The Search for QoS in Data Networks: A Statistical Approach

Pablo Belzarena and María Simon

Universidad de la República
Montevideo, Uruguay
{belza,msimon}@fing.edu.uy

Abstract. New Internet services like video on-demand, high definition IPTV, high definition video conferences and some real time applications have strong QoS requirements regarding losses, delay, jitter, etc. This work addresses the challenge of guaranteeing quality of service (QoS) in the Internet from a statistical point of view. Three lines of work are proposed. The first one is about the estimation of the QoS parameters from traffic traces (in the context of large deviation theory and effective bandwidth). The second one, address the admission control problem from results of the many sources and small buffer asymptotic. Finally, the third line focuses on the estimation of QoS parameters seen by an application based on end-to-end active measurements and statistical learning tools.
Keywords: *large deviations, statistical learning, admission control, active measurements, quality of service*

1 Introduction and motivation

Internet services with high quality of service requirements like video on-demand, high quality video conferences, high definition IPTV, telematic services with real time requirements, etc. have grown at a smaller rate than the initially hoped. One possible cause is the difficulty that exists in order to guarantee end-to-end quality of service (QoS) in IP networks. Another possible cause is that the operators have not deploy the different proposals developed during the last 15 years in order to assure QoS (IntServ, DiffServ, etc.). These difficulties have been recently increased by the heterogeneity of the access networks (xdsl, cablemodem, wifi, wimax, 2G, 3G, mesh networks, etc.). End-to-end QoS leads to another issue; in the general case, the end-to-end performance parameters can not be estimated from the performance parameters of each individual router in the path. This problem becomes even worst when the service operator offers its service over multiple domains. In this case, the nodes of the path are under the administration of different network operators.

An important issue in this context is the network admission control based on end-to-end QoS. In a network of "premium" services this kind of admission control allows the operator to control the end-to-end QoS. This issue is one of the main motivations of this work.

The focus of this work is on the estimation of the admission control region. We look for a simple and efficient procedure for such estimation which can be applied on line. A control admission tool using this estimation can decide if it accepts or not a new service request.

The admission control mechanisms proposed in the literature are mainly based on one link analysis [1]. We start analyzing admission control mechanisms where the link analysis is based on Large Deviation Theory (LDT). In the analysis of networks performance using LDT [2] three main asymptotic regimes have been described. These are the large buffer regime, the many sources asymptotic and the many source and small buffer asymptotic. In the first case the convergence rate to zero of some QoS parameter (e.g. loss probability) when the buffer size goes to infinity is studied. In the second one it is also studied the convergence rate to zero of the loss probability but when there are many independent and identically distributed sources arriving at the link and the link capacity and the buffer size both increases at the same rate as the number of sources. In the third case, there are many independent sources, the link capacity grows with the number of sources but the buffer size grows slower than the number of sources. The large buffer asymptotic can be applied only in the access networks where there are few sources and the buffer per source can be considered big. However, the large buffer asymptotic can only be applied to one isolated link because the output of that link does not verify the assumptions needed to apply the asymptotic to the following link in the path.

In networks such as an internet backbone, the many sources asymptotic approach is more reasonable than the large buffer one. In fact, in this kind of backbone, large numbers of flows from different sources arrive, the capacities are high and the buffer sizes per source are in general small, because they are intended to serve many sources but not many bursts at the same time.

We start analyzing in Section 2 a link based admission control. This mechanism is based on the many sources asymptotic and in particular on the effective bandwidth notion [3]. In this work we address an important issue for an on-line admission control mechanism: the estimation of the QoS performance parameters (particulary the buffer overflow probability) based on traffic traces.

Although we analyze the estimation of buffer overflow probability in the many source asymptotic, the results of this work can also be applied to estimate the large deviation rate function in the many sources and small buffer asymptotic introduced by Ozturk et al.[4]. The small buffer asymptotic presents more interesting results in order to analyze the end-to-end QoS and not only the QoS of an isolated link. For a service provider the most interesting issue is about admission control mechanisms based on the end-to-end QoS. In Section 3 we analyze end-to-end QoS applying the so called "fictitious network model". This model is based on the many sources and small buffer asymptotic. We will show that this model allows simple and on-line estimations of end-to-end QoS parameters, which will be in turn used to decide which flows can access the network.

Ozturk et al. find a useful way to analyze the overflow probability in a network interior link and show that when the fictitious network model is applied,

an overestimation is obtained. The fictitious network analysis gives then a simple and efficient yet conservative way to implement on-line admission control mechanisms. However, the overestimation can translate into wasted network resources. If a flow is admitted, its QoS is guaranteed but the link capacities can be under-used. In this work we analyze in detail the fictitious network model and we find conditions to assure that the fictitious network analysis in an interior link gives the same overflow probability than the real network analysis, being much simpler. We also find a method to bound the overestimation when these conditions are not fulfilled. In addition, since no model is assumed for the input traffic, we define an estimator of the end-to-end Loss ratio based on traffic measurements. We show that this estimator is a good one, i.e. it is consistent and verifies a Central Limit Theorem (CLT). These results allow us to define an admission control mechanism based on the expected end-to-end Loss Ratio that a flow traversing the network will obtain.

However, the many sources and small buffer asymptotic can only be applied to analyze an end-to-end path in a backbone network. If the end points are end users this asymptotic cannot be applied because the path goes through the backbone but also through the access network where the many sources asymptotic is not valid. The research community does not have yet a model in order to analyze an end-to-end path including the access and the backbone network. Therefore, a different approach must be applied if the access control mechanism must take a decision based on end-to-end QoS.

Some authors propose end-to-end admission control mechanism based on active measurements [5]. In the third part of this tutorial, in Section 4, we propose a different approach for an end-to-end admission control. This approach is based on active measurements and statistical learning tools. We analyze the application of an statistical learning approach in order to predict the quality of service seen by an application.

Although the end-to-end admission control problem is our main motivation, the different issues analyzed in this work can be applied to many other network operation and management problems like for example to share resources in a network, to continuous monitoring a Service Level Agreement (SLA), etc..

2 Effective bandwidth and link operation point estimation

2.1 Introduction

One of the main issues in QoS admission control is the estimation of the resources needed for guaranteed VBR communications, which cannot be the peak rate nor the mean rate of the service. Indeed, the mean rate would be a too optimistic estimation, that would cause frequent losses. On the other side, the peak rate would be too pessimistic and would lead to resource waste.

Effective bandwidth (EB) defined by F. Kelly in [3] is an useful and realistic measure of channel occupancy. The EB is defined as follows:

$$\alpha(s,t) = \frac{1}{st}\log \mathbf{E}\left(e^{sX_t}\right) \qquad 0 < s,t < \infty. \tag{1}$$

where X_t is the total amount of work arriving from a source in the time interval $[0,t]$, which is supposed to be a stochastic process with stationary increments. $\alpha(s,t)$ lies between the mean rate (for $s \to 0$) and the peak rate (for $s \to \infty$) of the input process.

Parameters s and t are referred to as the space and time parameters respectively. When solving for a specific performance guarantee, these parameters depend not only on the source itself, but on the context on which this source is acting. More specifically, s and t depend on the capacity, buffer size and scheduling policy of the multiplexer, the QoS parameter to be achieved, and the actual traffic mix (i.e. characteristics and number of other sources). The EB concept can be applied to sources or to aggregated traffic, as we find in a network's core link.

Under the *many sources asymptotic regime* discussed in [6], where it is assumed that, as the number of sources feeding a switch grows, the switch capacity and buffer size increase proportionally, the EB is related with the stationary loss probability through buffer overflow by the so called *inf sup* formula:

$$\Gamma = \inf_{t\geq 0}\sup_{s\geq 0}((B+Ct)s - Nst\alpha(s,t))$$

where C is the link capacity, B is its buffer size and N the number of incoming multiplexed sources of effective bandwidth $\alpha(s,t)$. If Q_N represents the stationary amount of work in the queue, the buffer overflow probability or loss probability is approximately given by:

$$\log \mathbf{P}(Q_N > B) \approx -\Gamma$$

We call s^* and t^* to the values of parameters s and t in which the *inf sup* is attained. These values s^* and t^* are called the link's *operating point*.

Therefore, a good estimation of s^* and t^* is useful for the network's design, for the Connection Admission Control (CAC) function, or for optimal operating.

We point out the need of a good estimation of the bandwidth in order to optimize resource sharing.

In section 2.2 we show how the operating point of a link can be estimated from its EB, the consistency of this estimation and its confidence interval. We observe that other well known estimators fit the necessary conditions for the validity of the theorem.

Analytical results are compared with numerical data in section 2.3. These numerical data were obtained independently from the analytical work from simulations models that are also explained in this section. In this framework, overflow probability estimation is a key topic, which makes necessary EB and link's operating point estimation.

2.2 Estimation

Estimating the operating point of a link, as defined in section 2.1 is closely related with its defining equation which we rewrite here on a *per source* basis:

$$\gamma = \inf_{t\geq 0} \sup_{s\geq 0}((b+ct)s - st\alpha(s,t)) \tag{2}$$

where γ is the asymptotic decay rate of the overflow probability as the number of sources increases, c and b are the link's capacity and buffer size *per source* and $\alpha(s,t)$ the effective bandwidth function (equation 1). With the present notation, stationary overflow probability in a switch multiplexing N sources, having capacity $C = Nc$ and buffer size $B = Nb$ verifies:

$$\lim_{N\to\infty} \frac{1}{N} \log \mathbf{P}\left(Q_N > B\right) = -\gamma \tag{3}$$

In general, the effective bandwidth function $\alpha(s,t)$ is unknown, and shall be estimated from measured traffic traces. The problem is how to estimate the moment generating function $\Lambda(s,t) = \mathbf{E}\left(e^{sX_t}\right)$ of the incoming traffic process X_t for each s and t.

Different approaches have been presented to solve this problem. One of them, presented in [7] and [8] is to estimate the expectation $\mathbf{E}\left(e^{sX_t}\right)$ as the time average given by:

$$\Lambda_n(s,t) = \frac{1}{n}\sum_{k=1}^{n} e^{s(X_{kt} - X_{(k-1)t})} \tag{4}$$

which is valid if the process increments are stationary and satisfy any weak dependence hypothesis that guarantees ergodicity. To estimate $\Lambda(s,t)$ a traffic trace of length $T = nt$ is needed. We can construct an appropriate estimator of the EB as $\alpha_n(s,t) = \frac{1}{st}\log(\Lambda_n(s,t))$.

When a model is available for incoming traffic, a parametric approach can be taken. In the case of a Markov Fluid model, i.e. when the incoming process is modulated by a continuous time Markov chain which dictates the rate of incoming work, explicit computation can be made as shown by Kesidis et al. in [9]. In this case, an explicit formula is given for $\Lambda(s,t)$ and $\alpha(s,t)$ in terms of the infinitesimal generator or Q-matrix of the Markov chain. In a previous work of our group [10], and based on the maximum likelihood estimators of the Q-matrix parameters presented in [11], an EB estimator and confidence intervals are developed.

Having an estimator of the function $\alpha(s,t)$, it seems natural to estimate γ, and the operating point s^*, t^* substituting the function $\alpha(s,t)$ by $\alpha_n(s,t)$ in equation (2) and solving the remaining optimization problem. The output would be some values of γ_n, s_n^* and t_n^*, and the question is under what conditions these values are good estimators of the real γ, s^* and t^*.

Therefore, we may discuss two different problems concerning estimation. The first one is, given a "good" estimator $\alpha_n(s,t)$ of $\alpha(s,t)$, find sufficient conditions

under which the estimators s_n^*, t_n^* and γ_n^* obtained by solving the optimization problem:

$$\gamma_n = \inf_{t\geq 0} \sup_{s\geq 0}((b+ct)s - st\alpha_n(s,t)) \tag{5}$$

are "good" estimators of the operating point s^*, t^* and the overflow probability decay rate γ of a link. This affirmation is not an obvious result because s^* and t^* are found from a non linear and implicit function. We remark that the reasoning applied to s^* and t^* can be also applied to other parameters that are deduced from the EB. Further in the article the parameters B and C are also studied.

The second problem is finding this good estimator of the EB and determining whether the conditions are met, so that the operating point can be estimated using equation (5).

The remaining part of the section addresses the first problem, where a complete answer concerning consistency and Central Limit Theorem (CLT) properties of estimators is given by theorem 1, based on regularity conditions of the EB function. At the end of the section we discuss the validity of the theorem for some known estimators and in section 2.3 we compare our analytical results with numerical ones.

Let us define:

$$g(s,t) = s(b+ct) - st\alpha(s,t)$$

which can be rewritten in terms of $\Lambda(s,t) = \mathbf{E}\left(e^{sX_t}\right)$. We have that $\frac{\partial}{\partial s}g(s,t) = 0$ if and only if:

$$\frac{\partial}{\partial s}g(s,t) = b + ct - \frac{\frac{\partial}{\partial s}\Lambda(s,t)}{\Lambda(s,t)} = 0 \tag{6}$$

Assuming that for each t there exists $s(t)$ such that, $\frac{\partial}{\partial s}g(s(t),t) = 0$, it is easy to show that $\sup_{s\geq 0} g(s,t) = g(s(t),t)$ because $g(s,t)$ is convex as a function of s. In that case, $\gamma = \inf_{t\geq 0} g(s(t),t)$, and:

$$\frac{\partial}{\partial t}g(s(t),t) = \frac{\partial}{\partial s}g(s(t),t)\dot{s}(t) + \frac{\partial}{\partial t}g(s,t)\Big|_{s=s(t)}$$

If there exists t^* such that: $\frac{\partial}{\partial t}g(s(t^*),t^*) = 0$ and the infimum is attained, it follows that: $\gamma = g(s(t^*),t^*)$.

If we define $s^* = s(t^*)$, we have that $\gamma = g(s^*,t^*)$ where:

$$\frac{\partial}{\partial s}g(s^*,t^*)\,\dot{s}(t^*) + \frac{\partial}{\partial t}g(s^*,t^*) = 0 \qquad \text{and} \qquad \frac{\partial}{\partial s}g(s^*,t^*) = 0$$

and then we have the relations:

$$\frac{\partial}{\partial s}g(s^*,t^*) = \frac{\partial}{\partial t}g(s^*,t^*) = 0 \tag{7}$$

Since:

$$\frac{\partial}{\partial t}g(s,t) = cs - \frac{\frac{\partial}{\partial t}\Lambda(s,t)}{\Lambda(s,t)} \tag{8}$$

it follows from (6), (7) and (8) that the operating point must satisfy the equations:

$$b + ct^* - \frac{\frac{\partial}{\partial s}\Lambda(s^*,t^*)}{\Lambda(s^*,t^*)} = 0 \qquad \text{and} \qquad cs^* - \frac{\frac{\partial}{\partial t}\Lambda(s^*,t^*)}{\Lambda(s^*,t^*)} = 0 \tag{9}$$

If we make the additional assumptions that interchanging the order of the differential and expectation operators is valid, and that $\dot{X}_t$ exists for almost every t we can write:

$$\frac{\partial}{\partial s}\Lambda(s,t) = \mathbf{E}\left(X_t e^{sX_t}\right) \qquad \frac{\partial}{\partial t}\Lambda(s,t) = \mathbf{E}\left(s\dot{X}_t e^{sX_t}\right) \tag{10}$$

Replacing the expressions of (10) in equations (9) we deduce an alternative expression for the solutions s^* and t^*:

$$b + ct^* - \frac{\mathbf{E}\left(X_{t^*} e^{s^* X_{t^*}}\right)}{\mathbf{E}\left(e^{s^* X_{t^*}}\right)} = 0 \qquad \text{and} \qquad cs^* - \frac{\mathbf{E}\left(s^*\dot{X}_{t^*} e^{s^* X_{t^*}}\right)}{\mathbf{E}\left(e^{s^* X_{t^*}}\right)} = 0 \tag{11}$$

Therefore, we can reformulate the optimization problem presented in (2). The operating point of the link can be calculated solving the system of equations (9), or (11) if the additional assumptions are valid. The first formulation, which is more general, is the one used in the main result of this work, which follows:

Theorem 1. *If $\Lambda_n(s,t)$ is an estimator of $\Lambda(s,t)$ such that both are C^1 functions and:*

$$\Lambda_n(s,t) \underset{n}{\longrightarrow} \Lambda(s,t) \qquad \frac{\partial}{\partial s}\Lambda_n(s,t) \underset{n}{\longrightarrow} \frac{\partial}{\partial s}\Lambda(s,t) \qquad \frac{\partial}{\partial t}\Lambda_n(s,t) \underset{n}{\longrightarrow} \frac{\partial}{\partial t}\Lambda(s,t) \tag{12}$$

almost surely and uniformly over bounded intervals, and if we denote s_n^* and t_n^* the solutions of:

$$b + ct_n^* - \frac{\frac{\partial}{\partial s}\Lambda_n(s_n^*,t_n^*)}{\Lambda_n(s_n^*,t_n^*)} = 0 \qquad cs_n^* - \frac{\frac{\partial}{\partial t}\Lambda_n(s_n^*,t_n^*)}{\Lambda_n(s_n^*,t_n^*)} = 0 \tag{13}$$

then (s_n^*, t_n^*) are consistent estimators of (s^*, t^*). Moreover, if a functional Central Limit Theorem (CLT) applies to $\Lambda_n - \Lambda$, i.e,

$$\sqrt{n}\left(\Lambda_n(s,t) - \Lambda(s,t)\right) \underset{n}{\overset{w}{\Longrightarrow}} G(s,t),$$

where $G(s,t)$ is a continuous gaussian process, then:

$$\sqrt{n}\left((s_n^*,t_n^*) - (s,t)\right) \underset{n}{\overset{w}{\Longrightarrow}} N(\mathbf{0},\Sigma) \tag{14}$$

where $N(\mathbf{0},\Sigma)$ is a centered bivariate normal distribution with covariance matrix Σ.

Proof. See [12].

Remark 1. The computation of Σ is not trivial. However, if replication is possible (for instance by taking large traces of weak-dependent signals), the previous result allows the estimation of Σ in terms of empirical covariances. Arguments of this type are used in section 2.3.

Remark 2. Since the convergence assured by theorem 1 is uniform over bounded intervals, it is also assured that γ_n given by:

$$\gamma_n = s_n^*(b + ct_n^*) - \Lambda_n(s_n^*, t_n^*)$$

inherits the properties of the s_n^* and t_n^* estimators. That is, $\gamma = F(s^*, t^*, \Lambda)$ where F is a differentiable function. Also, $\gamma_n = F(s_n^*, t_n^*, \Lambda_n)$. Therefore, if the estimator Λ_n verifies a functional CLT we have for γ_n:

$$\sqrt{n}\,(\gamma_n - \gamma) \underset{n}{\Longrightarrow} N(0, \sigma^2)$$

Remark 3. In a many source environment, expressions for the buffer size b and the link capacity c obtained by Courcoubetis [7] are similar to the *inf sup* equation. Therefore, the reasoning used in the previous theorem extends consistency and CLT results to b^* and c^*. Also, confidence intervals for these design parameters can be constructed in this way.

We address now the second question posed at the beginning of the section. As we can see, for the validity of theorem 1 it is necessary that the estimator $\Lambda_n(s,t)$ converge uniformly to the moment generating function over bounded intervals, as well as its partial derivatives. These conditions are reasonably general, and it can be verified that they are met by the estimator (4) presented in [7] and [8], and by the estimator for Markov Fluid sources presented in [10]. In both cases a CLT can be obtained so the CLT conclusion of the theorem is also valid. It should be noticed that a consistent but non-smooth estimator can be used with this procedure, if it is previously regularized by convolution with a suitable kernel.

2.3 Simulation and numerical results

EB and operation point estimation. In order to validate the results obtained in the previous section, we simulated traffic using a two state (ON-OFF) Markov Fluid model. In that model, a continuous time Markov chain drives the process. When the chain is in the ON state, the workload is produced at constant rate h_0, and when it is in the OFF state no workload is produced ($h_1 = 0$). Denoting by Q the Markov chain infinitesimal generator, by $\boldsymbol{\pi}$, its invariant distribution, and by H, the diagonal matrix with the rates h_i in the diagonal. The effective bandwidth for a source of this type is [9][3]:

$$\alpha(s,t) = \frac{1}{st} \log\left\{\boldsymbol{\pi} \exp\left[(Q + Hs)t\right] \mathbf{1}\right\} \tag{15}$$

where $\mathbf{1}$ is a column vector of ones.

In our simulations we generated three hundred traffic traces of length T samples, with the following Q-matrix:

$$Q = \begin{pmatrix} -0.02 & 0.02 \\ 0.1 & -0.1 \end{pmatrix}$$

The effective bandwidth for this process calculated through equation (15).

For each traffic trace we estimated EB using the following procedure. We divided the trace in blocks of length t and constructed the following sequence:

$$\tilde{X_k} = \sum_{i=(k-1)t}^{kt} x(i) \qquad 1 \leq k \leq \lfloor T/t \rfloor$$

where $x(i)$ is the amount of traffic arrived between samples and $\lfloor c \rfloor$ denotes the largest integer less than or equal to c.

EB can then be estimated by the time average proposed in [7], [8] as

$$\alpha_n(s,t) = \frac{1}{st} \log \left[\frac{1}{\lfloor T/t \rfloor} \sum_{j=1}^{\lfloor T/t \rfloor} e^{s\tilde{X}_j} \right] \tag{16}$$

where $n = \lfloor T/t \rfloor$. This is merely an implementation of the time average estimator in equation (4) based on a finite length traffic trace. When the values of t verify that $t \ll T$, the number of replications of the increment process within the trace is good enough to get a good estimation.

In order to find the operating point (s^*,t^*) of the theoretical Markov model, and its estimator (s_n^*,t_n^*) for each simulated trace, we solve the *inf sup* optimization problem of equation (2). In our case $\alpha(s,t)$ will be the previous theoretical equation (15) for the Markovian source or the $\alpha_n(s,t)$ estimated for each trace. The numerical solution has two parts. First, for a fixed t we find the $s^*(t)$ that maximize $g(s,t)$ as a function of s. It can be shown that $st\alpha(s,t)$ is a convex function of s. This convexity property is used to solve the previous optimization problem, that is reduced to find the maximum difference between a convex function and a linear function of s, and it can be done very efficiently. After the $s^*(t)$ is found for each t, it is necessary to minimize the function $g(s^*(t),t)$ and find t^*. For this second optimization problem, there are no general properties that let us make the search algorithm efficient and a linear searching strategy is used.

An important issue is to develop a confidence region for (s^*,t^*). We simulated 300 traces of length 100000(T) samples and constructed, for each simulated trace indexed by $i = 1,\ldots,K$ the corresponding estimator $(s_n^*(i), t_n^*(i))$. By theorem 1 the vector $\sqrt{n}((s_n^*,t_n^*) - (s^*,t^*))$ is asymptotically bivariate normal with $(0,0)$ mean and covariance matrix Σ. We estimated the matrix Σ using the empirical covariances of the observations $\{\sqrt{n}((s_n^*(i),t_n^*(i)) - (s^*(i),t^*(i)))\}_{i=1,\ldots,K}$ given

by:

$$\Sigma_K = \frac{n}{K}\begin{pmatrix} \sum_{i=1}^{K}\left(s_n^*(i)-\overline{s_n^*}\right)^2 & \sum_{i=1}^{K}\left(s_n^*(i)-\overline{s_n^*}\right)\left(t_n^*(i)-\overline{t_n^*}\right) \\ \sum_{i=1}^{K}\left(s_n^*(i)-\overline{s_n^*}\right)\left(t_n^*(i)-\overline{t_n^*}\right) & \sum_{i=1}^{K}\left(t_n^*(i)-\overline{t_n^*}\right)^2 \end{pmatrix}$$

where $\overline{s_n^*} = \frac{1}{K}\sum_{i=1}^{K} s_n^*(i)$ and $\overline{t_n^*} = \frac{1}{K}\sum_{i=1}^{K} t_n^*(i)$.

Therefore, we can say that approximately: $(s_n^*, t_n^*) \approx N\left((s^*,t^*), \frac{1}{n}\Sigma_K\right)$, from where a level α confidence region: $R_\alpha = (s_n^*, t_n^*) + \frac{A_K^t B\left(\mathbf{0}, \sqrt{\chi_\alpha^2(2)}\right)}{\sqrt{n}}$, being A_K the matrix that verifies $A_K^t A_K = \Sigma_K$, while $B(x,r)$ is the ball of center x and radius r.

To verify our results, we calculated the theoretical operating point (s^*, t^*) and simulated another 300 traces independent of those that were used to estimate Σ_K. We constructed then the 95% confidence region. If the results are right, approximately 95% of the times, (s^*, t^*) must fall inside that region, or equivalently and easier to check, approximately 95% of the simulated (s_n^*, t_n^*) must fall inside the region $R = (s^*, t^*) + \frac{1}{\sqrt{n}} A_K^t B\left(\mathbf{0}, \sqrt{\chi^2_{0.05}(2)}\right)$. Numerical results, plotted in figure 1 (left), verify that the confidence level is attained, 95.33% of the estimated values fall inside the predicted region.

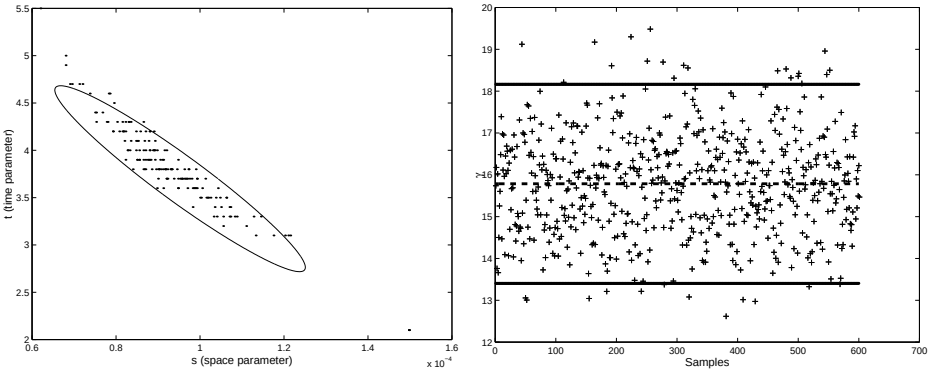


Fig. 1. Estimated operating points (left), γ_n and theoretical γ (right) and its confidence regions

QoS parameters estimation. We estimate the link operating point in order to estimate loss probability. As was said in section 2.2, if we have an EB estimator that verifies the hypotheses of theorem 1, then

$$\gamma_n = \inf_t \sup_s ((b+ct)s - st\alpha_n(s,t)) \tag{17}$$

is a consistent estimator and has CLT properties. From this estimator loss probability could be approximated by

$$q_n = P_n(Q_N > B) \approx \exp^{-N\gamma_n} \tag{18}$$

where Q_N is the queue size and N is the number of sources. Figure 1 (right) shows the estimations of γ_n for 600 simulated traces, its theoretical value and its confidence interval. Numerical results show that in this case 94.8% of the values fall in the 95% confidence interval.

3 End-to-end QoS, the fictitious network analysis

3.1 Introduction

As we have explained in the previous section, using Large Deviations Theory and in the many sources asymptotic Wischik [6] proves the following formula (called *inf sup* formula) for the overflow probability:

$$\log \mathbf{P}(Q_N > B) \approx -\inf_{t \geq 0} \sup_{s \geq 0}((B + Ct)s - Nst\alpha(s,t))$$

where Q_N represents the stationary amount of work in the queue, C is the link capacity, B is the buffer size and N is the number of incoming multiplexed sources of effective bandwidth $\alpha(s,t)$.

Wischik also shows in [13] that in the many sources asymptotic regime the aggregation of independent copies of a traffic source at the link output and the aggregation of similar characteristics at the link input, have the same effective bandwidth in the limit when the number of sources goes to infinity. This result allows to evaluate the end to end performance of some kind of networks like "in-tree" ones. Unfortunately this analysis can not be extended to networks with a general topology.

A slightly different asymptotic with many sources and small buffer characteristics was proposed by Ozturk, Mazumdar and Likhanov in [4]. They consider an asymptotic regime defined by N traffic sources, link capacity increasing proportionally with N but buffer size such that $\lim \frac{B(N)}{N} \to 0$. In their work they calculate the rate function for the buffer overflow probability and also for the end to end loss ratio. This last result can be used to evaluate the end to end QoS performance in a network backbone in contrast with the Wischick result explained before, where it is necessary to aggregate at each link N i.i.d. copies of the previous output link.

Ozturk et al. also introduce the "fictitious network" model. The fictitious network is a network with the same topology than the real one, but where each flow aggregate goes to a link on its path without being affected by the upstream links until that link. The fictitious network analysis is simpler and so, more adequate to on-line performance evaluation and traffic engineering. Ozturk et al. show that the fictitious network analysis overestimates the overflow probability. In this work we analyze when, for an interior network link, the overflow probability calculated using the fictitious network is equal to the overflow probability of the real network.

In the next section we summarize Ozturk et al. main results.

3.2 Many sources and small buffer asymptotic performance model

Ozturk, Mazumdar and Likhanov work. Consider a network of L links which is accessed by M types of independent traffic. Consider a discrete time fluid FIFO model where traffic arrives at time $t \in Z$ and is served immediately if buffer is empty and is buffered otherwise. Each link k has capacity NC_k and buffer size $B_k(N)$ where $B_k(N)/N \to 0$ with $N \to \infty$. Input traffic of type m=1,...,M, denoted $X^{m,N}$ is stationary and ergodic and has rate $X_t^{m,N}$ at time t (workload at time t of N sources of type m).

Let $\mu_m^N = \mathbf{E}(X_0^{m,N})/N$ and $X^{m,N}(t_1,t_2) = \sum_{t=t_1}^{t_2} X_t^{m,N}$. We assume that $\mu_m^N \underset{N\to\infty}{\to} \mu_m$ and $X^{m,N}(0,t)/N$ satisfies the following Large Deviation Principle (LDP) with *good rate function* $I_t^{X^m}(x)$:

$$-\inf_{x\in\Gamma^o} I_t^{X^m}(x) \leq \liminf_{N\to\infty} \frac{1}{N}\log\mathbb{P}\left(\frac{X^{m,N}(0,t)}{N}\in\Gamma\right) \tag{19}$$

$$\leq \limsup_{N\to\infty} \frac{1}{N}\log\mathbb{P}\left(\frac{X^{m,N}(0,t)}{N}\in\Gamma\right) \leq -\inf_{x\in\overline{\Gamma}} I_t^{X^m}(x) \tag{20}$$

where $\Gamma \subset \mathbb{R}$ is a Borel set with interior Γ^o and closure $\overline{\Gamma}$ and $I_t^{X^m}(x) : \mathbb{R} \to [0,\infty)$ is a continuous mapping with compact level sets. We also assume the following technical condition: $\forall\ m$ and $a > \mu_m$,

$$\liminf_{t\to\infty} \frac{I_t^{X^m}(at)}{\log t} > 0$$

Type m traffic has a fixed route without loops and its path is represented by the vector $\mathbf{k}^m = (k_1^m, ..., k_{l_m}^m)$, where $k_i^m \in (1,..,L)$. The set $\mathcal{M}_k = \{m : k_i^m = k, 1 \leq i \leq l_m\}$ denotes the types of traffic that goes through link k. To guarantee system stability it is assumed that

$$\sum_{m\in\mathcal{M}_k} \mu_m < C_k \tag{21}$$

The main result of Ozturk et al. work is the following theorem.

Theorem 2. *Let $X_{k,t}^{m,N}$ be the rate of type m traffic at link k at time t. There exist a continuous function $g_k^m : \mathbb{R}^M \to \mathbb{R}$ relating the instantaneous input rate at link k for traffic type m to all of the instantaneous external input traffic rates such that:*

$$\frac{X_{k,0}^{m,N}}{N} = g_k^m\left(\frac{X_0^{1,N}}{N}, ..., \frac{X_0^{M,N}}{N}\right) + o(1) \tag{22}$$

The buffer overflow probabilities are given by:

$$\lim_{N\to\infty} \frac{1}{N} \log P(\textit{overflow in link } k) = -\mathbf{I}_k =$$
$$-\inf\left\{ \sum_{m=1}^{M} I_1^{X^m}(x_m) : x = (x_m) \in \mathbb{R}^M, \sum_{m=1}^{M} g_k^m(x) \geq C_k \right\} \tag{23}$$

In (22), $o(1)$ verifies that $\lim_{N\to\infty} o(1) = 0$ since $\frac{B_k(N)}{N} \underset{N\to\infty}{\to} 0$. The function $g_k^m(x)$ is constructed in the proof of the theorem. Ozturk et al. prove that the continuous function relating the instantaneous input rate at link i for traffic m to all of the instantaneous external input traffic rates is the same function relating these variables in a no buffers network. The function relating the instantaneous output rate at link i for traffic m to all of the instantaneous input traffic rates at this link is:

$$f_i^m(x, C_i) = \frac{x_m C_i}{\max(\sum_{j\in\mathcal{M}_i} x_j, C_i)} \tag{24}$$

In a feed-forward network the function $g_k^m(x)$ can be written as composition of the functions of type (24) in a recursive way. Using equation (24) the buffer overflow probability can be calculated for any network link, by solving the optimization problem of equation (23). We need to know the network topology, the link capacities and, for each arrival traffic type m, the rate functions $I_1^{X^m}$.

Ozturk et al. define also the total (end to end) loss ratio as the ratio between the expected value of lost bits at all links along a route and the mean of input traffic in bits, for stream m identified by X^m. With the previous definition they find the following asymptotic for the loss ratio $\mathbf{L}^{m,N}$:

$$\lim_{N\to\infty} \frac{1}{N} \log \mathbf{L}^{m,N} = -\min_{k\in k^m} \mathbf{I}_k \tag{25}$$

The main problem of this approach is that the optimization problem of equation (23) could be very hard to solve for real-size networks. The calculation of the function $g_k^m(x)$ is recursive and so, when there are many links it becomes complex. In addition, the virtual paths can change during the network operation. Therefore, it is necessary to recalculate on-line the function $g_k^m(x)$. To solve equation (23), it is also necessary to optimize a nonlinear function under nonlinear constraints. In order to simplify this problem, Ozturk et al. introduce the "fictitious network" concept, that is simpler and gives conservative results. In the next section we find conditions to assure that there is no overestimation in the calculus of the link overflow probability in the fictitious network analysis. We also find a bound for the error (difference between the rate function calculated for the real network and the fictitious one) in those cases where the previous condition is not satisfied.

The aim of our work is to define an admission control mechanism. Such a mechanism is simple a set of rules to accept or reject a flow that intend to access

the network. This can be done by defining an acceptance region, i.e. which is the set of flows that can access the network. In [4] an acceptance region based on end-to-end QoS guarantees, is defined. This acceptance region is the traffic mix that can flow through the network without QoS violation. Assume that $X^{m,N}$ is the sum of Nn_m i.i.d. process. More formally, the acceptance region noted by $\mathcal{D}$ is the mix or collection $\{n_m\}_{m=1}^{M}$ of sources which can be flowing through the network while the QoS (loss ratio) for each class is met, that is:

$$\mathcal{D} = \{(n_m), m = 1, ..., M : \lim_{N\to\infty} \frac{1}{N} \log \mathbf{L}^{m.N} < -\gamma_m\} \quad \text{with} \quad \gamma_m > 0 \quad (26)$$

We will concentrate then in the estimation of this acceptance region. We aim not only to do it in a efficient way but also in a simple one in order to apply it on-line.

3.3 Fictitious network analysis

We analyze an interior network link k under the same assumptions that in Ozturk et al. work. $\mathcal{M}$ is the set of traffic types that access the network and $\mathcal{M}_i$ is the set of traffic types that go through link i. We suppose that the network is feed-forward, this means that each traffic type has a fixed route without loops. In the real network, the link k overflow probability large deviation function (or rate function) is given by:

$$I_k^R = \inf \left\{ \sum_{i\in\mathcal{M}} I_1^{X^i}(x_i) : x = (x_i)_{i\in\mathcal{M}}, \sum_{i\in\mathcal{M}} g_k^i(x) \geq C_k \right\} \quad (27)$$

In the fictitious network this function is given by

$$I_k^F = \inf \left\{ \sum_{i\in\mathcal{M}_k} I_1^{X^i}(x_i) : x = (x_i)_{i\in\mathcal{M}_k}, \sum_{i\in\mathcal{M}_k} x_i \geq C_k \right\} \quad (28)$$

In the following it is assumed that each traffic type is an aggregate of N $i.i.d$ sources. This implies that each rate function $I_1^{X^i}$ is convex and $I_1^{X^i}(\mu_i) = 0$ for all i. Then, (27) and (28) are convex optimization problems under constraints. The second one has the advantage that the constraints are linear and there are well known fast methods to solve it. The functions $I_1^{X^i}$ are continuous, so we solve the following problems corresponding to the real and fictitious network respectively.

$$P_R \begin{cases} \min \sum_{i\in\mathcal{M}} I_1^{X^i}(x_i) \\ \\ \sum_{i\in\mathcal{M}} g_k^i(x) \geq C_k \end{cases} \qquad P_F \begin{cases} \min \sum_{i\in\mathcal{M}_k} I_1^{X^i}(x_i) \\ \\ \sum_{i\in\mathcal{M}_k} x_i \geq C_k \end{cases}$$

Definition 1. *Consider two optimization problems*

$$P_1 \begin{cases} \min f_1(x) \\ x \in D_1 \end{cases} \quad \textit{and} \quad P_2 \begin{cases} \min f_2(x) \\ x \in D_2 \end{cases}$$

P_2 is called a relaxation of P_1 if $D_1 \subseteq D_2$ and $f_2(x) \leq f_1(x)$, $\forall x \in D_1$.

Proposition 1. *P_F is a relaxation of P_R.*

Proof. Since the functions $I_1^{X^i}$ are non negatives, it is clear that $\sum_{i \in \mathcal{M}_k} I_1^{X^i}(x_i) \leq \sum_{i \in \mathcal{M}} I_1^{X^i}(x_i)$ $\forall x = (x_i)_{i \in \mathcal{M}}$. Then, we have to prove that

$$\left\{ x : \sum_{i \in \mathcal{M}} g_k^i(x) \geq C_k \right\} \subseteq \left\{ x : \sum_{i \in \mathcal{M}_k} x_i \geq C_k \right\}$$

By definition, $g_k^i(x) = 0 \ \forall \ i \notin \mathcal{M}_k$ and $g_k^i(x) \leq x_i \ \forall \ i \in \mathcal{M}_k$ (since g_k^i can be written as composition of functions of type (24)) then

$$\sum_{i \in \mathcal{M}} g_k^i(x) = \sum_{i \in \mathcal{M}_k} g_k^i(x) \leq \sum_{i \in \mathcal{M}_k} x_i$$

and therefore $\sum_{i \in \mathcal{M}_k} g_k^i(x) \geq C_k$, implies $\sum_{i \in \mathcal{M}_k} x_i \geq C_k$.

Remark 4. If an optimum of the fictitious problem P_F verifies the real problem P_R constraints and the objective functions take the same value at this point, then it is an optimum of the real problem too.

The following theorem gives conditions over the network to assure that link k overflow probability rate function for the real and for the fictitious network are equal ($E = I_k^R - I_k^F = 0$). Since the network is feed forward, it is possible to establish an order between the links. We say that link i is "previous to" or "less than" link j if for one path, link i is found before than link j in the flow direction.

Theorem 3 (Sufficient Condition). *If $\widetilde{x} = (\widetilde{x}_i)_{i \in \mathcal{M}_k}$ is optimum for P_F, and the following condition is verified for all links i less than k:*

$$C_k - \sum_{j \in \mathcal{M}_k \setminus \mathcal{M}_i} \mu_j \leq C_i - \sum_{j \in \mathcal{M}_i \setminus \mathcal{M}_k} \mu_j \quad \forall \ i < k \tag{29}$$

then x^* defined by:

$$(x^*)_i = \begin{cases} \widetilde{x}_i & \textit{if } \ i \in \mathcal{M}_k \\ \mu_i & \textit{if } \ i \notin \mathcal{M}_k \end{cases}$$

is optimum for P_R.

Proof. See [14].

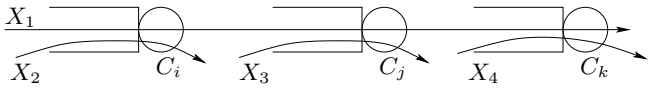


Fig. 2.

Example 1. Consider a network like in figure 2. We analyze the overflow probability at link k.
If condition (29) is attained for link k, then $E = I_k^R - I_k^F = 0$. This condition is:

$$\begin{cases} C_k - \mu_4 \leq C_i - \mu_2 \\ \\ C_k - \mu_4 \leq C_j - \mu_3 \end{cases}$$

Sufficient condition in terms of available bandwidth.

Definition 2. *For a traffic type m in a link j, it is defined the available bandwidth ABW_j^m as the difference between the link j capacity and the mean value of the transmission rate of the other traffic types in j.*

In terms of the previous definition, the theorem condition (29) assures that the overflow probability rate function at link k on real and fictitious network are the same if for all link $j < k$, and for all m traffic type in $\mathcal{M}_j \cap \mathcal{M}_k$, $ABW_j^m > ABW_k^m$. This condition is represented in figure 3 for a simple network with two links.

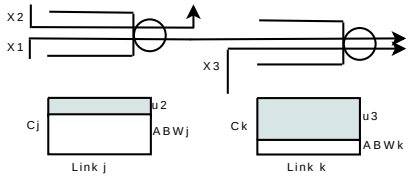


Fig. 3. Sufficient condition in terms of available bandwidth

Sufficient but not necessary condition. The theorem condition (29) is sufficient to assure that the overflow probability rate function at link k on real and fictitious networks are the same, but it is not a necessary condition. In fact, if $\widetilde{x}$ is optimum for the fictitious problem, and if x^* defined as:

$$(x^*)_i = \begin{cases} \widetilde{x}_i & \text{si } \ i \in \mathcal{M}_k \\ \mu_i & \text{si } \ i \notin \mathcal{M}_k \end{cases} \tag{30}$$

satisfies the real problem constraints, then x^* is optimum for the real problem. If x^* verifies the following condition

$$\sum_{j \in \mathcal{M}_i} (x^*)_j \leq C_i \quad \forall\, i < k \tag{31}$$

it also verifies the real problem constraints and therefore is optimum for the real problem.

Therefore, in the case that the theorem condition is not fulfilled, if we found $\widetilde{x}$ optimum for the fictitious problem, then is easy to check if the rate functions are equal or no. It is enough to check (31), where x^* is defined in (30).

Error bound. Since the functions $I_1^{X^i}$ are non negatives, it is clear that the rate function for the real problem is always greater than the fictitious one. Then the error $E = I_k^R - I_k^F$ is always non negative. This implies that the fictitious network overestimates the overflow probability. We are interested in finding an error bound for the overestimation of the fictitious analysis when conditions (29) and (31) are not satisfied. A simple way to get this bound is to find a point x which verifies the real problem constraints. In this case, we have that:

$$E = I_k^R - I_k^F \leq \sum_{i \in \mathcal{M}} I_1^{X^i}(x_i) - \sum_{i \in \mathcal{M}_k} I_1^{X^i}(\widetilde{x}_i)$$

To assure that x verifies the real problem constraints, we have already seen that it is enough to show that $\sum_{j \in \mathcal{M}_i} x_j \leq C_i \ \forall \ i < k$ and $\sum_{j \in \mathcal{M}_k} x_j \geq C_k$. Therefore, we have to solve this inequalities system. It can be seen that the optimum of the fictitious problem is in the boundary of the feasible region ($\sum_{i \in \mathcal{M}_k} \widetilde{x}_i = C_k$). Since we are looking for a point near the optimum of the fictitious problem in the sense that the error bound be as small as possible, we solve the following system:

$$\begin{cases} \sum_{j \in \mathcal{M}_i} x_j \leq C_i & \forall \ i < k \\ \\ \sum_{j \in \mathcal{M}_k} x_j = C_k \end{cases} \tag{32}$$

For the interesting cases, where there are losses at link k, this system always has a solution. In the following an algorithm to find a solution of this system is defined. We define the following point:

$$(x^*)_j = \begin{cases} \widetilde{x_j} & \text{if } \ j \in \mathcal{M}_k \\ 0 & \text{if } \ j \notin \mathcal{M}_k \end{cases}$$

If x^* verifies the conditions (32), we find a point that verifies the real problem constraints. In some cases this is not useful because $I_1^{X^j}(0) = \infty$ and we have that the error bound is infinite. If $P(X_1^{j,N} \leq 0) \neq 0$, the function $I_1^{X^j}(0) < \infty$ and a finite error bound is obtained. If x^* is not solution for system (32), then we redefine (by some small value) the coordinates where $\sum_{j \in \mathcal{M}_i} x_j > C_i$ in such a way that $\sum_{j \in \mathcal{M}_i} x_j = C_i$. The second equation must be verified too and, since some coordinates were reduced, others coordinates have to increase to get the total sum equal to C_k. Since the system is compatible, following this method, a solution is always found. There is no guarantee that the solution given by this method minimizes the error bound. However, this method has a very simple

implementation and gives reasonable error bounds as we can see in the numerical examples of the last section.

3.4 End-to-End Loss Ratio Evaluation

In the previous section we found sufficient conditions to assure that results on the fictitious and on the real network analysis coincide for an interior link. However, to define an admission control mechanism based on the end-to-end quality of service, we must find a condition that guarantees that the end-to-end loss ratio coincides for both networks. A natural answer is that the sufficient condition found in theorem 3 must be verified for all links in the considered path. However, as equation 25 suggest, we will show that this is not necessary since it is enough that the sufficient condition is verified for the link with minimum overflow probability rate function. This link must be then identified, and clearly we aim to do it within the fictitious network context. We must then be sure that the link with minimum rate function is the same for the real and the fictitious network. In the sequel we address this two issues.

Proposition 2. *Let k_f be the link with minimum overflow probability rate function in the fictitious network for traffic type m: $\overline{I}_{k_f} = \min_{k_i \in \mathbf{k}^m} \overline{\mathbf{I}}_{k_i}$*

If the conditions of theorem 3 are verified for link k_f, the minimum overflow probability rate function for traffic type m in the real network is also attained at link k_f.

Proof. See [14].

Proposition 3. *Let k be the link where $I_k = -\min_{k \in k^m} \mathbf{I}_k^m$ for the real network , i.e. the link where the minimum rate function of traffic type m is attained. Let $\overline{I}_k$ be the rate function of the same link k for the fictitious network. If the sufficient conditions of theorem 3 are verified for link k then $\mathbf{L}^m = \overline{\mathbf{L}}^m$, i.e. the end-to-end loss ratio for real and fictitious network coincide.*

Proof. See [14].

Remark 5. Previous propositions show that to evaluate the end-to-end loss ratio $\mathbf{L}^m$, it is enough that sufficient conditions of theorem 3 are verified by the link k where the minimum rate function of traffic type m path is attained. In this case, it results that $\mathbf{L}^m = \overline{\mathbf{L}}^m = \overline{I}_k$. If sufficient conditions are not verified, then the error bound obtained for the one link case can be applied.

3.5 Rate function estimation

In previous sections we show how to evaluate the end-to-end loos ratio in terms of the rate function for the fictitious network. In order to implement an on-line admission control based on this information, we must be able to accurately estimate the corresponding rate function. In this section we analyze how this estimation can be done using traffic traces of the input traffic.

Let $X_k^{m,N}(0,t)$ be the traffic type m workload at link k during the time interval $(0,t)$. We suppose that $X_k^{m,N}$ is the sum of $N\rho_m$ independent sources of type m:

$$X_k^{m,N}(0,t) = \sum_{i=1}^{N\rho_m} \widetilde{X}_k^{m,i}(0,t)$$

In this case, the instantaneous rate of traffic type m at time t is given by:

$$X_{k,t}^{m,N} = \sum_{i=1}^{N\rho_m} \widetilde{X}_{k,t}^{m,i}$$

Given the stationarity of the traffic, we can replace the t-index by 0 and for simplicity we omit the link index k. Then the instantaneous rate of total input traffic at link k is:

$$Z_0^N = \sum_{m\in\mathcal{M}_k} X_0^{m,N} = \sum_{m\in\mathcal{M}_k} \sum_{i=1}^{N\rho_m} \widetilde{X}_0^{m,i} = \sum_{j=1}^{N} \widetilde{Z}^j$$

where the variables $\widetilde{Z}^j$ are independent and identically distributed (*iid*) random variables. Each variable $\widetilde{Z}^j$ has the distribution of a mix of the variables $\widetilde{X}_0^{m,i}$ (given by the proportions ρ_m of each traffic type m present at link k). This means that instantaneous rate of input traffic at link k is the sum of N *iid* random variables and Cramer theorem (see for example [6]) can be applied. The variable $\frac{Z_0^N}{N}$ verifies then a large deviation principle with rate function:

$$I_t^Z(x) = \sup_{\theta\geq 0}\{\theta x - \Lambda(\theta)\} = \sup_{\theta\geq 0}\{\theta x - \log \mathbf{E}\left(e^{\theta\widetilde{Z}^1}\right)\} \tag{33}$$

Given the rate function of the LDP, $I_t^Z(x)$, we can calculate I_k^F:

$$\begin{aligned} I_k^F &= \inf\left\{I^Z(z) : z \geq C_k\right\} = \inf_{z\geq C_k} \sup_{\theta\geq 0}\left\{\theta z - \Lambda(\theta)\right\} \\ &= \sup_{\theta\geq 0}\left\{\theta C_k - \Lambda(\theta)\right\} \end{aligned} \tag{34}$$

Before solving the optimization problem 34, we must calculate or estimate $\Lambda(\theta)$. If some model is assumed for the traffic, $\Lambda(\theta)$ can be calculated analytically. In case no model is assumed as in our case, it must be estimated from measurements i.e. from traffic traces. A possible and widely used approach [7, 8] is to estimate the expectation as a temporal average of a given traffic trace $\{\widetilde{Z}^N(t)\}_{t=1:n}$:

$$\mathbf{E}\left(e^{\theta\widetilde{Z^1}}\right) = \mathbf{E}\left(e^{\theta\frac{Z_0^N}{N}}\right) \approx \frac{1}{n}\sum_{t=1}^{n} e^{\theta Z^N(t)/N}$$

Then $\Lambda(\theta)$ can be estimated by $\Lambda_n(\theta) = \log\left(\frac{1}{n}\sum_{t=1}^{n} e^{\theta Z^N(t)/N}\right)$

Now, the rate function I_k^F can be estimated as: $I_{k,n}^F = \sup_{\theta \geq 0} \{\theta C_k - \Lambda_n(\theta)\}$

However it remains unclear how good is this estimation. We will show that if $\Lambda_n(\theta)$ is a *good* estimator of $\Lambda(\theta)$, then $I_{k,n}^F$ is also a *good* estimator for the rate function I_k^F.

Theorem 4. *If $\Lambda_n(\theta)$ is an estimator of $\Lambda(\theta)$ such that both are C^1 functions and:*

$$\Lambda_n(\theta) \underset{n}{\longrightarrow} \Lambda(\theta) \qquad \frac{\partial}{\partial\theta}\Lambda_n(\theta) \underset{n}{\longrightarrow} \frac{\partial}{\partial\theta}\Lambda(\theta)$$

where the convergence is almost surely and uniformly over bounded intervals, then $I_{k,n}^F$ is a consistent estimator of I_k^F. Moreover, if a functional Central Limit Theorem (CLT) applies to $\Lambda_n - \Lambda$, i.e, $\sqrt{n}\left(\Lambda_n(\theta) - \Lambda(\theta)\right) \underset{n}{\Longrightarrow} G(\theta)$, where $G(\theta)$ is a continuous gaussian process, then: $\sqrt{n}\left(I_{k,n}^F - I_k^F\right) \underset{n}{\Longrightarrow} N(0,\sigma)$, where $N(0,\sigma)$ is a centered normal distribution with variance σ.

Proof. See [14].

From the previous analysis we conclude that the rate function and then the admission control region can be accurately estimated from traffic traces in a simple way. As we claimed before, this can be used in the definition of an admission control mechanism based in the end-to-end quality of service expected by the traffic.

3.6 Numerical example

In this section we present a numerical example to validate our results. Additional numerical examples can be found in [14].

There are many issues that could be evaluated to analyze the performance of an admission control mechanism. However, since the overall performance of our proposition depends on how accurate are the results obtained when the fictitious network model is considered, we will concentrate here only in this aspect.

Example 2. Consider a network like in figure 4. We analyze the overflow probability at link k, assuming that $C_i > C_k$.

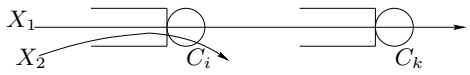


Fig. 4. Example 2-Network topology

If condition (29) is attained for link k, then $E = I_k^R - I_k^F = 0$.

This condition is: $C_k \leq C_i - \mu_2$.

If this condition is not satisfied, since $\widetilde{x} = C_k$ is optimum for P_F, we first verify if $x^* = (C_k, \mu_2)$ is optimum for (P_R). It is sufficient to show that x^* verifies the real problem constraints, i.e: $\begin{cases} C_k + \mu_2 \leq C_i \\ C_k = C_k \end{cases}$

If $C_k + \mu_2 > C_i$, we look for $x^* = (x_1^*, x_2^*)$ that verifies $\begin{cases} x_1^* + x_2^* \leq C_i \\ x_1^* = C_k \end{cases}$

It is possible to choose $x_1^* = C_k$ and $x_2^* = C_i - C_k > 0$ resulting in the following error bound:

$$E \leq I_1(C_k) + I_2(C_i - C_k) - I_1(\widetilde{x}_1) = I_2(C_i - C_k) \tag{35}$$

In the following numerical example, we calculate the overflow probability rate function for the real and fictitious network. Let $C_i = 16kb/s$ per source and C_k growing from 4 to $15.5kb/s$ per source. All traffic sources are on-off Markov processes. For X_1, the bit rate in the on state is $16kb/s$, and average times are $0.5s$ in the on state and $1.5s$ in the off state. For X_2, the bit rate in the on state is $16kb/s$, and average times are $1s$ in the on state and $1s$ in the off state. Since $\mu_1 = 4kb/s$ the stability condition is $C_k > \mu_1 = 4kb/s$. Using these values, the sufficient condition (29) is, $C_k \leq 8kb/s$. Figure 5 shows that while this condition is satisfied both functions match, but after $C_k \geq 8kb/s$ they separate. Figure 5 also shows the overestimation error ($E = I_k^R - I_k^F$) and the error bound (35) described before. In this case, the error bound is exactly the error.

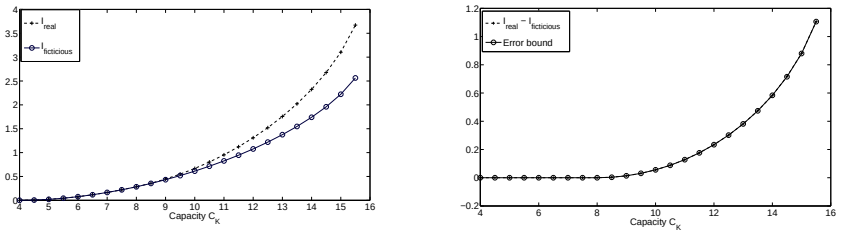


Fig. 5. Example 2-Rate functions and error bound

4 End-to-end QoS prediction based on active measurements and statistical learning

4.1 Introduction

The many sources and small buffer asymptotic analyzed in the previous section, can only be applied to analyze an end-to-end path in a backbone network. If the end points are end users this asymptotic cannot be applied because the path goes through the backbone but also through the access network where the many sources asymptotic is not valid. Therefore, a different approach must be applied if the access control mechanism must take a decision based on end-to-end QoS.

In this section we analyze another approach based on measurements and statistical learning in order to evaluate the end-to-end QoS parameters seen by applications like a video on demand service.

A possible measurement technique for such tool is to send the application traffic (a video for example) and to measure the video QoS parameters at the receiver. However, in many cases these application flows may have bandwidth requirements that are not negligible compared with links capacity. This technique could overload a congested link degrading the QoS perceived by clients using the system. This QoS degradation can be tolerated during short periods but the previous methodology cannot be used if the operator requires a permanent or frequent network monitoring.

Other measurement techniques estimate the QoS parameters seen by an application using light probe packets and without considering the particular characteristics of the application. These probe packets do not overload the network but this procedure assumes, for example, that the delay of a specific application can be approximated by the probe packets delay. This previous assumption is not always true because the QoS parameters depend on the statistical behavior of each traffic. Therefore, in many cases, this kind of estimation yields inaccurate results.

We propose a methodology that is an intermediate point between both approaches (to send a multimedia flow during long periods or to send light probe packets during short periods) and provides an accurate estimation of QoS parameters seen by an application without overloading the network during long periods.

Our goal is to learn the relation between the QoS parameters seen by an application and the probe packets interarrival times statistic. This statistic characterizes the network state. Once the model is learned, in order to predict the QoS parameters, it is necessary only to send light probe packets.

More formally, we consider the regression model

$$Y = \Phi(X) + \varepsilon \tag{36}$$

where X, Y and ε are random variables. The random variable X is an estimation of the state of network path, the response Y is the QoS parameter seen by the application (delay, jitter, loss rate) and ε is a centered random variable which represents an error, where ε and X are independent.

The previous formulation evidences two problems to be addressed in this work. First, it is necessary to find an accurate estimation of the state of the network path (the variable X). Second, it is necessary to estimate the function Φ. We propose to estimate this function learning Φ from samples of the random variables Y and X.

In order to estimate the state of the network path, we analyze a functional random variable X that is the empirical distribution of the probe packets interarrival times.

In order to estimate the function Φ we propose a statistical learning approach based on the Nadaraya-Watson estimator. Nadaraya-Watson, first introduced for real data [15], is used in this work mainly for functional regression [16]. We propose also an extension of theoretical results about the Nadaraya-Watson functional estimator in a nonstationary context. This non-parametric approach

is based on mapping data obtained from probe packets and any QoS parameter seen by an application.

4.2 Problem formulation and proposed solution

We first consider the case of a path with a single link. The multilink case is discussed later. We assume that the cross traffic, the link capacity and the buffer size are unknown. The QoS parameter seen by the application is called Y and it is a function of the link and traffic characteristics: $Y = F(X_t, V_t, C, B)$

where X_t is the cross traffic stochastic process, V_t is the video or other application traffic stochastic process, C is the link capacity and B is the buffer size. The link capacity C and the buffer size B are not known but it is assumed that both have constant values during the monitoring process. As the goal is to estimate a QoS parameter over the known process V_t (a video sequence for example), V_t can be considered as an input to our problem. Taking into account the previous considerations, we can say that $Y = F(X_t)$. At the end of this section we discuss these assumptions about C, B, and V_t.

The previous formulation pose two different problems that should be addressed. On one hand the estimation of the function that relates the cross traffic and the QoS parameter and on the other the estimation of the cross traffic process X_t. In order to take into account the multilink case, the last estimation is what we call the estimation of the state of the network path.

In order to estimate the cross traffic we send probe packets from the user equipment and measure the interarrival times. When two consecutive probe packets are queued in the same busy period at the link queue, as shown in figure 6 (left), the interarrival time is equal to $\frac{X_i}{C} + \frac{K}{C}$, where X_i is the amount of cross traffic that arrived at the queue between probe packets i and $i+1$, K is the probe packets size and C is the link capacity. Then, during the busy periods, the interarrival times are proportional to the cross traffic volume at least up to a constant.

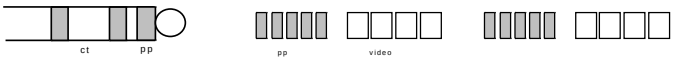


Fig. 6. Probe packets, probe video (video) and cross traffic (ct)

In the case where the packet $i+1$ is queued after the packet i leaves the queue, as we infer the cross traffic volume from the values t^i_{out}, we can conclude that there is a cross traffic volume larger than the real one.

Baccelli et al. [17] present a rigorous probabilistic approach to active probing methods for cross traffic estimation. They analyze the system identifiability and show that different cross traffic types can give rise to the same sequence of observed probe delays. Therefore, it is not always possible to determine the distribution of any desired aspect of the cross traffic using probes. However, we are not looking for an accurate estimator of the cross traffic. We are actually

looking for an estimation of Y. Therefore, our interest is only in finding an estimator that allow us to distinguish between possible states of the network. This state is represented by a variable X that is a function of the probe packets interarrival times.

We will estimate the function Φ in the regression model of equation 36 from the observations of the pairs (X, Y).

We divide the experiment in two phases. The first phase is called the learning phase. In the learning phase we send a burst of probe packets. The probe packets interdeparture time is a fixed value t_{in} and the packets have a fixed size K. Immediately after the probe packets we send a video sequence training sample during a short period.

This procedure is repeated periodically sending the probe packets and the video sequence alternatively as shown in figure 6 (right).

We build the variable X_j by measuring for each experiment j the interarrival times of the probe packets burst. We also measure the QoS parameter Y_j of the corresponding video sequence and we have a pair (X_j, Y_j) for each experiment. The problem is how to estimate the function $\Phi : \mathcal{D} \to \mathbb{R}$ by $\widehat{\Phi}$ from these observations, where $X \in \mathcal{D}$ and $\mathbb{R}$ is the real line.

The second phase is called the monitoring phase. During the monitoring phase we send only the probe packets. We build the variable X in the same way as in the learning phase. The QoS parameter $\widehat{Y}$ of the video sequence is estimated using the function $\widehat{\Phi}$ built in the learning phase by $\widehat{Y} = \widehat{\Phi}(X)$. We remark that this procedure does not load the network because it avoids sending the video sequence during the monitoring phase.

Remark 6. The previous discussion is based on the single link case. We discuss now some considerations about the multilink case. First, we must highlight that the multilink case can be reduced to the single link one in many important escenarios. For example, when the application service is offered by a server located at the ISP backbone (for example a video on demand server) and the user access is a cellular link or an ADSL link. In these cases the bottleneck is normally located at the access since the backbone is overprovisioned and it behaves as a single link.

However, there are cases where the packets must wait in more than one queue. In these cases the different queues modify the variable X that we use to characterize the cross traffic. This means that we estimate a variable X where the influence of all queues are accumulated. Nevertheless even in this case our method will work fine if it is possible to distinguish with this variable between different cross traffic processes observed in the path.

Remark 7. Another assumption was that the network path, the link capacities and the buffer sizes are fixed. For link capacities and buffer sizes this assumption is reasonable. However, the route between two points on the network can change. This problem can be solved because it is possible to verify periodically the route between two points using for example an application like trace-route. If a new route is detected two circumstances can arise. If the system has learned information about the new route, this information can be used for the estimation. If

the system has not learned information about the new route it is necessary to trigger a learning phase. Finally, we remark that in some cases a change in the route does not affect the measures, for example when the bottleneck is in the access link and the backbone is overprovisioned.

Remark 8. In this section we work with the assumption that the system is trained with a unique kind of video (we assume that V_t is a fixed sequence). This is not really an issue since the video QoS parameters depend on a set of characteristics like coding, bit-rate, frame-rate and motion level. Therefore, we can train the system with a set of video sequences that represent the different classes of videos. Later the system will use the corresponding training samples depending on the specific video that we want to monitor.

4.3 Statistical Learning, the Nadaraya-Watson estimator

In this section we discuss the mathematical tool selected to estimate Φ. We present a brief review of current results about Nadaraya-Watson estimations. We consider the regression model of equation 36. It is not assumed an explicit form for the function Φ that relates the state of the network with the QoS parameters, and it is not assumed either any particular probability distribution for the random variables involved in the model. For this reason the model is nonparametric.

There are several results on nonparametric regression for real random variables and for random variables in $\mathbb{R}^d$ since the works of Nadaraya and Watson [18]. The Nadaraya-Watson estimator for the real case is

$$\widehat{\Phi}_n(x) = \frac{\sum_{i=1}^{n} Y_i K\left(\frac{||x-X_i||}{h_n}\right)}{\sum_{i=1}^{n} K\left(\frac{||x-X_i||}{h_n}\right)} = \frac{\sum_{i=1}^{n} Y_i K_n\left(X_i\right)}{\sum_{i=1}^{n} K_n\left(X_i\right)} \tag{37}$$

K is a Kernel, which is a positive function that integrates one and $K_n(X_i) = K\left(\frac{||x-X_i||}{h_n}\right)$. h_n is a sequence that tends to zero and it is called the kernel bandwidth. This estimator is a weighted average of the samples $Y_1, \ldots, Y_n$. The weights are given by $K_n\left(X_i\right)$ taking into account the distance between x and each point of the sample $X_1, \ldots, X_n$.

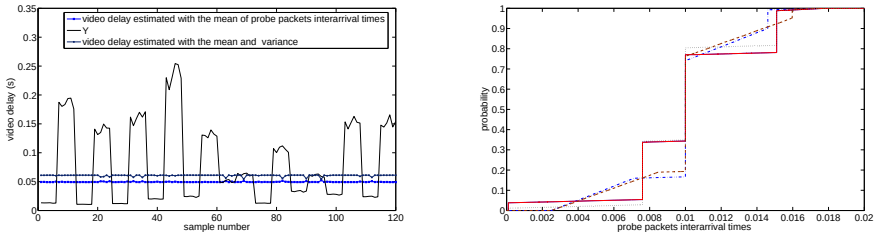
4.4 The empirical distribution of the probe packets interarrival times

In this section we select for the variable X, the empirical distribution of the probe packets interarrival times. This lead us to a functional regression model. In last years some theoretical results on the functional Nadaraya-Watson estimator were developed.

Why functional regression? We try to use as first option for the variable X the mean and/or the variance of the probe packets interarrival times.

In figure 7 (left) it can be observed the estimation of Y using these possible choices for X. We develop many experiments with simulated data and with data taken from operational networks and the estimations of Y are in all cases inaccurate. It is not possible to estimate Y from the mean and the variance.

In figure 7 (right) we show four empirical distribution functions for simulated data. Two of them were obtained in the presence of high cross traffic and the others with low cross traffic. These empirical distribution functions capture some network characteristics that allow us to distinguish between them. In the next section we analyze the QoS estimation using these empirical distribution functions.



Estimation of Y using mean and variance

Empirical distributions

Fig. 7. Probe packets inter-arrival times

Functional Nadaraya-Watson estimator. For functional random variables, i.e. when the regressor X is a random function Ferraty et al. [16] introduce a Nadaraya-Watson type estimator for Φ, defined by equation (37), where the difference with the real case is that $\|\cdot\|$ is a seminorm on a functional space $\mathcal{D}$. One of the main issues in the functional approach is the "curse of dimensionality". The estimation $\widehat{\Phi}_n(x)$ will be accurate if there are enough training samples near x. This issue becomes crucial when the observations come from an infinite dimensional vector space. This problem is addressed in the literature and we refer for example to [16, 19] for different approaches. These works state the convergence and the asymptotic distribution of the estimator for stationary and weakly dependent (for example mixing) functional random variables.

Extensions to the nonstationary case. The cross traffic X_t on the Internet is a dependent and non-stationary process. This topic has been studied by many authors during last ten years. Zhang et al. [20] show that many processes on the Internet (losses for example) can be well modelled as independent and identical distributed (i.i.d.) random variables within a "change free region", where stationarity can be assumed. They describe the overall network behavior as a series of piecewise-stationary intervals.

The nonstationarity has different causes. In all cases it is very important to have estimators that can be used with nonstationary traffic.

As our data comes from Internet data traffic and it is typically nonstationary, we extend previous results about functional nonparametric regression to this case. Instead of considering random variables X equally distributed we consider

a model introduced by Perera in [21] defined by $X_i = \varphi(\xi_i, Z_i)$ where ξ_i takes values in a seminormed vector space with a seminorm $||\cdot||$, and Z_i is a real random variable that takes values in a finite set $\{z_1, z_2, \ldots, z_m\}$. For each $k = 1, \ldots, m$ the sequence $(\varphi(\xi_i, z_k))_{i \geq 1}$ is weakly dependent and equally distributed, but the sequence Z_i may be nonstationary as in [21]. The model represents a mixture of weakly dependent stationary process, but the mixture is nonstationary and dependent. Here ξ represents the usual variations of the traffic, and the variable Z selects between different traffic regimes, and represents types of network traffic.

With this model two main theoretical issues appear: the convergence and the asymptotic distribution of the estimator. We prove in [23] the almost sure convergence of the estimator. The asymptotic distribution of the estimator for this model is discussed in [22].

4.5 First application to simulated data

In this section we analyze the accuracy of estimations with functional Nadaraya-Watson applied to simulated data. We analyze the estimation procedure by simulations using the ns-2 simulator software [24]. We simulate a link fed with a video trace, a simulated cross traffic and probe packets. The cross traffic corresponds to a model $X = \varphi(\xi, Z)$. We have two Markovian ON-OFF sources and Z is a random variable that takes values in $\{0, 1\}$ selecting periodically between this two sources. Fixing the value of Z we obtain stationary processes $\varphi(\xi, 0)$ and $\varphi(\xi, 1)$.

The first source (source 0) generates Markovian ON-OFF traffic corresponding to $\varphi(\xi, 0)$ with average bit rate varying from 150 Mb/s to 450 Mb/s and average time Ton in the ON state and Toff in the OFF state varying from 100 to 300 ms. The second source (source 1) generates Markovian ON-OFF traffic corresponding to $\varphi(\xi, 1)$ with average bit rate varying from 600 Mb/s to 900 Mb/s and average time Ton in the ON state and Toff in the OFF state varying from 200 to 500 ms. For each period an independent random variable is sampled to select the average bit rate. The payload of probe packets is 20 bytes and for the video packets is 1400 bytes. The video sequence has an average bit rate of 480 kbps. The link capacity is 1.6 Mbps.

We send this cross traffic to a network link together with the probe packets and the simulated video sequence. Each test consists on a probe packet burst with fixed interdeparture time t^*_{in}. After this burst we send a simulated video traffic (a video traffic trace). For each test j we compute from the probe packets the empirical distribution function of interarrival times X_j and we measure the average delay Y_j of the video packets.

The kernel is $K(x) = \begin{cases} (x^2 - 1)^2 & \text{if } x \in [-1, 1] \\ 0 & \text{if } x \notin [-1, 1] \end{cases}$

and we use the L^1 norm for the distance between the empirical distribution functions.

Concerning the time scales in our experiment the probe traffic is sent with fixed time t between consecutive probe packets. The aim is to find some criterion for choosing the best time scale in order to have accurate estimates. We consider

different sequences of observations for a finite set of time scales $\{t_1, t_2, \ldots, t_r\}$. In practice, as we send bursts of probe traffic with fixed time t between packets we have observations with time scales in the set $\{t, 2t, \ldots, rt\}$. Consider $n+m$ observations for each time scale $\left\{(X_i^{t_j}, Y_i^{t_j}) : 1 \leqslant i \leqslant n+m,\ 1 \leqslant j \leqslant r\right\}$

By dividing the sequence for a fixed time scale in two we can estimate the function Φ^{t_j} (for the time scale t_j) by $\widehat{\Phi}_n^{t_j}$ with the first n samples.

We then compute the difference ${\sigma^2}_{t_j}(n,m) = \frac{1}{m}\sum_{i=1}^{m}\left(\widehat{\Phi}_n^{t_j}(X_{n+i}^{t_j}) - Y_{n+i}^{t_j}\right)^2$, that gives a measure of the estimator performance for the time scale t_j. We choose $t^*_{n,m}$ such that minimize ${\sigma^2}_{t_j}(n,m)$

The kernel bandwidth is selected with a similar procedure.

In the simulations we have 360 values of (X, Y) and we divide the sample in two parts. The estimation of Φ is obtained from the last 300 samples and the accuracy of the estimation is evaluated over the first 60 samples by comparing $\Phi_n(X_j)$ with the measured average delay Y_j for $j = 1, \ldots, 60$. The relative error in each point j is computed by $\frac{|\widehat{\Phi}_n(X_j) - Y_j|}{Y_j}$. Figure 8 show the estimated and the measured value of the average delay, showing a good fitting.

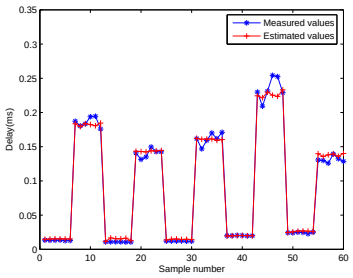


Fig. 8. Average delay estimation for simulated data.

4.6 Experimental Results.

In this section we show results of the procedures presented in this paper applied to different operational networks. The experiments were done with a measurement software tool specially developed for this purpose. In order to evaluate the practical limits of this methodology we analyze different scenarios that have different levels of complexity. In this paper we show only measurements using a cellular access network.

We analyze a cellular connection used with a PC and an cellular modem. The video sequences are downloaded from a server located at Facultad de Ingeniería, Universidad de la República. In this case the videos were codified at an average rate of 96 kbps. First of all, we take the first 30 samples in order to select the model. Next, we take the other 35 points not used to select the model in order to validate the model.

Figure 9 left and right show the video losses and its mean delay for the 35 points of the validation sample. The accuracy of the estimation is reasonable taking into account the variability of the data.

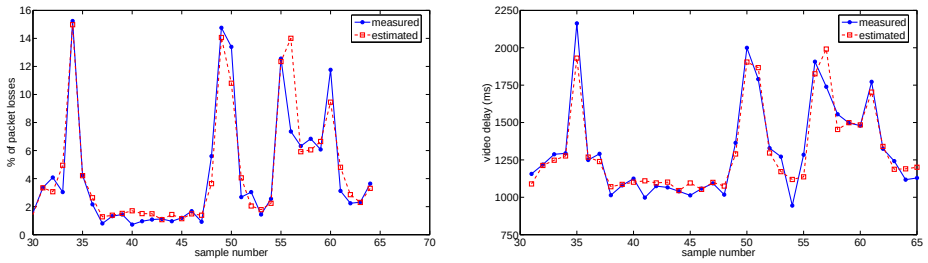


Fig. 9. Video packet losses (left) and mean delay (right) in the cellular case

5 Conclusions

This work addresses the challenge of guaranteeing quality of service (QoS) in the Internet from a statistical point of view. First, we have discussed the end-to-end QoS parameters estimations based models from the Large Deviation Theory. Later, we have analyzed the estimation of QoS parameters seen by an application based on end-to-end active measurements and statistical learning tools. We have discussed how these methodologies can be applied to different parts of the network in order to analyze its performance. We have obtained tight estimations applying both methodologies.

Acknowledgments. This work was partially supported by a grant of CSIC-UDELAR. The authors want to thank the following members of ARTES research group: L. Aspirot, B. Bazzano, P. Bermolen, P. Casas, A. Ferragut F. Larroca and G. Perera for their contributions to this work.

References

1. Breslau, L., Jamin, S., Shenker, S.: Comments on the performance of measurement-based admision control algorithm, IEEE INFOCOM 2000 (Tel Aviv, Israel), pp. 1233–1242, (2000).
2. Dembo, A., Zeitouni, O.: Large Deviations Techniques and its Applications, Jones and Bartlett, New York, (1993).
3. Kelly, F.: Notes on Effective Bandwidth, in Stochastic Networks: Theory and Applications, edited by Kelly, Zachary and Ziedins, Oxford University Press, (1996).
4. Ozturk, O., Mazumdar, R., Likhanov, N.: *Many sources asymptotics in networks with small buffers*, Queueing Systems (QUESTA) **46** , no. 1-2, 129–147, (2004).
5. Más, N., Karlsson, G.: Probe-based admission control for a differentiated-services internet, Computer networks **51**, 3902–3918, (2007).

6. Wischik, D.: Sample path large deviations for queues with many inputs, Annals of Applied Probability, No. 11, pp. 389-404, (2000).
7. Courcoubetis, C., Siris, V.A.: Procedures and tools for analysis of network traffic measurements, Elsevier Science, (2001).
8. Rabinovitch, P.: Statistical estimation of effective bandwidth, M.Sc.thesis, University of Cambridge, (2000).
9. Kesidis, G., Walrand, J., Chang, C.S.: Effective Bandwidths for Multiclass Markov Fluids and Other ATM Sources, IEEE/ACM Trans. Networking, No.1, pp. 424-428, (1993).
10. Pechiar, J., Perera, G., Simon, M.: Effective Bandwidth estimation and testing for Markov sources, Performance Evaluation, edited by D.D.Kouvatsos, Elsevier, New Holland, (2002).
11. Lebedev, E.A., Lukashuk, L.I.: Maximum likelihood estimation of the infinitesimal matrix of a Markov chain with continuous time, (Russian, English summary), Dokl. Akad. Nauk Ukr. SSR, Ser. A, No.1, pp. 12-14, (1986).
12. Aspirot, L., Belzarena, P., Bermolen, P., Ferragut, A., Perera, G., Simon, M.: Quality of service parameters and link operating point estimation based on effective bandwidths, Performance Evaluation, Elsevier, **59** , no. 2-3, pp. 103–120, (2005).
13. Wischik, D.: The output of a switch or effective bandwidths for network, Queueing Systems, **32** , no. 4, pp. 383–396, (1999).
14. Belzarena, P., Bermolen, P., Simon, M., Casas, P.: End-to-End Quality of Service-based Admission Control Using the Fictitious Network Analysis, Computer Communications (COMCOM) - The International Journal for the Computer and Telecommunications Industry, Special issue on 'Heterogeneous Networks: Traffic Engineering and Performance Evaluation, 2010 (to appear).
15. Nadaraya, E.A.: Nonparametric estimation of probability densities and regression curves, Mathematics and its Applications (Soviet Series), 20. Dordrecht: Kluwer Academic Publishers Group, (1989).
16. Ferraty, F. Vieu, P.: Nonparametric Functional data analysis: Theory and Practice, Springer Series in Statistics. New York: Springer, (2006).
17. Machiraju, S., Veitch, D., Baccelli, F., Nucci, A., Bolot, J.: Theory and practice of cross-traffic estimation, SIGMETRICS, pp. 400–401, (2005).
18. Nadaraya, E.A.: On estimating regression, Theory of Probability and its Applications, 9(1), pp. 141–142, (1961).
19. Masry, E.: Nonparametric regression estimation for dependent functional data: asymptotic normality. Stochastic Process. Appl., 115, pp. 155–177, (2005).
20. Zhang, Y. Duffield, N.: On the constancy of internet path properties, IMW '01: Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement, pp. 197–211, (2001).
21. Perera, G.: Irregular sets and central limit theorems. Bernoulli, 8, pp. 627–642, (2002).
22. Bertin, K., Aspirot, L.: Asymptotic normality of the Nadaraya-Watson estimator for non-stationary data. To appear in Journal of nonparametric statistics, (2009).
23. Aspirot, L., Belzarena, P., Bazzano, B., Perera, G.: End-To-End Quality of Service Prediction Based On Functional Regression. Proc. Third International Working Conference on Performance Modelling and Evaluation of Heterogeneous Networks (HET-NETs 2005), Ilkley, UK, (2005).
24. McCanne, S., Floyd, S.: ns network simulator, URL:http://www.isi.edu/nsnam/ns/ [Accessed 03/2009].