# A Mechanism for Pricing Service Guarantees

Bruce Hajek

Department of Electrical and Computer Engineering
and the Coordinated Science Laboratory
University of Illinois at Urbana-Champaign

Sichao Yang
Qualcomm Corporation
Bedminster, NJ

***Abstract*—The calculus of deterministic constraints on service and traffic streams offers a rich language to specify service guarantees. In particular, the service curve earliest deadline first (SCED) algorithm has an associated feasibility test given by linear constraints. Users sending streams of data through the server may have differing needs for delay and throughput. We suggest a way based on utility function maximization, subject to the linear constraints of the SCED algorithm, for allocation of service. In addition, a generalization of the SCED algorithm is given which does not require that the deadline sequences within streams be monotone nondecreasing.**

## I. Introduction

There are many philosophical issues related to design of mechanisms for allocating service over time. One is whether quality of service is based on payments, virtual or real, by users. In this paper, we consider the case that such payments are made, and focus on certain allocation mechanisms involving pricing. A second issue is whether state information is kept, either at the boundary of a network or at each node in a network, for each user. For example, the differentiated services model assumes no user state information is kept at the nodes within the network. A user can label packets with different quality of service requirements, and there could be prices associated with the different service classes, imposed at the boundary of the network. In contrast, the integrated services model involves each node of the network keeping state information for each user (i.e. each flow). In this paper we focus on only a single server, which does keep state information for each user. A possible application of the pricing mechanisms we discuss, however, would be to use them to set performance guarantees at several critical points along the path of a given network flow.

To make this discussion more concrete, consider the use of generalized processor sharing (GPS) [1, 2] by a rate $C$ server. Each flow $i$ receives, at a minimum, nearly constant rate service at rate $C_i = \phi_i / \sum_j \phi_j$, where $\phi_j$ is a weight assigned to each flow. A guarantees for flow $i$ could be absolute, such as a positive lower bound on $C_i$, or relative, such as a positive lower bound on $\phi_i$. Pricing could be used to mediate the choice between absolute and relative guarantees, with higher prices for absolute and longer term agreements, being higher.

Even within the space of pricing/allocation mechanisms for a single server, there are many design issues. A key choice is the time-horizon over which guarantees are extended. Another philosophical issue related to pricing is to know what the objective of the pricing is. One possible goal would be social efficiency: to maximize the sum, over all users, of the values that each user obtains from the resource (independently of the payments made), whereas another possible goal would be to maximize revenue. These two goals are related, to the extent that higher prices can be charged for a resource that provides more value. Yet another design issue is whether the users are assumed to anticipate the effects of their own individual actions on the prices offered. We shall focus on the case that users are price-takers, meaning they don't anticipate the effects of their own individual actions on the prices offered, and social welfare is to be maximized.

### A. Background on Network Calculus

This section briefly reviews some of the theory of deterministically constrained traffic, initiated by R. L. Cruz [3, 4], Parekh and Gallager [1, 2], and others. Specifically, upper constraints, regulators, and service curves are discussed. More detail and references can be found in the books of C.-S. Chang [5] and Leboudec and Thiran [6].

The case of equal length packets transmitted in discrete time will be considered. A packet stream can be described by a cumulative arrival sequence $A$, which is a nondecreasing, integer-valued function on the nonnegative integers $Z_+$, such that $A(0) = 0$. For each integer $t \geq 1$, $A(t)$ denotes the number of arrivals of the stream in slots $1, 2, \ldots, t$. Since we have no interest here in the actual contents of the packets, the cumulative arrival process $A$ is itself called a packet stream, or simply a stream.

Let $f$ be a nondecreasing function from $\mathbb{Z}_+$ to $\mathbb{R}_+$. A stream $A$ is said to be *$f$-upper constrained* if $A(t) - A(u) \le f(t-u)$ for all $u, t$ with $0 \le u \le t$. Equivalently, $A$ is $f$-upper constrained if $A \le A \star f$, where for two functions $f$ and $g$ defined on $\mathbb{Z}_+$, $f \star g$ denotes the function on $\mathbb{Z}_+$ defined by

$$(f \star g)(t) = \min_{0 \le u \le t} g(u) + f(t-u). \tag{1}$$

Some functions $f$ can be reduced, without changing the condition that a packet steam is $f$-upper constrained. The *subadditive integer closure,* $f^*$, is given by $f^*(0) = 0$, and for $u \ge 1$, $f^*(u) = \min\{\lfloor f(u_1) \rfloor + \cdots + \lfloor f(u_n) \rfloor\}$, where the minimum is over ways to write $u$ as a sum of positive integers: $u_1 + \cdots + u_n = u$, and $\lfloor x \rfloor$ denotes the greatest integer less than or equal to $x$. The function $f^*$ is subadditive: $f^*(u+t) \le f^*(u) + f^*(t)$ for $u, t \ge 0$, and $f^*$ is maximal among all integer-valued subadditive functions with initial value zero which are pointwise dominated by $f$. Then for any stream $A$, $A$ is $f$-upper constrained if and only if $A$ is $f^*$-upper constrained, or equivalently, if and only if $A = A \star f^*$. A *regulator* for an upper-constraint $f$ is a device such that for any input stream $A$ the corresponding output stream $B$ is $f$-upper constrained. A regulator is said to be a *maximal regulator* for $f$ if the following is true: For any input stream $A$, if $B$ is the output of the regulator for input $A$ and if $\tilde{B}$ is a stream such that $\tilde{B} \le A$ (flow condition) and $\tilde{B}$ is $f$-upper constrained, then $\tilde{B} \le B$.

*Proposition 1.1:* (See [5, 6].) A maximal regulator for $f$ is determined by the relation $B = A \star f^*$.

A stream $A$ is said to be *$(\sigma, \rho)$-upper constrained* if it is $f$-upper constrained for the function $f(t) = \sigma + \rho t$. An important property of the $(\sigma, \rho)$-upper constraint is that the maximal regulator for the $(\sigma, \rho)$-upper constraint can be implemented in a simple way using a single counter.

Let $C$ be a positive integer. A queue with a constant-rate, work-conserving server $C$ and input process $A$ has the output process $B = A \star f$, where $f(t) = Ct$. The queuelength process is $(q(t) : t \ge 0)$, where $q(t)$ is the number of packets carried over from slot $t$ to slot $t+1$. It is given by $q = A - B$. The time a packet spends in the queue does not include any service time of the packet (even if there is a nonzero service time), so that the queuing delay, or time-in-queue, for a packet, can be zero.

A *service curve* is a nondecreasing function from $\mathbb{Z}_+$ to $\mathbb{Z}_+$ with value zero at zero. Given a service curve $f$, a server is an *$f$-server* if for any input $A$, the output $B$ satisfies $B \ge A \star f$. An $f$ server is called *exact* if for any input $A$, the output $B$ satisfies $B = A \star f$. An exact $f$-server lets out the minimum number of packets necessary to meet the $f$-server constraint.

**Example 1.1:** (a) (Bounded delay server) Given an integer $d \ge 0$, let $O_d$ denote the function

$$O_d(t) = \begin{cases} 0 & \text{for } t \le d \\ +\infty & \text{for } t > d \end{cases}$$

Then a FIFO device is an $O_d$-server if and only if the delay of every packet is less than or equal to $d$, no matter what the input process.
(b) (Delay-rate function) Given $\delta, \beta \ge 0$, let $b(t) = \beta(t-\delta)_+$. A $b$-server serves at least as many packets by time $t$ as a constant rate $\beta$ server would serve by time $t - \delta$, for all $t \ge \delta$.
(c) A maximal regulator for a nondecreasing function $f : \mathbb{Z} \to \mathbb{R}_+$ is an exact $f^*$-server.

## II. Upper concentration and the SCED algorithm

This section generalizes the service curve earliest deadline first algorithm (SCED) of [7–9] to accommodate the case that the sequence of deadlines of packets within a data stream need not be monotone nondecreasing. Since deadlines can be put on individual packets by the source in real-time, this generalization allows more flexibility on the part of a user in determining a delay-throughput profile. Since there may be unknown transit times between a user and a server, the deadlines could instead be given as laxities, to be measured from the time of arrival of a packet at the server. Here we use deadlines and we do not include propagation delays.

Consider a packet stream described by $A$. Suppose the packets within the stream are $P_1, P_2, \ldots$, where the numbering is such that $P_{k+1}$ arrives after or at the same time as $P_k$, for all $k \ge 1$. The arrival time of $P_k$ is then $A^{-1}(k) = \min\{t : A(t) \ge k\}$. Let $D = (D(k) : k \ge 1)$ be a sequence of deadlines for the packets. We require that $D \ge A^{-1}$, but we do not require that the sequence $D$ be nondecreasing. The pair $(A, D)$ is called a *stream with deadlines.* Suppose $h$ is a nondecreasing function $h : \mathbb{Z} \to \mathbb{R}_+$. We introduce the following definition.

*Definition 2.1:* A stream with deadlines $(A, D)$ is *$h$-upper concentrated* if for any interval of $u \ge 1$ consecutive time slots, $[t-u+1 : t]$, the number of packets with both arrival time and deadline within the interval is less than or equal to $h(u)$.

Given an arrival stream $A$, a monotone nondecreasing sequence of deadlines $D$ can readily be assigned so that $(A, D)$ satisfies a prescribed upper concentration constraint (see the next two lemmas). Nomonotonic sequences of deadlines can be assigned, subject to upper concentration constraints, by running multiple sub-

streams, with monotone deadline sequences within each substream.

*Lemma 2.2:* (Service guarantees for a stream with nondecreasing deadlines) Suppose $(A, D)$ is a stream with deadlines, such that the sequence of deadlines $D$ is nondecreasing, and suppose $f$ is a nondecreasing function on $\mathbb{Z}_+$ such that $f(0) = 0$. Let $B$ denote the arrival stream corresponding to $D$ : $B(t) = \max\{k \geq 1 : D(k) \leq t\}$ and $D = B^{-1}$. Then
(a) A server that serves packets exactly at the deadlines in $D$, guarantees $\lfloor f \rfloor$-service if and only if $B \geq A \star \lfloor f \rfloor$.
(b) $(A, D)$ is $\lfloor f \rfloor$-upper concentrated if and only if $B \leq A \star \lfloor f \rfloor$.
(c) If $B = A \star \lfloor f \rfloor$ (i.e. $D = (A \star f)^{-1}$) $D$ is the latest nondecreaing sequence of deadlines guaranteeing $\lfloor f \rfloor$-service, and is the earliest nondecreasing sequence of deadlines such that $(A, D)$ is $\lfloor f \rfloor$-upper concentrated.

*Proof:* The definition of $\lfloor f \rfloor$-service implies (a). For and $u, t$ with $1 \leq u \leq t$, the number of packets that have arrival times and deadlines within $[t - u + 1, t]$ is $B(t) - A(t - u)$. So $(A, D)$ is $\lfloor f \rfloor$-upper concentrated if and only if $B(t) - A(t - u) \leq \lfloor f(u) \rfloor$ whenever $1 \leq u \leq t$, or equivalently, if and only if $B \leq A \star \lfloor f \rfloor$. Part (c) is implied by (a) and (b). ■

*Lemma 2.3:* (Concentration upper constraint for an upper-constrained stream with nondecreasing deadlines) Suppose $f, g : \mathbb{Z} \to \mathbb{R}_+$ are nondecreasing with $f(0) = 0$. Suppose $(A, D)$ is a stream with deadlines such that $A$ is $g$-upper constrained and the deadlines are given by $D = (A \star \lfloor f \rfloor)^{-1}$. (Note that the deadlines are nondecreasing.) Then $(A, D)$ is $h$-upper concentrated, where $h = g \star \lfloor f \rfloor$.

*Proposition 2.4:* (SCED guarantee, deadline sequence not necessarily monotone) Consider an EDF server with integer fixed rate $C$ that serves a set of streams with deadlines, $((A_s, D_s) : s \in S)$, and suppose that $(A_s, D_s)$ is $h_s$-upper concentrated for $s \in S$. If $\sum_{s \in S} h_s(u) \leq Cu$ for all $u \geq 0$, then each packet is served no later than its deadline.

*Proof:* A well-known property of EDF is that if all deadlines can be met by some service order, then all deadlines can be met by the EDF service order. It is thus sufficient to show that, for any $K \geq 1$, there is a schedule such that the first $K$ packets can be scheduled by their deadlines. Consider a bipartite graph, with vertices on the left given by packets $P_1, \ldots, P_K$, and vertices on the right given by $O_{v,l}$ for $v \geq 1$ and $1 \leq l \leq C$. Vertex $O_{v,l}$ denotes an opportunity for service at time $v$. Suppose there is an edge between $P_k$ and a vertex $O_{v,l}$ if and only if $v$ is in the closed interval bounded by the arrival time of $P_k$ and the deadline of $P_k$. Then the first $K$ packets can be scheduled by their deadlines if and only if there is a matching in the graph that covers all $K$ vertices on the left. Hall's theorem for matchings in bipartite graphs states that such a matching exists if (and only if) for any set of vertices $F$ on the right side, the number of vertices on the left that must be matched to one of those vertices is less than or equal to $|F|$. The assumptions of the proposition imply Hall's condition is true for $F$ of the form $\{O_{v,l} : (v, l) \in [t - u + 1 : t] \times [1 : C]\}$, from which Hall's condition follows for general $F$. Thus, there exists some schedule under which the first $K$ packets meet their deadline. ■

In view of Lemma 2.3, Proposition 2.4 reduces to the scheduling result for SCED for which service within each stream is FIFO:

*Corollary 2.5:* [7–9] Consider an EDF server with integer fixed rate $C$ that serves a set of streams with deadlines, $((A_s, D_s) : s \in S)$. Suppose $(A_s, D_s)$ for $s \in S$ is a packet stream such that $A_s$ is $g_s$-upper constrained and $D_s = (A_s \star \lceil f_s \rceil)^{-1}$. Then if

$$\sum_{s \in S} (g_s \star f_s)(t) \leq Ct \quad \text{for} \quad t \in \mathbb{Z}_+, \tag{2}$$

no packet is served after its deadline.

**Example 2.1:** For example, suppose $S = \{1, 2, 3\}$, $g_s(t) = (\sigma_s + \rho_s t) I_{\{t \geq 0\}}$ and $f_s(t) = O_{d_s}(t)$ for $1 \leq s \leq 3$, where $0 < d_1 < d_2 < d_3$. Then $g_s \star f_s = g_s(t - d_s)$, and the sufficient condition (2) is illustrated in Figure 1.

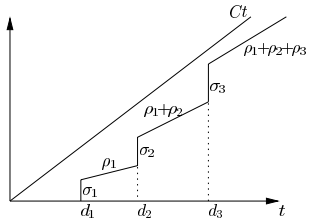


Fig. 1. Constraint set with three users

## III. Pricing Service Level Agreements

Let $S$ denote a set of users, such that each user $s$ sends a stream $A_s$ through the server. We turn to the problem of developing a mechanism for allocating functions $h_s = g_s \star f_s$ to the users based on prices, following the methodology of valuation functions. In particular, we would like to make available a decomposition of the system allocation problem into problems for the users, based on prices, and a network problem, based on bids from the users, along the lines of Kelly et al. [10, 11]. We use a vector $(g_s, f_s, w_s)$ to represent a

*service level agreement* between user $s$ and the network. The significance is that the user $s$ agrees that its input stream, $A_s$, will be $g_s$-upper constrained, and user $s$ agrees to pay at rate $w_s \geq 0$. In return, the server agrees to be an $\lfloor f_s \rfloor$-server for user $s$. Our approach is based on the following considerations.

The sufficient condition (2) is potentially infinite dimensional, because there is a constraint for each $t \geq 0$. But, Example 2.1 illustrates that if the functions $f_s$ and $g_s$ are piecewise linear, then the constraints can be reduced to a finite number of conditions. Hence, we will fix a set of delays, $0 < D_1 < D_2 < \cdots < D_J$, and require that the sufficient conditions (2) reduce to $J+1$ linear conditions, one for each $D_j$, $1 \leq j \leq J$, and one for the slope beyond $D_J$. For example, $J$ could be much smaller than $|S|$. Conceptually, we might allow the users to bid for $J+1$ dimensional allocation vectors, with each coordinate entering into one of the $J+1$ inequalities. But in order to reduce the complexity of the bidding process, we assume that each user $s$ specifies a small number of *shape vectors,* and then, is allocated an SLA based on a linear combination of the shape vectors, with coefficients in the linear combination determined through the bidding process.

For example, if there were three users as in Example 2, we could take $\{D_1, D_2, D_3\} = \{d_1, d_2, d_3\}$, and there would be four inequalities to be satisfied. We could suppose that each user $s$, requests an SLA of the form $(\sigma_s + \rho_s t, O_{d_s}, w_s)$, where a specific value of $d_s$ is specified (so the user is not elastic with respect to delay) but the values of $\sigma_s$ and $\rho_s$ are to be determined by a bidding process. If, furthermore, the user were to initially specify a fixed ratio for $\sigma_s/\rho_s$, then $g_s \star f_s$ would be determined up to a positive constant, in which case we would use a valuation function, $U_s$, of the one-dimensional variable $\rho_s$. If the ratios $\sigma_s/\rho_s$ aren't fixed in advance of the bidding process, then we would use a valuation function depending on two variables, namely, having the form $U_s(\sigma_s, \rho_s)$.

The framework of [10, 11] can be extended to multidimensional allocations based on utility functions, as shown by Gibbens and Kelly [12]. As pointed out in [12], pricing for differentiated services is akin to source routing, in which a packet source essentially specifies different routes for packets by labeling them with different quality-of-service marks. The routes in [12] correspond to shape vectors here, and the links in [12] correspond to constraints here. For each $s \in S$, the service to be allocated to user $s$ is to be represented by a vector $x_s = (y_r : r \in s)$. Here, $r$ indexes the coordinates of the allocation vector $x_s$. Let $R$ denote the union of the disjoint sets $\{r : r \in s\}$, indexed by $s \in S$.

For each $r \in s$ and $s \in S$, we assume there is a shape vector $(A_{j,r} : 1 \leq j \leq J+1)$, which specifies the contribution of $y_r$ to each of the $J+1$ constraints. Each user $s$ therefore has a shape matrix $(A_{j,r} : 1 \leq j \leq J+1, r \in s)$. Let $C_j = CD_j$ for $1 \leq j \leq J$ and $C_{J+1} = C$. The constraints are

$$\sum_r A_{j,r} y_r \leq C_j \quad 1 \leq j \leq J+1$$

The value of service vector $x_s$ to user $s$ is expressed as $U_s(x_s)$. The dimension of the demand of user $s$ is the length of $x_s$, or $|\{r : r \in s\}|$.

**Example 3.1:** A user $s$ seeking an SLA of the form $(g_s, f_s, w_s)$ with $g_s(t) = \sigma + \rho t$ and $f_s(t) = O_{D_{j_0}}(t)$ has $x_s = (y_{[s,\sigma]}, y_{[s,\rho]}) = (\sigma, \rho)$ and $2 \times (J+1)$ shape matrix given by

$$\begin{aligned} A_{j,[s,\sigma]} &= I_{\{j_o \leq j \leq J\}} \\ A_{j,[s,\rho]} &= (D_j - D_{j_o}) I_{\{j_0 \leq j \leq J\}} + I_{\{j=J+1\}}. \end{aligned}$$

**Example 3.2:** A user $s$ seeking $f_s(t) = \beta(t - D_{j_0})_+$ service with $\beta$ to be determined has $x_s = y_{[s,1]} = \beta$ and $1 \times (J+1)$ shape matrix given by

$$A_{j,[s,1]} = (D_j - D_{j_o}) I_{\{j_0 \leq j \leq J\}} + I_{\{j=J+1\}}.$$

This is the same as the second row for Example 3.1.

**Example 3.3:** A user $s$ seeking to obtain an SLA of the form $(h_s, w_s)$ for a stream of packets with deadlines subject to an $h$-upper concentration constraint for $h$ of the form $h_s(t) = \beta_1(\min\{t, D_{j_1}\} - D_{j_0})_+ + \beta_2 I_{\{t \geq D_{j_1}\}}$ has $x_s = (x_{[s,1]}, x_{[s,2]}) = (\beta_1, \beta_2)$ and $2 \times (J+1)$ shape matrix given by

$$\begin{aligned} A_{j,[s,1]} &= (\min\{D_j, D_{j_1}\} - D_{j_o}) I_{\{j_0 \leq j \leq J\}} \\ A_{j,[s,2]} &= (D_j - D_{j_1}) I_{\{j_1 \leq j \leq J\}} + I_{\{j=J+1\}} \end{aligned}$$

**Example 3.4:** One possibility for a concave valuation function of the form $U(\sigma, \rho)$ for an $\sigma, \rho$-upper constraint is

$$U(\sigma, \rho) = \begin{cases} \dfrac{\rho - \rho^{\sigma+1}}{1 - \rho^{\sigma+1}}, & \text{if } \rho \neq 1, \\ \dfrac{\sigma}{1+\sigma}, & \text{if } \rho = 1. \end{cases}$$

It can verified that the valuation function $U(\sigma, \rho)$ is a concave function of $(\sigma, \rho)$. The details are complicated. (Mathematica was used to algebraically check that the Hessian matrix is everywhere negative definite.) A heuristic motivation of this choice is that it is the throughput for a dropping bucket regulator with Poisson packet arrivals of rate one, and Poisson token arrivals of rate $\rho$ and maximum token backlog $\sigma$.

Define the vector $\boldsymbol{C} = (C_j : 1 \leq j \leq J+1)$ by $C_j = CD_j$ for $1 \leq j \leq J$ and $C_{J+1} = C$.

$SYSTEM(\boldsymbol{U}, A, \boldsymbol{C})$ :

$$\max \sum_{s \in S} U_s(x_s)$$
$$\text{over } y_r \geq 0, r \in R, \text{ subject to } A\boldsymbol{y} \leq \boldsymbol{C}$$
$$\text{where } x_s = (y_r : r \in s), s \in S.$$

The system problem is concave, and the Kuhn-Tucker conditions give rise to the following proposition, stated without proof.

*Proposition 3.1:* There is a solution to the system problem. A feasible allocation $\boldsymbol{x}$ is a solution if and only if there is a choice of $\boldsymbol{\mu} = (\mu_j : 1 \leq j \leq J+1)$ (i.e. Lagrange multipliers for the $J+1$ constraints) so that for all $s$

$$U_s^r(x_s) \leq \lambda_r \text{ with equality if } y_r > 0 \text{ for } r \in R$$
$$\boldsymbol{\lambda} = A\boldsymbol{\mu}$$
$$\mu_j \geq 0, \text{ with equality if } \sum_r A_{j,r} y_r = C_j, \forall j$$

### A. System problem decomposition for price-taking users

The form of the system problem falls within the multidimensional value optimization framework of Gibbens and Kelly [12], except here the entries in $A$ are not necessarily zero-one valued. The system problem can be decomposed as follows. Each user $s$ sends a signal $w_s$ to the server. After the server receives the signals from the users, it solves the following optimization problem:

$NETWORK(\boldsymbol{w}, A, \boldsymbol{C})$ :

$$\max \sum_{r \in R} w_r \log y_r$$
$$\text{over } y_r \geq 0, r \in R, \text{ subject to } A\boldsymbol{y} \leq \boldsymbol{C}$$
$$\text{where } x_s = (y_r : r \in s), s \in S.$$

The users take $\boldsymbol{\lambda}$ as fixed prices set by the server and update their signals in the direction of maximizing their payoffs:

$USER_s$ $(U_s, (\lambda_r : r \in s))$ :

$$\max U_s(x_s) - \sum_{r \in s} w_r$$
$$\text{over } w_r \geq 0, r \in R$$
$$\text{subject to } w_r = \lambda_r y_r, r \in s$$
$$\text{where } x_s = (y_r : r \in s).$$

The users and the server do the following iteration: The server solves the NETWORK problem based on users' updated signals and the users readjust their signals by new price feedback from the server. As in Kelly, Maulloo, Tan, [11], both primal and dual methods for solving relaxed versions of the network problem can be fashioned, and a user adaptation algorithm can be used for the user problem. Simultaneous solution to both yields a solution to the system problem.

## ACKNOWLEDGMENT

This work was supported in part by the National Science Foundation under grants NSF CNS 05-19691 and NSF ECS 06-21416.

## REFERENCES

[1] A. Parekh and R. Gallager, "A generalized processor sharing approach to flow control in integrated services networks: the single-node case," *IEEE/ACM Transactions on Networking*, vol. 1, no. 3, pp. 344–357, 1993.

[2] A. Parekh, R. Gallager, I. Center, and Y. Heights, "A generalized processor sharing approach to flow control in integrated services networks: the multiple node case," *IEEE/ACM Transactions on Networking*, vol. 2, no. 2, pp. 137–150, 1994.

[3] R. L. Cruz, "A calculus for network delay, part I: Network elements in isolation," *IEEE Transactions on Information Theory*, vol. 37, pp. 114–131, 1991.

[4] R. L. Cruz, "A calculus for network delay, part II: Network analysis," *IEEE Transactions on Information Theory*, vol. 37, pp. 132–141, 1991.

[5] C. Chang, *Performance Guarantees in Communication Networks*. Springer, 2000.

[6] J. Le Boudec and P. Thiran, *Network Calculus: A Theory of Deterministic Queuing Systems for the Internet*. Springer, 2001.

[7] R. L. Cruz, "Quality of service guarantees in virtual circuit switched networks," *IEEE J. Selected Areas in Communications*, vol. 13, pp. 1048–1056, 1995.

[8] H. Sariowan, R. Cruz, and G. Polyzos, "SCED: a generalized scheduling policy for guaranteeing quality-of-service," *IEEE/ACM Transactions on Networking (TON)*, vol. 7, no. 5, pp. 669–684, 1999.

[9] H. Sariowan, *A service-curve approach to performance guarantees in integrated-service networks*. PhD thesis, Dept. of Electrical & Computer Engineering, University of California San Diego, June 1996.

[10] F. P. Kelly, "Charging and rate control for elastic traffic," *European Transactions on Telecommunications*, vol. 8, pp. 33–37, 1997.

[11] F. P. Kelly, A. K. Maulloo, and D. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," *Journal of the Operational Research Society*, vol. 49, pp. 237–252, 1998.

[12] R. J. Gibbens and F. P. Kelly, "On packet marking at priority queues," *IEEE Transactions on Automatic Control*, vol. 47, pp. 1016–1020, June 2002.