VIRTUAL PATHS NETWORKS FAST PERFORMANCE ANALYSIS

P.BELZARENA, P.BERMOLEN, P.CASAS, M.SIMON
ARTES *

FACULTAD DE INGENIERÍA, UNIVERSIDAD DE LA REPÚBLICA
MONTEVIDEO, URUGUAY

Abstract. The performance analysis of a link is a well studied problem. However, for a service provider the most interesting issue is the end-to-end quality of service (QoS) evaluation. The focus of this work is to go from the link to the network analysis. This can be done in a complete but complex way or using an approximation to speedup the calculations. We analyse and compare both methods.

Large Deviations Theory applications to Data Networks are mainly based on the many sources asymptotic. This asymptotic is adequate for networks like Internet backbones, where the assumption that the network is fed by a large number of sources is reasonable.

Recently, Ozturk et al. have proposed a slightly different model called many sources and small buffer asymptotic. They give a formula to calculate the link overflow probability and the end-to-end Loss Ratio of traffic streams in a virtual path feed forward network of general topology. They also define the *fictitious network* concept. The fictitious network has the same topology than the real one, but each traffic stream goes across a link on its path without being affected by the upstream links until that one. So, in the fictitious network each internal link can be analysed as an external one. Therefore, the fictitious network usage simplifies dramatically the network performance analysis.

Our main motivation to simplify this task is to allow on-line performance analysis and traffic engineering algorithms in virtual path networks as MPLS or ATM.

Ozturk et al. show that the fictitious network overestimates the overflow probability and the end to end Loss Ratio. Therefore, decisions based on the fictitious network analysis are safe. However, this overestimation leads to network resources under-utilization.

Under certain conditions the real and the fictitious network analysis give the same results (there is no overestimation). In this work we establish sufficient conditions to assure that this coincidence arises. Those conditions are not necessary, and we give an easy way to check if exact results may be obtained even though sufficient conditions are not met. When the real and fictitious networks analysis give different results, we find a method to bound the overestimation.

Finally, we show some numerical examples to compare the performance analysis in the real vs. the fictitious network, and to validate our main results.

1. Introduction

MPLS (MultiProtocol Label Swiching)[10] is an architecture that enables to perform traffic engineering in IP networks. MPLS introduces the notion of Forwarding Equivalence Class (FEC), giving the network operator the possibility to split the traffic in aggregated flows according to the service model adopted by the Internet Service Provider (ISP).

The edge routers in an MPLS network (or LER for Label Edge Router) are responsible for establishing MPLS tunnels named LSPs (Label Switched Path) between the endpoints of the MPLS domain, and to send each arriving packet to the corresponding LSP.

Explicit Routing (ER), a typical MPLS feature, is the main function that enables Traffic Engineering (TE). Using ER the network operator can establish for each FEC one or more LSPs.

In order to satisfy end to end QoS guarantees in a MPLS network, a performance model is required. This model provides methods to analyse the end to end QoS performance parameters.

We are interested in a performance evaluation model simple enough to be used in on line algorithms. In this work we will analyse the model called the "fictitious network model" in the context of many sources and small buffer asymptotic.

* ARTES: Joint Research Group of the Electrical Engineering and Mathematics and Statistics Departments. Contact: artes@fing.edu.uy. This research was partially supported by PDT (Programa de Desarrollo Tecnológico, Préstamo 1293/OC-UR): S/C/OP/ 17/02, S/C/OP/17/03, CSIC and program FCE (Fondo Clemente Estable) 8079.

Ozturk et al.[9] find a useful way to analyse the overflow probability in a network interior link, in the many sources and small buffer asymptotic. Applying their model, the end-to-end Loss Ratio can also be evaluated. This model can be used for off-line performance evaluation and traffic engineering. They also introduce the fictitious network model and show that this latter overestimates the link overflow probability. The fictitious network is a simplified network model that can be used for on-line performance evaluation and traffic engineering. In this work we study 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 met.

In section (2) we summarise some related works in order to give a brief description of the problem´s context.

In section (3) we explain the model introduced by Ozturk et al. and we summarise their main results.

In section (4) we introduce our main results.

In section (5) we show some numerical examples, comparing the real and the fictitious network based performance analysis and evaluate our results.

Finally, in section (6) we summarise and comment the main results.

2. Related works

Network designers and operators need a model to evaluate the end to end network performance in an Internet core backbone. Since losses are "rare" events, some researchers have proposed the use of Large Deviations methods for network performance evaluation.

In this context, the effective bandwidth notion was introduced some years ago. The notion of equivalent bandwidth was formerly used to study the access control problem for some networks, as ATM. Many contributions following this approach were done during the 90's to analyse the access control in some networks based on the IntServ model or others. In that situation, the access node receives a connection request and has to estimate the resources it requires, in order to allow or deny the new connection. Kelly's Effective Bandwidth (EB) [7] may be used in such situations as the "equivalent capacity" needed by the new connection. In this context, the flow to be statistically characterized is an individual flow, and may be directly related with the data source (for instance, voice or video codecs). This situation was studied using the so called large buffer asymptotic, in which the link buffer grows to infinity, and its filling above some threshold is analysed. This approach cannot be used in backbone links, where buffers are not devised to store bursts but to resolve simultaneous packet arrival, being consequently small. The application of Large Deviations Theory to the analysis of the MPLS backbone must be performed on the basis of the many sources asymptotic. In this regime we take buffer size $B = Nb$ (whit N the number of sources), output capacity $C = Nc$ and make N go to infinity. Results about loss probability in this regime can be found in [13], [2], [11], [5].

Using Large Deviations, Wischik [13] 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 [12] 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" networks. Unfortunately this analysis can not be extended to networks like a MPLS backbone.

Eun y Shroff [6], have recently shown that in the many sources asymptotic regime, the probability of the buffer size to be grater than zero goes to zero when the number of sources goes to infinity. This result is valid for a discrete time queue and is valid for a continuous time queue if the source is bounded or can be expressed as an integral of a stationary stochastic process.

Recently, a slightly different asymptotic with many sources and small buffer characteristics was proposed by Ozturk, Mazumdar and Likhanov in [9]. They consider an asymptotic regime defined by N traffic sources, link capacity increasing proportionally with N but buffer size such $\lim \frac{B(N)}{N} \to 0$. 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 MPLS 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 analyse when, for an interior network link, the overflow probability calculated using the fictitious network is equal to the overflow probability of the real network. Ramon Casellas [1] has also studied the overestimation problem in the fictitious network. He found a condition to assure that there is no overestimation. This condition is a particular case of the sufficient condition proven in this work.

In the next section we introduce some concepts of the Large Deviations Theory and summarize Ozturk et al. work.

3. Many sources and small buffer asymptotic performance model

3.1. **Large Deviations Principle.** [4]

Definition 3.1. (1) $I : \chi \to [0, \infty)$*, with* (χ, B) *a measure space is a* rate function *(R.F) if it is a lower semicontinous function.*

(2) $I : \chi \to [0, \infty)$*, with* (χ, B) *a measure space is a* good rate function *(G.R.F) if all the level set of* I *(*$\Psi_\alpha(I) = \{x : I(x) \leqslant \alpha\}$*) are compact sets.*

Definition 3.2. *A probability measure family* μ_n *in* (χ, B) *satisfies a large deviation principle(LDP) with* rate function I *if*

$$-\inf_{x \in \Gamma^o} I(x) \leq \liminf_{n\to\infty} \frac{1}{n} \log \mu_n(\Gamma) \leq \limsup_{n\to\infty} \frac{1}{n} \log \mu_n(\Gamma) \leq -\inf_{x \in \overline{\Gamma}} I(x) \tag{1}$$

$\forall\ \Gamma \in B$ *with interior* Γ^o *and closure* $\overline{\Gamma}$*.*

3.2. **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* $I_t^{X^m}(x)$:

$$-\inf_{x \in \Gamma^o} I_t^{X^m}(x) \leq \liminf_{N\to\infty} \frac{1}{N} \log \mathbb{P}\left(\frac{X^{m,N}(0,t)}{N} \in \Gamma\right) \tag{2}$$

$$\leq \limsup_{N\to\infty} \frac{1}{N} \log \mathbb{P}\left(\frac{X^{m,N}(0,t)}{N} \in \Gamma\right) \leq -\inf_{x \in \overline{\Gamma}} I_t^{X^m}(x) \tag{3}$$

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 (as in MPLS and ATM networks) 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

$$\sum_{m \in \mathcal{M}_k} \mu_m < C_k \tag{4}$$

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

Theorem 3.3. *Let $X^{m,N}_{k,t}$ be the rate of type m traffic at link k at time t. There exist a continuous function $g^m_k : \mathbb{R}^M \to \mathbb{R}$ relating the instantaneous input rate at link k for traffic m to all of the instantaneous external input traffic rates such that:*

$$\frac{X^{m,N}_{k,0}}{N} = g^m_k\left(\frac{X^{1,N}_0}{N}, ..., \frac{X^{M,N}_0}{N}\right) + o(1) \tag{5}$$

The buffer overflow probabilities are given by:

$$\begin{aligned} &\lim_{N\to\infty} \frac{1}{N} \log P(\textit{overflow in link } k) = -\mathbf{I}_k = \\ &-\inf\left\{\sum_{m=1}^{M} I_1^{X^m}(x_m) : x = (x_m) \in \mathbb{R}^M, \sum_{m=1}^{M} g^m_k(x) > C_k\right\} \end{aligned} \tag{6}$$

In (5), $o(1)$ verifies that $\lim\limits_{N\to\infty} o(1) = 0$ since $\frac{B_k(N)}{N} \underset{N\to\infty}{\to} 0$. The function $g^m_k(x)$ 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:

$$f^m_i(x, C_i) = \frac{x_m C_i}{\max(\sum\limits_{j\in\mathcal{M}_i} x_j, C_i)} \tag{7}$$

In a feed-forward network the function $g^m_k(x)$ can be written as composition of the functions of type (7) in a recursive way. Using equation (7) the buffer overflow probability can be calculated for any network link, by solving the optimisation problem of equation (6). We need to know the network topology, the link's 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 $\mathbf{L}^{m,N}$:

$$\lim_{N\to\infty} \frac{1}{N} \log \mathbf{L}^{m,N} = -\min_{k\in k^m} \mathbf{I}_k \tag{8}$$

However, in a big network, the optimisation problem of equation (6) could be very hard to solve. The calculation of the function $g^m_k(x)$ 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 $g^m_k(x)$. To solve equation (6), it is also necessary to optimise 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 (overestimation) in those cases where the previous condition is not satisfied.

4. Fictitious network analysis

We analyse an interior network link k 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:

$$I^R_k = \inf\left\{\sum_{i\in\mathcal{M}} I_1^{X^i}(x_i) : x = (x_i)_{i\in\mathcal{M}}, \sum_{i\in\mathcal{M}} g^i_k(x) > C_k\right\} \tag{9}$$

In the fictitious network this function is given by

$$I_k^F = \inf\left\{\sum_{i\in\mathcal{M}_k} I_1^{X^i}(x_i) : x = (x_i)_{i\in\mathcal{M}_k}, \sum_{i\in\mathcal{M}_k} x_i > C_k\right\} \tag{10}$$

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, (9) and (10) are convex optimisation 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 4.1. *Consider two optimisation problems*

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

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 4.2. *If P_2 is a relaxation of P_1 and x_2 is optimum for P_2 such $x_2 \in D_1$ and $f_2(x_2) = f_1(x_2)$, then x_2 is optimum for P_1.*

Proof. $f_1(x_2) = f_2(x_2) \leq f_2(x) \leq f_1(x)$ $\forall x \in D_1 \subseteq D_2$, so x_2 is optimum for P_1 because it minimizes f_1 and belongs to D_1. ☐

Proposition 4.3. *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 (7)) 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 4.4. *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. In the next remark we find the optimality conditions for P_F.*

Remark 4.5. *The optimality conditions (KKT [1]) for the fictitious problem*

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

are the following:

[1]Karush-Khum-Tucker [8]

(1) $\nabla \left(\sum_{i \in \mathcal{M}_k} I_1^{X^i}(x_i) + \lambda(C_k - \sum_{i \in \mathcal{M}_k} x_i) \right) = 0$, *with* λ *Lagrange multiplier.*

(2) $\lambda \geq 0$.

(3) $\sum_{i \in \mathcal{M}_k} x_i \geq C_k$.

(4) $\lambda \left(C_k - \sum_{i \in \mathcal{M}_k} x_i \right) = 0$.

The first condition implies that

$$\frac{\partial I_1^{X^i}}{\partial x_i}(x_i) = \lambda \quad \forall\, i \in \mathcal{M}_k$$

and by the second one,

$$\frac{\partial I_1^{X^i}}{\partial x_i}(x_i) = \lambda \geq 0$$

If $\lambda = 0$, $x_i = \mu_i$ $\forall i$. *In this case* $\sum_{i \in \mathcal{M}_k} I_1^{X^i}(\mu_i) = 0$ *and it is not considered. Then we supposed that* $\frac{\partial I_1^{X^i}}{\partial x_i}(x_i) > 0$, *which implies* $x_i > \mu_i$. *Finally, since* $\lambda \neq 0$, *the last condition implies that*

$$C_k - \sum_{i \in \mathcal{M}_k} x_i = 0$$

Then, $\widetilde{x} = (\widetilde{x}_i)_{i \in \mathcal{M}_k}$ *optimum for* P_F *verifies:*

$$\left\{ \begin{array}{l} \widetilde{x}_i > \mu_i \quad \forall\, i \in \mathcal{M}_k \\ \\ \sum_{i \in \mathcal{M}_k} \widetilde{x}_i = C_k \end{array} \right. \tag{11}$$

The following theorem gives conditions over the network to assure that the 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 4.6. *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:*

$$C_k - \sum_{j \in \mathcal{M}_k \setminus \mathcal{M}_i} \mu_j \leq C_i - \sum_{j \in \mathcal{M}_i \setminus \mathcal{M}_k} \mu_j \quad \forall\, i < k \tag{12}$$

then x^* *defined by:*

$$(x^*)_i = \left\{ \begin{array}{ll} \widetilde{x}_i & \text{if } i \in \mathcal{M}_k \\ \mu_i & \text{if } i \notin \mathcal{M}_k \end{array} \right.$$

is optimum for P_R.

Proof. The objective functions of the optimisation problems (9) and (10) take the same values at x^* because $I_1^{X^i}(\mu_i) = 0 \quad \forall i$:

$$\sum_{i \in \mathcal{M}} I_1^{X^i}(x_i^*) = \sum_{i \in \mathcal{M}_k} I_1^{X^i}(\widetilde{x}_i) + \sum_{i \in \mathcal{M} \setminus \mathcal{M}_k} I_1^{X^i}(\mu_i) = \sum_{i \in \mathcal{M}_k} I_1^{X^i}(\widetilde{x}_i)$$

Considering proposition (4.3), it is enough to prove that x^* satisfy the real problem constraints:

$$\sum_{i \in \mathcal{M}} g_k^i(x_i^*) \geq C_k$$

By definition $g_k^i(x^*) = 0\ \forall\, i \notin \mathcal{M}_k$. Moreover the function g_k^i can be written as composition of function of type (7), so if $\sum_{j \in \mathcal{M}_i} (x_j^*) \leq C_i\ \forall\, i$, then $g_k^i(x^*) = (x^*)_i\ \forall\, i \in \mathcal{M}_k$ and

$$\sum_{i \in \mathcal{M}} g_k^i(x^*) = \sum_{i \in \mathcal{M}_k} (x^*)_i = \sum_{i \in \mathcal{M}_k} \widetilde{x}_i = C_k$$

proving the theorem. In the last equality we use that $\widetilde{x}$ verifies (11), since it is optimum for P_F. Then, it is sufficient to prove that $\sum_{j \in \mathcal{M}_i} (x^*)_j \leq C_i \quad \forall\, i < k$. Separating the sum,

$$\sum_{j \in \mathcal{M}_i} (x^*)_j = \sum_{j \in \mathcal{M}_i \cap \mathcal{M}_k} \widetilde{x}_j + \sum_{j \in \mathcal{M}_i \setminus \mathcal{M}_k} \mu_j \leq C_i \quad \forall\, i < k \tag{13}$$

and then we have to guarantee that

$$\sum_{j \in \mathcal{M}_i \cap \mathcal{M}_k} \widetilde{x}_j \leq C_i - \sum_{j \in \mathcal{M}_i \setminus \mathcal{M}_k} \mu_j \quad \forall\, i < k$$

Since $\widetilde{x}$ is optimum for P_F, it satisfy $C_k = \sum_{j \in \mathcal{M}_k} \widetilde{x}_j$, and therefore

$$\sum_{j \in \mathcal{M}_k \cap \mathcal{M}_i} \widetilde{x}_j = C_k - \sum_{j \in \mathcal{M}_k \setminus \mathcal{M}_i} \widetilde{x}_j$$

Also, $\widetilde{x}_j > \mu_j \ \forall\, j \in \mathcal{M}_k$

$$\sum_{j \in \mathcal{M}_k \cap \mathcal{M}_i} \widetilde{x}_j \leq C_k - \sum_{j \in \mathcal{M}_k \setminus \mathcal{M}_i} \mu_j$$

Using the hypothesis, we have that:

$$\sum_{j \in \mathcal{M}_k \cap \mathcal{M}_i} \widetilde{x}_j = C_k - \sum_{j \in \mathcal{M}_k \setminus \mathcal{M}_i} \mu_j \leq C_i - \sum_{j \in \mathcal{M}_i \setminus \mathcal{M}_k} \mu_j \quad \forall\, i < k$$

which proves (13) and the theorem.

□

Example 4.7. *Consider a network like in figure 1. We analise the overflow probability at link k.*

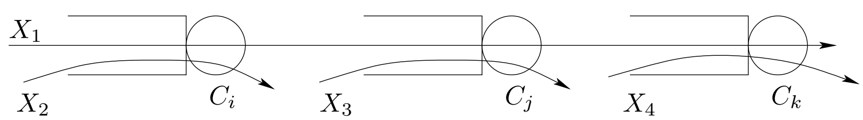

FIGURE 1

If condition (12) is attained for link k, then $E = I_k^R - I_k^F = 0$. This condition is:

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

4.1. **Theorem 4.6 in terms of available bandwith.**

Definition 4.8. *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 (12) 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 2 for a simple network with two links.

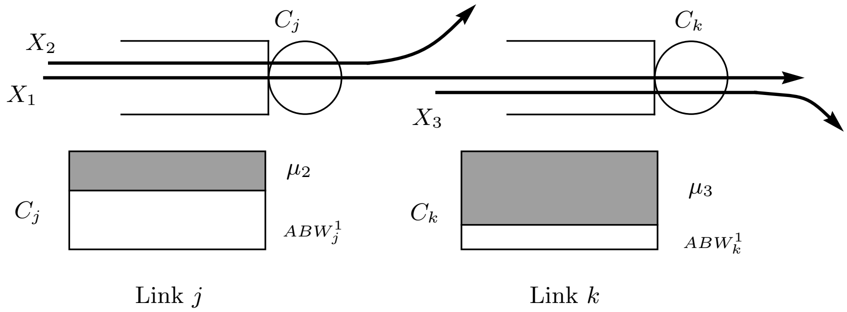

FIGURE 2

4.2. **Sufficient but not necessary condition.** The theorem condition (12) 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:

$$(x^*)_i = \begin{cases} \widetilde{x}_i & \text{si } i \in \mathcal{M}_k \\ \mu_i & \text{si } i \notin \mathcal{M}_k \end{cases} \tag{14}$$

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

$$\sum_{j \in \mathcal{M}_i} (x^*)_j \leq C_i \quad \forall\, i < k \tag{15}$$

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 (15), where x^* is defined in (14).

4.3. **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 (12) and (15) 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

$$\begin{cases} \sum_{j \in \mathcal{M}_i} x_j \leq C_i \quad \forall\, i < k \\ \\ \sum_{j \in \mathcal{M}_k} x_j \geq C_k \end{cases}$$

Therefore, we have to solve this inequalities system. From remark (4.5), 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:

$$(16) \qquad \begin{cases} \sum_{j\in\mathcal{M}_i} x_j \leq C_i \quad \forall\, i<k \\ \\ \sum_{j\in\mathcal{M}_k} x_j = C_k \end{cases}$$

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 (16), 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 (16), 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 minimises 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.

4.4. **Error bound in a particular case.** We analyse a particular case in which the following conditions are verified:

(1) $\mathcal{M}_i\backslash\mathcal{M}_k\neq\emptyset\ \forall\, i<k$ this means that for all link i less than k, there exists at least one traffic type going through link i and not arriving at link k.

(2) $C_i - \left(C_k - \sum_{j\in\mathcal{M}_k\backslash\mathcal{M}_i} \mu_j\right) \geq 0 \quad \forall\, i.$

Consider $\widetilde{x}=(\widetilde{x}_i)_{i\in\mathcal{M}_k}$ optimum for (P_F) and x^* defined by:

$$(17) \qquad (x^*)_j = \begin{cases} \widetilde{x}_j & \text{if } j\in\mathcal{M}_k \\ x_j^* & \text{if } j\notin\mathcal{M}_k \end{cases}$$

If x^* verify the real problem constraints, the following error bound is obtained:

$$(18) \qquad E \leq \sum_{i\in\mathcal{M}} I_1^{X^i}((x^*)_i) - \sum_{i\in\mathcal{M}_k} I_1^{X^i}(\widetilde{x}_i)$$

$$(19) \qquad = \sum_{i\in\mathcal{M}_k} I_1^{X^i}(\widetilde{x}_i) + \sum_{i\mathcal{M}\backslash\mathcal{M}_k} I_1^{X^i}(x_i^*) - \sum_{i\in\mathcal{M}_k} I_1^{X^i}(\widetilde{x}_i)$$

$$(20) \qquad = \sum_{i\in\mathcal{M}\backslash\mathcal{M}_k} I_1^{X^i}(x_i^*)$$

By definition of x^* and the optimality conditions for the fictitious problem, it follows that:

$$\sum_{j\in\mathcal{M}_k} (x^*)_j = \sum_{j\in\mathcal{M}_k} (\widetilde{x})_j = C_k$$

Therefore, to prove that x^* verify the real problem constraints (16), it is enough to show that x^*, verify:

$$(21) \qquad \sum_{j\in\mathcal{M}_i} (x^*)_j = \sum_{j\in\mathcal{M}_i\cap\mathcal{M}_k} \widetilde{x}_j + \sum_{j\in\mathcal{M}_i\backslash\mathcal{M}_k} x_j^* \leq C_i \quad \forall\, i$$

In this particular case, by the second condition, it is possible to define x_j^* for $j\in\mathcal{M}_i\backslash\mathcal{M}_k$ such that

$$\sum_{j\in\mathcal{M}_i\backslash\mathcal{M}_k} x_j^* \leq C_i - \left(C_k - \sum_{j\in\mathcal{M}_k\backslash\mathcal{M}_i} \mu_j\right) \quad \forall\, i$$

and therefore

$$\sum_{j\in\mathcal{M}_i}(x^*)_j \le \sum_{j\in\mathcal{M}_i\cap\mathcal{M}_k}\widetilde{x}_j + C_i - C_k + \sum_{j\in\mathcal{M}_k\setminus\mathcal{M}_i}\mu_j$$

On the other hand, since $\widetilde{x}$ is optimum for P_F, $\sum_{j\in\mathcal{M}_k}\widetilde{x}_j = C_k$ and

$$\sum_{j\in\mathcal{M}_i\cap\mathcal{M}_k}\widetilde{x}_j = C_k - \sum_{j\in\mathcal{M}_k\setminus\mathcal{M}_i}\widetilde{x}_j$$

Replacing in the previous equation results

$$\begin{aligned}\sum_{j\in\mathcal{M}_i}(x^*)_j &\le C_k - \sum_{j\in\mathcal{M}_k\setminus\mathcal{M}_i}\widetilde{x}_j + C_i - C_k + \sum_{j\in\mathcal{M}_k\setminus\mathcal{M}_i}\mu_j \\ &= C_i + \sum_{j\in\mathcal{M}_k\setminus\mathcal{M}_i}(-\widetilde{x}_j + \mu_j) < C_i\end{aligned}$$

since from (11), $\widetilde{x}_j > \mu_j \quad \forall\, j\in\mathcal{M}_k$.

Then x^* verifies (16) and therefore is optimum for the real problem. The error bound obtained is (20). We can found $(x^*)_{j\in\mathcal{M}\setminus\mathcal{M}_k}$ such that, the error bound (20) be minimum in the set of $(x^*)_{j\in\mathcal{M}}$ defined in (17) that verifies the real problem constraints. It is necessary to solve the following convex optimisation problem:

$$\begin{cases} \min \sum_{i\in\mathcal{M}\setminus\mathcal{M}_k} I_1^{X^i}(x_i) \\ \\ \sum_{j\in\mathcal{M}_i\setminus\mathcal{M}_k} x_j \ge C_i - \sum_{j\in\mathcal{M}_i\cap\mathcal{M}_k}\widetilde{x}_j \quad \forall\, i=1,\cdots,L \end{cases}$$

Once again it is sufficient to find $(x^*)_{j\in\mathcal{M}}$ that verifies the KKT optimality conditions:

(1) $\frac{\partial}{\partial x_j} I_1^{X^j}(x_j) + \sum_{i\in k^j}\lambda_i = 0\ \forall\, j\in\mathcal{M}\setminus\mathcal{M}_k$.
(2) $\lambda_i \ge 0\ \forall\, i$.
(3) $\sum_{j\in\mathcal{M}_i\setminus\mathcal{M}_k} x_j \ge C_i - \sum_{j\in\mathcal{M}_i\cap\mathcal{M}_k}\widetilde{x}_j \quad \forall\, i$.
(4) $\lambda_i\left(\sum_{j\in\mathcal{M}_i\setminus\mathcal{M}_k} x_j - \left(C_i - \sum_{j\in\mathcal{M}_i\cap\mathcal{M}_k}\widetilde{x}_j\right)\right) = 0\ \forall\, i$.

We will define an algorithm to find such point. If for $j\in\mathcal{M}\setminus\mathcal{M}_k$, there is a link $i\in k^j$ that verifies

$$C_i - \sum_{h\in\mathcal{M}_i\cap\mathcal{M}_k}\widetilde{x}_h \le \sum_{h\in\mathcal{M}_i\setminus\mathcal{M}_k}\mu_h$$

we define x^* as follows:

$$\sum_{h\in\mathcal{M}_i\setminus\mathcal{M}_k} x_h = C_i - \sum_{j\in\mathcal{M}_i\cap\mathcal{M}_k}\widetilde{x}_j$$

This determines a linear equations system that always has a solution, but it can be undetermined. The choice in that case it is not important because the optimum obtained is global. For the coordinates $j\in\mathcal{M}\setminus\mathcal{M}_k$, that are not determined with the previous equations, we define $x_j = \mu_j$. It is easy to check that $(x^*)_{j\in\mathcal{M}\setminus\mathcal{M}_k}$ defined by this algorithm verifies the KKT optimality conditions. Then we have defined an algorithm that gives the minimum error bound for this particular case.

5. Numerical examples

Example 5.1. *Consider a network like in figure 3. We analise the overflow probability at link k, assuming that $C_i > C_k$.*

If condition (12) is attained for link k, then $E = I_k^R - I_k^F = 0$. This condition is:

$$C_k \le C_i - \mu_2$$

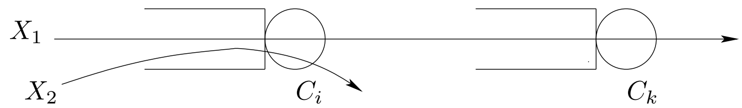

FIGURE 3

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:

$$E \leq I_1(C_k) + I_2(C_i - C_k) - I_1(\widetilde{x}_1) = I_2(C_i - C_k) \tag{22}$$

For 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 (12) is, $C_k \leq 8kb/s$. Figures 4 and 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 (22) described before. In this case, the error bound is exactly the error.

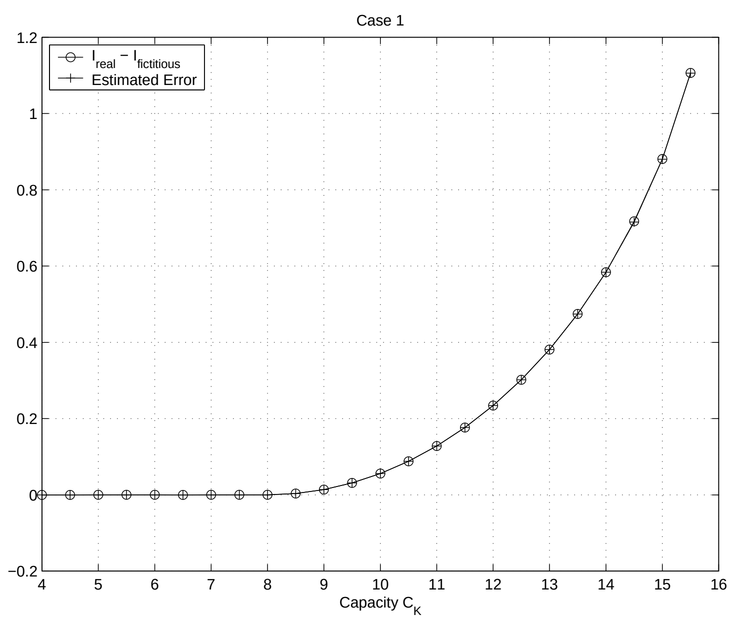

FIGURE 4

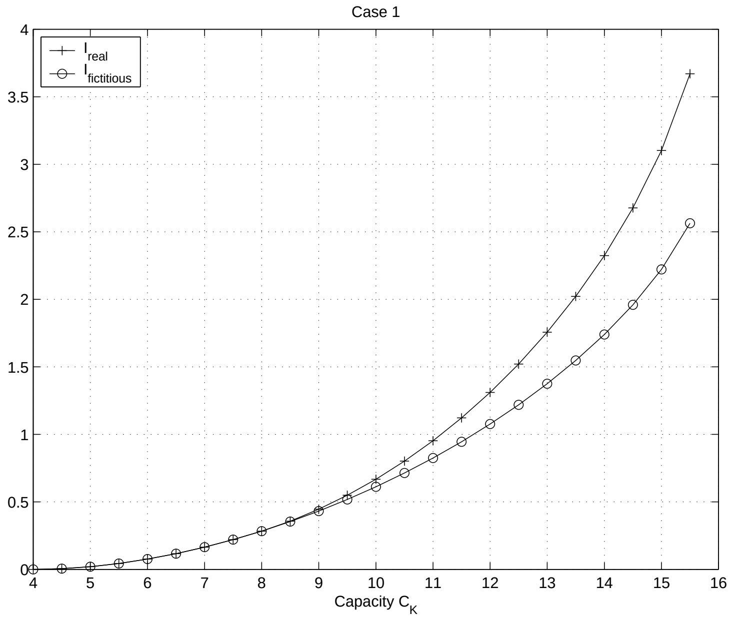

FIGURE 5

Example 5.2. *Consider a network like in figure 6. We analise the overflow probability at link k.*

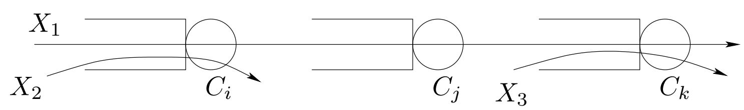

FIGURE 6

If condition (12) is attained for link k, then $E = I_k^R - I_k^F = 0$. This condition is:

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

If this condition is not satisfied, and $\widetilde{x} = (\widetilde{x}_1, \widetilde{x}_3)$ is optimum for P_F, we first verify if $x^* = (\widetilde{x}_1, \mu_2, \widetilde{x}_3)$ is optimum for P_R. It is sufficient to show that x^* verifies the real problem, i.e:

$$\begin{cases} \widetilde{x}_1 + \mu_2 \le C_i \\ \widetilde{x}_1 \le C_j \\ \widetilde{x}_1 + \widetilde{x}_3 = C_k \end{cases}$$

If these conditions are not satisfied, we look for $x^* = (x_1^*, x_2^*, x_3^*)$ that satisfies:

$$\begin{cases} x_1^* + x_2^* \le C_i \\ x_1^* \le C_j \\ x_1^* + x_3^* = C_k \end{cases}$$

We choose $x_1^* = \min(\widetilde{x}_1, C_i, C_j)$. Three different cases are identified. For the first case $x_1^* = \widetilde{x}_1$, we choose:

$$\begin{cases} x_1^* = \widetilde{x}_1 \\ x_2^* = C_i - \widetilde{x}_1 \\ x_3^* = C_k - \widetilde{x}_1 \end{cases} \tag{23}$$

In this case the error bound is:

$$\begin{aligned} E &\le I_1(\widetilde{x}_1) + I_2(C_i - \widetilde{x}_1) + I_3(C_k - \widetilde{x}_1) - I_1(\widetilde{x}_1) - I_3(\widetilde{x}_3) \\ &= I_2(C_i - \widetilde{x}_1) + I_3(C_k - \widetilde{x}_1) - I_3(\widetilde{x}_3) \end{aligned} \tag{24}$$

Using that $\widetilde{x}_1 + \widetilde{x}_3 = C_k$, we have another possibility for determining an error bound. We can rewrite the first equation as $C_k - \widetilde{x}_3 + x_2^* \leq C_i$. And, since $\widetilde{x}_3 > \mu_3$ we can choose $x_2^* = C_i - (C_k - \mu_3)$ (or any lower value). In this case the error bound is:

$$\begin{aligned} E &\leq I_1(\widetilde{x}_1) + I_2(C_i - (C_k - \mu_3)) + I_3(C_k - \widetilde{x}_1) - I_1(\widetilde{x}_1) - I_3(\widetilde{x}_3) \\ &= I_2(C_i - (C_k - \mu_3)) + I_3(C_k - \widetilde{x}_1) - I_3(\widetilde{x}_3) \end{aligned} \tag{25}$$

The best error bound depends on the relative position of the points $C_i - (C_k - \mu_3)$ and $C_i - \widetilde{x}_1$. Since $C_i - \widetilde{x}_1 \leq C_i - (C_k - \mu_3)$, if both are less than μ_2 then the best error bound is (24).

For the second case $x_1^* = C_i$ $(C_i \leq C_j)$, we choose:

$$\begin{cases} x_1^* = C_i \\ x_2^* = 0 \\ x_3^* = C_k - C_i \end{cases} \tag{26}$$

In this case the error bound is:

$$E \leq I_1(C_i) + I_2(0) + I_3(C_k - C_i) - I_1(\widetilde{x}_1) - I_3(\widetilde{x}_3) \tag{27}$$

For the last case $x_1^* = C_j$ $(C_j \leq C_i)$, we choose:

$$\begin{cases} x_1^* = C_j \\ x_2^* = C_i - C_j \\ x_3^* = C_k - C_j \end{cases} \tag{28}$$

In this case the error bound is:

$$E \leq I_1(C_j) + I_2(C_i - C_j) + I_3(C_k - C_j) - I_1(\widetilde{x}_1) - I_3(\widetilde{x}_3) \tag{29}$$

For the following numerical example, we calculate the overflow probability rate function for the real and the fictitious network. Let $C_i = 5.5kb/s$, $C_j = 7kb/s$ per source and C_k ranging from 7 to $25kb/s$ per source. All traffic sources are on-off Markov processes. For X_1, the bit rate in the on state is $8kb/s$. For X_2, the bit rate in the on state is $10kb/s$. For X_3 the bit rate in the on state is $20kb/s$. The average time for all traffic types are $0.5s$ on the on state and $1.5s$ in the off state. Since $\mu_1 = 2kb/s$ and $\mu_3 = 5kb/s$ the stability condition is $C_k > \mu_1 + \mu_3 = 7kb/s$. Using these values, the sufficient conditions (12) are:

$$\begin{cases} C_k \leq C_i - \mu_2 + \mu_3 = 8kb/s \\ C_k \leq C_j + \mu_3 = 12kb/s \end{cases}$$

The condition are satisfied for values of C_k less than $8kb/s$. Figure 7 shows that both functions match even after the condition is not satisfied and up to $C_k \simeq 15kb/s$. The reason is that $x^* = (\widetilde{x}_1, \mu_2, \widetilde{x}_3)$ is optimum for P_R. From this point the functions begin to separate. Figures 7 and 8 also shows the functions I_k^R, I_k^F and $I_k^F + E'$, where E' is the error bound. Until $C_k = 24kb/s$, E' is calculated using (23), and then using (26). It is important to note that when $C_k > 14kb/s$ per source, the link utilization falls to less than 50% and therefore, as it can be seen in figure 8 the rate function $I_1^{X^k}$ takes values bigger than 0.5. If for example the number of sources feeding the network is $N = 100$, the losses are near 10^{-22}. Finally, we have seen that the estimated error bound is tight and when the error is big, the link overflow probability is small and, therefore, these links are not relevant for the QoS evaluation.

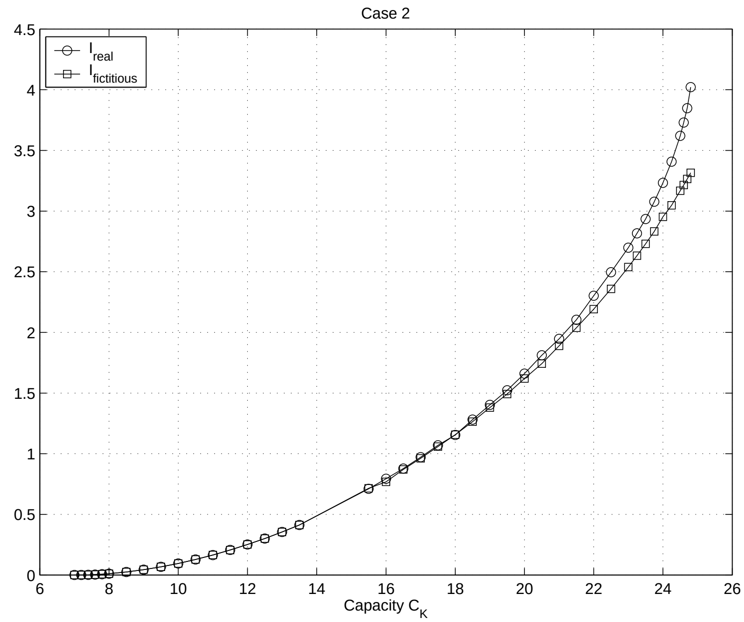

FIGURE 7

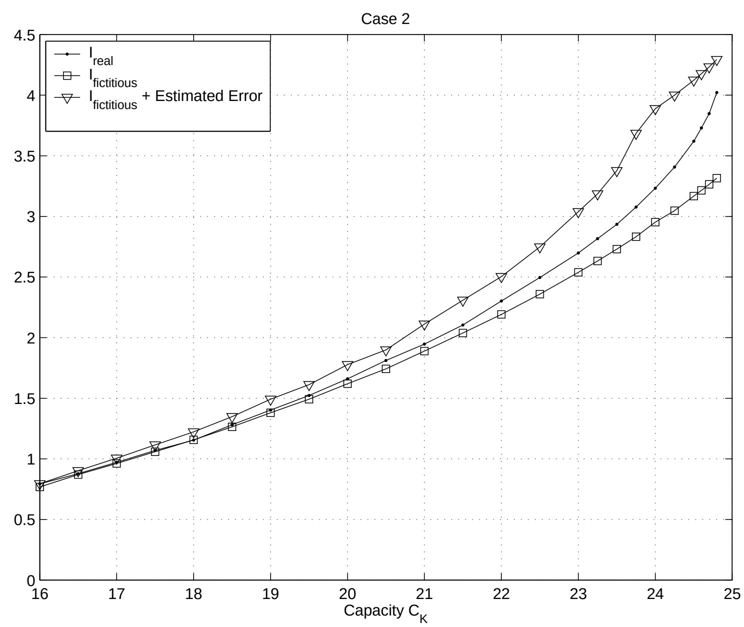

FIGURE 8

Example 5.3. *Consider a network like in figure 9. We analyse the overflow probability at link k, assuming that $C_i + C_j > C_k$.*

If condition (12) is attained for link k, then $E = I_k^R - I_k^F = 0$. This condition is:

$$\begin{cases} C_k - \mu_3 \leq C_i - \mu_1 \\ C_k - \mu_2 \leq C_j - \mu_4 \end{cases}$$

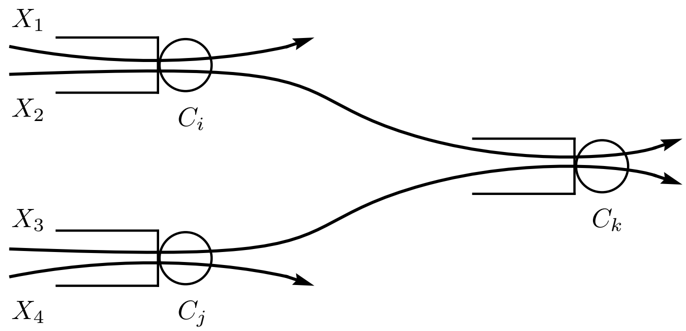

FIGURE 9

If this condition is not satisfied, since $\widetilde{x} = (\widetilde{x}_2, \widetilde{x}_3)$ is optimum for P_F, we first verify if $x^* = (\mu_1, \widetilde{x}_2, \widetilde{x}_3, \mu_4)$ is optimum for P_R. It sufficient to show that x^* verifies the real problem constraints, i.e:

$$\left\{ \begin{array}{l} \widetilde{x}_2 + \mu_1 \leq C_i \\ \widetilde{x}_3 + \mu_4 \leq C_j \\ \widetilde{x}_2 + \widetilde{x}_3 = C_k \end{array} \right. \tag{30}$$

If these conditions are not satisfied, we look at first for $x^* = (x_1^*, \widetilde{x}_2, \widetilde{x}_3, x_4^*)$ that satisfies

$$\left\{ \begin{array}{l} \widetilde{x}_2 + x_1^* \leq C_i \\ \widetilde{x}_3 + x_4^* \leq C_j \\ \widetilde{x}_2 + \widetilde{x}_3 = C_k \end{array} \right.$$

If $\widetilde{x}_2 > C_i$ or $\widetilde{x}_3 > C_j$ then it is not possible to choose such point. So, we look for $x^* = (x_1^*, x_2^*, x_3^*, x_4^*)$ that verifies

$$\left\{ \begin{array}{l} x_1^* + x_2^* = C_i \\ x_3^* + x_4^* = C_j \\ x_1^* + x_3^* = C_k \end{array} \right.$$

One possible choice is:

$$\left\{ \begin{array}{l} x_1^* = C_i \\ x_2^* = 0 \\ x_3^* = C_k - C_i \\ x_4^* = C_j - (C_k - C_i) \end{array} \right. \tag{31}$$

For the following numerical example, we calculate the overflow probability rate function for the real and fictitious network. Let $C_i = 12kb/s$, $C_j = 14kb/s$ per source and C_k growing from 8 to $25.5kb/s$ per source. All traffic sources are on-off Markov processes. For X_1, the bit rate in the on state is $20kb/s$. For X_2, the bit rate in the on state is $16kb/s$. For X_3, the bit rate in the on state is $16kb/s$. For X_4, the bit rate in the on state is $12kb/s$. The average times for all traffic types are $0.5s$ in the on state and $1.5s$ in the off state. Since $\mu_2 = 4kb/s$ and $\mu_3 = 4kb/s$ the stability condition is $C_k > \mu_1 + \mu_3 = 8kb/s$. Using these values, the sufficient condition (12) are:

$$\left\{ \begin{array}{l} C_k < C_i - \mu_1 + \mu_3 = 11kb/s \\ C_k < C_j - \mu 4 + \mu_2 = 15kb/s \end{array} \right.$$

Figures 10 and 11 shows that both functions match even after the condition is not satisfied and up to $C_k \simeq 15kb/s$. The reason is that $x^* = (\mu_1, \widetilde{x}_2, \widetilde{x}_3, \mu_4)$ is optimum for the real problem. From this point the functions begin to separate.

Figure 11 also shows the functions I_k^R, I_k^F and $I_k^F + E'$, where E' is the error bound. Until $C_k = 24kb/s$, E' is calculated using (30), and then using (31). As in the previous example, when $C_k > 16kb/s$, the link utilisation is less than 50% and that the error bound is tight for the relevant cases in the QoS evaluation.

All calculations of overflow probability rate functions were done with a software package developed by our group available in the web [3].

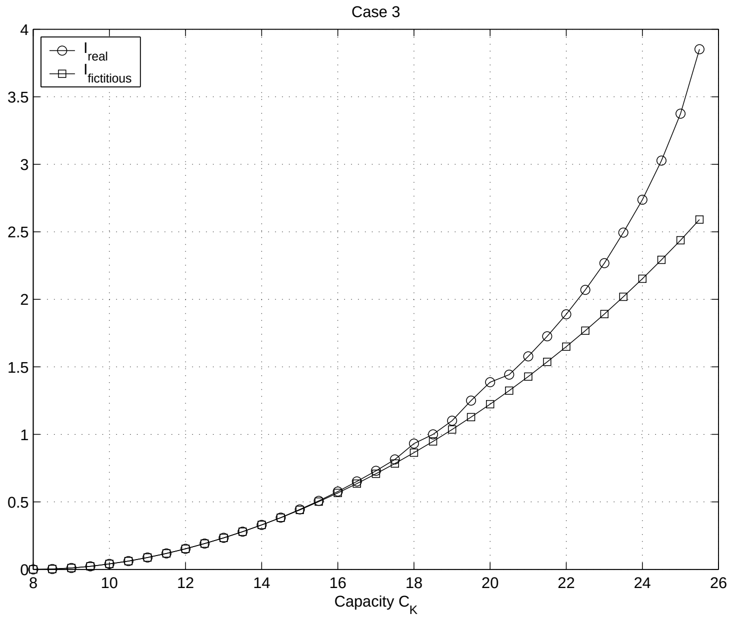

FIGURE 10

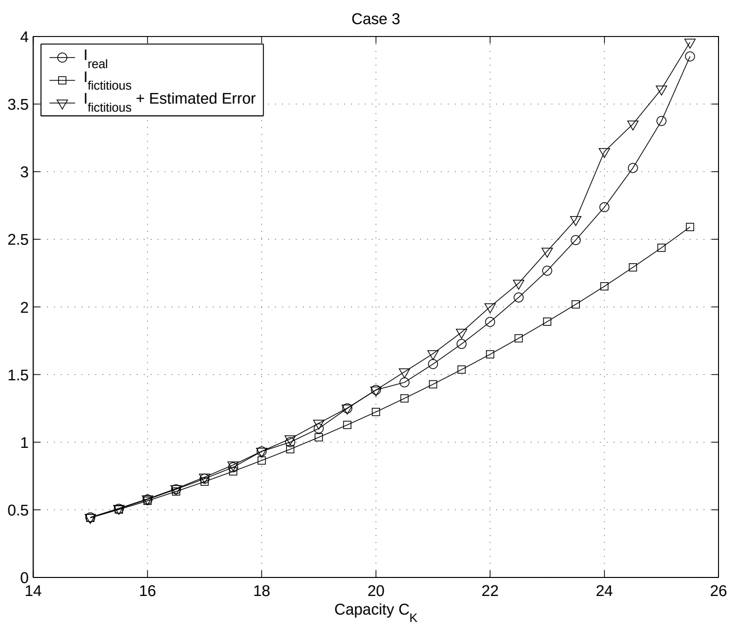

FIGURE 11

6. Conclusions

In this paper we have explained the fictitious network analysis. We have seen that the calculus of the overflow probability rate function of an interior network link is simpler and faster than the equivalent task in the real network. For this reason, on-line performance analysis and on-line traffic engineering will be easier (or even feasible instead of impossible) using the fictitious network analysis.

Generally, the fictitious network analysis overestimates the overflow probability and the end-to-end Loss Ratio. Therefore, this approach is safe but network resources can be under-utilized. To solve this problem we have found a condition that depends only on link´s capacities and mean traffic rates. If this condition is satisfied, the overflow probability calculated using the fictitious network has the same value that the one calculated in the real one. When this condition is not satisfied, the rate function of the link overflow probability calculated in the fictitious network can be smaller or equal than the same rate function calculated in the real network. We have shown that once the fictitious rate function is calculated, it is very simple to verify if both rate functions are equal or not. If they are not equal, we have found a simple algorithm to find an error bound.

In the numerical examples we have found that the error bound is tight. In these examples, it can be seen that when the error is big, the link overflow probability is very small and, therefore, these links are not relevant for the QoS evaluation. In spite of this, we can affirm that when the overflow probability at link k in the fictitious network is very small, even if the error is big, these link is not considered for the QoS evaluation.

7. Acknowledgements

The authors wish to thank the referees for their careful review and helpful comments of this work.

References

1. Ramon Casellas, *MPLS traffic engineering*, Ph.D. thesis, ENST, Paris, 2002.
2. C. Courcoubetis and R.Weber, *Buffer overflow asymptotics for a switch handling many traffic sources*, Journal of Applied Probability (1996).
3. A. Vazquez P. Belzarena D.Buschiazzo, A. Ferragut, *ARCA*, http://iie.fing.edu.uy/investigacion/grupos/artes/arca, 2004.
4. Amir Dembo and Ofer Zeitouni, *Large deviations techniques and applications*, 1993.
5. N.G. Duffield and Neil O'Connell, *Large deviations and overflow probabilities for the general single server queue, with applications*, Math. Proc. Cam. Phil. Soc., 118 (1995), 363–374.
6. D. Y. Eun and N. Shroff, *Network decomposition in the many-sources regime*, Advances in Applied Probability (submitted, 2002).
7. F.P. Kelly, *Notes on effective bandwidth*, Stochastic Networks: Theory and Applications, Oxford University Press (1996).
8. Sherali C. M. Shetty Mokhtar S., Bazaraa Hanif D., *Nonlinear programming theory and algorithms*, John Wiley & Sons Inc, 1993.
9. Ozturk O., Mazumdar R., and Likhanov N., *Many sources asymptotics in networks with small buffers*, Queueing Systems (QUESTA) (Vol.46, Nos. 1-2, 2004, pp.129-147).
10. E. Rosen and A. Viswanathan, *Multiprotocol label switching architecture, RFC3031*, IETF (January 2001).
11. A. Simonian and J. Guibert, *Large deviations approximations for fluid queues fed by a large number of on/off sources*, IEEE JSAC, 13(7) (August 1995.), 1017–1027.
12. Damon Wischik, *The output of a switch or effective bandwidths for networks*, Queueing Systems **32** (1999).
13. ———, *Sample path large deviations for queues with many inputs*, Annals of Applied Probability (2000).