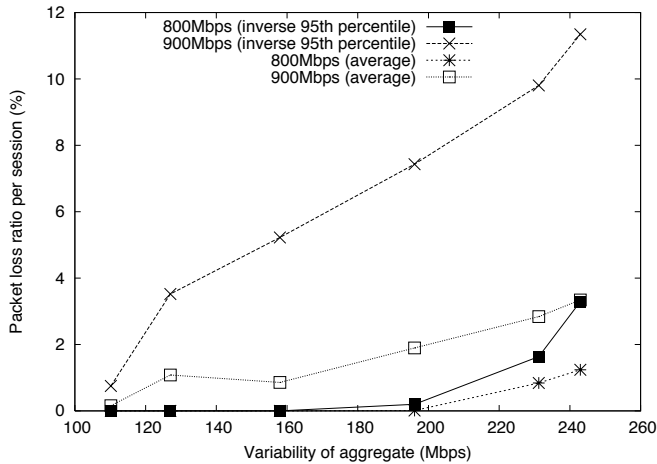


Fig. 10. Observed packet loss ratio of the 800 Mbps and 900 Mbps static algorithms for an increasing variability. In the other implementations, no packet loss was observed.

admission control systems to lower the amount of allowed sessions. This is an obvious behaviour as, for a high variability, there will be higher bandwidth peaks causing packets to be marked, and consequently, sessions to be blocked. All static implementations result in more or less the same behaviour with higher bandwidth thresholds allowing more sessions. This is not the case for the dynamic algorithm: a higher variability also causes the number of allowed sessions to go down but the decrease is much larger when compared to the static algorithms. This is because a higher variability will lead to a lower bandwidth threshold, leading to a stricter admission control mechanism. At an aggregate variability of 200 Mbps, the dynamic algorithm drops below the 800 Mbps algorithm and further decreases closer to the line of the 700 Mbps algorithm.

This drop below the 800 Mbps mark at an aggregate variability of 200 Mbps is necessary to avoid congestion. This is illustrated in Figure 10, where the observed packet loss

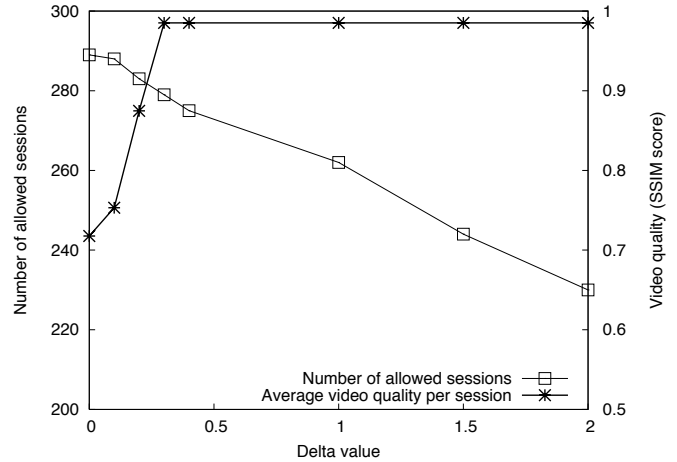


Fig. 11. Impact of the  $\Delta$  value on the performance of the dynamic algorithm in terms of video quality. A too low  $\Delta$  value leads over admission and a drop in video quality.

ratio is shown for an increasing variability. For the dynamic algorithm and the static algorithm with a bandwidth threshold of 700 Mbps no packet loss was observed. However, for the static algorithm at 800 Mbps, the packet loss ratio starts to appear and further increases above the 200 Mbps mark. For the static algorithm at 900 Mbps, packet loss is already observed at a variability of 100 Mbps and increases up to an average of 3.3%. The worst case scenario, denoted through the inverse 95th percentile, is even a lot higher with a packet loss ratio of 11.3%. These values show that, for an increasing variability, the dynamic algorithm effectively decreases its bandwidth threshold in order to avoid congestion. Hence, for every variability value, the dynamic algorithm is the algorithm leading to the highest number of sessions allowed without causing any congestion.

As discussed in Section V-A, a higher  $\Delta$  value will lower the bandwidth threshold and leave more room for occasional increases in the variability of the aggregate. Figure 11 illustrates the influence of the  $\Delta$  value in the dynamic algorithm on the number of admitted sessions and the video quality. Each test was repeated 30 times resulting in standard deviation values of 0.2 and lower for the number of admitted sessions and 0.09 and lower for the video quality scores. While the  $\Delta$  value was varied, the  $T_{mon}$  value was set to 5 seconds. In this case, the video quality will be affected by the occurrence of packet loss. We measure the video quality through the Structural Similarity (SSIM) [11]. The SSIM is an objective Full-Reference quality metric based upon the assumption that the Human Visual System is more specialised in the extraction of structural information from scenes. The SSIM model takes the original and the distorted signal as input and produces a score between 0 and 1, where 1 stands for perfect quality. Although SSIM is originally an image quality metric, it is increasingly used for video quality evaluation.

As can be seen in Figure 11, a trade-off can be found in determining an optimal  $\Delta$  value. If the  $\Delta$  value is chosen too small, the configured bandwidth threshold will be too high

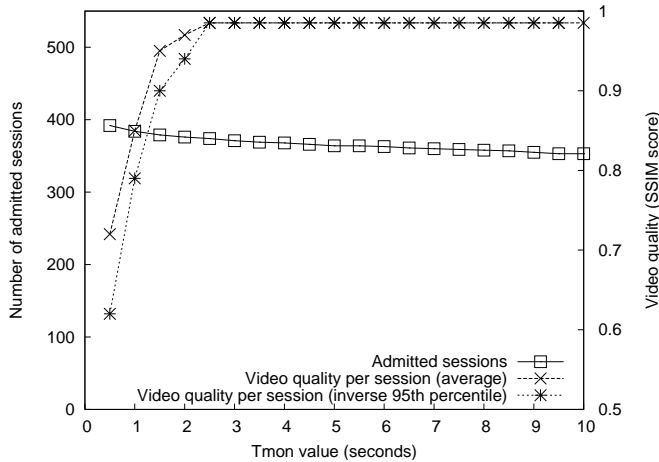


Fig. 12. Impact of the  $T_{mon}$  value on the performance of the dynamic algorithm. The graph allows to determine the minimum value of  $T_{mon}$ .

and an occasional packet drop can occur. This has a negative effect on the video quality, as can be seen through the average SSIM value per session. For a  $\Delta$  value lower or equal than 0.1 the SSIM score is lower than 0.75, which corresponds to a poor video quality. Increasing the  $\Delta$  value leads to a restoration of the video quality but if the  $\Delta$  value is too high, less sessions will be allowed resulting in a decrease in the total video quality, which can be calculated by taking the sum of the SSIM scores of all allowed sessions.

Figure 12 illustrates the impact of the  $T_{mon}$  value on the performance of the dynamic algorithm. Here, the SSIM value per session and the number of admitted sessions is shown for a varying  $T_{mon}$  value. Again, the test was repeated 30 times resulting in low standard deviation values. For the number of admitted sessions, the standard deviation is always lower than 0.25. For the SSIM values, standard deviation values are approximately 0.04 for both the average and percentile values. As can be seen, a  $T_{mon}$  value lower than 4 seconds has a negative impact on the performance of the dynamic algorithm. In this case, the  $T_{mon}$  measurement window is too small to obtain a representative view on the variability of the aggregate. This leads to an under estimation of the variability and thus to an over admission of sessions resulting in packet loss and ultimately video degradation. A  $T_{mon}$  value of 4 seconds and higher results in a good performing admission control mechanism. Here, the video quality can be seen as perfect with a SSIM value of 0.9852. By increasing the  $T_{mon}$  value, the SSIM value per sessions remains the same but the number of allowed sessions decreases. This is because a large peak is longer taken into account in the variability calculation. However, this only affects a small number of sessions: for a  $T_{mon}$  value of 4 seconds, 366 sessions are admitted, while for a  $T_{mon}$  value of 10 seconds, 353 sessions are admitted.

## VI. CONCLUSIONS & FUTURE WORK

In this paper, we evaluated the performance of a PCN based admission control system in protecting the transmission of video services, which have a bursty traffic characterisation.

We investigated two different measurement algorithms, a token bucket based approach as originally presented by the IETF and a novel bandwidth metering approach. We showed that both algorithms can be configured to protect the transmission of video services, a kind of traffic which is not considered in the original PCN work.

While a working admission control mechanism can be realised, the bursty traffic has impacts on the configuration resulting in an under utilisation of the network. Therefore, we propose a dynamic algorithm which autonomically adjusts the rate, one of the parameters of the measurement algorithm, based on the measured variability of the aggregate. Extensive evaluation results showed that this algorithm provides a better link utilisation than static implementations, while guaranteeing no congestion in the network.

In future work, we are investigating the influence of a change in session mix. For example, by introducing several HD videos in a session mix where only SD videos are present. Furthermore, we are investigating the impact of sessions being stopped once admitted (e.g. because the video is finished or the user stops the video).

## ACKNOWLEDGMENT

The research was performed partially within the framework of the EUREKA CELTIC RUBENS project; The authors would like to thank all RUBENS partners for their valuable contributions and feedback. Steven Latré is funded by Ph.D grant of the Fund for Scientific Research, Flanders (FWO-V).

## REFERENCES

- [1] P. Eardley, "Pre-Congestion Notification (PCN) Architecture (draft-ietf-pcn-architecture-09)," [online] <http://tools.ietf.org/html/draft-ietf-pcn-architecture-09>.
- [2] ETSI TS 182 019, "Resource and Admission Control Sub-system (RACS); Function Architecture."
- [3] K. Nichols, V. Jacobson, and L. Zhang, "A Two-bit Differentiated Services Architecture for the Internet," RFC 2638 (Informational), Jul. 1999. [Online]. Available: <http://www.ietf.org/rfc/rfc2638.txt>
- [4] J. Rezgui, A. Hafid, and M. Gendreau, "A distributed admission control scheme for wireless mesh networks," in *5th International Conference on Broadband Communications, Networks and Systems, 2008. BROADNETS 2008.*, 2008.
- [5] Z. Xia, W. Hao, I.-L. Yen, and P. Li, "A distributed admission control model for qos assurance in large-scale media delivery systems," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 16, no. 12, pp. 1143–1153, Dec. 2005.
- [6] J. Aracil, J. Hernandez, A. Elizondo, R. Duque, and O. de Dios, "On local cac schemes for scalability of high-speed networks," *Transparent Optical Networks, 2008. ICTON 2008. 10th Anniversary International Conference on*, vol. 3, pp. 175–177, June 2008.
- [7] "Congestion and Pre-Congestion Notification Working Group," [online] <http://tools.ietf.org/wg/pcn>.
- [8] M. Menth and F. Lehrieder, "Performance evaluation of pcn-based admission control," in *Quality of Service, 2008. IWQoS 2008. 16th International Workshop on*, 2008, pp. 110–120. [Online]. Available: <http://dx.doi.org/10.1109/IWQOS.2008.19>
- [9] M. Menth and H. M., "Threshold configuration and routing optimization for pcn-based resilient admission control," *Journal of Computer Networks*, 2009.
- [10] "Ns-2, The Network Simulator," [online] <http://www.isi.edu/nsnam/ns/>.
- [11] Z. Wang, L. Lu, and A. C. Bovik, "Video quality assessment based on structural distortion measurement," *Signal Processing: Image Communication*, vol. 19, no. 2, pp. 121–132, February 2004.