A LAGRANGEAN RELAXATION APPROACH FOR IP NETWORK CFA PROBLEMS WITH END-TO-END QOS PERFORMANCE CONSTRAINTS

Emilio C. G. Wille, Marco Mellia, Emilio Leonardi, and Marco Ajmone Marsan

Abstract - The traditional approaches to optimal design and planning of packet-switching networks focus on the network-layer infrastructure, thus neglecting end-to-end Quality of Service (e2e QoS) issues, and Service Level Agreement (SLA) guarantees. This is quite inappropriate since the Internet today carries a wide range of critical telecommunication services. In this paper, we propose a packet network design and planning approach that considers the dynamics of packet networks, as well as the effect of protocols at the different layers of the Internet architecture on the e2e QoS experienced by end-users. Our proposed approach maps the end-user performance constraints into transport-layer performance constraints first, and then into network-layer performance constraints. This translating process is then considered together with a refined TCP/IP traffic modeling technique that is both simple and capable of producing accurate performance estimates for general-topology packet networks subject to realistic traffic patterns. We illustrate an example of its application to the optimization of link capacities and routing in a corporate VPN (Virtual Private Network) where traffic is mainly due to TCP connections. An efficient Lagrangean relaxation based heuristic procedure is developed to find bounds and solutions for the considered problem. Numerical results for a variety of problem instances are reported.

Keywords: Network design and planning, Queueing theory, Mathematical programming/optimization, Lagrangean relaxation.

Resumo - As abordagens tradicionais para o projeto de redes por chaveamento de pacotes concentram-se na infra–estrutura da camada de rede, ignorando assim garantias de qualidade de serviço (*Quality of Service* - QoS) para usuários finais e fatores como, por exemplo, o acordo de nível de serviço (*Service Level Agreement* - SLA). Esta abordagem é inapropriada, dado que a Internet oferece, hoje, diversos serviços de telecomunicações com restrições em termos de vazão, confiabilidade, retardo, etc. Neste artigo, propõe-se uma metodologia para o dimensionamento de redes de pacotes que considera a dinâmica de tráfego e o efeito dos protocolos nas diferentes camadas da pilha de protocolos na QoS experimentada pelo usuário final. A metodologia proposta mapeia restrições de desempenho para usuários finais, primeiramente em restrições da camada de transporte, e finalmente em restrições da camada de rede. A proposta baseia-se, também, em um modelo de rede que, apesar de simples, é capaz de estimar de forma acurada o desempenho de seus elementos sujeitos a padrões reais de tráfego. Ilustra-se a aplicação da metodologia no dimensionamento conjunto de *links* e roteamento em uma rede privada virtual (VPN) com tráfego tipo TCP. Uma heurística baseada no método da relaxação Lagrangeana é utilizada para a obtenção de limites e soluções viáveis para o problema. Diversos resultados numéricos são apresentados.

Palavras-chave: Projeto e planejamento de redes de telecomunicações, Teoria de filas, Otimização, Relaxação Lagrangeana.

1. INTRODUCTION

The new generation of packet-switching networks are expected to support a wide range of communication-intensive realtime multimedia applications. These applications will have their own different quality-of-service (QoS) requirements in terms of throughput, reliability, and bounds on end-to-end (e2e) delay, jitter, and packet loss ratio. It is technically a challenging and complicated problem to deliver multimedia information in a timely, synchronized manner over a decentralized, shared network environment, especially one that was originally designed for best-effort traffic such as the Internet.

Accordingly, a key issue in this area is how to devise reasonable packet–switching network design methodologies that allow the choice of the most adequate set of network resources for the delivery of a given mix of services with the desired level of e2e QoS and, at the same time, consider the traffic dynamics of today's packet networks. The traditional approaches to optimal design and planning of packet networks, extensively investigated in the early days of packet networks [1, 2], focus on the network-layer infrastructure thus neglecting e2e QoS issues, and Service Level Agreement (SLA) guarantees.

From the end-user's point of view, QoS is driven by end-to-end performance parameters, such as data throughput, web page latency, transaction reliability, etc. Matching the user-layer QoS requirements to the network-layer performance parameters is not a straightforward task. The QoS perceived by end-users in their access to Internet services is mainly driven by TCP, the reliable transport protocol of the Internet, whose congestion control algorithms dictate the latency of information transfer. Indeed, it is well known that TCP accounts for a

Emilio C. G. Wille is with the CEFET/PR - CPGEI, Av. Sete de Setembro 3165, CEP 80230-901, Curitiba-PR, Brazil. Fax: +55 41 310-4683. He was supported by a CAPES Foundation scholarship from the Ministry of Education of Brazil. e-mail: ecgwille@cefetpr.br.

M. Mellia, E. Leonardi, and M. Ajmone Marsan are with the Politecnico di Torino - DELEN, Corso Duca degli Abruzzi 24, I-10129, Torino, Italy. e-mail: {mellia, emilio, ajmone}@mail.tlc.polito.it.

great amount of the total traffic volume in the Internet [3, 4], and among all the TCP flows, a vast majority is represented by short-lived flows (also called mice), while the rest is represented by long-lived flows (also called elephants).

The description of traffic patterns inside the Internet is a particularly delicate issue, since it is well known that IP packets do not arrive at router buffers following a Poisson process; instead of a high degree of correlation exists [5]. Traditionally, either $M/M/1$ or $M/M/1/B$ queueing models were considered as good representations of packet queueing elements in the network. However, the traffic flowing in IP networks is known to exhibit Long Range Dependent (LRD) behaviors, which cause queue dynamics to severely deviate from the above model predictions. For these reasons, the usual approach of modeling packet networks as networks of $M/M/1$ queues [6, 7, 8] appears now inadequate for the design of such networks. Unfortunately, explicitly considering LRD traffic models is not practical. Indeed, queues driven by LRD processes are very difficult to study, and only asymptotic results exist. To the best of our knowledge, no closed–form expression exists for queues fed by LRD processes, which relates the queue performance to input parameters.

In this paper, we propose a packet network design and planning approach that considers the dynamics of packet networks, as well as the effect of protocols at the different layers of the Internet architecture on the e2e QoS experienced by end-users. Our proposed approach maps the end-user performance constraints into transport-layer performance constraints first, and then into network-layer performance constraints. A refined TCP/IP traffic modeling technique, already presented in [9], that is both simple and capable of producing accurate performance estimates for general-topology packet networks loaded by realistic traffic patterns, is considered. When explicitly considering TCP traffic it is also necessary to tackle the Buffer Assignment (BA) problem, for which we propose an efficient solution for the droptail case as well as for more advanced Active Queue Management (AQM) schemes, like RED [10].

Designing a packet network today may have quite different meanings, depending on the type of network that is being designed. If we consider the design of the physical topology of the network of a large Internet Service Provider (ISP), the design must very carefully account for the existing infrastructure, for the costs associated with the deployment of a new connection or for the upgrade of an existing link, and for the very coarse granularity in the data rates of high-speed links. Instead, if we consider the design of a corporate Virtual Private Network (VPN), where connections are leased from a long distance carrier, then the set of leased lines is not a critical legacy, costs are directly derived from the leasing fees, and the data rate granularity is much finer. While the general methodology for packet network design and planning that we describe in this paper can be applied to both contexts, as well as others, in this paper we concentrate on the design of corporate VPNs.

Traditionally, packet network design focused on optimizing either network cost or performance by tuning link capacities and routing strategies. Since the routing and link capacities optimization problems are closely interrelated, it is appropriate to jointly solve them in what is called the Capacity and Flow Assignment (CFA) problem. The literature focusing on the routing problem, where link capacities are assumed to be known, is abundant; see, for example, [2, 7, 11]. Papers where the routing and capacity assignment problems are treated simultaneously include [2, 6, 8, 12, 13].

Gersht and Weihmayer [8] presented a mixed integer/linear programming (MILP) formulation of the optimal network design and facility engineering problem, which corresponds to finding network topologies that minimize the total network cost while selecting facility types, allocating capacity, and routing traffic to accommodate traffic demands and performance requirements. The MILP formulation is decomposed into two subproblems, which can be solved sequentially. The solution of the first subproblem yields the topological design, facility selection, and flow assignment. The second subproblem corresponds to the capacity assignment. This work uses $M/M/1$ queueing systems to model the network behavior, and the average network-wide packet delay in the problem formulation.

Ng and Hoang [12] proposed a global optimal solution technique for the CFA problem. A continuous lower bound of the average network-wide packet delay is used in the formulation of the cost objective function. They consider an x-$M/M/1$ queueing system to model the network behavior (where a link is implemented by x transmission lines, each of capacity C); therefore, the objective function is shown to be convex with respect to the network multicommodity flow. The convexity property ensures the global optimal solution of the CFA problem, that is obtained using the Flow Deviation method [2].

Cheng and Lin [6] consider the problem of minimizing the maximum end-to-end delay in the network. They propose a two-phase algorithm to solve the CFA problem, where in a first phase a minimum-hop heuristic routing is used, and in a second phase the capacity assignment problem is solved. They also adopt $M/M/1$ queueing systems to model the network.

Medhi and Tipper [13] proposed four approaches based on the Lagrangean relaxation with subgradient optimization method and genetic algorithms to obtain solutions to a multihour combined capacity design and routing problem, neglecting however packet delay constraints.

Recently, in [14], the authors for the first time abandon the Markovian assumption in favor of a fractional Brownian motion model, i.e., an LRD traffic model. They solve the discrete capacity assignment problem under network e2e delay constraints only, using simulated annealing metaheuristic. However, it is difficult to extend this approach to consider more general CFA problems, because the relation among traffic, capacity and queueing delay is not expressed by a closed–form expression.

To the best of our knowledge, no previous work solves the CFA problem for packet networks taking into account user-layer e2e QoS constraints considering more realistic traffic models. In this paper, we present a nonlinear mixed-integer programming formulation for the CFA problem and solve it in the case of corporate VPNs. An efficient Lagrangean relaxation based heuristic procedure is developed to find bounds

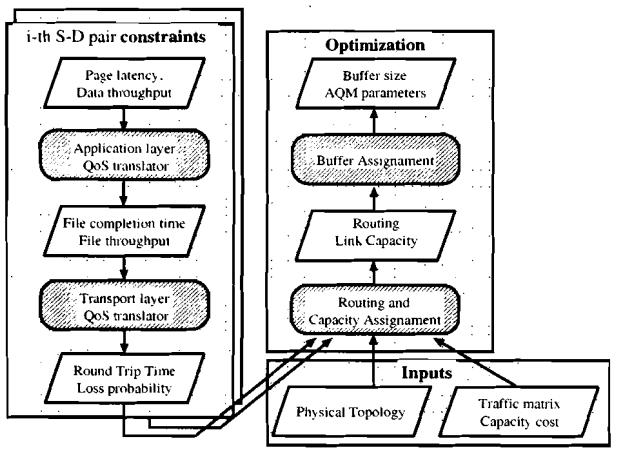

Figure 1. Schematic Flow Diagram of the Network Design Methodology

and solutions. Numerical results for a variety of problem instances are reported.

The rest of the paper is organized as follows. Section 2 describes the general design methodology and provides the formulation of the optimization problems. Section 3 illustrates a Lagrangean relaxation of the problem, as well as a heuristic solution procedure. Numerical results are discussed and compared against results of ns-2 simulations in Section 4. Conclusions are given in Section 5.

2. THE IP NETWORK DESIGN METHODOLOGY

The IP network design methodology that we propose in this paper is based on a "Divide and Conquer" approach, in the sense that it corresponds to several subtasks, which are solved separately. Such an approach is a necessity, because, even if the resulting methodology provides near-optimal solutions, the complexity of the problem makes a global solution impossible.

Fig. 1 shows the flow diagram of the design methodology. Shaded, rounded boxes represent function blocks, while white parallelograms represent input/output of functions. There are three main blocks, which correspond to the classic blocks in constrained optimization problems: *constraints* (on the left), *inputs* (on the bottom right) and *optimization procedure* (on the top right). As constraints we consider, for every source/destination pair, the specification of user-layer QoS parameters, e.g., download latency for web pages or perceived quality for realtime applications. Thanks to the definition of *QoS translators*, all the user-layer QoS constraints are mapped into lower-layer performance constraints, down to the network layer, where performance metrics are typically expressed in terms of average delay and loss probability.

The optimization procedure needs as inputs the description of the physical topology, the traffic matrix, and the expression of the cost as function of link capacities. The objective of the optimization is to find the minimum cost solution that satisfies the user-layer QoS constraints. The solution identifies link capacities, flow assignment (i.e. routing) and buffer sizes (or AQM parameters).

In our methodology we decouple the CFA problem from the BA problem. The optimization starts with the CFA subproblem, solved considering infinite buffers. A second optimization is then performed to solve the BA subproblem. Motivations for this choice are given in the following sections, where we briefly comment on the main steps of the design methodology, and we provide a formal description for the optimization problems.

2.1 QOS TRANSLATORS

The process of translating QoS specifications between different layers of the protocol stack is called QoS translation. According to the Internet protocol architecture, at least two QoS mapping procedures should be considered in our case; the first translates the application-layer QoS constraints into transport-layer QoS constraints, and the second translates transport-layer QoS constraints into network-layer QoS constraints.

2.1.1 APPLICATION-LAYER QOS TRANSLATOR

This module takes as inputs the application-layer QoS constraints, such as web page transfer latency, data throughput, audio quality, etc. Assuming that for each application we know which transport protocol is used, i.e., either TCP or UDP, this module maps the application-layer QoS constraints into transport-layer QoS constraints. Given the multitude of Internet applications it is not possible to devise a generic procedure to solve this problem. Hence, in this paper, we will focus on ad-hoc solutions depending on the application.

Real Time Applications - UDP For realtime applications over UDP, the output of the application-layer translator is given in terms of packet loss probability, and maximum network e2e delay. Considering *Voice over UDP*, high-level QoS constraints, such as the Mean Opinion Score (MOS), are expressed in terms of transport-layer performance constraints. For example, good vocal perceived quality is associated with an average packet loss probability of the order of 1%, and a maximum e2e delay smaller than 100 ms [16]. This case will not be considered further in this work.

Elastic Traffic - TCP For elastic applications exploiting TCP, the output of the application-layer translator is still a set of high-level constraints, expressed as *file transfer latency* (L_t), or *throughput* (T_h). Considering *Web page download*, a desired download time is expressed in terms of TCP latency (or throughput) constraint. For example, given a desired web page download time smaller than 1.5 s, a web page which contains 20 objects, downloaded using 4 parallel TCP connections at most, each object must be transferred with a TCP connection of average duration smaller than 0.3 s.

2.1.2 TRANSPORT-LAYER QOS TRANSLATOR

The transport-layer QoS translator maps transport-layer performance constraints into network-layer performance constraints; the translator in this case must be tailored to the transport protocol used: either UDP or TCP.

Real Time Applications - UDP The translation from transport-layer performance constraints into network-layer performance constraints in the case of real-time UDP applications is rather straightforward, since the transport-layer performance constraints are usually expressed in terms of packet loss probability and maximum e2e network delay, which can be directly used also as network-level performance parameters. The only effect of UDP that must be taken into account is related to the protocol overhead, which increases the offered load to the network. This effect may be significant, specially for applications like voice, that use small packets. Experiments with UDP will be considered in a future work.

Elastic Traffic - TCP The translation from transport-layer QoS constraints to network-layer QoS parameters, such as *Round Trip Time* (RTT) and *Packet Loss Probability* (P_{loss}), is more difficult. This is mainly due to the complexity of the TCP protocol, and in particular to the error, flow and congestion control algorithms. The TCP QoS translator accepts as inputs either the maximum file transfer latency, or the minimum file transfer throughput. We impose that all flows shorter than a given threshold (i.e., TCP mice) meet the maximum file transfer latency constraint, while longer flows (i.e., TCP elephants) are subjected to the throughput constraint. For example, from measurements of the *flow length distribution* over the Internet [4], it is possible to say that 85% of all TCP flows are shorter than 20 segments. For these flows, we impose that the latency constraint must hold. Instead, for flows longer than 20 segments we impose that the throughput constraint must be met. Obviously, the most stringent constraint must be considered in the translation. The maximum RTT and P_{loss} that satisfy both constraints constitute the output of this translator.

To solve the translation problem, we exploit recent research results in the field of TCP modeling. Usually, TCP models take network-layer parameters as inputs, i.e., RTT and P_{loss}, and give as output either the average throughput or the file transfer latency. Our approach is based on the inversion of two TCP models, taking as input either the connection throughput or the file transfer latency, and obtaining as outputs RTT and P_{loss}. When considering file transfer latency, we use the TCP latency model described in [17], which offers a good tradeoff between computational complexity and accuracy of performance predictions. We will refer to this model as the CSA model (from the last names of the authors). When considering throughput, we instead exploit the formula given in [18]. We will refer to this formula as the PFTK formula (from the last names of the authors).

The inversion of TCP models is not simple, since there are at least two parameters that impact TCP throughput and latency, i.e., RTT and P_{loss}. An infinite number of possible solutions for these two parameters satisfies a given constraint at the TCP level. We decided therefore to fix the P_{loss} parameter, and leave RTT as the free variable. This choice is due to the fact that the loss probability has a larger impact on the latency of very short flows, and that it impacts the network load due to retransmissions. Furthermore, P_{loss} is also constrained by realtime applications. Finally, fixing the value of the loss probability allows us to decouple the CFA problem from the BA problem, as shown in Section 2.2.1. Therefore, after choosing a value for P_{loss}, a set of curves can be derived, showing the behavior of RTT as a function of file latency and throughput. From these curves it is then possible to derive the maximum allowable RTT. The inversion of the CSA and PFTK formulas is obtained using numerical algorithms.

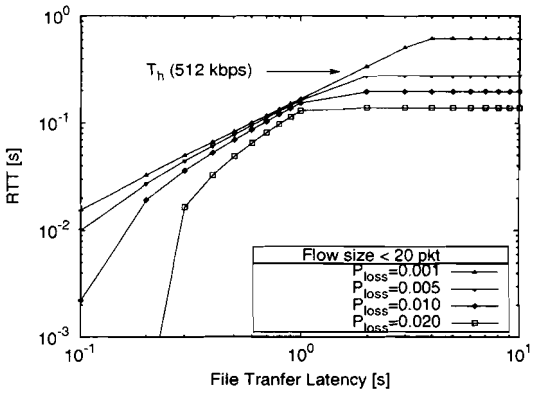

Figure 2. Maximum admissible RTT which satisfies the most stringent constraint (latency or throughput) for different values of P_{loss} (considering 20-packet flows)

For example, given a maximum file transfer latency and a minimum throughput $T_h = 512$ kbps constraint, the curves of Fig. 2 report the maximum admissible RTT which satisfies the most stringent constraint for different values of P_{loss}.

2.2 PROBLEM DESCRIPTION AND FORMULATION

A network is modeled as a directed, connected graph $G = (V, E)$ where V is a finite set of vertices (network nodes) and E is the set of edges (network links) representing connection of these vertices. Let n be the number of network nodes and m be the number of network links. A path from source node s to a destination node d is a sub-graph of G and it is modeled as an open network of queues, where each queue represents an output interface of an IP router with its buffer. Four non-negative real value functions are associated with each link: link cost, buffer cost, average packet delay, and average packet loss probability. The link and buffer cost functions may be either monetary cost or any measure of the resource utilization, which must be optimized. The average packet delay is considered to be the sum of queueing and transmission delays. The average packet delay and average packet loss probability functions define the criteria that must be constrained (bounded).

2.2.1 TRAFFIC AND QUEUEING MODEL

In order to obtain a useful formulation of the CFA problem, it is necessary on one side to be accurate in the prediction of the performance metrics of interest (average delay, packet loss probability), while on the other side to adopt a low complexity model (i.e., we adopt models allowing a simple closed-form solution).

In [9], a simple and quite effective expedient was proposed to accurately predict the performance of network elements subject to TCP traffic, using Markovian queueing models. The main idea behind the approach in [9] corresponds to reproduce the effects of traffic correlations on network queueing elements by means of Markovian queueing models with *batch arrivals*. The choice of using batch arrivals following a Poisson process has the advantage of combining the nice characteristics of Poisson processes (analytical tractability in the first place) with the possibility of capturing the burstiness of the IP traffic. Hence, we model network queueing elements using $M_{[X]}/M/1$ queues. The batch size varies between 1 and W with distribution $[X]$, where W is the maximum TCP window size expressed in segments. The distribution $[X]$ is obtained considering the number of segments that TCP sources send in one RTT for a given flow length distribution. The Markovian assumption for the batch arrival process is mainly justified by the Poisson assumption for the TCP connection generation process (when dealing with TCP mice), as well as the fairly large number of TCP connections simultaneously present in the network. Given the flow length distribution, a stochastic model of TCP (described in [9]) is used to obtain the batch size distribution $[X]$. The distribution $[X]$ is obtained only once before starting the optimization process.

The average (busy-hour) traffic requirements between nodes can be represented by a requirement matrix $\hat{\Gamma} = \{\hat{\gamma}_{sd}\}$, where $\hat{\gamma}_{sd}$ is the average packet transfer rate from source s to destination d. The $\hat{\Gamma}$ matrix can be derived from a higher-level description of the (maximum) traffic requests, expressed in terms of "pages per second", or "flows per second" for a given source/destination pair. We consider as traffic offered to the network $\gamma_{sd} = \frac{\hat{\gamma}_{sd}}{1-P_{loss}}$, to take into account the retransmissions due to the losses that flows experience along their path to the destination. Recall that P_{loss} is the desired e2e loss probability.

The decision of fixing *a-priori* the loss probability allows us to decouple the CFA solution from the BA solution. We first solve the CFA problem (properly selecting the capacity of links and routing of flows) considering the e2e delay constraints only. Then, we enforce the loss probability to meet the P_{loss} constraints by properly choosing buffer sizes. In the first optimization, a $M_{[X]}/M/1/\infty$ queueing model will be used, i.e., a queueing model with infinite buffers. This provides a pessimistic estimate of the queueing delay that packets suffer with finite buffers, which will results from the second optimization step, during which an $M_{[X]}/M/1/B$ queueing model is used.

The following notation is necessary for developing a mathematical model for the CFA and BA problems:

C_{ij}	the capacity of link (i,j).
f_{ij}	the average data flow on link (i,j).
d_{ij}	the physical length of link (i,j).
RTT_{sd}	the Round Trip Time of path (s,d).
B_{ij}	the buffer size on link (i,j).
δ_{ij}^{sd}	auxiliary variables (which are one if link (i,j) is on the path (i,j), and zero otherwise).

CFA formulation Different formulations of the CFA problem result by selecting i) the cost functions $g(C_{ij})$, ii) the routing model, and iii) the capacity constraints. It is important to note that for a given set of options a specific optimization technique must be applied to solve the problem. In this paper we focus on the VPN case, in which common assumptions are i) linear cost, i.e., $g(C_{ij}) = d_{ij}C_{ij}$, ii) non-bifurcated routing, and iii) continuous capacities. Solution techniques for this subcase are presented in Section 3. Optimization techniques for the design of physical topologies, i.e., the non-continuous capacity case, are currently being investigated.

Our goal is to minimize the total link costs while determining the best route for the traffic that flows on each source/destination path, and meeting the maximum e2e packet delay constraint. The following optimization problem is thus formulated:

$$Z_{CFA} = \min \sum_{i,j} g(C_{ij}) \tag{1}$$

subject to:

$$\sum_j \delta_{ij}^{sd} - \sum_j \delta_{ji}^{sd} = \begin{cases} 1 & \text{if } i = s \\ -1 & \text{if } i = d \\ 0 & \text{otherwise} \end{cases} \quad \forall\,(i,s,d) \tag{2}$$

$$K_1 \sum_{i,j} \frac{\delta_{ij}^{sd}}{C_{ij} - f_{ij}} \leq RTT_{sd} - K_2 \sum_{i,j} \delta_{ij}^{sd} d_{ij} \qquad \forall\,(s,d) \tag{3}$$

$$f_{ij} = \sum_{s,d} \delta_{ij}^{sd} \gamma_{sd} \qquad \forall\,(i,j) \tag{4}$$

$$\delta_{ij}^{sd} \in \{0,1\} \quad \forall\,(i,j),\ \forall\,(s,d); \quad C_{ij} \geq f_{ij} \geq 0 \quad \forall\,(i,j) \tag{5}$$

The objective function (1) represents the total link cost, which is the sum of the cost functions of link (i,j), $g(C_{ij})$. Constraint set (2) contains the flow conservation equations, which define routes for the traffic of each source/destination pair. The formulation considers non-bifurcated routing, i.e. the traffic of a given source/destination pair follows one path. Constraints (3) are the e2e packet delay constraints for each source/destination pair. It says that the total amount of delay experienced by all the flows routed on a path should not exceed the maximum RTT minus the propagation delay of the route. The average queueing delay is expressed by considering an $M_{[X]}/M/1/\infty$ queue [19]:

$$E[T] = \frac{K}{\mu} \frac{1}{C - f} \tag{6}$$

$$K = \frac{m'_{[X]} + m''_{[X]}}{2m'_{[X]}} \tag{7}$$

where $m'_{[X]}$ and $m''_{[X]}$ are the first and second moments of the batch size distribution $[X]$ and $1/\mu$ is the average packet length (which is assumed to be exponentially distributed [1, 2]).

Equation (4) defines the average data flow on the link. Constraints (5) are integrity and non-negativity constraints. Finally, $K_1 = K/\mu$, and K_2 is a constant to convert distance in time.

BA formulation As final step in our methodology, we need to dimension buffer sizes, i.e., to solve the following problem:

$$Z_{BA} = \min \sum_{i,j} h(B_{ij}) \tag{8}$$

Subject to:

$$\sum_{ij} \delta_{ij}^{sd} p(B_{ij}, C_{ij}, f_{ij}, [X]) \leq P_{loss} \quad \forall\ (s,d) \tag{9}$$

$$B_{ij} \geq 0 \quad \forall\ (i,j) \tag{10}$$

The objective function (8) represents the total buffer cost, which is the sum of the cost functions $h(B_{ij}) = B_{ij}$. Equation (9) is the loss probability constraint for each source/destination pair. Constraints (10) are non-negativity constraints. $p(B_{ij}, C_{ij}, f_{ij}, [X])$ is the average loss probability for the $M_{[X]}/M/1/B$ queue, which is evaluated by solving its Continuous Time Markov Chain (CTMC) model.

In the previous formulation we have considered the following upper bound on the value of P_{loss} (constraint (9)).

$$\begin{aligned} \hat{P}_{loss} = \ & 1 - \prod_{i,j} \left(1 - \delta_{ij}^{sd} p(B_{ij}, C_{ij}, f_{ij}, [X])\right) \\ & \leq \sum_{ij} \delta_{ij}^{sd} p(B_{ij}, C_{ij}, f_{ij}, [X]) \end{aligned} \tag{11}$$

Notice also that the first part of equation (11) is based on the assumption that link losses are independent. Therefore, the solution of the BA problem is a conservative solution to the full problem.

The proof that the BA problem is a convex optimization problem is not a straightforward task. The difficulty in this proof derives from the need of showing that $p(B, C, f, [X])$ is convex. Since, to the best of our knowledge, no closed-form expression for the $M_{[X]}/M/1/B$ stationary distribution is known, no closed-form expression for $p(B, C, f, [X])$ can be derived. However, we conjecture that the BA problem is a convex optimization problem by considering that: i) for an $M/M/1/B$ queue, $p(B, C, f)$ is a convex function (see [20]); and ii) approximating $p(B, C, f, [X]) = \sum_{i=B}^{\infty} \pi_i$, where π_i is the stationary distribution of an $M_{[X]}/M/1/\infty$ queue, the loss probability is a convex function of B. We can thus classify the BA problem as a multivariable constrained convex minimization problem; therefore, the global minimum can be found using convex programming techniques. We solve this minimization problem applying first a constraints reduction procedure which reduces the set of constraints by eliminating redundancies. Then, the solution of the BA problem is obtained via the *logarithm barrier method* [21] (it gives a solution whose accuracy is known *a priori*).

Setting the AQM parameters The output of the BA problem is the buffer size B_{ij} for each router interface, assuming a droptail behavior. If more advanced AQM schemes are deployed by network providers to enhance the TCP performance, it is possible to derive guidelines for the configuration of the AQM parameters as well. In this paper, we consider Random Early Detection (RED) [10] as an example, and discuss how to set its parameters.

The original RED algorithm has three static parameters min_th, max_th, max_p, and one state variable avg. When the average queue length avg exceeds min_th, an incoming packet is dropped with a probability that is a linear function of the average queue length. In particular, the packet dropping probability increases linearly form 0 to max_p, as avg increases from min_th to max_th. When the average queue size exceeds max_th, all incoming packets are dropped.

Ideally, the buffer size should be sufficiently large to avoid that packets are dropped at the queue due to buffer overflow. Therefore, we choose $B_{ij} = \alpha.max\_th$, $\alpha > 1$, e.g., $\alpha = 2$ as suggested in the "gentle" variation of RED.

Therefore, the RED parameter dimensioning problem can be solved by imposing that:

$$p(B_{ij}, C_{ij}, f_{ij}, [X]) = \frac{E_{ij}[N] - min\_th_{ij}}{max\_th_{ij} - min\_th_{ij}} max\_p_{ij} \tag{12}$$

Note that (12) fixes max_p_{ij} by imposing that the average RED dropping probability evaluated at the average queue length $E_{ij}[N]$ (obtained considering the $M_{[X]}/M/1/B$ queue) satisfies the P_{loss} constraint in (9). Finally, we set $min\_th_{ij} = \beta.max\_th_{ij}$, $\beta < 1$. In the numerical examples discussed in this work, we selected $\alpha = 2$, $\beta = 1/16$ (these values produced the best results in our tests).

3. CFA PROBLEM: THE VPN CASE

The resulting CFA problem is a nonlinear mixed-integer programming problem, which is difficult in general. Except for the nonlinear constraint (3), this is basically a multicommodity flow problem [22], since each source/destination pair transmits a different quantity of traffic over the network. Multicommodity flow problems belong to the class of NP-hard problems. In [6] it is proved that also the continuous relaxation of the integrity constraints (5) leads to a non-convex programming problem by verifying the Hessian of (3). As a consequence of this property, in general, several local minima exist.

In the following, we propose a composite upper and lower bounding procedure based on a Lagrangean relaxation of the problem. A Lagrangean relaxation is created by removing (relaxing) a set of constraints, weighting them with Lagrangean multipliers and then placing them in the objective function. The purpose is to obtain a relaxed problem, called Lagrangean subproblem, which is easier to solve than the original problem. The objective value from the Lagrangean relaxation problem, for any given set of multipliers, provides a lower bound (in the case of minimization) for the optimal solution to the original problem. The best lower bound can be derived by solving the Lagrangean dual. Since the dual

function most often is non-differentiable there is a need to use a special method for this class of problems. A frequently and efficient method is subgradient optimization [23]. Information obtained from the Lagrangean relaxation is then often used by application-dependent heuristics to construct feasible solutions and hence upper bounds to the original problem.

3.1 LAGRANGEAN RELAXATION

The CFA problem is complicated by the nonlinear constraints (3). We first apply the change of variable $w_{ij} = \frac{1}{C_{ij}-f_{ij}}$, obtaining:

$$Z_{CFA} = min\left(\sum_{i,j}\frac{d_{ij}}{w_{ij}} + \sum_{i,j}\sum_{s,d} d_{ij}\delta_{ij}^{sd}\gamma_{sd}\right) \quad (13)$$

subject to:

$$K_1\sum_{i,j}\delta_{ij}^{sd}w_{ij} + K_2\sum_{i,j}\delta_{ij}^{sd}d_{ij} \leq RTT_{sd} \qquad \forall\,(s,d) \quad (14)$$

$$w_{ij} \geq 0 \qquad \forall\,(i,j) \quad (15)$$

$$\delta_{ij}^{sd} \in \{0,1\} \qquad \forall\,(i,j),\ \forall\,(s,d) \quad (16)$$

and (2).

Our next step toward obtaining a lower bound on the cost of the full problem is to linearize constraints (14) (by using a logical constraint). We use auxiliary variables w_{ij}^{sd} (whose dimension is seconds per bit) for each link (i,j) on path (s,d). Thus we have the equivalent problem:

$$Z_{CFA} = min\left(\sum_{i,j}\frac{d_{ij}}{w_{ij}} + \sum_{i,j}\sum_{s,d} d_{ij}\delta_{ij}^{sd}\gamma_{sd}\right)$$

subject to:

$$w_{ij}^{sd} \leq M_{sd}\delta_{ij}^{sd} \qquad \forall\,(i,j),\ \forall\,(s,d) \quad (17)$$

$$K_1\sum_{i,j} w_{ij}^{sd} + K_2\sum_{i,j}\delta_{ij}^{sd}d_{ij} \leq RTT_{sd} \qquad \forall\,(s,d) \quad (18)$$

$$w_{ij}^{sd} \geq 0 \qquad \forall\,(i,j),\ \forall\,(s,d) \quad (19)$$

and (2), (15), (16).

Note that constraints (17) force the packet delay of link (i,j) on path (s,d) to be 0 if the link is not used. The constant M_{sd} corresponds to the minimum value of w_{ij}^{sd} that is able to satisfy the packet delay constraints for path (s,d). We have $M_{sd} = RTT_{sd}/K_1$. We refer to this problem as problem P in the rest of this paper.

Feasible solutions as well as lower bounds for the optimal solution of problem P, can be obtained by using Lagrangean relaxation. First, constraints in (17) and (18) are relaxed, and the corresponding Lagrangean problem is constructed; next, a subgradient optimization procedure is used in order to improve the quality of the Lagrangean lower bound.

Consider the Lagrangean relaxation of problem P obtained by dualizing constraints (17) and (18) using the non-negative multipliers α_{ij}^{sd} and β_{sd}, respectively.

$$\begin{aligned} L(\alpha,\beta) \quad = min\Bigg\{ & \sum_{i,j}\frac{d_{ij}}{w_{ij}} + \sum_{i,j}\sum_{s,d} d_{ij}\delta_{ij}^{sd}\gamma_{sd} \\ & + \sum_{s,d}\sum_{i,j}\alpha_{ij}^{sd}\left(w_{ij}^{sd} - M_{sd}\delta_{ij}^{sd}\right) \\ & + \sum_{s,d}\beta_{sd}\left(\sum_{i,j}(K_1 w_{ij}^{sd} + K_2\delta_{ij}^{sd}d_{ij}) - RTT_{sd}\right)\Bigg\} \end{aligned} \quad (20)$$

subject to (2), (15), (16) and (19).

Problem $L(\alpha,\beta)$ can now be decomposed into two independent subproblems as follows:

Subproblem $L_1(\alpha,\beta)$:

$$L_1(\alpha,\beta) = min\sum_{s,d}\sum_{i,j}\delta_{ij}^{sd}(d_{ij}\gamma_{sd} - M_{sd}\alpha_{ij}^{sd} + K_2 d_{ij}\beta_{sd}) \quad (21)$$

subject to (2) and (16).

This subproblem can be further decomposed into $n \times (n-1)$ shortest path problems (one for each source/destination pair) and solved using the classic Bellman-Ford algorithm.

Subproblem $L_2(\alpha,\beta)$:

$$\begin{aligned} L_2(\alpha,\beta) = \quad & min\sum_{i,j}\left(\frac{d_{ij}}{w_{ij}} + \sum_{s,d} w_{ij}^{sd}(\alpha_{ij}^{sd} + K_1\beta_{sd})\right) \\ & - \sum_{s,d}\beta_{sd}RTT_{sd} \end{aligned} \quad (22)$$

subject to (15) and (19).

This subproblem can be further decomposed into m independent subproblems (one for each link) in the following way. Let the auxiliary variables y_{ij}^{sd} be seen as estimates of the network routing. Thus, the variables w_{ij}^{sd} can be substituted by $w_{ij} \times y_{ij}^{sd}$, obtaining:

$$L_2^{(i,j)}(\alpha,\beta) = min\left(\frac{d_{ij}}{w_{ij}} + w_{ij}\sum_{s,d} y_{ij}^{sd}(\alpha_{ij}^{sd} + K_1\beta_{sd})\right) \quad (23)$$

where the unknown variables are w_{ij} and y_{ij}^{sd}.

Subproblem $L_2^{(i,j)}(\alpha,\beta)$ is minimized by minimizing $\sum_{s,d} y_{ij}^{sd}(\alpha_{ij}^{sd} + K_1\beta_{sd})$. It is straightforward to see that at least one variable y_{ij}^{sd} must be 1, for all (s,d); otherwise w_{ij} tends to infinity. The solution corresponds to setting $y_{ij}^{sd} = 1$ for (s,d) of minimum $(\alpha_{ij}^{sd} + K_1\beta_{sd})$, and $y_{ij}^{sd} = 0$ otherwise. Finally, the optimal values of w_{ij} are given by:

$$w_{ij}^{*} = \sqrt{\frac{d_{ij}}{\sum_{s,d} y_{ij}^{sd}(\alpha_{ij}^{sd} + K_1\beta_{sd})}} \quad (24)$$

3.2 SOLVING THE LAGRANGEAN DUAL PROBLEM

The Lagrangean Dual problem typically produces solutions that after recovering primal feasibility tend to be close to optimal. Like for all relaxation procedures, the success of the approach depends mainly on the ability to generate good Lagrangean multipliers. In order to solve the Lagrangean dual problem, we employ a subgradient algorithm to search for "good" multipliers, while to recover primal feasibility we employ a heuristic.

The value of the Lagrangean for any set of multipliers $v = (\alpha, \beta)$ will be equal to the sum of the optimal solutions to the subproblems, $L(v) = L_1(v) + L_2(v)$. It is well known from optimization theory, by using the weak Lagrangean duality theorem [24], that for any vector of multipliers, $L(v)$ is a lower bound for the objective function value of the original problem, i.e., $L(v) \leq Z_{CFA}; \forall\, v \geq 0$. We are interested in obtaining the tightest possible lower bound, i.e., in the multipliers vector v^*, that corresponds to $L(v^*) = max_v\{L(v)\}$ (the Lagrangean Dual problem).

3.3 SUBGRADIENT OPTIMIZATION PROCEDURES

Subgradient method is a common approach for solving Lagrangean Dual problems. It can be described as follows. Given a set of Lagrange multipliers, the relaxed problem is solved, generating a lower bound, and subgradients associated with relaxed constraints are calculated. Then, the subgradients are used to update the set of Lagrange multipliers, in order to obtain a better lower bound. This procedure is repeated until lower and upper bounds are equal (or almost equal) or a duality gap is detected. For problem P, given a relaxed problem solution $(w_{ij}^{sd}, \delta_{ij}^{sd})$, the subgradient ξ_{ij}^{sd} can be calculated as:

$$\xi_{ij}^{sd} = \left(w_{ij}^{sd} - M_{sd}\delta_{ij}^{sd},\ \sum_{i,j}(K_1 w_{ij}^{sd} + K_2 \delta_{ij}^{sd} d_{ij}) - RTT_{sd} \right) \tag{25}$$

During p–th iteration, a new set of Lagrange multipliers is obtained by:

$$v^{p+1} = max\{0, v^p + t^p \xi^p\} \tag{26}$$

where t^p is a positive scalar (step size) and ξ^p is the subgradient.

The convergence of subgradient method is closely related to the step size, t^p. And, in this work, we have adopted a traditional approach [23]:

$$t^p = s^p \frac{\overline{Z_{FCA}} - L(v^p)}{\| \xi^p \|^2} \tag{27}$$

where the relaxation parameter s^p varies between 2 and 0. $\overline{Z_{FCA}}$ is the value of the best feasible solution found so far (upper bound), and $L(v^p)$ represents the value of lower bound calculated in p-th iteration. We update the value of $\overline{Z_{CFA}}$ using feasible solutions from a primal heuristic algorithm (see the next subsection). Finally, $\| \xi^p \|^2$ is a norm, usually Euclidian norm, of subgradient ξ^p. The value of s^p should be updated during the procedure. First s^p is set to 2 and, if there is no lower bound improvement in $MaxImp$ iterations, s^p is divided by 2. But every time that an lower bound improvement is found, s^p is set to 2 again.

In practice, there are several stopping criteria that may be used to terminate the algorithm. If $\| \xi^p \|^2 \leq \epsilon$ or $t^p \leq \epsilon$, for some very small $\epsilon > 0$, the algorithm should stop, since v^p is not varying enough. It is also possible to stop the procedure if the difference between the best upper and lower bounds is smaller than a pre-specified percentage. We also use a maximum number of iterations ($MaxItr$) as stopping criterion.

For the forthcoming analysis, the parameters were set as follows: v^0 is a vector of random numbers (between 0.1 and 10), $MaxImp = 20$, $MaxItr = 500$, and $\epsilon = 10^{-3}$.

3.4 OBTAINING FEASIBLE SOLUTIONS

Because of the used decision variables and stopping criterion, the solution to the dual problem is generally associated with an infeasible project, i.e., some of the end-to-end packet delay constraints and/or routing constraints may be violated.

Hence, we use information obtained from the Lagrangean relaxation, at each of the iterations of the subgradient procedure, to construct a feasible solution (Primal Heuristic). At each iteration, we test if the routing obtained from the solution of subproblem $L_1(v)$ can generate a feasible solution to the primal problem P. The test corresponds to verifying whether RTT is strictly greater than $K_2 \sum_{i,j} \delta_{ij}^{sd} d_{ij}$ for all source/destination pairs; in this case the values for w_{ij}^{sd} can be obtained. The algorithm stops if no feasible solution can be found, so that the requirements of the dimensioning problem must be relaxed.

Therefore, given the routing, a capacity assignment (CA) solver provides the values for the C_{ij}. We apply two techniques to solve the CA problem: i) a approximate solution which is described in the next subsection; and ii) a second approach using the logarithmic barrier method [21]. Consequently, the value of the primal objective function can be obtained. As the iteration progresses, we check for decreases in the primal objective value, and store the best primal solution, and the best primal cost so far.

3.5 APPROXIMATE SOLUTION TO THE CA PROBLEM

If we assume that the routing is known, problem P reduces to the following CA problem (where the second part of the objective function is now constant).

$$Z_{CA} = min\left(\sum_{i,j} \frac{d_{ij}}{w_{ij}} + \sum_{i,j} \sum_{s,d} d_{ij} \delta_{ij}^{sd} \gamma_{sd} \right)$$

subject to:

$$\sum_{i,j} w_{ij}^{sd} \leq b_{sd} \qquad \forall\, (s,d) \tag{28}$$

and (19).
where:

$$b_{sd} = \frac{1}{K_1}\left(RTT_{sd} - K_2 \sum_{i,j} \delta_{ij}^{sd} d_{ij}\right) \qquad \forall\,(s,d) \quad (29)$$

A simple heuristic can be derived to obtain solutions to this problem. The main idea is to decompose the problem into $n \times (n-1)$ single constrained problems (one for each path (s,d)). Let I_{sd} be the set of links which compose path (s,d). To solve each single path problem we apply the Lagrangean multiplier method obtaining:

$$L^{(s,d)}(\psi) = min\left(\sum_{(i,j)\in I_{sd}} \frac{d_{ij}}{w_{ij}^{sd}} + \psi\Big(\sum_{(i,j)\in I_{sd}} w_{ij}^{sd} - b_{sd}\Big)\right) \quad (30)$$

subject to (19).

Now it is necessary to minimize equation (30) with respect to the variables w_{ij}^{sd}. Upon differentiation and simplification we obtain:

$$w_{ij}^{sd} = \sqrt{\frac{d_{ij}}{\psi}} \quad (31)$$

The substitution of equation (31) into (28) (considering the equality) yields the Lagrangean multiplier ψ. By turning back to the equation (31) we obtain the solutions:

$$w_{ij}^{sd} = \frac{b_{sd}\sqrt{d_{ij}}}{\sum_{(k,l)\in I_{sd}} \sqrt{d_{kl}}} \quad (32)$$

Knowing the values for the variables w_{ij}^{sd} (in the single path problem) we obtain admissible values for the variables w_{ij} (in the original CA problem) assigning:

$$w_{ij} = \min_{s,d}\{w_{ij}^{sd}\} \quad (33)$$

Finally, the capacities are computed using $C_{ij} = \frac{1}{w_{ij}} + f_{ij}$.

4. NUMERICAL RESULTS

In order to prove the effectiveness of the design methodology, we run numerical experiments and computer simulations. We consider a mixed traffic scenario where the flow length follows the distribution shown in Fig. 3, which is derived from one-week long measurements [4]. We have one curve for each year (from 2000 to 2002). In particular, we report the discretized cumulative distribution function (CDF), obtained by splitting the flow length distribution in 15 groups with the same number of flows per group, from the shortest to the longest flow, and then computing the average flow length in each group. The large plot reports the discretized CDF using bytes as unit, while the small plot inside reports the same distribution taking today's most common maximum segment size (MSS) of 1460 bytes as unit.

We present results obtained considering several topologies, which have been generated using the BRITE topology generator with the router level option [25]. Random traffic matrices were generated by picking the traffic intensity of each

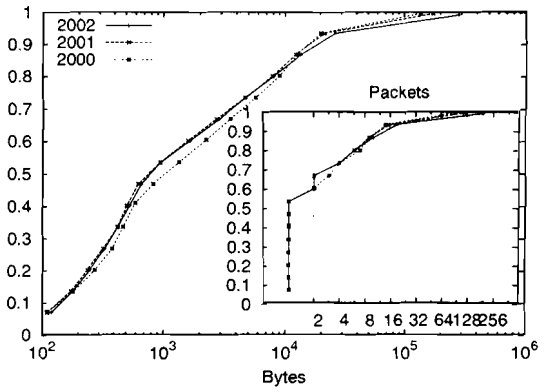

Figure 3. TCP connection length cumulative distribution derived from one-week long measurements in three different time periods

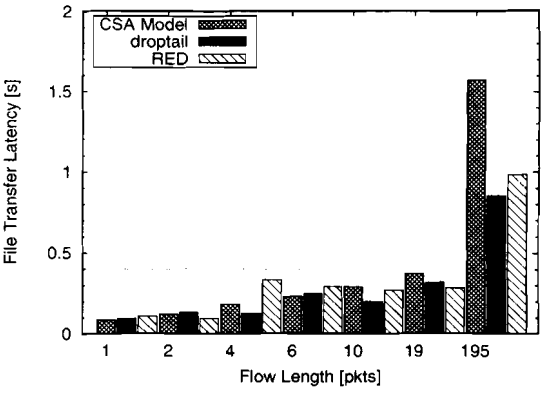

Figure 4. File transfer latency for all flow size classes (5-link path from the 10-node network; $L_t \leq 0.4$ s; $T_h \geq 512$ Kbps)

source/destination pair from a uniform distribution. For each topology, we solved both the CFA and BA problems using the approach described in previous sections. Simulation experiments at the packet level were run using the ns-2 simulator.

4.1 10-NODE NETWORKS

In this section, we present results obtained considering a 10-nodes, 20-links network topology. In the design we considered the following target QoS constraints for all source/destination pairs: i) file latency $L_t \leq 0.4$ s for TCP flows shorter than 20 segments, ii) throughput $T_h \geq 512$ kbps for TCP flows longer than 20 segments. Selecting $P_{loss} = 0.01$, we obtain a network-level design constraint equal to $RTT \leq 0.052$ s (see Fig. 2) for all source-destination pairs. Each traffic relation offers an average aggregate traffic equal to $\hat{\gamma}_{sd} = 1$ Mbps. Link propagation delays range from 0.25 ms to 1.5 ms, i.e., link lengths vary between 50 km and 300 km. After solving the CFA problem, we solved the BA problem in both the droptail and RED cases.

To verify the accuracy of the IP network design produced by the methodology, we performed packet-level simulations to check whether the QoS constraints are actually met. In the

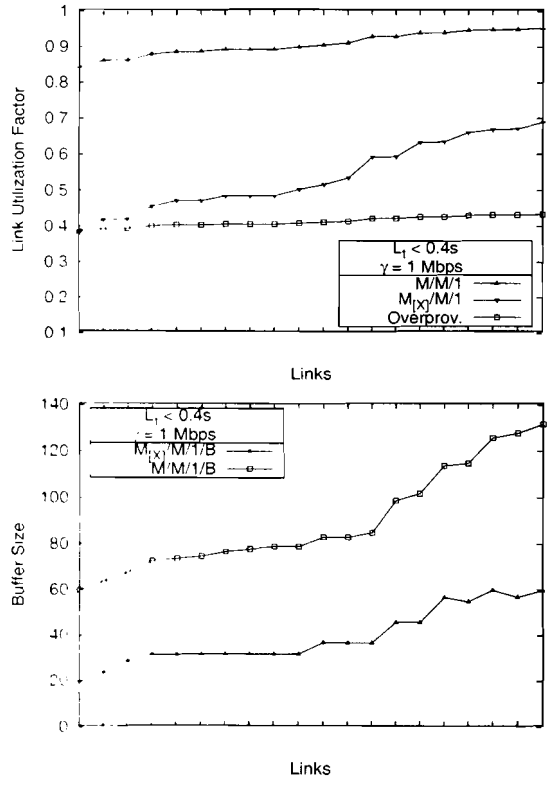

Figure 5. Link utilization factor and Buffer size for the 10-node network (classical and new approach)

experiments we assumed that TCP New Reno is adopted, and that TCP connections are established at instants described by a Poisson process, choosing at random a server-client pair. Connection opening rates are determined so as to meet the offered traffic, γ_{sd}. The amount of data to be transferred by each TCP connection (i.e., the flow length) is expressed in number of packets according to the measured values. We performed *path simulations* rather than simulating the entire network, i.e., we selected a path referring to a single source/destination pair, and simulated only links in that path, considering also interfering cross traffic. This approach is necessary due to scalability problems in *ns*-2, which did not allow us to simulate the entire topology. Moreover, results obtained in path simulations are worst-case results with respect to entire network simulations, because cross traffic is more aggressive, since it is directly injected into the simulated path, without traversing all links along its path, hence not suffering losses or shaping.

Among all possible source/destination pairs, we selected the longest path in the network, which comprises 5 links. Results are plotted in Fig. 4, which reports the file transfer latency for all flow size classes. The QoS constraint of 0.4 s for the maximum latency is also shown. We can clearly see that model predictions and simulation results are in perfect agreement with specifications, since the latency constraint is satisfied for all flows shorter than 20 segments. The flow transfer latency constraint for mice is more stringent than the throughput constraint for elephants, represented by 195 packet long flows, therefore the throughput of the latter is

Table 1. $E[T]$ and P_{loss} predicted by the $M_{[X]}/M/1/B$ model and measured in simulations (5-link path)

10-Node Network				
	$M_{[X]}/M/1/B$		*ns*-2 (RED)	
Link	$E[T]$	P_{loss}	$E[T]$	P_{loss}
1	0.006	0.0031	0.007	0.0023
2	0.008	0.0015	0.010	0.0016
3	0.008	0.0018	0.010	0.0018
4	0.006	0.0018	0.008	0.0016
5	0.009	0.0016	0.012	0.0038
tot	0.037	0.0098	0.047	0.0111

2.2 Mbps in the RED case, instead of the minimum desired 512 kbps. Notice that the predicted throughput obtained by applying the CSA model is a pessimistic estimate. This is due to the limit in the CSA model itself, and not to a mismatch in the network-layer parameters between model and simulation. Indeed, Table 1 reports the average packet delay $E[T]$, and the average packet loss probability P_{loss} predicted by the $M_{[X]}/M/1/B$ queueing model and measured in simulations (with RED buffers), for the selected 5-link path. The simulation margin of error for $E[T]$ and P_{loss}, for each link, are about $\pm 13.5\%$ and $\pm 20\%$, respectively. As it can be observed, there is a very good match between model predictions and simulation results.

To complete the evaluation of our methodology, we compare the link utilization factor and buffer size obtained when considering the classical model [1, 2] instead of the $M_{[X]}/M/1$ model. Fig. 5 shows the link utilizations and buffer sizes, respectively, obtained with our method and with the classical model. It can be immediately noticed that considering the burstiness of IP traffic radically changes the network design. Indeed, the link utilizations obtained with our methodology are much lower (i.e., capacities are much longer) than those produced by the classical approach, and buffers are much longer.

It is important to observe that the test of the QoS perceived by end users in a network dimensioned using the classical approach cannot be performed, since simulations fail even to run, because the dropping probability experienced by TCP flows is so high that retransmissions cause the offered load to become larger than 1 for some links. This means that a network designed with the classical approach is not capable of supporting the offered load, and therefore cannot satisfy the QoS constraints (because the classical approach considers Poisson arrivals and uses the average network-wide packet delay in the problem formulation).

In addition, in Fig.5 we also compare our results to those of an overprovisioned network, in which the capacities obtained by using the classic model are multiplied a posteriori by the minimum factor which allows the QoS constraints to be met. The overprovisioning factor was estimated by a trial and error procedure based on path simulations at the packet level. Since it is difficult to define an overprovisioning factor for the BA problem, we fixed *a priori* the buffer size to be equal to 150. The final overprovisioned network is capable of satisfying the QoS constraints, but a larger cost is incurred, which is directly proportional to the increase in link capacities (that is observed by the reduction in the link utilization factors).

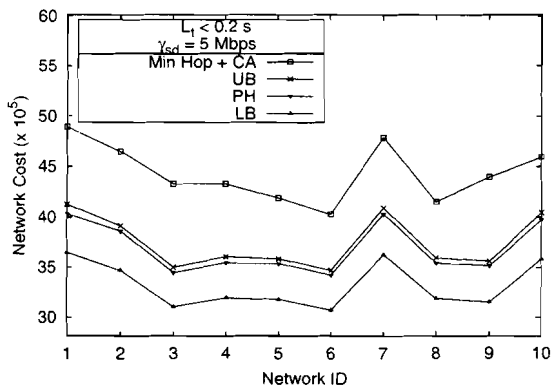

Figure 6. Network cost for 40-node networks with random topologies (considering: Lagrangean relaxation (LB), primal heuristic with logarithmic barrier CA solution (PH), primal heuristic with approximate CA solution (UB), CA with minimum-hop routing (MinHop+CA)

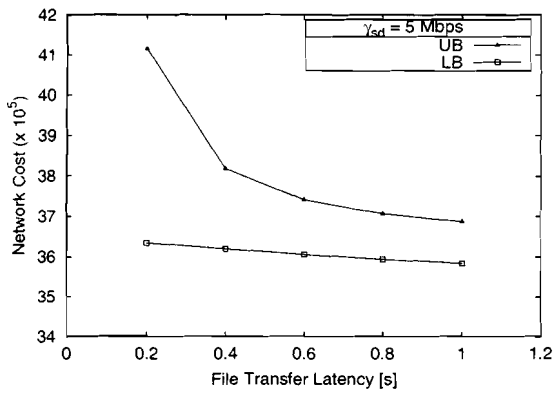

Figure 7. LB and UB network costs as a function of latency constraints (40-node, 160-link network)

Note also that the heuristic used to find the minimum overprovisioning factor cannot be applied for large/high-speed networks, due to scalability problem of packet level simulators.

4.2 40-NODE NETWORKS

In this section we present results for 40-node, 160-link network topologies where link propagation delays are uniformly distributed between 0.5 and 1.5 ms, i.e., where link lengths vary between 100 and 300 km.

Two sets of experiments were performed. In the first set of experiments, we compare the results obtained with four different techniques: i) Lagrangean relaxation (LB), ii) primal heuristic with logarithmic barrier CA solution (PH), iii) primal heuristic with approximate CA solution (UB), iv) CA with minimum-hop routing (MinHop+CA). Results for 10 random topologies are presented in Fig. 6. The average source/destination traffic requirement is set to $\hat{\gamma}_{sd} = 5$ Mbps. For all source/destination pairs, the target QoS constraints are: i) latency $L_t \leq 0.2$ s for TCP flows shorter than 20 segments, ii) throughput $T_h \geq 512$ kbps for TCP flows

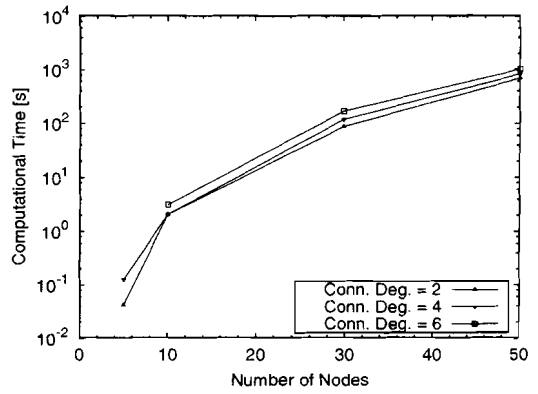

Figure 8. Computation times for the CFA problem (with different numbers of nodes and different connection degrees)

longer than 20 segments. Selecting $P_{loss} = 0.001$ and using the transport-layer QoS translator, we obtain the equivalent network-layer performance constraint $RTT \leq 0.032$ s, which derives from the most stringent latency constraint ($L_t \leq 0.2$ s for 20-segment flows).

First of all, the results allow us to conclude that ignoring the routing optimization when solving the CA problem (MinHop+CA) leads to poor results. In addition, we can observe that the feasible solutions (PH) and suboptimal solutions (UB) for all considered topologies always fall rather close to the lower bound (LB). The gap between UB and LB is about 16%. Using the PH solution, the gap is reduced to 13%.

The second set of experiments was conduced to investigate the impact of the latency constraints on the optimized network cost. Fig. 7 shows the LB and UB values for latency constraint values ranging from 0.2 to 1.0 s. The plots clearly show the tradeoff between cost and latency; as expected, costs grow when the latency constraints become tighter. It is interesting to observe that when the latency constraints become very tight (latencies become close to zero), the sensitivity of the network cost increases.

4.3 COMPUTATION TIMES

Finally, we briefly discuss the computation times needed to solve the CFA problem. The optimization algorithms (the subgradient algorithm and the heuristic) were implemented using the C language and run on a workstation with 1 GHz processor running Linux. The computation times (in CPU seconds) for several CFA problems are presented in Fig. 8. We tested our approach on networks with different numbers of nodes and different connection degrees (number of ingoing/outgoing links in a node). As can be seen, CPU times range from less than 1 second to more than 15 minutes.

It is straightforward to obtain the time complexity of the subgradient algorithm delineated in section 3.3. At each iteration it is necessary to solve subproblems $L_1(v)$ and $L_2(v)$. The subproblem $L_1(v)$ requires $O(n^3m)$ operations, since $n \times (n-1)$ shortest paths are found using the Bellman-Ford algorithm of complexity $O(nm)$; and subproblem $L_2(v)$ re-

quires $O(n^2m)$ operations, since we need to solve $n \times (n-1)$ simple problems in order to calculate each one of the m links. If the number of iterations, $MaxItr$, is the only stopping criterion used, the resulting time complexity is $O(n^3m\ MaxItr)$.

5. CONCLUSIONS

In this paper, we have considered the QoS design of packet-switching networks, and in particular the joint Capacity and Flow Assignment problem where both routing and capacity assignments are considered to be decisions variables. Our new formulation to the CFA problem differs in two important points from previous formulations. First, the novelty of our approach is that it considers end-to-end QoS constraints for all source/destinations pairs on the network. A second important improvement with respect to earlier approaches is the use of a refined IP traffic modeling technique that provides an accurate description of the traffic dynamics in multi-bottleneck networks subject to TCP mice and elephants. By explicitly considering TCP traffic, we also need to consider the impact of finite buffers, therefore facing the Buffer Assignment problem.

We have formulated the problem as a nonlinear mixed-integer programming problem. A Lagrangean Relaxation approach was used to obtain both lower bounds and feasible solutions in the VPN case. A subgradient method was used to find the optimal Lagrangean multipliers. Examples of application of the proposed design methodology to different networking configurations have been discussed. The network target performances are validated against detailed simulation experiments. Numerical results suggest that the proposed methodology provides a quite efficient approach to obtain near-optimal solutions with small computational effort.

REFERENCES

[1] L. Kleinrock, "Queueing Systems, Volume II: Computer Applications," *Wiley Interscience*, New York, 1976.

[2] M. Gerla and L. Kleinrock, "On the Topological Design of Distributed Computer Networks," *IEEE Transactions on Communications*, Vol. 25, pp. 48-60, Jan. 1977.

[3] K. Claffy, Greg Miller, and Kevin Thompson, "The nature of the beast: Recent traffic measurements from an Internet backbone," in *Proc. INET '98*, Geneva, Switzerland, July 1998.

[4] M. Mellia, A. Carpani, R. Lo Cigno, "Measuring IP and TCP behavior on Edge Nodes," *Proceddings of IEEE Globecom 2002*, Taipei, TW, Nov. 2002.

[5] V. Paxson, S. Floyd, "Wide-Area Traffic: The Failure of Poisson Modeling," *IEEE/ACM Transactions on Networking*, Vol. 3, No. 3, pp. 226-244, Jun. 1995.

[6] K.T. Cheng, F.Y.S. Lin, "Minimax End-to-End Delay Routing and Capacity Assignment for Virtual Circuit Networks," *IEEE Globecom 95*, Singapore, pp. 2134-2138, Nov. 1995.

[7] E. Rolland, A. Amiri, R. Barkhi, "Queueing Delay Guarantees in Bandwidth Packing," In *Computers and Operations Research*, Vol. 26, pp. 921-935, 1999.

[8] A. Gersht, R. Weihmayer, "Joint optimization of data network design and facility selection," *IEEE Journal on Selected Areas in Communications*, Vol. 8, No. 9, pp. 1667-1681, Dec. 1990.

[9] M. Garetto, D. Towsley, "Modeling, Simulation and Measurements of Queuing Delay under Long-tail Internet Traffic," *ACM Sigmetrics 2003*, pp. 47-57, San Diego, CA, June 2003.

[10] S. Floyd, V. Jacobson, "Random early detection gateways for congestion avoidance", *IEEE/ACM Transactions on Networking*, Vol. 1, No. 4, pp. 397-413, 1993.

[11] H. Pirkul, A. Amiri, "Routing in Packet-switched Communication Networks", *Computer Communications*, Vol. 17, No.5, pp. 307-316, May 1994.

[12] T. Ng, D. Hoang, "Joint Optimization of Capacity and Flow Assignment in a Packet Switched Communication Network", *IEEE Transaction on Communications*, Vol. 35, No. 2, pp. 202-209, Feb. 1987.

[13] D. Medhi, D. Tipper, "Some approaches to solving a multi-hour broadband network capacity design problem with single-path routing", *Telecommunication Systems*, Vol. 13, No. 2, pp. 269-291, 2000.

[14] C. Fraleigh, F. Tobagi, C. Diot, "Provisioning IP Backbone Networks to Support Latency Sensitive Traffic," *IEEE Infocom 03*, San Francisco, CA, Mar 2003.

[15] H. Knoche, H. de Meer, "Quantitative QoS Mapping: A Unifying Approach," *5th Int. Workshop on Quality of Service (IWQoS97)*, New York, NY, pp. 347-358, May 1997.

[16] A. Markopoulou, F. Tobagi, M. Karam, "Assessment of VoIP Quality over Internet Backbones," *IEEE Infocom 02*, pp. 150-159, New York, NY, June 2002.

[17] N. Cardwell, S. Savage, T. Anderson, "Modeling TCP Latency," *IEEE Infocom 00*, pp. 1742-1751, Tel Aviv, IS, March 2000.

[18] J. Padhye, V. Firoiu, D. Towsley, J. Kurose, "Modeling TCP Reno performance: a simple model and its empirical validation," /EEE/ACM Transactions on Networking, Vol. 8, No. 2, pp. 133-145, Apr. 2000.

[19] X. Chao, M. Miyazawa, M. Pinedo, *Queueing Networks, Customers, Signals and Product Form Solutions*, John Wiley, 1999.

[20] R. Nagarajan, D. Towsley, "Note on the Convexity of the Probability of a Full Buffer in the M/M/1/K queue," *CMPSCI Technical Report TR 92-85*, Aug. 92.

[21] M. Wright, "Interior methods for constrained optimization", *Acta Numerica*, Vol. 1, pp. 341-407, 1992.

[22] B. Gendron, T.G. Crainic, A. Frangioni, "Multicommodity Capacitated Network Design". B. Sansò and P. Soriano (eds), *Telecommunications Network Planning*, pp. 1-19, Kluwer, MA, 1998.

[23] M. L. Fisher, "The Lagrangean relaxation method for solving integer programming problems," *Management Science*, Vol. 27, pp. 1-18, 1981.

[24] A.M. Geoffrion, "Lagrangean relaxation and its uses in integer programming," *Mathematical Programming Study*, Vol. 2, pp. 82-114, 1974.

[25] A. Medina, A. Lakhina, I. Matta, J. Byers, "BRITE: Boston university representative internet topology generator," Boston University, http://cswww.bu.edu/brite, April 2001.

Emilio C. G. Wille received his degree in Electronic Engineering in February 1989 and a M.Sc in Electronic and Telecommunications Engineering in July 1991, both from *Centro Federal de Educação Tecnológica do Paraná* - CEFET/PR (Curitiba - Brazil). Since October 1991 he is with Electronics Department of CEFET/PR as an Assistant Professor. His teaching duties at CEFET/PR comprise graduate and undergraduate-level courses on electronic and telecommu-

nication theory. From February 2001 until February 2004 he was with the Electronics Department of *Politecnico di Torino* (Italy) as a Ph.D student. He was supported by a CAPES Foundation scholarship from the Ministry of Education of Brazil. His research interests are centered upon the application of optimization algorithms for telecommunication networks design and planning, Markov processes, queueing models, and performance analysis of telecommunication systems.

Marco Mellia received his degree in Electronics Engineering in 1997, and a Ph.D in Telecommunications Engineering in 2001, both from *Politecnico di Torino*, Torino, Italy. From March to October 1999 he was with the CS department at Carnegie Mellon University, Pittsburgh, PA, as a visiting scholar. Since April 2001, he has been with Electronics Department of the *Politecnico di Torino* as Assistant Professor. His research interests are in the fields of all-optical networks, traffic measurement and modeling, switching architectures, and QoS routing algorithms.

Emilio Leonardi received the Dr. Ing. degree in Electronics Engineering in 1991, and a Ph.D in Telecommunications Engineering in 1995, both from *Politecnico di Torino*, Torino, Italy. He is currently an Assistant Professor in Electronics Department of the *Politecnico di Torino*. In 1995, he spent one year at the Computer Science Department of the University of California, Los Angeles (UCLA), where he was involved in the Supercomputer-SuperNet (SSN) project, aimed at the design of a hierarchical network comprising an optical backbone that interconnects several high speed wormhole routed networks. In the Summer 1999 he visited the "High Speed Networks Research Group", at Bell labs. - Lucent, where he collaborated to the design of efficient scheduling policies for high capacity input queued switches. His research interests are in the fields of all-optical networks, queueing theory, switching architectures, and wireless communications.

Marco Ajmone Marsan received degrees in Electronics Engineering from the *Politecnico di Torino*, Torino, Italy, and the University of California, Los Angeles (UCLA). He is currently a Full Professor at the Electronics Department of the *Politecnico di Torino*. During the summers of 1980 and 1981, he was with the Research in Distributed Processing Group, Computer Science Department, UCLA. During the summer of 1998 he was an Erskine Fellow at the Computer Science Department of the University of Canterbury in New Zealand. He has coauthored over 300 journal and conference papers in Communications and Computer Science, as well as the two books "Performance Models of Multiprocessor Systems", published by the MIT Press, and "Modelling with Generalized Stochastic Petri Nets", published by John Wiley. In 2002, he was awarded a Honoris Causa Degree in Telecommunications Networks from the Budapest University of Technology and Economics. His current interests are in the performance evaluation of communication networks and their protocols.