Experience-Based Admission Control in the Presence of Traffic Changes

Jens Milbrandt, Michael Menth, and Jan Junker
University of Würzburg, Institute of Computer Science, Germany
Email: {milbrandt, menth, junker}@informatik.uni-wuerzburg.de

***Abstract*— This article investigates the transient behavior of *experience-based admission control* (EBAC) in case of traffic changes. EBAC is a robust and resource-efficient admission control (AC) mechanism used for reservation overbooking of link capacities in packet-based networks. Recent analyses gave a proof of concept for EBAC and showed its efficiency and robustness through steady state simulation on a single link carrying traffic with constant properties. The contribution of this paper is an examination of the memory from which EBAC gains its experience and which strongly influences the behavior of EBAC in stationary and non-stationary state. For the latter, we investigate the transient behavior of the EBAC mechanism through simulation of strong traffic changes which are characterized by either a sudden decrease or increase of the traffic intensity. Our results show that the transient behavior of EBAC partly depends on its tunable memory and that it copes well with even strongly changing traffic characteristics.**

***Index Terms*— admission control, reservation overbooking, quality of service, resource & traffic management**

I. Introduction

Internet service providers operating next generation networks (NGNs) are supposed to offer quality of service (QoS) to their customers. As packet-based Internet protocol (IP) technology becomes more and more the basis of these networks, QoS in terms of limited packet loss, packet delay, and jitter is required to support real-time services. There are two fundamentally different methods to implement QoS: capacity overprovisioning (CO) and admission control (AC). With CO the network has so much capacity that congestion becomes very unlikely [1], [2], but this also implies that its utilization is very low even in the busy hour. Although CO is basically simple, it requires traffic forecasts and capacity provisioning must be done on a medium or long time scale.

In contrast, AC works on a smaller time scale. It grants access to flows with QoS requirements if the network load is sufficiently low and rejects excessive flow requests to shelter the network from overload before critical situations occur. QoS is thus realized by flow blocking during overload situations. Compared to CO, AC requires less capacity and yields better resource utilization at the expense of more signaling, coordination and state management [3]–[5] especially in the context of a network-wide AC [6].

A. Conventional Link Admission Control

In this work, we focus on link-based AC, i.e., on AC methods that protect a single link against overload. These methods are usually extended for application in networks on, e.g. a link-by-link basis. AC approaches can coarsely be classified into parameter-based AC (PBAC), measurement-based AC (MBAC), and derivatives thereof.

1) Parameter-Based Admission Control: PBAC, also known as (a priori) traffic-descriptor-based AC, is an approach appropriate for guaranteed network services [7], i.e., for traffic with imperative QoS requirements. It relies solely on traffic descriptors that are signaled by a source/application and describe the traffic characteristics of a flow, e.g. its mean and peak rate together with token bucket parameters. If an admission request succeeds, bandwidth is reserved and exclusively dedicated to the new flow. PBAC is often inefficient regarding its resource utilization since the traffic descriptors usually overestimate the actual rate to avoid packet delay and loss due to spacing or policing. With PBAC, traffic is limited either by deterministic worst case considerations like network calculus [8], [9] or by stochastic approaches such as effective bandwidth [10]. In addition, PBAC calculations for heterogeneous traffic mixes can be very complex.

2) Measurement-Based Admission Control: MBAC, in contrast, is an AC method adequate for controlled load network services [11], i.e. for traffic with less stringent QoS requirements. It measures the current link or network load in real-time and takes a rate estimate of the new flow to make the admission decision. The determination of the traffic characteristics is thus shifted from a source/application to the network and the specified traffic descriptor, e.g. the peak rate, can be very simple. MBAC can be classified into different approaches.

- *Aggregate-oriented MBAC (A-MBAC)* Most MBAC approaches measure the traffic properties of the entire traffic aggregate admitted to the link. The effective bandwidths of a flow is only required for the initial admission decision, when the requested bandwidth is compared to the available link capacity.

This paper is based on "Performance of Experience-Based Admission Control in the Presence of Traffic Changes," by J. Milbrandt, M. Menth, and J. Junker, which appeared in the Proceedings of the 5^{th} IFIP-TC6 Networking Conference (NETWORKING), Coimbra, Portugal, May 2006. © 2006 Springer.
This work was funded by Siemens AG, Munich, and the German Research Foundation (DFG). The authors alone are responsible for the content of the paper.

For that purpose, the rate of the admitted traffic aggregate is sufficient. A-MBAC has two advantages. The traffic measurement is simple since no per flow measurement states have to be managed and the statistical properties of a stationary traffic aggregate are more stable. On the other hand, the admission of new flows and the termination of others make the traffic aggregate a non-stationary process which must be carefully observed [12]. Comparisons of different A-MBAC approaches can be found in [13]–[19].

- *Flow-oriented MBAC (F-MBAC)* Some MBAC approaches use flow-specific measurements to assess the bandwidth consumption of each traffic flow individually. The initial effective bandwidth of a new flow is calculated based on its declared traffic descriptor. As soon as the confidence in the measurements of an admitted flow is high enough, its initial bandwidth is substituted by a calculated update based on the measured traffic parameters. Examples of F-MBAC methods are given in [20]–[23].

All presented MBAC methods use real-time measurements and admit traffic as long as enough capacity is available. The downside of MBAC is its sensitivity to measurement accuracy and its susceptibility to traffic prediction errors which can occur, e.g., during QoS attacks, i.e., when admitted traffic flows are temporarily "silent" to provoke an underestimation of the admitted traffic rate and congest the link later by simultaneously sending at high bitrate.

B. Experience-Based Admission Control

To the best of our knowledge, experience-based AC (EBAC) is the first hybrid AC approach that takes advantage of traffic measurements without real-time requirements. It uses information about previously admitted traffic and past measurements to make current admission decisions. The concept of EBAC is introduced in [24]. Its details are described in [25] and can be summarized as follows: with EBAC, a new flow is admitted to a link at time t if its peak rate together with the peak rates of already admitted flows does not exceed the link capacity multiplied by an overbooking factor $\varphi(t)$. The overbooking factor is calculated based on the reservation utilization of the admitted flows in the past. Hence, this method relies on experience. EBAC also requires traffic measurements to compute the reservation utilization, but these measurements do not have real-time requirements and thus influence the admission decision only indirectly. The proof of concept for EBAC is given in [25] by simulations and corresponding waiting time analyses of the admitted traffic. In particular, the steady state performance of EBAC is investigated for traffic with static characteristics. Since MBAC methods are sensitive to traffic variability, we investigate the behavior of EBAC in the presence of traffic changes in [26], [27].

The remainder of this article is organized as follows. In Section II, we review the EBAC concept. Section III describes our simulation design and the applied traffic model. In Section IV, we investigate the behavior of EBAC in case of sudden traffic changes. Further work on EBAC is briefly presented in Section V. Section VI summarizes this work and gives a conclusion.

II. Fundamentals of Experience-Based Admission Control (EBAC)

Experience-based AC (EBAC) is a hybrid approach combining functional elements of PBAC and MBAC in a novel AC concept. It therefore implements link admission control but can be easily extended to a network-wide scope. EBAC relies on peak rate traffic descriptors which may be significantly overestimated in the signaled flow requests. The utilization of the overall reserved capacity gives an estimate for the peak-to-mean rate ratio (PMRR) of the traffic aggregate and allows for the calculation of a factor to overbook the link capacity. The idea is simple but safety margins are required to provide sufficient QoS and questions arise regarding its robustness against variable traffic flows. In this section, we elaborate the EBAC concept and describe its basic functional components.

A. Admission Decision on a Single Link

EBAC makes an admission decision as follows. An AC entity limits the access to a link l with capacity c_l and records the admitted flows $\mathcal{F}(t)$ at any time t together with their requested peak rates $\{r_f : f \in \mathcal{F}(t)\}$. When a new flow f_{new} arrives, it requests for a peak rate $r_{f_{new}}$. If

$$r_{f_{new}} + \sum_{f \in \mathcal{F}(t)} r_f \leq c_l \cdot \varphi(t) \cdot \rho_{max} \quad (1)$$

holds, admission is granted and f_{new} joins $\mathcal{F}(t)$. Otherwise, the new flow request is rejected. Flows are removed from $\mathcal{F}(t)$ on termination. The experience-based overbooking factor $\varphi(t)$ is calculated by statistical analysis and indicates how much more bandwidth than c_l can be safely allocated for reservations. The maximum link utilization threshold ρ_{max} limits the traffic admission such that the expected packet delay W exceeds an upper threshold W_{max} only with probability p_W.

B. Calculation of the Maximum Link Utilization Threshold

The value of ρ_{max} depends significantly on the traffic characteristics and the capacity c_l of the EBAC-controlled link. Most prominent solutions are based on the $M/M/1-\infty$ and the $N \cdot D/D/1-\infty$ queuing system. Real-time traffic produced from, e.g., voice or video applications has a rather constant output rate that can be controlled by a spacer such that a maximum flow rate is enforced. Therefore, we calculate the threshold ρ_{max} based on the $N \cdot D/D/1-\infty$ approach, which assumes N homogeneous traffic flows in $\mathcal{F}$, each sending packets of constant size B (in bits) and with constant packet inter-arrival times A (in seconds). An analysis method for this queueing system is presented Section 15.2.4 of [10]. More details on the computation of ρ_{max} can be found in [25]. For the ease of simulation of changing traffic, we set the maximum link utilization to a conservative and constant value of $\rho_{max}=0.95$.

C. Calculation of the Overbooking Factor

The overbooking factor $\varphi(t)$ depends on the admitted traffic $\mathcal{F}(t)$ which, in turn, depends on time t because new flows are admitted and existing ones terminate. For the computation of $\varphi(t)$, we define $R(t)=\sum_{f\in\mathcal{F}(t)} r_f$ as the reserved bandwidth of all admitted flows at time t and $C(t)$ denotes their unknown cumulated mean rate. EBAC measures the consumed link bandwidth $M(t)$ of the overall reservation $R(t)$. To obtain $M(t)$, we use measurements based on equidistant, disjoint intervals such that for an interval $I(t_i)=[t_i, t_i+\Delta]$ with length Δ, the measured rate $M(t_i)=\frac{\Gamma(t_i)}{\Delta}$ is determined by metering the traffic volume $\Gamma(t_i)$ sent during $I(t_i)$. For the rates $R(t)$ and $M(t)$, a time statistic for the reservation utilization $U(t)=\frac{M(t)}{R(t)}$ is collected. The values $U(t)$ are sampled in constant time intervals and are stored as hits in bins for a time-dependent histogram $P(t,U)$. From this histogram, the time-dependent p_u-percentile $U_p(t)$ of the empirical distribution of $U(t)$ can be derived by

$$U_p(t)=\min_u\{u : P(t, U\leq u)\geq p_u\}. \tag{2}$$

Since traffic characteristics change over time, the reservation utilization statistic must forget obsolete data to reflect the properties of the new traffic conditions. Therefore, we record new samples of $U(t)$ by incrementing the corresponding histogram bin by one and devaluate the contents of all histogram bins in regular devaluation intervals I_d by a constant devaluation factor f_d. The devaluation process determines the memory of EBAC. The reciprocal of the reservation utilization percentile is the overbooking factor

$$\varphi(t)=\frac{1}{U_p(t)} \tag{3}$$

which is computed each time a new value $U(t)$ is recorded in the histogram. To avoid an underestimation of $U_p(t)$ and an overestimation of $\varphi(t)$, enough statistical data must be collected before Equation (3) yields a reliable overbooking factor.

D. Peak-to-Mean Rate Ratio (PMRR)

The intrinsic idea of EBAC is the exploitation of the peak-to-mean rate ratio (PMRR) of the traffic aggregate admitted to the link. With EBAC, the signaled peak rate r_f of an admitted flow f is enforced by a traffic shaper. In contrast to reality, the mean rate c_f of a flow is known a priori in our simulations. We define the PMRR of a flow by $k_f=\frac{r_f}{c_f}$. Analogously, $K(t)=\frac{R(t)}{C(t)}$ denotes the PMRR of the entire traffic aggregate admitted to the link at time t. $K(t)$ is an upper limit for the achievable overbooking factor $\varphi(t)$.

E. Memory of EBAC

1) Implementation as Simple Histogram: The histogram $P(t,U)$, i.e. the collection and the aging of statistical reservation utilizations, implements the memory of EBAC. This memory correlates successive flow admission decisions and consequently influences the adaptation of the overbooking factor $\varphi(t)$ to changing traffic conditions on the link. The statistic aging process, characterized by the devaluation interval I_d and the devaluation factor f_d, makes this memory forget about reservation utilizations in the past. The parameter pairs (I_d, f_d) yield typical half-life periods T_H after which collected values $U(t)$ have lost half of their importance in the histogram. Therefore, we have $\frac{1}{2}=f_d^{T_H/I_d}$ and define the EBAC memory based on its half-life period

$$T_H(I_d, f_d)=I_d\cdot\frac{-ln(2)}{ln(f_d)}. \tag{4}$$

With Equation (4), different combinations of devaluation parameters (I_d, f_d) and $(I_{d'}, f_{d'})$ yield equal half-life periods if either $I_{d'} = ln(f_{d'})/ln(f_d^{(1/I_d)})$ or $f_{d'} = f_d^{(I_{d'}/I_d)}$ holds. However, these equations guarantee only that the two resulting histograms are identical at times $t = LCM(I_d, I_{d'})$, where LCM denotes the least common multiple. The reservation utilizations obtained in the intervals I_d and $I_{d'}$ are devaluated differently for the two parameter sets due to the length of the devaluation interval and the different devaluation factors. This leads to intermediate deviations between the two histograms and consequently to different overbooking factors.

2) Implementation as Time Exponentially-Weighted Moving Histogram: To express the performance of the EBAC memory by only its characteristic half life period, we introduce the method of time exponentially-weighted moving histogram (TEWMH) [28] which improves the timeliness of the overbooking factor calculation. This method follows the principle of time exponentially-weighted moving average (TEWMA) [29] used to improve the timeliness of rate measurements, and it logically extends TEWMA for application to statistical histograms.

Based on the EBAC memory defined in Equation (4), we define the aging rate $a=\frac{ln(f_x)}{I_x}, x\in\{d, d'\}$. Rate a is constant for two parameter sets (I_d, f_d) and $(I_{d'}, f_{d'})$ if they yield the same half-life period T_H. Instead of incrementing the histogram bins by one, we weight the reservation utilization hits in the time interval $[t_i, t_i+I_x]$ exponentially by the weight factor $\frac{1}{e^{at}}$ and use the result as an increment for the bins. Parameter $t\in[0, I_x-1]$ thereby denotes the time-offset of the sampled reservation utilization in seconds since the last devaluation. This way, newer values $U(t)$ experienced in the interval $[t_i, t_i+I_x]$ become more important than older values and, as a consequence, all reservation utilizations gathered in this interval are evenly devaluated. In addition, the histograms of both parameter sets are comparable at any time and always lead to identical overbooking factors depending only on the half-life period T_H.

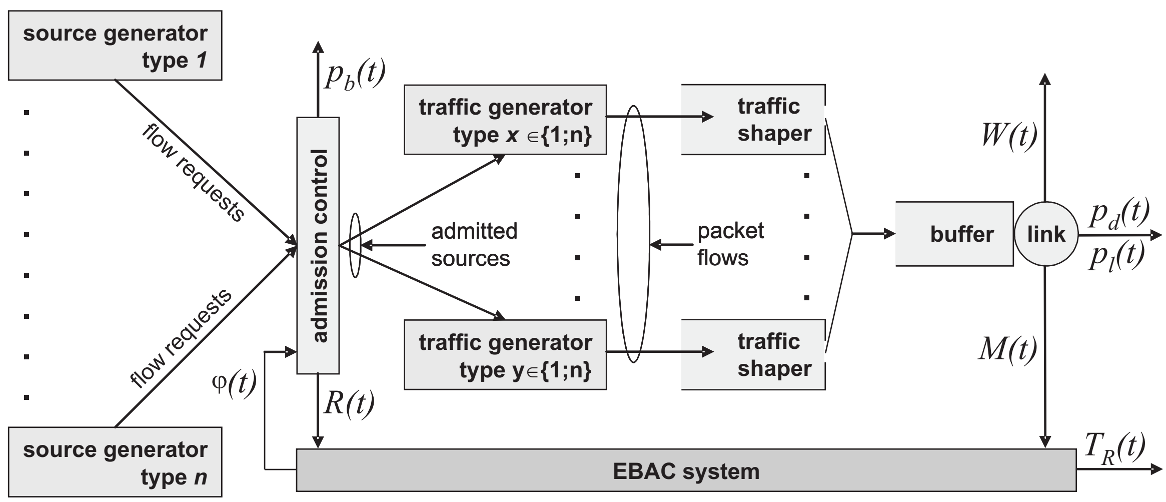


Fig. 1. Simulation design for EBAC performance evaluation.

In Section IV-B.1, the advantage of the TEWMH-based memory implementation of EBAC becomes visible in the presence of traffic changes, when we compare the overbooking performance of EBAC depending on its memory with and without TEWMH.

III. EBAC Performance Simulation

In this section, we first present the simulation design of EBAC on a single link and then describe the traffic model we used on the flow and packet scale level.

A. Simulation Design

We evaluate the performance of EBAC on a single link by discrete event simulation. The simulator is implemented in Java^{TM} and based on a simulation library called *JSimLib* which has been developed at the Chair of Distributed Systems of the University of Würzburg in the past years.

The design of the simulation is shown in Figure 1. Different types of traffic *source generators* produce flow requests that are admitted or rejected by the *admission control* entity. The flows request reservations of different bandwidths which leads to different request-dependent blocking probabilities on a heavily loaded link. To avoid this, we apply trunk reservation [30], i.e., a flow is admitted only if a flow request with maximum reservation size could also be accepted. For an admission decision, the AC entity takes the overbooking factor $\varphi(t)$ into account and admits a flow if Equation (3) holds. In turn, the AC entity provides information regarding the reservations $R(t)$ to the *EBAC system* and yields flow blocking probabilities $p_b(t)$. For each admitted source, a *traffic generator* is instantiated to produce a packet flow that is shaped to its contractually defined peak rate. Traffic flows leaving the *traffic shapers* are then multiplexed on the buffered *link* l with capacity c_l. The link provides information regarding the measured traffic $M(t)$ to the EBAC system and yields packet delay probabilities $p_d(t)$, packet loss probabilities $p_l(t)$, and the packet waiting time $W(t)$. In the presence of traffic changes, an important performance measure for the EBAC mechanism is the overall response time $T_R(t)$, i.e., the time-span required by the EBAC system to fully adapt to a new traffic situation.

B. Traffic Model

In our simulations, the traffic controlled by EBAC is modeled on two levels, i.e. the flow scale level and the packet scale level. While the flow level controls the inter-arrival times of flow requests and the holding times of admitted traffic flows, the packet level defines the inter-arrival times and the sizes of packets of individual flows.

1) Flow Level Model: On the flow level, we distinguish different traffic source types, each associated with a characteristic peak-to-mean rate ratio (PMRR) and corresponding to a source generator type in Figure 1. The inter-arrival time of flow requests and the holding time of admitted flows both follow a Poisson model [31], i.e., new flows arrive with rate λ_f and the duration of a flow is controlled by rate μ_f. The mean of the flow inter-arrival time is thus denoted by $1/\lambda_f$ and the holding time of a flow is exponentially distributed with a mean of $1/\mu_f$. Provided that no blocking occurs, the overall offered load $a_f = \frac{\lambda_f}{\mu_f}$ is the average number of simultaneously active flows measured in Erlang. To saturate an EBAC-controlled link with traffic, the load is set to $a_f \geq 1.0$. The latter assumption allows for an evaluation of the EBAC performance under heavy traffic load such that some flow requests are rejected.

2) Packet Level Model: On the packet level model, we abstract from the wide diversity of packet characteristics induced by the application of different transmission layer protocols. Since we are interested in the basic understanding of the behavior of EBAC, we abstain from real traffic patterns and define a flow of consecutive data packets simply by a packet size distribution and a packet inter-arrival time distribution. Both contribute to the rate variability within a flow that is produced by a traffic generator in Figure 1. To keep things simple, we assume a fixed packet size per flow and use a Poisson arrival process to model a packet inter-arrival time distribution with rate λ_p. We are aware of the fact that Poisson is not a suitable model to simulate Internet traffic on the packet level [32]. We therefore generate Poisson packet streams and subsequentially police the individual flows with peak-rate traffic shapers (cf. Figure 1). The properties of the flows are significantly influenced by the configuration of these shapers. In practice, the peak rate r_f of a flow f is limited by an application or a network element and the mean rate c_f is often unknown. In our simulations, however, the mean rate is known a priori and, therefore, we can control the rate of flow f by its PMRR $k_f = \frac{r_f}{c_f}$.

3) Traffic Variations: Traffic variations may be due to changes of traffic characteristics on the flow- and/or the packet-level. Variations on the flow-level signify a change of the traffic mix. Variations on the packet-level are caused by individual traffic sources that change their sending behavior. This article studies the performance of EBAC for traffic changes on the packet scale level. In the corresponding simulations, the PMRRs k_f of admitted flows vary over time which directly impacts the traffic load on the link. We investigate the transient behavior of EBAC through simulation of traffic changes which are characterized by either a decrease or increase of the traffic intensity.

IV. EBAC Performance under Traffic Changes

This section discusses the transient behavior of EBAC in the presence of traffic changes, i.e., when the traffic characteristics of the EBAC-controlled flows change. We investigate the response time T_R required by the EBAC system to provide a new appropriate overbooking factor $\varphi(t)$ after a decrease or increase of the traffic intensity. We consider sudden changes of the traffic characteristics to have worst case scenarios and to obtain upper bounds on T_R. We simulate them with only two types of traffic flows since only the properties of the entire admitted traffic aggregate are of interest for the calculation of $\varphi(t)$. The simulations allow for an examination of the memory from which EBAC gains its experience and which influences the behavior of EBAC in both stationary and non-stationary state. Our results show that the transient behavior of EBAC partly depends on its adjustable memory and that EBAC copes well with even strongly changing traffic characteristics.

A. Evaluation Issues of EBAC in Transient State

For the performance evaluation of EBAC in case of traffic changes, we use the simulation design shown in Figure 1. EBAC is simulated under heavy traffic load, i.e., we saturate the EBAC-controlled link with flow requests. To achieve traffic saturation, we set the traffic characteristics $\lambda_f = \frac{1}{750 \text{ ms}}$ and $\mu_f = \frac{1}{90 \text{ s}}$ on the flow level of the traffic model. For all simulations, we use a link capacity of $c_l = 10$ Mbit/s, a reservation utilization percentile $p_u = 0.99$, and a measurement interval $\Delta = 1$ s. The maximum link utilization is set to $\rho_{max} = 0.95$.

The primary performance measure of our non-stationary EBAC simulations is the overall response time T_R, i.e., the time span required by the EBAC system to fully adapt the overbooking factor $\varphi(t)$ to a new traffic situation. We evaluate T_R for different settings of the EBAC memory T_H which depends on the histogram devaluation interval I_d and the devaluation factor f_d. We use the time exponentially-weighted moving histogram (TEWMH) method (cf. Section II-E.2) to avoid multiple simulations for different parameter combinations (I_d, f_d) yielding the same half-life period T_H. We use the packet delay $W(t)$ obtained from link l to derive time-dependent packet delay probabilities $p_d(t)$. Together with the time-dependent flow blocking probabilities $p_b(t)$ determined by the admission control process, they serve as indicators for potential QoS degradation.

The change of the traffic intensity is achieved by simultaneously adjusting the rates of all active flows in the simulation. The flow rates are controlled on the packet level of the traffic model by the setting of rate λ_p for the packet inter-arrival time distribution. Increasing λ_p thereby decreases the rate of a flow and vice versa.

B. Decrease of the Traffic Intensity

We first investigate the change of the traffic intensity from a high to a low value which corresponds to an increase of the PMRR $K(t)$ of the simulated traffic under control of EBAC.

1) Slow Decrease of the Traffic Intensity: We start with a slow decrease of the traffic intensity and thereby show the advantage of the TEWMH over the simple histogram method. The former uses adaptive increments as calculated in Section II-E.2 while the latter uses simple increments of constant size 1 to indicate a hit in the reservation utilization histogram $P(t, U)$. The content of $P(t, U)$ controls the overbooking factor $\varphi(t)$.

In our simulation, the traffic intensity, i.e., the PMRR of the simulated flows is controlled by the rate function $\lambda_p(t)$ for the packet inter-arrival time distribution:

$$\lambda_p(t) = \begin{cases} \lambda_p^0 & \text{for } t \leq t_0 \\ \lambda_p^0 + \frac{t - t_0}{t_1 - t_0} \cdot (\lambda_p^1 - \lambda_p^0) & \text{for } t_0 < t < t_1 \\ \lambda_p^1 & \text{for } t \geq t_1. \end{cases} \quad (5)$$

Equation (5) defines a linear decrease of the traffic intensity that starts at time t_0 with rate λ_p^0 and ends at time t_1 with rate λ_p^1. A traffic intensity decrease corresponds to an increase of the PMRR $K(t)$ as illustrated in Figures 2(a) and 2(b). At simulation time $t_0 = 230$ s, the PMRR starts to increase from $K(t) = 2$ to $K(t) = 4$ at $t_1 = 590$ s, i.e., all traffic sources slow down and the rates of the generated packet flows are steadily reduced. Figures 2(a) and 2(b) show simulation results averaged over 50 runs for different combinations of histogram devaluation intervals I_d and devaluation factors f_d which yield equal half-life periods of $T_H = 20$ s.

For a small devaluation interval $I_d = 10$ s in Figure 2(a), the evolution of the overbooking factor (OBF) $\varphi(t)$ is rather smooth. At simulation time $t_0 = 230$ s, the measured rate $M(t)$ decreases according to the traffic reduction. With a certain delay, EBAC increases $\varphi(t)$ and, therefore, more flows are accepted such that the reserved rate $R(t)$ is rising and $M(t)$ increases again to almost its former level. For a long devaluation interval $I_d = 360$ s in Figure 2(b), the evolution of $\varphi(t)$ equals a step function. At time $t = 230$ s, $M(t)$ starts to decrease like before. At time $t = 360$ s, EBAC devaluates the contents of the histogram $P(t, U)$ for the first time and strongly increases $\varphi(t)$ according to the changed traffic situation. In a short period of time, a large number of new flows are admitted by EBAC and $R(t)$ rises quickly. For the next 360 s, $\varphi(t)$ remains rather constant although the traffic intensity is still decreasing. Hence, $M(t)$ decreases again. At time $t = 720$ s, $P(t, U)$ is devaluated once more, and $\varphi(t)$ and $R(t)$ suddenly increase like for the last devaluation. Finally, the EBAC system reaches a new stable state after the decrease of the traffic intensity is finished.

The stepwise development of $\varphi(t)$ in Figure 2(b) is due to the fact that at times of devaluation, the contents of the reservation utilization histogram $P(t, U)$ are strongly devaluated by factor $f_d = 3.815 \cdot 10^{-6}$ such that the bins in $P(t, U)$ are almost empty. Therefore, each new sample $U(t)$ in $P(t, U)$ that enters shortly after a devaluation has a strong effect on the reservation utilization percentile $U_p(t)$ and, hence, on $\varphi(t) = \frac{1}{U_p(t)}$. As a consequence, the steps of $\varphi(t)$ are determined by the current reservation utilization $U(t) = \frac{M(t)}{R(t)}$ at times t of devaluation. At these time instants, the corners of the steps of $\varphi(t)$ approach the PMRR $K(t)$ and, hence, the safety margin between them diminishes which may lead to QoS violations. After the first devaluation, the utilizations $U(t)$ inserted into $P(t, U)$ decrease since $R(t)$ increases quickly and $M(t)$ continues to decrease. The 99%-percentile $U_p(t)$ thereby decreases only very slowly which keeps $\varphi(t)$ on a rather constant level until the next devaluation.

We replace the simple histogram by a TEWMH histogram to avoid the step function for $\varphi(t)$. With the TEWMH method, all combinations (I_d, f_d) yielding a half-life period $T_H = 20$ s lead to the same smooth development of $\varphi(t)$ as in Figure 2(a). Instead of incrementing the bins in $P(t, U)$ by one, we add weighted increments that give more importance to newer reservation utilization values (cf. Section II-E.2). As a consequence, the samples $U(t)$ are evenly devaluated.

The simulation results for an extended set of EBAC memory parameters with a half-life period of $T_H(I_d, f_d) = 20$ s are summarized in Table I for the simple (SIMPLE) and the TEWMH method. The average link utilization $E[U_l] = \text{avg}_t\{\frac{c_l}{M(t)}\}$ and the minimum deviation $\delta_{min} = \min_t\{K(t) - \varphi(t)\}$ for $t \in [200\ s, 800\ s]$ assess the performance of both approaches. For the simple method, $E[U_l]$ decreases for large values of I_d which is disadvantageous in a situation where traffic is blocked. Increasing I_d also reduces the safety margin between $\varphi(t)$ and its natural upper limit $K(t)$. For $I_d = 310$ s and $I_d = 360$ s we have $\delta_{min} < 0$ and, therefore, the QoS of admitted traffic is jeopardized for too long devaluation intervals. In contrast, applying the TEWMH method provides rather constant values $E[U_l]$ and δ_{min}, regardless of the settings of I_d and f_d.

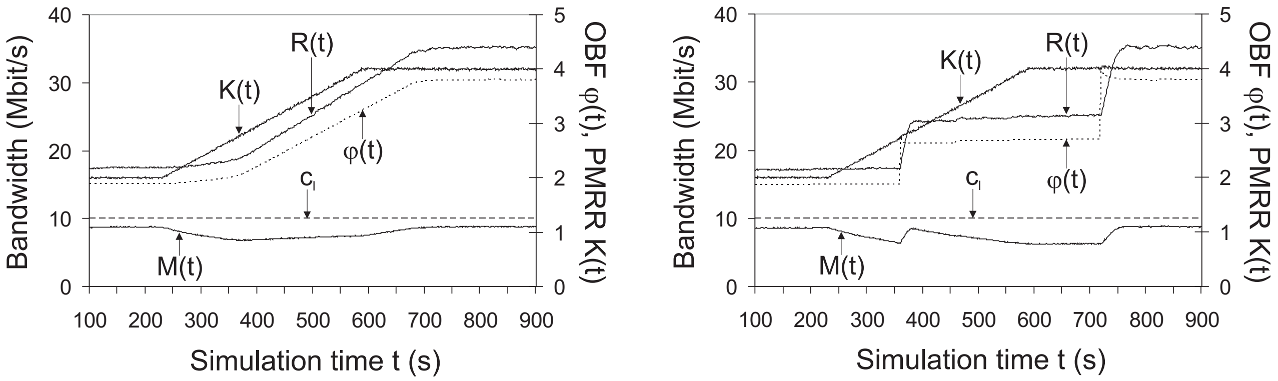
TABLE I
PERFORMANCE COMPARISON OF SIMPLE HISTOGRAM (SIMPLE) AND TEWMH METHOD FOR A SLOW TRAFFIC INTENSITY DECREASE AND DIFFERENT EBAC MEMORY SETTINGS.

		$E[U_l]$		δ_{min}	
I_d (s)	f_d	SIMPLE	TEWMH	SIMPLE	TEWMH
10	0.707	0.781	0.781	0.096	0.087
60	0.125	0.780	0.781	0.096	0.095
110	$2.209 \cdot 10^{-2}$	0.773	0.780	0.086	0.097
160	$3.906 \cdot 10^{-3}$	0.790	0.782	0.096	0.089
210	$6.905 \cdot 10^{-4}$	0.778	0.781	0.047	0.088
260	$1.221 \cdot 10^{-4}$	0.764	0.781	0.073	0.092
310	$2.158 \cdot 10^{-5}$	0.756	0.780	-0.002	0.097
360	$3.815 \cdot 10^{-6}$	0.720	0.781	-0.002	0.089

In summary, the presented results show that the simple histogram method is well applicable, but it must be carefully parameterized, i.e., its devaluation interval I_d must not be chosen too long compared to the half-life period T_H. Very short intervals increase the computational overhead for the devaluation of the histogram. The TEWMH is preferable since it does not require any other parameters besides the half-life period T_H. Its percentile $U_p(t)$ reacts rather quickly even for long devaluation intervals I_d. This improves the timeliness of the histogram without sacrificing the statistical significance of its values. Therefore, TEWMH is our preferred method for the implementation of the EBAC memory and it is used for all further simulations.

2) Sudden Decrease of the Traffic Intensity: We now investigate a sudden decrease of the traffic intensity, i.e., all currently and future admitted traffic sources simultaneously reduce their sending rate from one moment to the next. The simulation is designed similar to the slow traffic intensity decrease and the TEWMH method is again used to implement the EBAC memory. At simulation time $t_0 = 250$ s, the PMRR suddenly increases from $K(t) = 2$ to $K(t) = 3$.

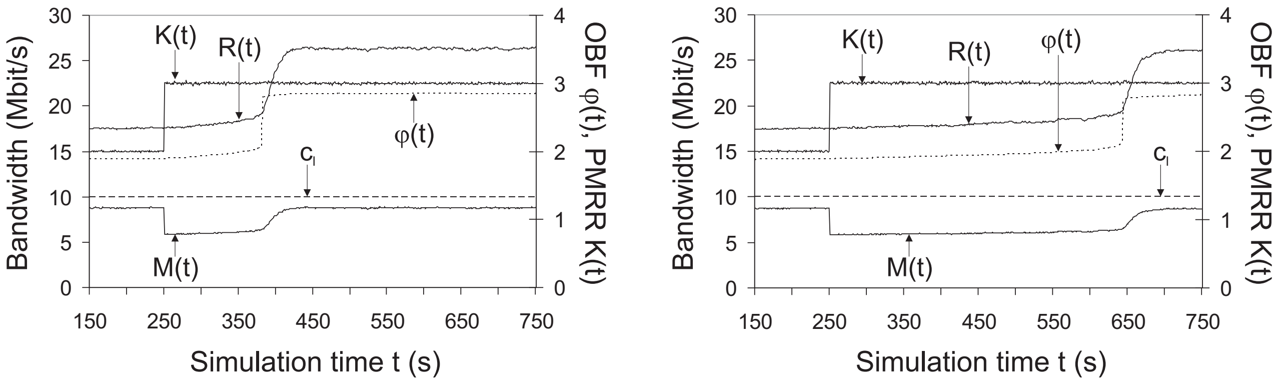
Figures 3(a) and 3(b) illustrate simulations averaged over 50 runs for different EBAC memories with half-life periods $T_H = 20$ s and $T_H = 60$ s. The primary y-axis indicates the link capacity c_l, the overall reserved bandwidth $R(t)$, and the consumed link bandwidth $M(t)$. The sudden increase of the PMRR results in an immediate decrease of $M(t)$ which also decreases the reservation utilization $U(t) = \frac{M(t)}{R(t)}$. Over time, the histogram $P(t, U)$ collects more and more low utilization values. As a consequence, the 99%-percentile $U_p(t)$ decreases which leads to a higher overbooking factor $\varphi(t) = \frac{1}{U_p(t)}$. Hence, more traffic sources are admitted to the link and the reserved



(a) Smooth function $\varphi(t)$ for devaluation interval $I_d = 10$ s and devaluation factor $f_d = 0.707$.

(b) Stepwise function $\varphi(t)$ for devaluation interval $I_d = 360$ s and devaluation factor $f_d = 3.815 \cdot 10^{-6}$.

Fig. 2. Impact of different combinations of histogram devaluation parameters with equal half-life period $T_H = 20$ s on overbooking factor.



(a) EBAC memory with half-life period $T_H = 20$ s.

(b) EBAC memory with half-life period $T_H = 60$ s.

Fig. 3. Impact of different EBAC memory half-life periods on time-dependent overbooking factor.

bandwidth $R(t)$ rises. Finally, the EBAC system stabilizes again with an expected overbooking factor $\varphi(t) \approx 3$. The speed of the adaptation process is obviously influenced by the EBAC memory parameter T_H.

To measure the duration of the transient phase, i.e., the time until $\varphi(t)$ reaches a new stable value, we calculate the difference between the PMRR $K(t)$ and the overbooking value $\varphi(t)$. If $K(t) - \varphi(t) < \varepsilon$, the transition between the two traffic scenarios is completed and the EBAC system is in steady state again. We therefore define the EBAC response time

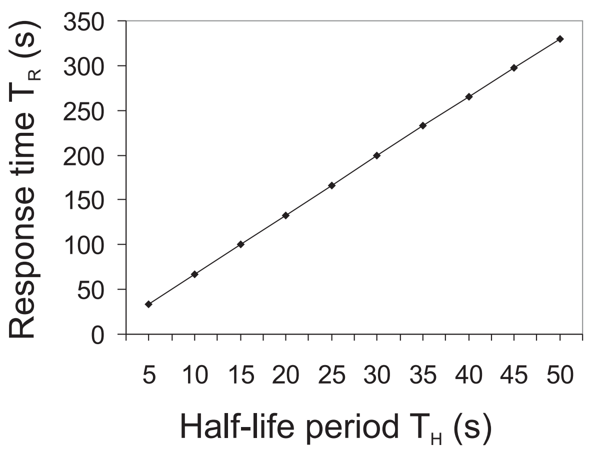
$$T_R = \min\{t_i - t_0 : K(t_i) - \varphi(t_i) < \varepsilon \wedge t_i > t_0\} \quad (6)$$

and set the threshold $\varepsilon = 0.2$ in our simulations. This value is specific to our experiments and seems to be appropriate with regard to the asymptotic convergence of $\varphi(t)$ to $K(t)$. Using the TEWMH method for the EBAC memory, $\varphi(t) \leq K(t)$ always holds. The statistical significance of our results is assured by calculating the 95% confidence intervals of the overbooking factor $\varphi(t)$ within 50 iterations of the simulation. As a result, the confidence intervals turn out to be so narrow that we omit them in Figures 3(a) and 3(b) for the sake of clarity.

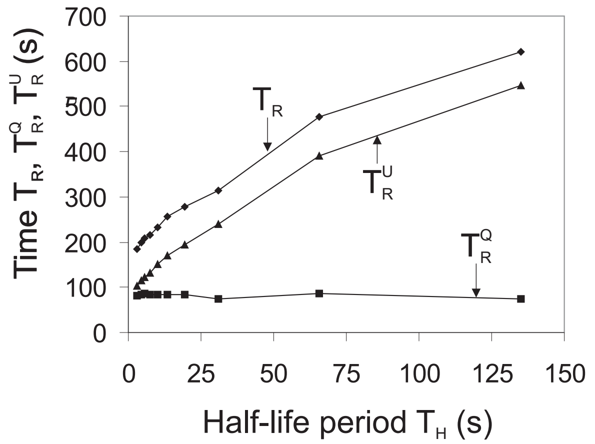
The different progressions of the overbooking factor $\varphi(t)$ in Figures 3(a) and 3(b) show that for a sudden traffic intensity decrease, the EBAC response time T_R strongly depends on the EBAC memory represented by the half-life period T_H. To investigate the correlation between the parameters T_R and T_H, we perform a series of experiments with varying half-life periods and measure the EBAC response times. Figure 4(a) shows that there is an almost linear dependency between the EBAC response time T_R and the half-life period T_H of the EBAC memory.

C. Increase of the Traffic Intensity

We now change the traffic intensity from a low to a high value which corresponds to a decrease of the aggregate PMRR $K(t)$, i.e., all admitted and future traffic sources simultaneously raise their sending rate from one moment to the next. This corresponds to a collaborative QoS attack. In contrast to the previous experiment, the QoS is at risk here as the link suddenly gets overloaded and the packet delay and flow blocking probabilities increase as expected during a QoS attack. To blind out the impact of the link buffer (cf. Figure 1) on the EBAC response time, we set its value to infinity. The QoS attack experiment is designed analogous to the decrease of the traffic intensity, but the sending rates of all traffic sources are increased such that the PMRR decreases from $K(t) = 3$ to $K(t) = 2$.



(a) Correlation between half-life period of EBAC memory and EBAC response time for decreasing traffic intensity.



(b) Impact of EBAC memory half-life period on overall response time and QoS/utilization restoration times of EBAC.

Fig. 4. Impact of different half-life periods on EBAC response time for changing traffic conditions.

Figures 5(a) and 5(b) show the overbooking and QoS performance of EBAC for a short half-life period of $T_H = 5.76$ s while Figures 6(a) and 6(b) show the same results for $T_H = 65.79$ s. At time $t_0 = 250$ ms, the QoS attack starts. As the link becomes overutilized, the fill level of the link buffer increases and the packet delay probability $p_d(t) = P(\text{packet delay} > 50 \text{ ms})$ and the flow blocking probability $p_b(t)$ raise to 100% (cf. Figures 5(b) and 6(b)). As another consequence, the overbooking factor $\varphi(t)$ decreases due to a rising reservation utilization percentile $U_p(t)$ and all new flows are blocked by EBAC. Over time, some admitted flows expire and their reserved bandwidth is released. However, $\varphi(t)$ is further decreased as long as the packet delay and the link load are high. Finally, the overbooking factor decreases below its target value of $\varphi(t) \approx 2$ (cf. Figures 5(a) and 6(a)). When enough flows have expired, the link buffer empties and the QoS is restored as a result of the decreased overbooking factor. Figures 5(b) and 6(b) show that the time T_R^Q required to restore QoS is almost the same for the short and the long EBAC memory, respectively. After a certain time span T_R^U, the overestimated reservation utilizations in the histogram are faded out by statistic aging. Simultaneously, the overbooking factor $\varphi(t)$ and the link utilization $U_l(t)$ converge to stable values when the EBAC system reaches its steady state again.

In contrast to Equation (6), we now define the EBAC response time as

$$T_R = T_R^Q + T_R^U, \tag{7}$$

where $T_R^Q = \min\{t_i - t_0 : p_d(t_i) = 0 \wedge t_i > t_0\}$ is the QoS restoration time and $T_R^U = \min\{t_j - (t_0 + T_R^Q) : K(t_i) - \varphi(t_i) < \varepsilon \wedge t_j > t_0 + T_R^Q\}$ is the utilization restoration time.

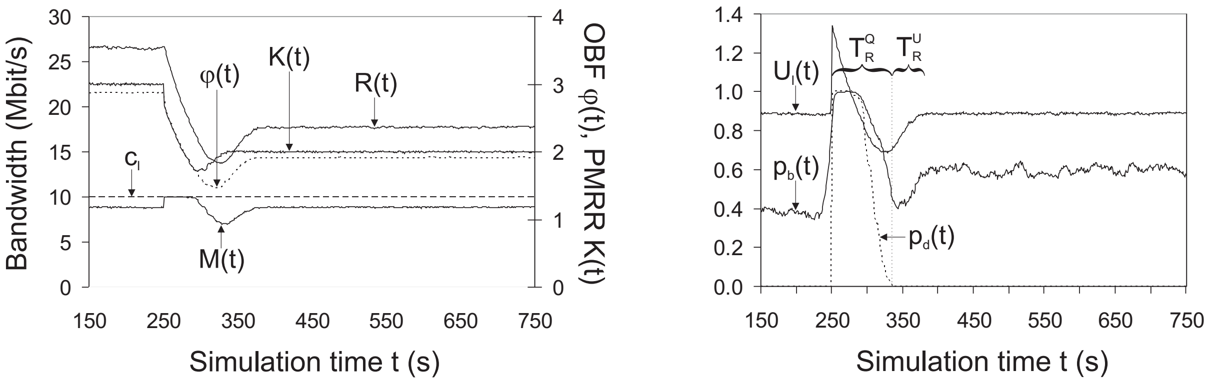
To investigate the correlation between T_R and T_H, we simulate a sudden traffic intensity increase for various half-life periods T_H. Our simulation results compiled in Figure 4(b) show that T_H influences the overall response time T_R of EBAC after a QoS attack. However, it does not influence the time T_R^Q required to restore the QoS.

TABLE II
IMPACT OF MEAN FLOW HOLDING TIME $E[1/\mu_f]$ AND BUFFER SIZE B ON COMPOUND EBAC RESPONSE TIME $T_R = T_R^Q + T_R^U$ FOR HALF-LIFE PERIOD $T_H = 20$ S.

$E[1/\mu_f]$ (s)	60	90	120		90	
B (ms)		1000		500	1000	2000
T_R (s)	105	119	132	70	118	143
T_R^Q (s)	45	62	77	54	62	64
T_R^U (s)	60	57	55	16	56	79

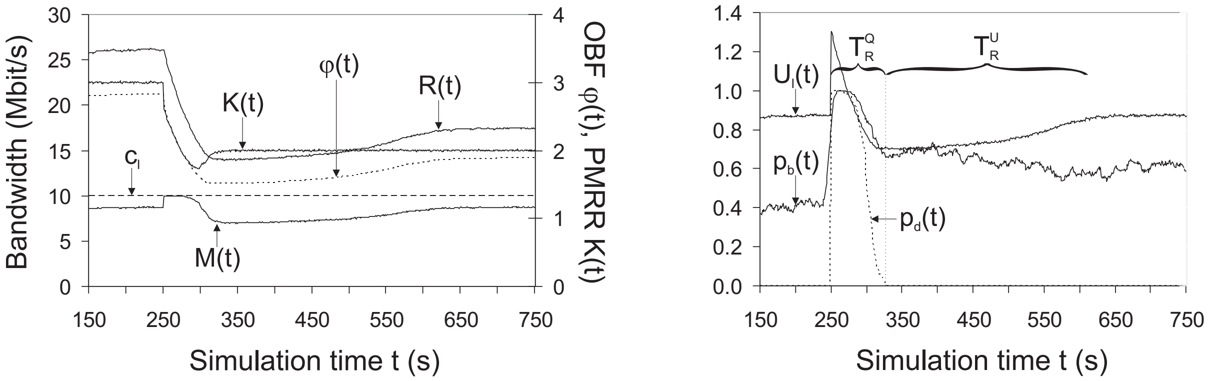
For the sake of completeness, we perform further experiments to investigate the impact of the mean flow holding time $E[1/\mu_f]$, the link buffer size B, and the link capacity c_l on the behavior of EBAC in case of a QoS attack. Table II shows that larger values for $E[1/\mu_f]$ and B both extend the EBAC response time T_R. However, increasing the mean flow holding time $E[1/\mu_f]$ primarily extends the QoS restoration time T_R^Q. This is reasonable since the restoration of QoS requires the termination of some admitted flows which are active on average for a longer time. Increasing the link buffer size B particularly affects the utilization restoration time T_R^U. A large buffer requires a longer time to be emptied. Therefore, the measured rate $M(t)$ is kept high for a longer time, more overestimated reservation utilizations $U(t)$ are sampled into the histogram $P(t, U)$, and a longer time T_R^U is required to fade them out.

Figures 7(a) and 7(b) illustrate the overbooking and the QoS performance of EBAC during a sudden increase of the traffic intensity on a link with capacity $c_l = 100$ Mbit/s. The EBAC memory is set to $T_H = 65.79$ s and, therefore, the results are directly comparable to Figures 6(a) and 6(b). The link capacity c_l has no remarkable effect on the overall EBAC response time T_R or its components T_R^Q and T_R^U. The temporary underestimation of $\varphi(t)$ and, hence, the end of the transient phase of the overbooking



(a) Overbooking performance. (b) QoS performance.

Fig. 5. Time-dependent EBAC performance during a QoS attack for an EBAC memory with half-life period $T_H = 5.76$ s.



(a) Overbooking performance. (b) QoS performance.

Fig. 6. Time-dependent EBAC performance during a QoS attack for an EBAC memory with half-life period $T_H = 65.79$ s.

adaptation are clearly more visible on the link with capacity $c_l = 100$ Mbit/s. This is due to the large number of flows that are multiplexed on the link and that allow for a more precise overbooking of the link resources.

The above statements concerning the impact of the mean flow holding time $E[1/\mu_f]$, the buffer size B, and the link capacity c_l hold for arbitrary settings of the EBAC memory parameter T_H.

V. Further Work on EBAC

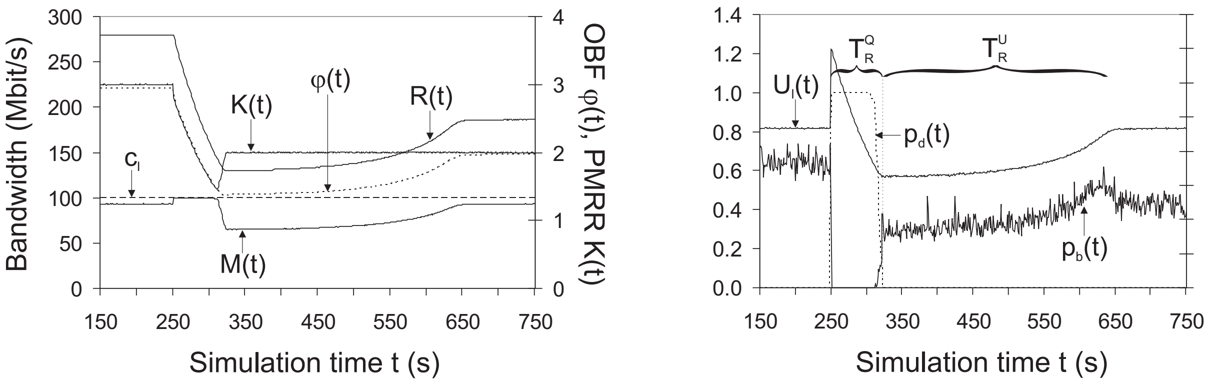
This section briefly presents further work on experience-based admission control. We illustrate the performance of EBAC in steady state, extend the original concept to type-specific overbooking (TSOB), and show a feasible application of EBAC in a network scope.

A. EBAC in Steady State

The intrinsic idea of EBAC is the exploitation of the peak-to-mean rate ratio $K(t)$ of the traffic aggregate admitted to the link. In [25], we simulate EBAC on a single link with regard to its behavior in steady state, i.e., when the properties of the traffic aggregate are rather static. These simulations provide a first proof of concept for EBAC. We show for different peak-to-mean rate ratios that EBAC achieves a high degree of resource utilization through overbooking while packet loss and packet delay are well limited. Further simulation results allow us to give recommendations for the EBAC parameters such as measurement interval length and reservation utilization percentile to obtain appropriate overbooking factors $\varphi(t)$. They furthermore show that the EBAC mechanism is robust against traffic variability in terms of packet size and inter-arrival time distribution as well as to correlations thereof.

B. EBAC with Type-Specific Overbooking (TSOB)

For the calculation of the overbooking factor $\varphi(t)$, only the traffic characteristics of the entire aggregate of admitted flows are considered in the original EBAC concept. In [27], we propose EBAC with type-specific overbooking (TSOB) which extends the original EBAC concept. EBAC with TSOB uses additional information about the characteristics of individual traffic types and about the composition of the admitted traffic mix to calculate a compound overbooking factor $\varphi_c(t)$. We thus consider different traffic types subsuming flows with similar peak-to-mean rate ratio and also their share in the currently admitted traffic mix. The concept of TSOB improves EBAC and can be



(a) Overbooking performance.

(b) QoS performance.

Fig. 7. Time-dependent EBAC performance during a QoS attack for a 100 Mbit/s link and an EBAC memory with half-life period of $T_H = 65.79$ s.

well implemented since it does not require type-specific traffic measurements. In [27], we give a proof of concept for EBAC with TSOB, describe the system extension, and show how the compound overbooking factor $\varphi_c(t)$ can be calculated without type-specific traffic measurements. We also compare the performance of EBAC with TSOB to conventional EBAC for traffic changes on the flow level. The simulation results show that EBAC with TSOB leads to better resource utilization under normal conditions and also to faster response times for changing traffic mixes.

C. Application of EBAC in a Network Scope

The performance of EBAC has intensively been studied by simulations on a single link. A simple deployment of EBAC in a network scope is the link-by-link application of the concept. However, this method requires a lot of signaling which may lead to scalability problems. Another option is to perform AC only at the network border by using separate instances of EBAC at all network ingress routers. This approach guarantees scalability but requires further investigation of the resulting distributed network admission control (NAC) system.

A prototype applying EBAC at the network border exists for the purely IP-based network architecture of the KING (Key components for the Internet of the Next Generation) project testbed [33], [34]. Its implementation requires the collection, synchronization, and correlation of many distributed network information about, e.g., resource reservations of flows admitted at the ingress routers, traffic measurements on the links, and routing and load balancing in the network. As a consequence, the network-wide admission decisions cannot be made independently of each other since they have a correlated impact on the link loads in the network.

However, if a network architecture fulfills certain requirements, the application of EBAC in the scope of a network is well feasible. The border-to-border (b2b) budget-based network admission control (BBB-NAC) presented in [6] is a feasible approach. The BBB-NAC implements admission control at the border of a network and uses directed b2b tunnels with pre-determined capacities. In a simple network-scoped implementation, these tunnel capacities can be overbooked by EBAC like physical link capacities, i.e., the tunnels are considered as virtual links. This approach requires separate EBAC instances for all capacity tunnels and, hence, the complexity of the problem is now reduced to appropriate tunnel dimensioning and to b2b aggregate-specific traffic measurements. If the network is based on, e.g. the (generalized) multi-protocol label switching ((G)MPLS) architecture [35], [36], tunnels can be implemented as label switched paths (LSPs) between label edge routers (LERs) and traffic can be easily measured per tunnel. Traffic matrices determined with help of the label distribution protocol (LDP) [37] provide the necessary traffic measurements per LSP.

VI. Conclusion

In this article, we gave a general overview of experience-based admission control (EBAC) and, in particular, investigated its transient behavior in the presence of traffic changes on the packet scale level.

EBAC is a new link admission control paradigm [24] and represents a hybrid solution between parameter-based and measurement-based admission control. We briefly reviewed the EBAC system and explained the simulation design and the traffic model used for the analyses of the transient behavior of EBAC. We simulated EBAC under changing traffic conditions on the packet level and showed the corresponding results as the main contribution of this article. Finally, we summarized further work on EBAC regarding its steady state behavior [25], its extension towards type-specific (TSOB) overbooking [27], and its application in a network-wide scope.

Since EBAC partly relies on traffic measurements, it is susceptible to changes of the traffic characteristics. There are certain influencing parameters coupled with this problem. Among them are the implementation and the length of the EBAC memory which is specified by its half-life period T_H. We first simulated a slow traffic intensity decrease and thereby showed that using a time exponentially-weighted moving histogram (TEWMH) [28] instead of a simple histogram to implement the EBAC memory facilitates the simulation of

EBAC with different memory settings. We then tested the impact of the EBAC memory on a sudden decrease and increase of the traffic intensity which was expressed by the change of the peak-to-mean rate ratio of the simulated traffic flows. We showed that, in case of a changing traffic intensity, the response time T_R required to adapt the overbooking factor to the new traffic conditions depended linearly on the half-life period T_H. For a decreasing traffic intensity, the QoS of the traffic was not at risk. For a suddenly increasing traffic intensity, however, it was compromised for a certain time span T_R^Q which was less than the average flow holding time and independent of the EBAC memory length. Note that the respective experiment used an unlimited link buffer and investigated the performance of EBAC under very extreme traffic conditions that correspond to a collaborative and simultaneous QoS attack by all traffic sources.

In conclusion, the resources controlled by admission control mechanisms in next generation networks (NGNs) must be exclusively dedicated to admitted traffic to guarantee QoS. For that purpose, robust and efficient concepts for the management of network resource are required to control the requested bandwidth with regard to the available transmission capacity. Sophisticated admission control will be a key function for the management of network resources in NGNs and, therefore, efficient concepts like experience-based admission control as presented in this article are appealing for NGN solutions.

ACKNOWLEDGMENT

We thank Prof. Tran-Gia for the stimulating environment which was a prerequisite for this work and the reviewers for their detailed comments.

REFERENCES

[1] R. Martin, M. Menth, and J. Charzinski, "Comparison of Link-by-Link Admission Control and Capacity Overprovisioning," in *Proc. of International Teletraffic Congress (ITC)*, August 2005.

[2] R. Martin, M. Menth, and J. Charzinski, "Comparison of Border-to-Border Budget Based Network Admission Control and Capacity Overprovisioning," in *Proc. of IFIP-TC6 Networking Conference (NETWORKING)*, May 2005, pp. 1056–1068.

[3] R. Braden *et al.*, "RFC 2205: Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification," September 1997.

[4] R. Yavatkar, D. Pendarakis, and R. Guerin, "RFC 2753: A Framework for Policy-based Admission Control," January 2000.

[5] R. Yavatkar, D. Hoffman, Y. Bernet, F. Baker, and M. Speer, "RFC 2814: SBM (Subnet Bandwidth Manager): A Protocol for RSVP-based Admission Control over IEEE 802-style networks," May 2000.

[6] M. Menth, "Efficient Admission Control and Routing for Resilient Communication Networks," PhD thesis, University of Würzburg, July 2004.

[7] S. Shenker, C. Partridge, and R. Guerin, "RFC 2212: Specification of Guaranteed Quality of Service," September 1997.

[8] J. L. Boudec, "Application of Network Calculus to Guaranteed Service Networks," *IEEE Network Magazine*, vol. 44, no. 3, May 1998.

[9] M. Fidler and V. Sander, "A Parameter Based Admission Control for Differentiated Services Networks," *Computer Networks*, vol. 44, no. 4, pp. 463–479, 2004.

[10] J. Roberts, U. Mocci, and J. Virtamo, *Broadband Network Teletraffic - Final Report of Action COST 242*. Springer, 1996.

[11] J. Wroclawski, "RFC 2211: Specification of the Controlled-Load Network Element Service," September 1997.

[12] M. Grossglauser and D. Tse, "A Time-Scale Decomposition Approach to Measurement-Based Admission Control," *IEEE/ACM Transactions on Networking*, vol. 11, no. 4, pp. 550–563, August 2003.

[13] S. Jamin, S. Shenker, and P. Danzig, "Comparison of Measurement-Based Call Admission Control Algorithms for Controlled-Load Service," in *Proc. of IEEE Conference on Computer Communications (INFOCOM)*, March 1997, pp. 973–980.

[14] Z. Turanyi, A. Veres, and A. Olah, "A Family of Measurement-Based Admission Control Algorithms," in *Proc. of IFIP Conference on Performance of Information and Communication Systems*, May 1998.

[15] L. Breslau, S. Jamin, and S. Shenker, "Comments on the Performance of Measurement-Based Admission Control Algorithms," in *Proc. of IEEE Conference on Computer Communications (INFOCOM)*, March 2000, pp. 1233–1242.

[16] K. Shiomoto, N. Yamanaka, and T. Takahashi, "Overview of Measurement-Based Connection Admission Control Methods in ATM Networks," *IEEE Communications Surveys & Tutorials*, vol. 2, no. 1, pp. 2–13, January 1999.

[17] H. van den Berg and M. Mandjes, "Admission Control in Integrated Networks: Overview and Evaluation," in *Proc. of International Conference on Telecommunication Systems, Modeling and Analysis (ICTSM)*, 2000, pp. 132–151.

[18] J. Qiu and E. Knightly, "Measurement-Based Admission Control with Aggregate Traffic Envelopes," *IEEE/ACM Transactions on Networking*, vol. 9, no. 2, pp. 199–210, April 2001.

[19] S. Georgoulas, P. Trimintzios, and G. Pavlou, "Joint Measurement- and Traffic Descriptor-based Admission Control at Real-Time Traffic Aggregation Points," in *Proc. of IEEE International Conference on Communications (ICC)*, June 2004.

[20] S. Jamin, P. Danzig, S. J. Shenker, and L. Zhang, "Measurement-Based Admission Control Algorithms for Controlled-Load Services Packet Networks," in *Proc. of ACM SIGCOMM Symposium on Communications Architectures & Protocols*, 1995.

[21] R. Gibbens and F. Kelly, "Measurement-Based Connection Admission Control," in *Proc. of International Teletraffic Congress (ITC)*, June 1997.

[22] T. Lee, M. Zukerman, and R. Addie, "Admission Control Schemes for Bursty Multimedia Traffic," in *Proc. of IEEE Conference on Computer Communications (INFOCOM)*, April 2001, pp. 478–487.

[23] M. Dabrowski and F. Strohmeier, "Measurement-Based Admission Control in AQUILA Network and Improvements by Passive Measurements," in *Proc. of Architectures for Quality of Service in the Internet (Art-QoS)*, March 2003.

[24] J. Milbrandt, M. Menth, and S. Oechsner, "EBAC - A Simple Admission Control Mechanism," in *Proc. of IEEE International Conference on Network Protocols (ICNP)*, October 2004.

[25] M. Menth, J. Milbrandt, and S. Oechsner, "Experience Based Admission Control (EBAC)," in *Proc. of IEEE Symposium on Computers and Communications (ISCC)*, June 2004.

[26] J. Milbrandt, M. Menth, and J. Junker, "Performance of Experience-Based Admission Control in the Presence of Traffic Changes," in *Proc. of IFIP-TC6 Networking Conference (NETWORKING)*, May 2006.

[27] J. Milbrandt, M. Menth, and J. Junker, "Experience-Based Admission Control with Type-Specific Overbooking," in *Proc. of IEEE International Workshop on IP Operations and Management (IPOM)*, October 2006.

[28] M. Menth, J. Milbrandt, and J. Junker, "Time-Exponentially Weighted Moving Histograms (TEWMH) for Self-Adaptive Systems," in *Proc. of IEEE Global Telecommunications Conference (GLOBECOM)*, November 2006.

[29] R. Martin and M. Menth, "Improving the Timeliness of Rate Measurements," in *Proc. of GI/ITG Conference on Measuring, Modelling and Evaluation of Computer and Communication Systems (MMB) together with Polish-German Teletraffic Symposium (PGTS)*, September 2004.

[30] F. Hübner and P. Tran-Gia, "An Analysis of Multi-Service Systems with Trunk Reservation Mechanisms," University of Würzburg, Institute of Computer Science, Technical Report No. 40, April 1992.

[31] A. M. Law and W. D. Kelton, *Simulation Modeling and Analysis*. McGraw-Hill, 2000.

[32] V. Paxson and S. Floyd, "Wide-Area Traffic: The Failure of Poisson Modeling," *IEEE/ACM Transactions on Networking*, vol. 3, no. 3, pp. 226–244, June 1995.

[33] N. Heldt, "Re-Engineering the Internet Gently," *I&C World*, vol. 12, December 2002.

[34] C. Hoogendoorn, K. Schrodi, M. Huber, C. Winkler, and J. Charzinski, "Towards Carrier-Grade Next Generation Networks," in *Proc. of International Conference on Communication Technology (ICCT)*, April 2003.

[35] E. Rosen, A. Viswanathan, and R. Callon, "RFC 3031: Multiprotocol Label Switching Architecture," January 2001.

[36] E. Mannie, "RFC 3945: Generalized Multi-Protocol Label Switching (GMPLS) Architecture," October 2004.

[37] S. Schnitter and M. Horneffer, "Traffic Matrices for MPLS Networks with LDP Traffic Statistics," in *Proc. of International Telecommunication Network Strategy and Planning Symposium (Networks)*, June 2004, pp. 231–236.

Jens Milbrandt studied computer science and economics in Wurzburg/Germany and received his diploma degree in 2001. He finished his Ph.D. on performance evaluation and resource management in next generation networks at the University of Wurzburg (Institute of Computer Science, Chair of Distributed Systems) in 2006.

He now works as a scientist and developer in telecommunications for Alacatel/Lucent in Stuttgart/Germany and is concerned with network architecture and control plane issues in optical transport network.

Michael Menth studied computer science and mathematics at the Universities of Wurzburg/Germany and Austin/Texas. He has been working as a researcher at the Universities of Ulm/Germany and Wurzburg where he is currently assistant professor and heading the research group on next generation networks. His special interests are performance analysis, optimization of communication networks, resource management, and resilience issues. Dr. Menth holds about 25 patent applications and has received various scientific awards for innovative work.

Jan Junker studied computer science at the University of Wurzburg (Institute of Computer Science, Chair of Distributed Systems) in Germany. In his diploma thesis, he dealt with performance evaluation and quality of service of IP networks.

After having received his diploma in computer science, he now works as consultant in the department for research and development of the PASS Consulting Group. There he focuses on the sectors of model-driven software development, service-oriented architecture and software engineering automation based on code generation.