A Methodology for Virtual Network Partitioning: The Deterministic Approach

Shahrokh Valaee*

August 20, 2004

Abstract:
This paper introduces a methodology for virtual network partitioning in a single node with a generalized processor sharing scheduler. The approach is based on the worst-case service curve provisioning. The true service curve is a function of the number of users and their traffic fluctuations. In this paper, we define a deterministic universal service curve as a lower bound to the true service curve. The universal service curve is time-invariant and is independent of traffic characteristics. We design a novel call admission controller to guarantee that the true service curve is always lower bounded by the deterministic universal service curve. This is achieved by adapting a suitable regulator to the input traffic of each user and allocating appropriate scheduler parameters. The universal service curve is then used to quantify the performance of the network in terms of the maximum delay and the maximum backlog. We further use the universal service curve to propose a virtual subnetwork decomposition. The assignment of an input traffic to a certain subnetwork is based on the proximity of its upper envelope function to the universal service curve. We use min-plus algebra and network calculus to concoct our approach. By using min-plus algebra, we establish a duality between the maximum delay and the maximum backlog.

Keywords: Guaranteed quality-of-service, network partitioning, min-plus algebra, service curve, upper envelope process, generalized processor sharing, traffic regulation.

1 Introduction

Guaranteed quality-of-service (QoS) has been the focus of recent literature in high-speed networking. Providing QoS in general, and guaranteed QoS in particular, impose a significant restriction on network management. The *worst-case design* approach to network provisioning has been utilized as a means to restrain the network operation to a range of behavior induced by "greedy" sources. A greedy source tries to consume all available resources by producing a traffic that meets its worst-case behavior. The worst-case network provisioning commences with the seminal work of Cruz [1] [2]. This approach has later on been explored in several comprehensive subsequent works [3], [4], [5]. The main results, reported in the literature, quantify the worst-case network performance by monitoring the maximum delay and/or the maximum backlog, the scheduling strategy in the nodes, and the parameters of prescribed *regulators*—usually located at the network boundary for *shaping* and/or *filtering* the injected traffic into the network.

The two concepts of traffic *burstiness* and *service curve* have been widely discussed in the literature. The traffic burstiness constructs a bound on the worst-case behavior—in the sense of worst induced backlog—of the input traffic [6] [7]. The burstiness of a source is usually used to anticipate the worst-case behavior of the source in the network and to take appropriate steps to minimize its destructive effect. The service curve, on the other hand, is used to set up a guaranteed performance for the network when exposed to an injected

*The author is with the Department of Electrical and Computer Engineering, University of Toronto, 10 King's College Road, Toronto, Ontario Canada M5S 3G4. Email: valaee@comm.utoronto.ca

traffic. Indeed, the service curve of a user reflects the tendency of the server to the submitted traffic and quantifies the minimum service given to the corresponding user.

The worst-case network design has created a new set of techniques usually referred to as the *network calculus* [8] [9] [10] [11]. Network calculus uses the *min-plus algebra* [12] to instrument the worst-case network design. In min-plus algebra, the addition operator of the conventional algebra is replaced by minimization and the multiplication is replaced by addition. The min-plus algebra provides a systematic approach to traffic regulation and service curve formulation by devising a complying filtering theory [9]. The filtering theory in min-plus algebra imitates its counterpart in the classical signal processing and creates a structured framework for network calculus.

The objective of this paper is two-fold. We introduce a new methodology for *virtual network partitioning* and also devise a novel *call admission controller*. Our approach is to make use of the concept of *service curve*. The service curve has been utilized in network calculus as a means to quantify the minimum guaranteed output for a backlogged traffic. In [3], the service curve has been used as a metric that represents the output traffic in a single node network with a *generalized processor sharing* (GPS) scheduler and a regime of all-greedy leaky bucket regulated traffics. The authors show that if the input traffics are regulated by leaky bucket regulators, the service curve will be a continuous piecewise linear convex function. Service curve, in general, is a function of the scheduler used in the network, the projected data flow, and the volume of data infused by other sources as a crossing traffic. Hence, the service curve is a multivariate functional moving over a range of behavior circumscribed by some unknown time-varying parameters. This observation suggests that using the service curve to construct a framework for guaranteed QoS can be challenging.

Cruz [5] generalizes the concept of service curve by defining the service curve as a function assigned by the network controller to each source, such that the stability condition is satisfied: addition of all service curves at any time instant is smaller than the total capacity. In [13], for a given service curve, the Service Curve Earliest Deadline first (SCED) algorithm has been introduced (see also [14]). Although [5] relaxes the one-to-one relationship between the source regulator parameters and the induced service curve, it does not provide a mechanism for service curve allocation. The network controller should choose the service curve of all connections based on the required QoS for each source. This approach suffers from the scalability problem. Nevertheless, the stability condition requires that the selected service curves be coupled. Therefore, any change on the parameters of one connection may require a rearrangement of the service curve of other connections.

Our approach in this paper is different. We envision the service curve as a random functional changing with the network load. However, if the input traffics satisfy certain constraints, the stochastic behavior of the service curve can be confined to a region with a non-stochastic lower bound. We devise such constraints in this paper. Our technique generalizes the results of [5] by allowing the service curves move arbitrarily inside the stability region. We then compare a weighted version of each service curve with a lower bound. When a service curve coincides with the lower bound, the maximum traffic load is obtained and no new connection is allowed into the network. We show that the proposed methodology solves the scalability problem of [5]. Furthermore, the proposed approach can be applied to distributed call admission control. We have proposed such a call admission controller for wireless LANs in [15].

It has been observed in [16] that it is very challenging to simultaneously achieve fairness and guaranteed service curve. The challenge arises from the fact that the service curves are selected so as to satisfy the stability condition. Therefore, to satisfy this condition, the SCED algorithm should be modified, resulting in a violation of the service curve [16]. Indeed, [16] shows that if fairness is necessary, using only the stability condition does not provide an appropriate set of service curves. In this paper, we solve this problem by allowing the service curves be arbitrary functions that move with the requirements of the input traffic and the fairness indices.

The present paper focuses on assuring guaranteed QoS for input traffics. Here, the network is provisioned to guarantee certain QoS parameters—defined here as the maximum delay and the maximum backlog—for input traffics. We propose to impose a deterministic lower bound on the service curve which we call it the *universal service curve*. The definition of the universal service curve in this paper is different from that used in [3]. In our technique, the universal service curve is independent of input traffics and is selected by the network manager and used for the network life time. The universal service curve acts as a reference curve where all users can refer to quantify their QoS parameters. The application of the universal service curve produces an effective approach to network provisioning by decoupling the interaction among individual sources and reverting it only to the coordination of each user with the universal service curve.

We use the universal service curve as a lower bound to the service provided by the network. Admission of a new traffic into the network is then allowed if the prevailed service curve is secured below by the universal service curve. This bound is then used as a benchmark that can be utilized to capture the worst-case behavior of the projected traffic and to give a framework for a call admission procedure. It also produces a methodology for network partitioning.

In this paper, we use the maximum delay and the maximum backlog to specify the requested QoS and to measure the performance in the presence of greedy sources. The sources are assumed to be greedy in order to reflect the worst-case behavior of the input traffic. As a means to classify the injected traffic, a virtual network partitioning is proposed. We emphasize a "virtual" partitioning since the classified traffics are not totally segregated into isolated subnetworks. We propose a nested network partitioning in which the aggregates of input traffics are treated in a hierarchical manner—unused resources in a higher class can be utilized by a lower hierarchy. This approach is also useful for the *differentiated services* (DiffServ) standard of the internet engineering task force (IETF) [17]. In DiffServ, a *relative* service differentiation is used in which network resources are distributed into certain classes so as to satisfy a relative QoS ordering [18].

Virtual network partitioning is performed on the basis of the proximity of greedy traffic patterns to the universal service curve. We decompose the "support" of universal service curve into non-overlapping intervals. All the traffic upper envelopes intersecting the universal service curve in a given interval are allocated to the same subnetwork. We will show that the maximum delay and the maximum backlog for an input traffic allocated to a certain subnetwork, are bounded in a prescribed region.

We also use GPS or one of its non-preemptive variants [3]. In GPS, the backlogged traffic of each user is handled by a server, with a specified *fair share*, usually dictated by a non-negative real number, indicating the relative proportions of the served traffics of the sources. GPS provides a relative fairness among the users sharing a common infrastructure. The GPS weight is only valid over a single node, therefore we do not consider a multi-node GPS assignment as suggested in [19]. In this paper, a multi-node case can be treated using an approach similar to [2]: the total maximum delay is the summation of the maximum delays in all nodes along the path. Therefore, analyzing each node is independent from the rest of the network. This assumption decouples the interaction between tandem nodes and justifies the study of virtual network decomposition in a single node.

2 Background

The set of all non-decreasing functions over $\mathbb{R}^+$ is denoted by $\mathcal{J} \triangleq \{a(t)\,|\,0 \leq a(t_1) \leq a(t_2),\ \text{for } 0 \leq t_1 \leq t_2\}$. Define also $\mathcal{J}_0 \triangleq \{a(t)\,|\,a(t) \in \mathcal{J}, a(0) = 0\}$. Let f_c and f_v be, respectively, the set of all concave and convex functions. We use the *min-plus algebra* [12] to handle the algebraic manipulation of sequences in $\mathcal{J}$. In min-plus algebra, the addition operator of the conventional algebra is replaced by point-wise minimization, denoted here by the notation $\oplus$, and the multiplication is replaced by point-wise addition, represented by $\otimes$. In fact, for $a(t), b(t) \in \mathcal{J}$, $a(t) \oplus b(t) \triangleq \min\{a(t), b(t)\}$, and $a(t) \otimes b(t) \triangleq a(t) + b(t)$. The sets $\mathcal{J}$ and $\mathcal{J}_0$ are closed under $\oplus$ and $\otimes$ operations. It is also possible to show that $(\mathcal{J}, \oplus, \otimes)$ is a *complete dioid* [20]. The *scalar projection* of a function $a(t)$ onto $b(t)$ was defined in [21] as

$$\langle a(t), b(t)\rangle = \sup_{t\geq 0}[a(t) - b(t)] \vee 0. \tag{1}$$

Definition 1 For $a(t) \in \mathcal{J}$, we define the *adjoint mapping* as $a^*(t) \triangleq t + \inf\{d\,|\,t \leq a(t+d)\}$. The reverse mapping is defined as $a(t) = t - \inf\{d\,|\,a^*(t-d) \leq t\}$.

It is straightforward to show that there is a one-to-one relationship between $a(t)$ and $a^*(t)$. In other words, for each $a(t) \in \mathcal{J}$ there exists a unique $a^*(t) \in \mathcal{J}$. Fig. 1 illustrates an example with two functions $a(t) \in \mathcal{J}_0$

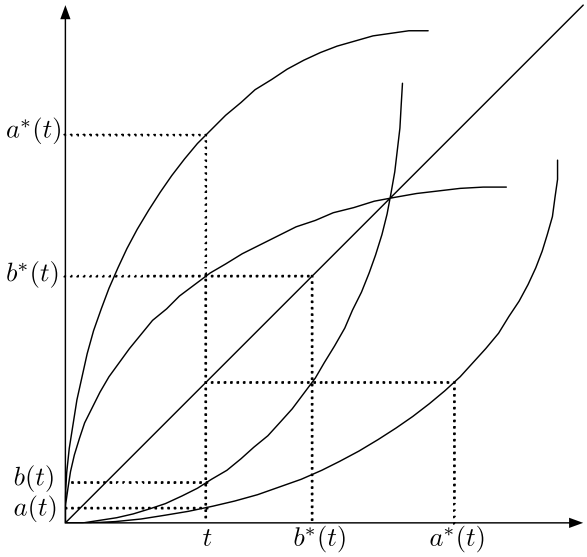


Figure 1: The adjoint mapping for functions $a(t) \in \mathcal{J}_0$ and $b(t) \in \mathcal{J}_0$.

and $b(t) \in \mathcal{J}_0$ and their adjoint mapping. The adjoint mapping of a function is the reflection of that function about the line $a(t) = t$.

The cumulative input traffic for source i over the interval $[0, t]$ is denoted by a non-decreasing function $A_i(t) \in \mathcal{J}_0$. Assume that the input $A_i(t)$ is served by a node which assigns to this traffic, the service curve $S_i(t; \underline{A}, \Phi) \in \mathcal{J}_0$, in which $\underline{A} \triangleq \Big(A_1(t), \ldots, A_N(t)\Big)$ is the vector of all traffics, N is the total number of connections, and Φ is the scheduler used in the node. The network guarantees the service curve $S_i(t; \underline{A}, \Phi)$ for the input traffic $A_i(t)$ if the output of the server, $B_i(t; \underline{A}, \Phi)$, satisfies

$$B_i(t; \underline{A}, \Phi) \geq \inf_{0 \leq s \leq t} \{A_i(s) + S_i(t - s; \underline{A}, \Phi)\}. \tag{2}$$

In min-plus algebra, this inequality can be represented by

$$B_i(t; \underline{A}, \Phi) \geq A_i(t) \star S_i(t; \underline{A}, \Phi), \tag{3}$$

where $\star$ is the convolution operator defined as

$$a(t) \star b(t) = \bigoplus_{0 \leq s \leq t} \{a(s) \otimes b(t - s)\}. \tag{4}$$

A traffic $A_i(t)$ is called a_i-*upper constrained* [9], if

$$A_i(t) \leq A_i(t) \star a_i(t). \tag{5}$$

where $a_i(t)$ is the *upper envelope process.* The source is called *greedy* if $A_i(t) = a_i(t)$. Define the aggregated input traffic

$$A(t) \triangleq \sum_{i=1}^{N} A_i(t). \tag{6}$$

If the input traffics, $A_i(t)$, are a_i-upper constrained for all $i = 1, \ldots, N$, the aggregated traffic $A(t)$ will be a-upper constrained with

$$a(t) = \sum_{i=1}^{N} a_i(t). \tag{7}$$

The multiplexed traffic, $A(t)$, is called *all-greedy* if $A(t) = a(t)$. Throughout the paper, we assume that the input traffic satisfies

$$\limsup_{t\to\infty} \frac{a(t)}{t} < C. \tag{8}$$

We further assume that the node uses a GPS scheduler [3] with the parameters $\phi_i, i = 1, \ldots, N$, satisfying

$$\sum_{i=1}^{N} \phi_i < 1. \tag{9}$$

In GPS, the service provided for a session i, which is continuously backlogged over the interval $[0, t]$, satisfies

$$\frac{S_i(t; \underline{A}, \Phi)}{\phi_i} \geq \frac{S_j(t; \underline{A}, \Phi)}{\phi_j}. \tag{10}$$

A session i is said to be *backlogged* at time t, if $q_i(t; \underline{A}, \Phi) > 0$; the server is backlogged at t, if there exists an $i \in \{1, \ldots, N\}$ such that $q_i(t; \underline{A}, \Phi) > 0$. Equality holds in (10), if both sessions i and j are backlogged in $[0, t]$. Therefore, the normalized service curve of all backlogged connections are identical. This is an important property of the GPS scheduler since a unique service curve can quantify the service given to all backlogged connections. We represent the set of all backlogged sessions at time t by $\mathcal{B}(t)$. The complementary set of $\mathcal{B}(t)$—the set of unbacklogged sessions—is represented by $\mathcal{B}^c(t)$. In a GPS scheduler, the service provided for a backlogged session i at time t is given by

$$S_i(t; \underline{A}, \Phi) = \phi_i \frac{Ct - \sum_{j\in\mathcal{B}^c(t)} A_j(t)}{\sum_{j\in\mathcal{B}(t)} \phi_j}. \tag{11}$$

We use the *maximum delay* and the *maximum backlog* as the QoS parameters. The delay, $d_i(t; \underline{A}, \Phi)$, and the backlog, $q_i(t; \underline{A}, \Phi)$, for session i, with the input traffic $A_i(t)$ at time t, are defined as

$$d_i(t; \underline{A}, \Phi) \triangleq \inf\{d \geq 0 \mid A_i(t) \leq S_i(t + d; \underline{A}, \Phi)\}, \tag{12}$$

$$q_i(t; \underline{A}, \Phi) \triangleq A_i(t) - S_i(t; \underline{A}, \Phi). \tag{13}$$

The objective is to give a set of constraints over $\underline{A}$ and Φ such as to guarantee

$$\sup_{t\geq 0} d_i(t; \underline{A}, \Phi) \leq D_i, \tag{14}$$

$$\sup_{t\geq 0} q_i(t; \underline{A}, \Phi) \leq Q_i. \tag{15}$$

The bounds D_i and Q_i in (14) and (15)—indicated as the prescribed QoS parameters—are the maximum delay and the maximum backlog for source i, respectively. Using (1) and Definition 1, (14) and (15), are given by

$$\langle S_i^*(t; \underline{A}, \Phi)\,,\, A_i^*(t)\rangle \leq D_i, \tag{16}$$

$$\langle A_i(t)\,,\, S_i(t; \underline{A}, \Phi)\rangle \leq Q_i. \tag{17}$$

Note that (16) and (17) manifest a duality for the maximum delay and the maximum backlog. The maximum backlog is the scalar projection of the traffic curve onto the service curve. Similarly, the maximum delay can be interpreted as the projection of the adjoint service curve onto the adjoint traffic curve.

3 Universal Service Curve

Assume a scenario with N sources using a node with capacity, C, and a work-conserving GPS scheduler. The GPS weights are assigned by the node during the call admission process. The traffic of each source i is guaranteed by a service curve $S_i(t; \underline{A}, \Phi)$. Since the total service given in a time unit cannot exceed the link bandwidth, we should have [13]

$$\sum_{i=1}^{N} A_i(t) \star S_i(t; \underline{A}, \Phi) \leq Ct. \tag{18}$$

Equality holds if the server is work-conserving and the node is continuously backlogged over the interval $[0, t]$.

Inequality (18) shows that the service curves $S_i(t; \underline{A}, \Phi), i = 1, \ldots, N$, are dependent and cannot be selected independently. Hence, altering the traffic descriptors and/or the scheduling parameters of a single source might affect the performance of all sources. Indeed, admitting a new traffic $A_i(t)$ into the network will reduce the service given to some of the backlogged sessions. The maximum delay and the maximum backlog for session i in an all-greedy regime are defined as

$$d_i(\underline{a}, \Phi) \triangleq \langle S_i^*(t; \underline{a}, \Phi), a_i^*(t) \rangle, \tag{19}$$

$$q_i(\underline{a}, \Phi) \triangleq \langle a_i(t), S_i(t; \underline{a}, \Phi) \rangle \tag{20}$$

where $\underline{a} \triangleq (a_1(t), \ldots, a_N(t))$ is the vector of all traffic upper envelope processes, and $S_i(t, \underline{a}, \Phi)$ is the service curve of user i under an all-greedy regime.

We denote the *depletion time*, T_i, for each session i in an all-greedy regime, by

$$T_i \triangleq \inf\{t > 0 \,|\, a_i(t) \leq S_i(t, \underline{a}, \Phi)\}. \tag{21}$$

Without loss of generality, we assume that the sessions are numbered in the increasing order of depletion times, $T_1 \leq T_2 \leq \ldots \leq T_N$. Let also $T_0 = 0$. The service curve $S_i(t; \underline{a}, \Phi)$, $i = 1, \ldots, N$, for a GPS scheduler is then represented as

$$S_i(t; \underline{a}, \Phi) = \begin{cases} \phi_i \frac{Ct - \sum_{j=1}^{k} a_j(t)}{\sum_{j=k+1}^{N} \phi_j} & \text{for } t \in [T_k, T_{k+1}),\ k = 0, \ldots, i-1, \\ \infty & \text{for } t \geq T_i. \end{cases} \tag{22}$$

Theorem 1 *Let $A(t) = a(t)$.*

(i) If $a_j(t)$ is concave for all $j = 1, \ldots, N$, the service curve $S_i(t; \underline{a}, \Phi), i = 1, \ldots, N$, will be convex in $[0, T_i]$;

(ii) If $a_j(t)$ is u.s.c.[1] for all $j = 1, \ldots, N$, the service curve $S_i(t; \underline{a}, \Phi), i = 1, \ldots, N$, will be l.s.c. in $[0, T_i]$;

(iii) If $a_j(t)$ is sub-additive for all $j = 1, \ldots, N$, the service curve $S_i(t; \underline{a}, \Phi), i = 1, \ldots, N$, will be super-additive in $[0, T_i]$.

Proof: See Appendix A. □

Theorem 1 can be considered as the generalization of the results of [3]. The concave upper envelope processes $a_i(t)$, used in Theorem 1, are of practical importance; regulators of the form $\min_i\{\sigma_i + \rho_i t\}$ provide concave upper envelope processes. These type of regulators are used in the ATM Forum [22] and the IETF IntServ [23] standards. Note also that concave functions are subadditive [8], a requirement for upper envelope processes [4].

[1]A function $f(t)$ is called *upper semi-continuous* (u.s.c.) at t_0 if $f(t_0) = \limsup_{t \to t_0} f(t)$. It is called *lower semi-continuous* (l.s.c.) at t_0 if $f(t_0) = \liminf_{t \to t_0} f(t)$.

Since the service curves $S_i(t;\underline{a},\Phi), i=1,\ldots,N$, satisfy $\frac{S_i(t;\underline{a},\Phi)}{\phi_i} = \frac{S_j(t;\underline{a},\Phi)}{\phi_j}$ for $t \le T_i \oplus T_j$, one can verify that

$$\Big\langle \frac{S_i(t;\underline{a},\Phi)}{\phi_i}, \frac{S_j(t;\underline{a},\Phi)}{\phi_j} \Big\rangle = \begin{cases} 0 & i \ge j, \\ \infty & i < j. \end{cases} \tag{23}$$

Hence, $\left\{\frac{S_i(t;\underline{a},\Phi)}{\phi_i}\right\}$ is a set of ordered orthogonal bases [21]. The span of $\left\{\frac{S_i(t;\underline{a},\Phi)}{\phi_i}\right\}$ is the set of all linear combinations, $\bigoplus_{i=1}^{N} \Big(\alpha_i \otimes \frac{S_i(t;\underline{a},\Phi)}{\phi_i}\Big)$ with $\alpha_i \in I\!R^+$.

We can define the *minimum service curve* $S(t;\underline{a},\Phi)$ as

$$S(t;\underline{a},\Phi) \triangleq \bigoplus_{1\le i\le N} \frac{S_i(t;\underline{a},\Phi)}{\phi_i}. \tag{24}$$

The minimum service curve, $S(t;\underline{a},\Phi)$, is a linear combination of the bases $\left\{\frac{S_i(t;\underline{a},\Phi)}{\phi_i}\right\}$ with the $\alpha_i = 0$ (note that $\alpha_i = 0$ is the identity element in min-plus algebra). Hence, $S(t;\underline{a},\Phi) \in \text{span}\left\{\frac{S_i(t;\underline{a},\Phi)}{\phi_i}\right\}$. Since $(\mathcal{J},\oplus,\otimes)$ is a complete dioid, $S(t;\underline{a},\Phi) \in \mathcal{J}$ (the property also holds for $N \to \infty$). Representation (24) can also be interpreted as a combination of N parallel filters operating in min-plus algebra [9]. Since $S(t;\underline{a},\Phi)$ is a function of the number of sources, N, and the upper envelope processes, $a_i(t), i = 1,\ldots,N$, it is a time-varying filter.

Lemma 1 *For $A(t) = a(t)$, and $a_j(t)$ concave and u.s.c. for all $j \in \{1,\ldots,N\}$, the minimum service curve $S(t;\underline{a},\Phi)$ is convex and l.s.c.*

Proof: Use (23) to get $\frac{S_N(t;\underline{a},\Phi)}{\phi_N} \le \frac{S_i(t;\underline{a},\Phi)}{\phi_i}$, for all $i = 1\ldots,N-1$. Hence $S(t;\underline{a},\Phi) = \frac{S_N(t;\underline{a},\Phi)}{\phi_N}$ for $0 \le t \le T_N$. In Theorem 1, we proved that $S_i(t;\underline{a},\Phi)$ is convex and l.s.c. in $[1,T_i]$ for all $i=1,\ldots,N$. □

Definition 2 The *universal service curve*, $\bar{S}(t) \in \mathcal{J}_0$, is defined as a positive, increasing, l.s.c., convex function, independent of the traffic bounds $\underline{a}$ and the scheduler Φ, that satisfies for all t,

$$\bar{S}(t) < S(t;\underline{a},\Phi). \tag{25}$$

If the lower bound (25) could be found, the delay and the backlog will be bounded above by

$$d_i(t;\underline{A},\Phi) \le \inf\{d \ge 0 \,|\, A_i(t) \le \phi_i\bar{S}(t+d)\}, \tag{26}$$
$$q_i(t;\underline{A},\Phi) \le A_i(t) - \phi_i\bar{S}(t). \tag{27}$$

Then, the QoS parameters, D_i and Q_i, are guaranteed for the input traffic $A_i(t)$, if the following constraints hold

$$\langle \bar{S}^*(\frac{t}{\phi_i}), A_i^*(t)\rangle \le D_i, \tag{28}$$
$$\langle A_i(t), \phi_i\bar{S}(t)\rangle \le Q_i, \tag{29}$$
$$S_i(t;\underline{A},\Phi) \in \mathcal{S}_i, \tag{30}$$

where $\mathcal{S}_i$ is defined as $\mathcal{S}_i \triangleq \{S(t) \,|\, S(t) \in \mathcal{J}_0, S(t) \ge \phi_i\bar{S}(t)\}$. A call can be accepted into the network if (28)-(30) hold for all i. This requires that $A_i(t)$ be constrained to the region $\phi_i\bar{S}(t) \le S_i(t;\underline{A},\Phi) \le A_i(t) \le \phi_i\bar{S}(t\otimes D_i) \oplus (\phi_i\bar{S}(t) \otimes Q_i)$ as illustrated in Fig. 2.

In practice, for any $S(t;\underline{a},\Phi)$ it is possible to find a universal service curve $\bar{S}(t)$. However, in this paper, we address this problem from a different perspective. We assume that the network manager chooses an appropriate $\bar{S}(t)$ and then only those calls are accepted into the network that satisfy (25). Note that $\bar{S}(t)$

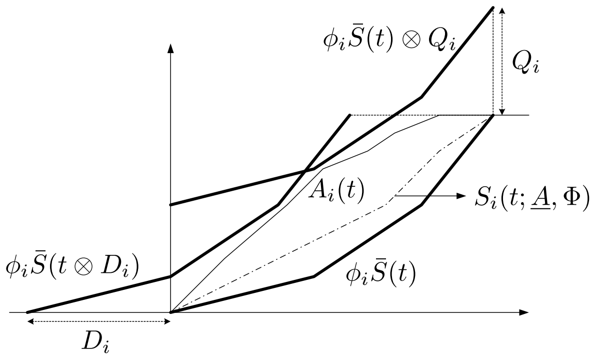


Figure 2: The deterministic lower boundary of the service curve.

is a parameter of design that should be properly selected by the network manager. There might be a large number of curves that can be selected as the universal service curve. There is not a unique approach to select such a curve. The universal service curve acts as the lower bound to the service provided by network. Therefore, selecting an appropriate universal service curve is related to the call initiation process, guaranteed QoS, and pricing. The selection of universal service curve is an off-line procedure that should be performed once for the network life time. In general, an appropriate universal service curve can be determined by a tradeoff on the guaranteed QoS of all sources and the network usage (number of connections). These two factors are usually contradictory. In one hand, accepting more connections usually increases the total revenue of the network. On the other hand, more connections will produce a degraded service. Therefore, the network controller should select a universal service curve that guarantees a required QoS for all connections and also increases the network usage. In Section 7, we will give an example on how a proper universal service curve can be selected. For the moment, we assume that a universal service curve is given and we continue to use it in our network decomposition technique and call admission control.

Using the definition of the minimum service curve $S(t; \underline{a}, \Phi)$, we have $\left\langle \bar{S}(t), \frac{S_i(t;\underline{a},\Phi)}{\phi_i} \right\rangle = 0$, for all $i = 1, \ldots, N$. If we also take $\left\langle \frac{S_i(t;\underline{a},\Phi)}{\phi_i}, \bar{S}(t) \right\rangle = \infty$, then $\left\{ \frac{S_1(t;\underline{a},\Phi)}{\phi_1}, \ldots \frac{S_N(t;\underline{a},\Phi)}{\phi_N}, \bar{S}(t) \right\}$ will be a set of ordered orthogonal bases. Hence, for every $a(t) \in \mathcal{J}_0$, we will have $\langle a(t), \frac{S_i(t;\underline{a},\Phi)}{\phi_i} \rangle \leq \langle a(t), \bar{S}(t) \rangle$ and $\langle \bar{S}^*(t), a^*(t) \rangle \leq \langle S_i^*(\phi_i t; \underline{a}, \Phi), a^*(t) \rangle$. In the following theorem, we will find the necessary and sufficient conditions under which (25) is satisfied.

Theorem 2 *Let $S(t) \in \mathcal{J}_0$ and $\bar{S}(t) \in \mathcal{J}_0$ be two positive, increasing, l.s.c., convex functions, and assume $\dot{S}(0^+) > \dot{\bar{S}}(0^+)$. A necessary and sufficient condition for $\bar{S}(t) < S(t)$, $0 < t < T$, is that there exists a countable set of time indices $0 = t_0 < t_1 < t_2 < \ldots$, such that*

$$S(t_j) + \dot{S}(t_j^+)(t_{j+1} - t_j) > \bar{S}(t_{j+1}), \tag{31}$$

for all $j = 0, 1, 2, \ldots$ and $\lim_{j \to \infty} t_j \geq T$.

Proof: See Appendix B. □

The condition (31) of Theorem 2 can be interpreted as follows. Assume two affine functions $\Delta_j(t) \triangleq S(t_j) + \dot{S}(t_j^+)(t - t_j)$ and $\bar{\Delta}_j(t) \triangleq \bar{S}(t_j) + \frac{\bar{S}(t_{j+1}) - \bar{S}(t_j)}{t_{j+1} - t_j}(t - t_j)$ acting, respectively, as the lower bound of $S(t)$ and the upper bound of $\bar{S}(t)$ over the interval $[t_j, t_{j+1}]$ (see Fig. 3 for an illustration.) We have $\Delta_j(t) = \inf\{\tilde{S}(t) \,|\, \tilde{S}(t) \in f_v \text{ and } \tilde{S}(t_j) = S(t_j), \dot{\tilde{S}}(t_j^+) \geq \dot{S}(t_j^+)\}$, and $\bar{\Delta}_j(t) = \sup\{\tilde{S}(t) \,|\, \tilde{S}(t) \in f_v \text{ and } \tilde{S}(t_j) = \bar{S}(t_j), \tilde{S}(t_{j+1}) = \bar{S}(t_{j+1})\}$. Theorem 2 states that the necessary and sufficient conditions for $S(t) \geq \bar{S}(t)$ is that there exists a set of time indices $0 = t_0 < t_1 < t_2 < \ldots$, such that $\Delta_j(t) \geq \bar{\Delta}_j(t)$ for $t_j \leq t \leq t_{j+1}, j = 0, 1, \ldots$.

The following corollary gives a sufficient condition on $\bar{S}(t) \leq S(t), 0 \leq t \leq T$.

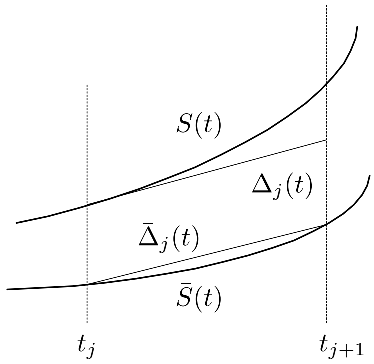


Figure 3: An illustration for Theorem 2.

Corollary 1 *Let $S(t) \in \mathcal{J}_0$ and $\bar{S}(t) \in \mathcal{J}_0$ be two positive, increasing, l.s.c., convex functions. A sufficient condition for $\bar{S}(t) \leq S(t), 0 \leq t \leq T$ is to have $\dot{\bar{S}}(t) \leq \dot{S}(t)$ for all $0 \leq t \leq T$.*

4 Hierarchical Network Partitioning

Here, we devise a framework for bandwidth provisioning based on a partitioning of the link capacity into a hierarchy of subchannels. Network partitioning is performed on the basis of an observation made in Theorem 2: if we could select a set of time indices $0 = t_0 < t_1 < \ldots, < t_L$, such that (31) is satisfied over this set, we will have $S(t; \underline{a}, \Phi) \geq \bar{S}(t)$ for $0 \leq t \leq t_L$. We proceed as follows.

We decompose the support of $\bar{S}(t)$ into L non-overlapping intervals. The end points of the intervals are indicated by, $0 = \bar{T}_0 < \bar{T}_1 < \ldots < \bar{T}_L$, where, without loss of generality, we might have $\bar{T}_L = \infty$. The L hierarchical virtual subnetworks are denoted here by $\Gamma_\ell, \ell = 1, \ldots, L$. Define $\bar{S}_\ell \triangleq \bar{S}(\bar{T}_\ell)$.

Definition 3 We say that the connection i belongs to the virtual network Γ_ℓ if

$$\inf \left\{ t > 0 \,\middle|\, b_i(t) \leq \bar{S}(t) \right\} \leq \bar{T}_\ell \tag{32}$$

where $b_i(t) \triangleq \frac{a_i(t)}{\phi_i}$ is the *normalized upper envelope process* of source i.

Fig. 4 illustrates an example with a leaky bucket regulated traffic. The universal service curve in this figure is selected as a piecewise linear convex function. Note that the left-hand-side of (32) is the intersection of the upper envelope $b_i(t)$ and the universal service curve $\bar{S}(t)$. Since $b_i(t)$ is a concave increasing function, and $\bar{S}(t)$ is a convex function, $i \in \Gamma_\ell$ if and only if $b_i(\bar{T}_\ell) \leq \bar{S}_\ell$. It is also possible to show that if $b_i(\bar{T}_\ell) \leq \bar{S}_\ell$, then $b_i(\bar{T}_p) \leq \bar{S}_p$ for all $\ell \leq p \leq L$. This suggests that the virtual networks Γ_ℓ are nested as

$$\Gamma_1 \subseteq \ldots \subseteq \Gamma_L. \tag{33}$$

In a nested structure, any unused bandwidth in a virtual network can be consumed by the sessions of a higher subnetwork.

Our definition of subnetworks is different from the conventional definitions. Although the subnetworks are nested, the connections in a subnetwork with a smaller index do not necessarily perform better—in terms of maximum delay and maximum backlog—than the connections in subnetworks with higher indexes. In our approach, each connection can have a smaller maximum delay or maximum backlog if it is accepted into a subnetwork with a smaller index. However, a connection $k \in \Gamma_j$ might have a smaller maximum delay and maximum backlog than another connection $k' \in \Gamma_i$ with $\Gamma_i \subset \Gamma_j$. In other words, the performance of each

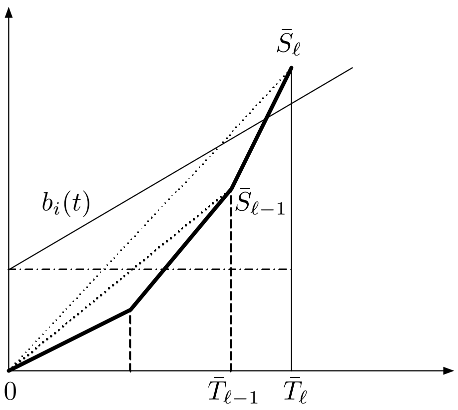


Figure 4: An example with a leaky bucket regulated traffic and a piecewise linear convex universal service curve.

connection in any subnetwork can only be compared to the performance of the same connection in other subnetworks. This observation relates to the fact that using the universal service curve for call admission decouples the connections and removes the dependency among their QoS indexes.

In the sequel, we derive several sufficient conditions which satisfy $\bar{S}(t) \leq S(t; \underline{a}, \Phi)$, for $0 \leq t \leq \bar{T}_L$.

In Corollary 1, the derivative of the universal service curve should be smaller than the derivative of the true service curve over the whole interval $[0, \bar{T}_L)$. We can reduce the complexity by applying a virtual subnetwork decomposition. Let $\bar{T}_0 = 0$ and assume that $[0, \bar{T}_L)$ is decomposed into L non-overlapping intervals $[\bar{T}_{\ell-1}, \bar{T}_\ell), \ell, = 1, \ldots, L$ with $\bar{T}_0 < \bar{T}_1 < \ldots < \bar{T}_L$. Using the sufficient conditions of Theorem 2, we can show that $S(t; \underline{a}, \Phi) \geq \bar{S}(t)$ for $0 < t < T_L$, if

$$S(\bar{T}_{\ell-1}; \underline{a}, \Phi) + \dot{S}(\bar{T}^{+}_{\ell-1}; \underline{a}, \Phi)(\bar{T}_\ell - \bar{T}_{\ell-1}) \geq \bar{S}_\ell. \tag{34}$$

for all $\ell = 1, \ldots, L$. The conditions (34) only use the value of the true service curve and its derivative on L points over the whole interval $[0, \bar{T}_L]$ to verify whether it is lower bounded by the universal service curve. We can also use $\dot{S}(\bar{T}^{+}_{\ell-1}; \underline{a}, \Phi)(\bar{T}_{\ell-1} - \bar{T}_{\ell-2}) \geq S(\bar{T}_{\ell-1}; \underline{a}, \Phi) - S(\bar{T}_{\ell-2}; \underline{a}, \Phi)$ to show that if

$$S(\bar{T}_{\ell-1}; \underline{a}, \Phi)(\bar{T}_\ell - \bar{T}_{\ell-2}) - S(\bar{T}_{\ell-2}; \underline{a}, \Phi)(\bar{T}_\ell - \bar{T}_{\ell-1}) \geq \bar{S}_\ell(\bar{T}_{\ell-1} - \bar{T}_{\ell-2}) \tag{35}$$

for all $\ell = 2, \ldots, L$, and $\dot{S}(0^+; \underline{a}, \Phi) \geq \dfrac{\bar{S}_1}{\bar{T}_1}$, then $S(t; \underline{a}, \Phi) \geq \bar{S}(t)$, for $0 < t < T_L$.

Equations (34) and (35) give sufficient conditions for $S(t; \underline{a}, \Phi) \geq \bar{S}(t)$. Note that these conditions depend on the service curve $S(t; \underline{a}, \Phi)$. It is possible to show that for affine regulators the conditions (34) and (35) are translated to user traffic descriptors [24].

4.1 An Example

Here, we give an example to further study the virtual network partitioning technique. Assume that we have a set of fictitious leaky bucket regulated connections with the parameters $\{(\bar{\rho}_\ell, \bar{\sigma}_\ell, \bar{\phi}_\ell); \ell = 1, \ldots, L\}$ where $\sum_{\ell=1}^{L} \bar{\phi}_\ell = 1$. It is possible to find a piecewise linear convex service curve $\bar{S}(t)$, which is the normalized service curve of an all greedy GPS node with the leaky bucket parameters $\{\bar{\rho}_\ell, \bar{\sigma}_\ell; \ell = 1, \ldots, L\}$ and the GPS weights $\{\bar{\phi}_\ell; \ell = 1, \ldots, L\}$ [3]. Let us represent the break points of $\bar{S}(t)$ by $\bar{T}_1 < \ldots < \bar{T}_L$ and let $\bar{T}_0 = 0$. It is possible to show that $\bar{S}(t)$ is the only service curve for the connections $\{(\bar{\rho}_\ell, \bar{\sigma}_\ell, \bar{\phi}_\ell); \ell = 1, \ldots, L\}$ [24]. Therefore, there is a one-to-one relationship between the set of parameters $\{(\bar{\rho}_\ell, \bar{\sigma}_\ell, \bar{\phi}_\ell); \ell = 1, \ldots, L\}$ and the service curve $\bar{S}(t)$.

Assume that the total capacity of the link is normalized to 1 and define

$$\bar{\rho}(\ell) = \sum_{m=1}^{\ell} \bar{\rho}_m \tag{36}$$

$$\bar{\phi}(\ell) = \sum_{m=\ell+1}^{L} \bar{\phi}_m. \tag{37}$$

Now let us choose $\bar{S}(t)$ as the universal service curve of the network. The network accepts a set of leaky bucket regulated traffics with parameters $\{\sigma_i, \rho_i, \phi_i; i = 1, \ldots, N\}$ into the network. The universal service curve $\bar{S}(t)$ decomposes the traffics into L virtual subnetworks. For all $\ell = 1, \ldots, L$, we also define

$$\rho(\Gamma_\ell) = \sum_{i \in \Gamma_\ell} \rho_i \tag{38}$$

$$\phi(\Gamma_\ell) = \sum_{i \in \Gamma_\ell} \phi_i. \tag{39}$$

Since the true service curve is a piecewise linear convex function [3] given by

$$\frac{1}{\phi_i} S_i(t; \underline{a}, \Phi) = \frac{t - \sum_{j \in \mathcal{B}^c(t)} (\sigma_j + \rho_j t)}{\sum_{j \in \mathcal{B}(t)} \phi_j}, \tag{40}$$

using Corollary 1, it is possible to show that if for all $\ell = 1, \ldots, L$,

$$\rho(\Gamma_\ell) \leq \bar{\rho}(\ell) \tag{41}$$

$$\phi(\Gamma_\ell) \geq \phi(\Gamma_L) - \bar{\phi}(\ell) \tag{42}$$

$$\phi(\Gamma_L) \leq 1, \tag{43}$$

then $S(t; \underline{a}, \phi) \geq \bar{S}(t)$ for all $0 \leq t \leq \bar{T}_L$.

This example shows an interesting observation on the virtual network decomposition technique. Let $\bar{\rho}(\ell)$ represent the capacity of the ℓth subnetwork. We can also indicate by $\rho(\Gamma_\ell)$, the total accepted bandwidth into the ℓth subnetwork. Equation (41) illustrates the stability condition of the ℓth subnetwork; that is, the total accepted load into each subnetwork should be smaller than the capacity of that subnetwork. Similar interpretation could be proposed for the GPS weights. If $\sum_{m=1}^{\ell} \bar{\phi}_m \leq \phi(\Gamma_\ell)$, then (42) will also hold. Therefore, the total service given to the connections in a subnetwork should be more than that of the subnetwork. Fig. 5 depicts an instance at which a number of calls are allocated to 3 subnetworks with the ordered capacities $\bar{\rho}(1) < \bar{\rho}(2) < \bar{\rho}(3)$. The total bandwidth assigned to each subnetwork is shown by $\rho(\Gamma_1)$, $\rho(\Gamma_2)$, and $\rho(\Gamma_3)$. In this figure $\rho(\Gamma_1) < \bar{\rho}(1)$, $\rho(\Gamma_2) < \bar{\rho}(2)$, and $\rho(\Gamma_3) < \bar{\rho}(3)$. If we can further assume that $\phi(\Gamma_i) \geq \phi(\Gamma_3) - \bar{\phi}(i)$, for $i = 1, 2, 3$, then we can conclude that $S(t; \underline{a}, \phi) \geq \bar{S}(t)$ for all $0 \leq t \leq \bar{T}_3$. Note that unused bandwidth in Γ_1 can be utilized in Γ_2 and unused bandwidth in Γ_2 can be consumed in Γ_3. Therefore, the subnetworks are truly nested.

4.2 QoS Bounds

In this subsection, we find upper bounds and lower bounds on the maximum delay and maximum backlog of a session with the upper envelope function $b_i(t)$ that belongs to a virtual network Γ_ℓ.

Proposition 1 *If $i \in \Gamma_\ell$ and $b_i(t) \in f_c$, the delay, $d_i(a_i, \bar{S}) \triangleq \langle \bar{S}^*(t), b_i^*(t) \rangle$, and the backlog, $\frac{q_i(a_i, \bar{S})}{\phi_i} \triangleq \langle b_i(t), \bar{S}(t) \rangle$, will satisfy*

$$\langle \frac{\bar{T}_\ell}{\bar{S}_\ell} t, b_i^*(t) \rangle \leq d_i(a_i, \bar{S}) \leq \bar{T}_\ell, \tag{44}$$

$$\langle b_i(t), \frac{\bar{S}_\ell}{\bar{T}_\ell} t \rangle \leq \frac{q_i(a_i, \bar{S})}{\phi_i} \leq \bar{S}_\ell. \tag{45}$$

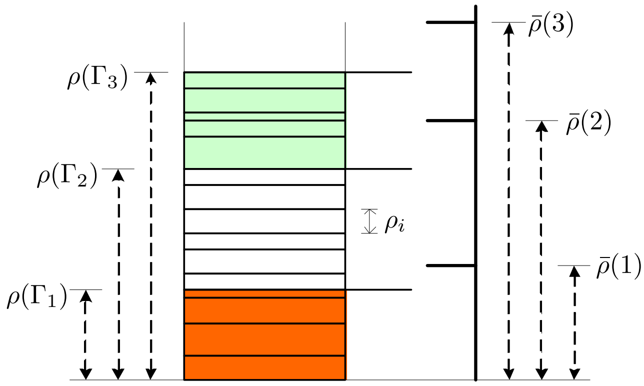


Figure 5: Nested virtual subnetworks in an example with leaky-bucket regulated traffics. Here, $i \in \Gamma_2$. The aggregated bandwidth in each subnetwork should be smaller than the total capacity of that subnetwork, that is, $\rho(\Gamma_\ell) \leq \bar{\rho}(\ell)$, $\ell = 1, 2, 3$.

For the proof note that if $i \in \Gamma_\ell$, then $\frac{\bar{S}_\ell}{\bar{T}_\ell} t \geq \bar{S}(t) \geq \bar{O}_\ell(t)$ for $0 \leq t \leq \bar{T}_\ell$, where $\bar{O}_\ell(t)$ is defined as $\bar{O}_\ell(t) = 0$ for $0 \leq t \leq \bar{T}_\ell$, and $\bar{O}_\ell(t) = \infty$ for $t \geq \bar{T}_\ell$.

Fig. 4 illustrates an example with an affine upper envelope—as used in leaky bucket regulated traffics—and a piecewise linear convex universal service curve. The lower bounds can simply be found by calculating the maximum vertical and horizontal distances between the traffic upper envelope and the line $\frac{\bar{S}_\ell}{\bar{T}_\ell} t$. Note that for each subnetwork it suffices to store the pair $(\bar{T}_\ell, \bar{S}_\ell)$.

We observe in Fig. 4 that the bounds of Proposition 1 can be very loose. Tighter bounds can be found if we define

$$\hat{S}(t) \triangleq \bigvee_{0 \leq \ell \leq L} \left\{ \bar{S}_\ell + \dot{\bar{S}}(\bar{T}_\ell)(t - \bar{T}_\ell) \right\} \tag{46}$$

$$\check{S}(t) \triangleq \bigvee_{1 \leq \ell \leq L} \left\{ \bar{S}_{\ell-1} + \frac{\bar{S}_\ell - \bar{S}_{\ell-1}}{\bar{T}_\ell - \bar{T}_{\ell-1}}(t - \bar{T}_{\ell-1}) \right\} \tag{47}$$

where $\dot{\bar{S}}(\bar{T}_\ell)$ is the derivative[2] of $\bar{S}(t)$ at $\bar{T}_\ell$. Since $\hat{S}(t) \leq \bar{S}(t) \leq \check{S}(t)$, it is straightforward to prove the following proposition.

Proposition 2 *If $i \in \Gamma_\ell$, then*

$$\langle \check{S}^*(t), b_i^*(t) \rangle \leq d_i(a_i, \bar{S}) \leq \langle \hat{S}^*(t), b_i^*(t) \rangle \tag{48}$$

$$\langle b_i(t), \check{S}(t) \rangle \leq \frac{q_i(a_i, \bar{S})}{\phi_i} \leq \langle b_i(t), \hat{S}(t) \rangle. \tag{49}$$

It is also possible to find lower bounds for the maximum delay and maximum backlog if we know that a call does not belong to a virtual network. This fact has been investigated in the following proposition.

Proposition 3 *If $i \notin \Gamma_\ell$ and $a_i(t) \in f_c$, the maximum delay, $d_i(a_i, \bar{S})$, and the maximum backlog, $\frac{q_i(a_i, \bar{S})}{\phi_i}$, will satisfy*

$$d_i(a_i, \bar{S}) \geq \langle \bar{S}^*(t), \frac{\bar{T}_\ell}{\bar{S}_\ell} t \rangle \tag{50}$$

$$\frac{q_i(a_i, \bar{S})}{\phi_i} \geq \langle \frac{\bar{S}_\ell}{\bar{T}_\ell} t, \bar{S}(t) \rangle. \tag{51}$$

[2]If the derivative, $\dot{\bar{S}}(\bar{T}_\ell)$, is not defined, the left and the right derivatives at $\bar{T}_\ell$ are used in (46).

The proof is straightforward by noting that if $i \notin \Gamma_\ell$, then $\frac{\bar{S}_\ell}{\bar{T}_\ell} t \le b_i(t)$ for $0 \le t \le \bar{T}_\ell$ (see Fig. 4).

The interesting aspect of the bounds in Propositions 1–3 is that they only depend on the upper envelope function of connection i and the universal service curve. In fact, the QoS parameters of each connection can be obtained independent from the rest of the traffic. Therefore, a true decoupling of the traffics is achieved.

5 Vector Representation

In this section, we introduce a vector representation of virtual network partitioning. Define the functions $\bar{O}_\ell(t), \ell = 1, \dots, L$, as

$$\bar{O}_\ell(t) = \begin{cases} 0 & t \le \bar{T}_\ell, \\ \infty & t > \bar{T}_\ell. \end{cases} \tag{52}$$

$\{\bar{O}_\ell(t), \ell = 1, \dots, L\}$ is a set of ordered orthogonal bases [21],

$$\langle \bar{O}_\ell(t), \bar{O}_p(t) \rangle = \begin{cases} 0 & \ell \ge p, \\ \infty & \ell < p. \end{cases} \tag{53}$$

$\bar{O}_\ell(t)$ is, in fact, the transfer function of a delay line with the delay of $\bar{T}_\ell$ seconds. Let us also define $\mathcal{O}_L \triangleq \text{span}\{\bar{O}_1(t), \dots, \bar{O}_L(t)\}$.

A session i belongs to the virtual network Γ_ℓ if and only if the projection of its traffic function, $b_i(t)$, onto $\bar{O}_\ell(t)$ is smaller than the projection of the universal service curve $\bar{S}(t)$ onto the same basis, that is

$$i \in \Gamma_\ell \iff \langle b_i(t), \bar{O}_\ell(t) \rangle \le \langle \bar{S}(t), \bar{O}_\ell(t) \rangle. \tag{54}$$

Due to the duality between the maximum delay and the maximum backlog, it suffices to solely use backlog to define virtual network partitioning.

In (54), $\langle b_i(t), \bar{O}_\ell(t) \rangle$ indicates the maximum normalized traffic that can be generated in $\bar{T}_\ell$ seconds by source i. The inequality (54) denotes that a source i belongs to Γ_ℓ if the maximum normalized traffic generated by that source is smaller than an amount indicated for that subnetwork.

Let a generic point in $\mathcal{O}_L$ be represented by the L-topple $\mathbf{x} = (x_1, \dots, x_L)$ where x_ℓ is the projection onto the basis $\bar{O}_\ell(t)$. Since $\bar{S}(t) \in f_v$, its projection onto $\mathcal{O}_L$ will be in the region

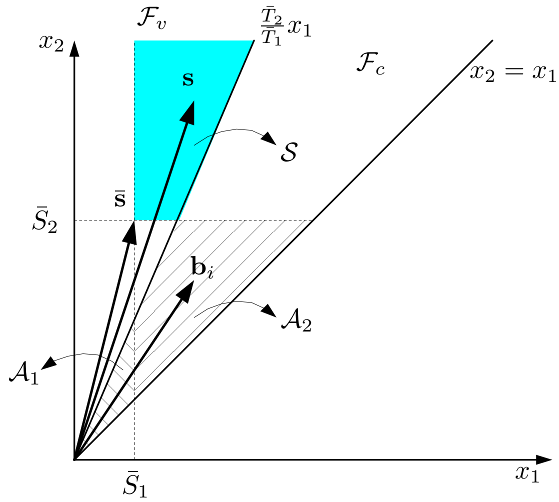
$$\mathcal{F}_v \triangleq \left\{ (x_1, \dots, x_L) \,\middle|\, x_n \ge \frac{\bar{T}_n}{\bar{T}_m} x_m \ge 0, \text{ for } 1 \le m < n \le L \right\}. \tag{55}$$

A similar argument holds for the projection of $S(t; \underline{a}, \Phi)$. Furthermore, since $\bar{S}(t) \le S(t; \underline{a}, \Phi)$, we should have $\langle \bar{S}(t), \bar{O}_\ell(t) \rangle \le \langle S(t; \underline{a}, \Phi), \bar{O}_\ell(t) \rangle, \ell = 1, \dots, L$. Define also

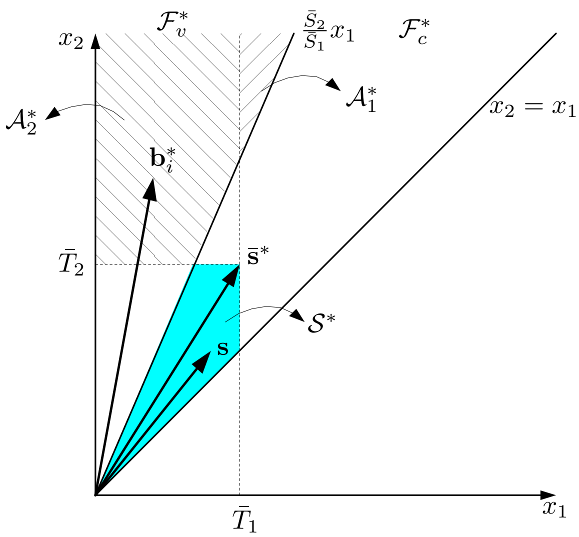
$$\mathcal{F}_c \triangleq \left\{ (x_1, \dots, x_L) \,\middle|\, x_m \le x_n \le \frac{\bar{T}_n}{\bar{T}_m} x_m, \text{ for } 1 \le m < n \le L \right\}. \tag{56}$$

Since the normalized upper envelope process, $b_i(t)$, is concave and non-decreasing, its projection onto $\mathcal{O}_L$, denoted here as $\mathbf{b}_i \triangleq \Big(b_i(\bar{T}_1), \dots, b_i(\bar{T}_L) \Big)$, will be in $\mathcal{F}_c$. We further assume that the projection of the normalized upper envelope process onto $\bar{O}_L(t)$ is smaller than the projection of the universal service curve onto this function. This assumption guarantees that the projection of the normalized upper envelope process is, indeed, in $\mathcal{F}_c \bigcap \{(x_1, \dots, x_L) \,|\, x_L \le \bar{S}_L\}$.

The projection of the normalized upper envelope process, $\mathbf{b}_i$, the universal service curve, $\bar{\mathbf{s}} \triangleq (\bar{S}_1, \dots, \bar{S}_L)$, and the true service curve, $\mathbf{s} \triangleq \Big(S(\bar{T}_1; \underline{a}, \Phi), \dots, S(\bar{T}_L; \underline{a}, \Phi) \Big)$ onto $\mathcal{O}_L$ have been illustrated in Fig. 6-(a) for $L = 2$. To satisfy $S(\bar{T}_\ell; \underline{a}, \Phi) \ge \bar{S}_\ell$ for all $\ell = 1, \dots, L$, the true service curve should be constrained to the shaded region denoted as $\mathcal{S}$ in Fig. 6-(a).



(a)



(b)

Figure 6: (a) The projection of the universal service curve, $\bar{\mathbf{s}}$, the true service curve, $\mathbf{s}$, and the normalized upper envelope process, $\mathbf{b}_i$, onto $\mathcal{O}_2$. (b) The projection of the adjoints of the universal service curve, $\bar{\mathbf{s}}^*$, the true service curve, $\mathbf{s}^*$, and the normalized upper envelope process, $\mathbf{b}_i^*$, onto the span of $\{\delta_1(t), \delta_2(t)\}$.

Based on the observation above, the virtual network partitioning is also simplified. For $\ell = 1, \ldots, L$, define

$$\mathcal{A}_\ell \triangleq \Big\{(x_1, \ldots, x_L) \Big| (x_1, \ldots, x_L) \in \mathcal{F}_c, \text{ and } \bar{S}_{\ell-1} \leq x_{\ell-1}, x_\ell \leq \bar{S}_\ell\Big\}. \tag{57}$$

Then $\mathbf{b}_i \in \bigcup_{1 \leq m \leq \ell} \mathcal{A}_m \iff i \in \Gamma_\ell$. Note that $\mathcal{A}_\ell$ is a closed convex set. In practice, the true service curve is a function of the vector of upper envelope processes $\mathbf{b}_i, i = 1, \ldots, N$. Therefore, by varying the parameters of $\mathbf{b}_i, i = 1, \ldots, N$, we can move $\mathbf{s}$ inside $\mathcal{F}_v$ and specifically in the shaded region. For instance, for GPS scheduler and leaky bucket regulated traffics, the normalized service curve for a connection $i \in \Gamma_\ell$ is a piecewise linear convex function given by

$$\begin{aligned} \frac{1}{\phi_i} S_i(\bar{T}_\ell; \underline{a}, \Phi) &= \frac{\bar{T}_\ell - \sum_{j \in \mathcal{B}^c(\bar{T}_\ell)} (\sigma_j + \rho_j \bar{T}_\ell)}{\sum_{j \in \mathcal{B}(\bar{T}_\ell)} \phi_j} \\ &\leq \frac{\bar{T}_\ell - \sum_{j \in \Gamma_\ell} (\sigma_j + \rho_j \bar{T}_\ell)}{\sum_{j \in \Gamma_L} \phi_j - \sum_{j \in \Gamma_\ell} \phi_j}. \end{aligned} \tag{58}$$

Now for any $j \in \Gamma_\ell$, if we increase σ_j or ρ_j, then $S_k \triangleq \frac{1}{\phi_i} S_i(\bar{T}_k; \underline{a}, \Phi)$ will decrease for all $k \geq \ell$. Similarly, if we increase ϕ_j, then S_k will decrease for all $k < \ell$. Therefore, by varying the paramters of an appropriately selected set of $\mathbf{b}_i$, we can select the true service curve inside a region where $S_k > \bar{S}_k$ for all $k = 1, \ldots, L$.

A similar approach can be used in the dual space. Define for $\ell = 1, \ldots, L$,

$$\delta_\ell(t) = \begin{cases} 0 & t \leq \bar{S}_\ell \\ \infty & t > \bar{S}_\ell \end{cases} \tag{59}$$

$\{\delta_1(t), \ldots, \delta_L(t)\}$ forms a set of ordered orthogonal bases with $\langle \bar{S}^*(t), \delta_\ell(t) \rangle = \bar{T}_\ell$, $\ell = 1, \ldots, L$. The bases $\{\delta_1(t), \ldots, \delta_L(t)\}$ can be used to construct a projection subspace (as shown in Fig. 6-(b) for $L = 2$). Let $\bar{\mathbf{s}}^* \triangleq (\bar{T}_1, \ldots, \bar{T}_L)$. Note $\bar{\mathbf{s}}^* \in \mathcal{F}_c^*$, where

$$\mathcal{F}_c^* \triangleq \Big\{(x_1, \ldots, x_L) \Big| x_m \leq x_n \leq \frac{\bar{S}_n}{\bar{S}_m} x_m, \text{ for } 1 \leq m < n \leq L\Big\}. \tag{60}$$

Similarly, $\mathbf{b}_i^* \in \mathcal{F}_v^*$, where

$$\mathcal{F}_v^* \triangleq \Big\{(x_1, \ldots, x_L) \Big| x_n \geq \frac{\bar{S}_n}{\bar{S}_m} x_m \geq 0, \text{ for } 1 \leq m < n \leq L\Big\}. \tag{61}$$

The hierarchical network partitioning can be used by assigning input traffics to the regions,

$$\mathcal{A}_\ell^* \triangleq \Big\{(x_1, \ldots, x_L) \Big| (x_1, \ldots, x_L) \in \mathcal{F}_v^*, \text{ and } \bar{T}_{\ell-1} \geq x_{\ell-1}, x_\ell \geq \bar{T}_\ell\Big\}, \tag{62}$$

for $\ell = 1, \ldots, L$. Here, the call admission controller should be designed such as to guarantee that the maximum distance between the elements of $\bar{\mathbf{s}}^*$ and $\mathbf{b}_i^*$ be smaller than D_i, the maximum prescribed delay.

We may also include $\bar{O}_0(t)$ to the analysis above to consider the traffic upper envelopes that have a jump at the origin.

6 Call Admission Control

A call set-up is initiated by a source requesting a connection through the network with some specified QoS parameters. A call admission controller is then used to allocate appropriate bandwidth and buffer-size to the requested call. The call is accepted if (i) enough bandwidth and buffers are available (ii) the requested QoS can be guaranteed (iii) admission of the new call will not drive the QoS for the ongoing sessions below their prescribed thresholds. In fact, the call admission controller should verify that (28)-(30) are satisfied for all i, once the new call is accepted. In this section, we propose a call admission controller for a node which has been decomposed into L subnetworks with a FIFO queue allocated to each subnetwork.

The call admission controller might optimize a suitable cost function by shaping the source traffic, $A_i(t)$—provided that it is allowed by the user—and selecting the scheduler parameter, ϕ_i. Upon a new call request, the network manager solves

$$\begin{aligned} \max \quad & U(i,\ell) && (63)\\ \text{s.t.} \quad & b_i(\bar{T}_\ell) \leq \bar{S}_\ell, && (64)\\ & S(t;\underline{a},\Phi) \geq \bar{S}(t) && (65)\\ & \langle b_i(t), \bar{S}(t)\rangle \leq \frac{Q_i}{\phi_i} && (66)\\ & \langle \bar{S}(t)^*, b_i^*(t)\rangle \leq D_i && (67) \end{aligned}$$

where $U(i,\ell)$ is an appropriate utility function [25] selected to maximize the network revenue, which in general is a function of the requested call i and the subnetwork ℓ to which the call is assigned to. For instance, the network provider might be willing to accept a call into the largest subnetwork (least demanding) for which the constraints (64)-(67) hold. In such a case, the utility function will be the maximum index of all subnetworks that satisfy (64)-(67).

In a core network, traffic flows are usually aggregated into certain hierarchical classes. In such a network, per-user quality of service guarantee is not prescribed. For instance, the *differentiated services* (DiffServ) standard [17] of the Internet Engineering Task Force (IETF) provides a framework for the assignment of an aggregate of flows into several classes of service called the per-hop behaviour (PHB). Here, a direct application of a GPS scheduling for a single traffic flow may not be feasible. For such cases, one can assume that the traffics of all connections for a certain class share a common queue with a first-in-first-out (FIFO) discipline. The scheduler transmits the traffic of all queues in a pre-selected GPS framework.

Assume there exist L queues represented by Π_ℓ, $\ell = 1,\ldots,L$. The traffic in queue Π_ℓ is served by a GPS scheduler with parameter

$$\Phi_\ell \triangleq |\Pi_\ell|\phi_\ell \tag{68}$$

where $|\Pi_\ell|$ is the total number of connections assigned to queue Π_ℓ (all traffics in Π_ℓ have the same ϕ_ℓ). The GPS parameters ϕ_ℓ are ordered as $\phi_1 > \ldots > \phi_L$. Hence, the traffic in Π_1 is in the highest hierarchy and receives the best relative service in terms of the maximum delay and the maximum backlog. The queues are allocated to virtual networks in a nested structure.

We represent the aggregated traffic in the ℓth queue by $\hat{a}_\ell(t) \triangleq \sum_{i\in\Pi_\ell} a_i(t)$. Assume that the service given to the aggregated traffic $\hat{a}_\ell(t)$ is controlled by the fairness coefficient Φ_ℓ. We assume that the GPS parameter of each queue is adapted to the number of connections assigned to that queue. Since $\frac{\hat{a}_\ell(\bar{T}_\ell)}{\Phi_\ell} = \frac{1}{|\Pi_\ell|}\sum_{i\in\Gamma_\ell}\frac{a_i(\bar{T}_\ell)}{\phi_\ell} \leq \bar{S}_\ell$, we can prove the following proposition.

Proposition 4 *The aggregated traffic, $\hat{a}_\ell(t)$, belongs to the virtual network Γ_ℓ.*

Proposition 4 illustrates an interesting property of the GPS scheduler. It shows that the relative fairness among all connections will remain constant if the GPS parameters are weighted with the number of connections in each queue. In other words, admission of a new connection into a queue will preserve the relative performance of all connections. This can also be quantified in terms of the guaranteed QoS parameters.

Let $\hat{d}_\ell$ and $\hat{q}_\ell$ be, respectively, the maximum delay and the maximum backlog of the aggregated traffic $\hat{a}_\ell(t)$. Therefore, we have

$$\begin{aligned} \hat{d}_\ell &\leq \langle \bar{S}^*(t), \hat{a}_\ell^*(\Phi_\ell t)\rangle, && (69)\\ \hat{q}_\ell &\leq \langle \hat{a}_\ell(t), \bar{S}(t)\Phi_\ell\rangle. && (70) \end{aligned}$$

In the following theorem, we show that the maximum delay and the maximum backlog for a subnetwork are bounded by the maximum delay and the maximum backlog of a session that represents the worst-case behavior inside that group.

Theorem 3 *Using a weighted GPS scheduler, we will have, for all subnetworks $\ell = 1, \dots, L$,*

$$\langle \bar{S}^*(t), \hat{a}^*_\ell(\Phi_\ell t)\rangle \leq \max_{i\in\Gamma_\ell}\langle \bar{S}^*(t), a^*_i(\phi_\ell t)\rangle \tag{71}$$

$$\langle \hat{a}_\ell(t), \bar{S}(t)\Phi_\ell\rangle \leq \max_{i\in\Gamma_\ell}\langle a_i(t), \phi_\ell \bar{S}(t)\rangle \tag{72}$$

Proof: Replace all the connections in Γ_ℓ with the one for which $\langle a_i(t), \phi_\ell \bar{S}(t)\rangle$ is maximized. Similar argument holds for the maximum delay. □

Theorem 3 can be used to propose a call admission controller that satisfies for all $\ell \in \{1, \dots, L\}$,

$$\langle \bar{S}^*(t), \hat{a}^*_\ell(\Phi_\ell t)\rangle \leq \min_{i\in\Gamma_\ell} D_i \tag{73}$$

$$\langle \hat{a}_\ell(t), \bar{S}(t)\Phi_\ell\rangle \leq \min_{i\in\Gamma_\ell} Q_i, \tag{74}$$

where D_i and Q_i are the requested QoS parameters of session i. Thus, in general, upon initiation of a new connection request with the QoS parameters D_i and Q_i, the call will be accepted into $\Gamma_{\bar{\ell}}$ if $\bar{\ell}$ is the solution of

$$\max \quad \ell \tag{75}$$

$$\text{s.t.} \quad b_i(\bar{T}_\ell) \leq \bar{S}_\ell, \tag{76}$$

$$S(t; \underline{a}, \Phi) \geq \bar{S}(t) \tag{77}$$

$$\langle \bar{S}^*(t), \hat{a}^*_\ell(\Phi_\ell t)\rangle \leq \min_{i\in\Gamma_\ell} D_i, \tag{78}$$

$$\langle \hat{a}_\ell(t), \bar{S}(t)\Phi_\ell\rangle \leq \min_{i\in\Gamma_\ell} Q_i, \tag{79}$$

$$\textstyle\sum_{\ell=1}^{L} \Phi_\ell \leq 1 \tag{80}$$

where $S(t; \underline{a}, \Phi)$ is defined as

$$S(t; \underline{a}, \Phi) = \frac{Ct - \sum_{\ell\in\mathcal{G}(t)} \hat{a}_\ell(t)}{\sum_{\ell\in\mathcal{G}^c(t)} \Phi_\ell} \tag{81}$$

with $\mathcal{G}(t)$ being the set of all subnetworks satisfying

$$\mathcal{G}(t) \triangleq \left\{ \ell \,\middle|\, \frac{\hat{a}_\ell(t)}{\Phi_\ell} \leq S(t; \underline{a}, \Phi) \right\} \tag{82}$$

and $\mathcal{G}^c(t)$ the complementary set of $\mathcal{G}(t)$.

7 Assignment of Universal Service Curve

In this section, we will use a numerical example to devise a procedure to designate an appropriate universal service curve. The universal service curve is selected at network setup. The selection is an off-line procedure and might be computationally involved. Here, we use an exhaustive search approach. A numerical example is used to illustrate the procedure.

Let the input traffic to a node be composed of two types of leaky bucket regulated traffics with the normalized parameters ($\sigma_1 = 0.1, \rho_1 = 0.08, \phi_1 = 0.01$) and ($\sigma_2 = 0.7, \rho_2 = 0.1, \phi_2 = 0.03$). For the sake of simplicity, we consider two traffic types. Generalization to larger number of traffic types is straightforward.

For given ρ_1, the maximum number of Type 1 traffics that can be supported by the network in the absence of Type 2 traffic is $N_1 = 1/\rho_1$ (assuming that $\phi_1 \leq \rho_1$). Similarly, in the absence of Type 1 traffic, a maximum of $N_2 = 1/\rho_2$ can be supported by the network ($\phi_2 \leq \rho_2$). Now, for all combinations (n_1, n_2), $0 \leq n_1 \leq N_1, 0 \leq n_2 \leq N_2$, we find the true service curve. Note that for $n_1\rho_1 + n_2\rho_2 > 1$, the system is unstable and the maximum delay is infinity.

Table 1: The maximum delay as a function of the number of Type 1 and Type 2 users.

(d_1, d_2)		n_2						
		0	1	2	3	4	5	6
n_1	0	(—,—)	(— ,0.7)	(— ,1.4)	(—, 2.1)	(—,2.8)	(— ,3.5)	(— ,4.2)
	1	(0.1,—)	(0.4,0.9)	(0.7,1.6)	(1.0,2.3)	(1.5,3.0)	(3.0,3.7)	(5.4,4.4)
	2	(0.2,—)	(0.5,1.0)	(0.8,1.9)	(1.1,2.6)	(1.9,3.3)	(3.7,4.0)	(6.5,4.7)
	3	(0.3,—)	(0.6,1.3)	(0.9,2.1)	(1.2,2.8)	(2.4,3.5)	(4.5,4.2)	(7.9,4.9)
	4	(0.4,—)	(0.7,1.6)	(1.0,2.3)	(1.5,3.0)	(3.0,3.7)	(5.4,4.4)	(9.6,5.1)
	5	(0.5,—)	(0.8,1.9)	(1.1,2.6)	(1.9,3.3)	(3.7,4.0)	(6.5,4.7)	(11.8,5.4)
	6	(0.6,—)	(0.9,2.1)	(1.2,2.8)	(2.4,3.5)	(4.5,4.2)	(7.9,5.0)	(∞, ∞)
	7	(0.7,—)	(1.0,2.3)	(1.5,3.0)	(3.0,3.7)	(5.4,4.4)	(∞, ∞)	(∞, ∞)
	8	(0.8,—)	(1.1,2.6)	(1.9,3.3)	(3.7,4.0)	(∞, ∞)	(∞, ∞)	(∞, ∞)
	9	(0.9,—)	(1.2,2.8)	(2.4,3.5)	(∞, ∞)	(∞, ∞)	(∞, ∞)	(∞, ∞)
	10	(1.0,—)	(1.5,3.0)	(3.0,3.7)	(∞, ∞)	(∞, ∞)	(∞, ∞)	(∞, ∞)
	11	(1.1,—)	(1.9,3.3)	(∞, ∞)	(∞, ∞)	(∞, ∞)	(∞, ∞)	(∞, ∞)
	12	(1.2,—)	(∞, ∞)	(∞, ∞)	(∞, ∞)	(∞, ∞)	(∞, ∞)	(∞, ∞)

We use these service curves to compute the maximum delay for both traffic types; for simplicity we assume that the traffics are not backlog-constrained—similar approaches can be used for backlog-constrained traffics. Any combination of (n_1, n_2) for which the maximum delay for each traffic type is smaller than the maximum allowable delay can be considered as an appropriate service curve.

In the present example, in the absence of Type 2 traffic, a total of 10 Type 1 sources can be accepted into the network. Similarly, in the absence of Type 1 traffic, a total of 12 Type 2 traffics can be accepted. The maximum delay for this example has been illustrated in Table 7 for $0 \leq n_1 \leq 12$ and $0 \leq n_2 \leq 6$. The mnth entry of the table represents the maximum delay incurred if m Type 1 users and n Type 2 users are admitted into the network. As expected, for the values of m and n with $n_1\rho_1 + n_2\rho_2 > 1$ the maximum delay is infinity. In these cases, the service curve does not intersect the load line.

The maximum delays for both traffics are illustrated in Fig. 7. Now assume that the maximum tolerable delays for the two traffic types are given by $D_1 = 1.0$ and $D_2 = 2.5$. These values are represented by horizontal lines in Fig. 7. A service curve, corresponding to a point located below the dashed lines, can be considered as a candidate for the universal service curve. It is obvious that the points closer to the dashed lines will result in the service curves that are smaller and therefore can accept more calls into the network.

Fig. 8 illustrates the lattice of all combinations of $0 \leq n_1 \leq N_1$, $0 \leq n_2 \leq N_2$, and $n_1\rho_1 + n_2\rho_2 \leq 1$. Note that all points which $n_1\rho_1 + n_2\rho_2 > 1$ will provide unbounded d_1 and d_2 and therefore are not shown in Fig. 8. In the figure, all points corresponding to $d_1 \leq D_1$ and $d_2 \leq D_2$ are illustrated by dark circles. Note that if (n_1, n_2) is an acceptable point—that is $d_1 \leq D_1$ and $d_2 \leq D_2$—all points (n'_1, n'_2), $n'_1 \leq n_1$, $n'_2 \leq n_2$, will also be acceptable. Furthermore, the service curve associated to a combination (n'_1, n'_2) with $n'_1 < n_1$ or $n'_2 < n_2$, will be larger than the service curve associated to (n_1, n_2).

It is obvious that the service curve of any black circle in Fig. 8 can be used as the universal service curve. It should be noted however that an appropriate universal service curve will be the one that is a lower bound to the service curve of all dark circles in Fig. 8. Using the observation in the previous paragraph, one might prefer to choose the service curve of (n_1, n_2) instead of the ones for which $0 \leq n'_1 < n_1$ and $0 \leq n'_2 < n_2$. In this example, three points can be used as the candidates for the universal service curve: $(n_1, n_2) = (1, 3)$, $(n_1, n_2) = (4, 2)$, and $(n_1, n_2) = (7, 1)$. The three service curves are illustrated in Fig. 9. The smallest service curve corresponds to $(n_1, n_2) = (7, 1)$. Therefore, the universal service curve can be selected as the service curve of $(n_1, n_2) = (7, 1)$.

In this example, the service curves of all candidate points can be ordered monotonically. This results in one of the service curves being considered as the lower bound to the other service curves. In some cases, this requirement may not be satisfied and the service curves of the candidate points cannot be ordered over the

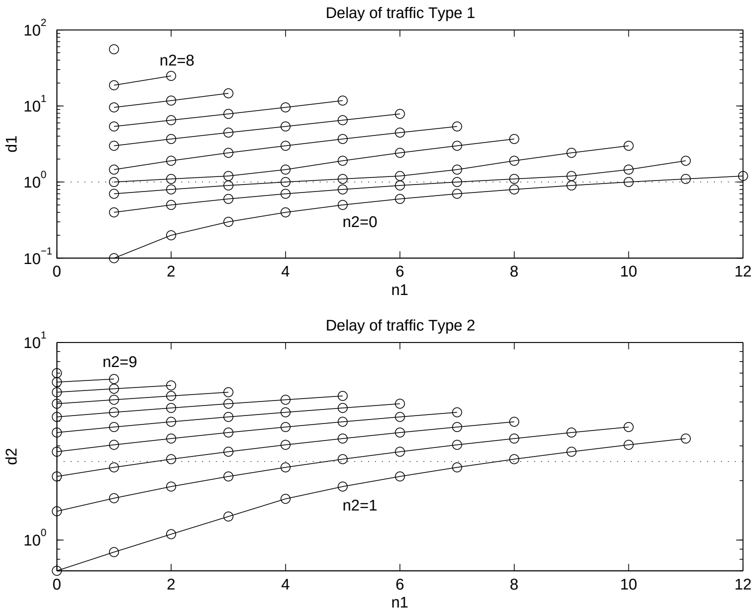


Figure 7: The delay lines for all combinations of $0 \le n_1 \le N_1$ and $0 \le n_2 \le N_2$.

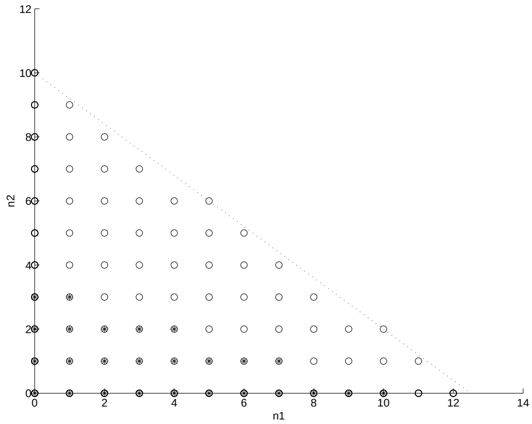


Figure 8: The lattice points for all combinations of $0 \le n_1 \le N_1$ and $0 \le n_2 \le N_2$. The black circles correspond to the pairs (n_1, n_2) for which $d_1 \le D_1$ and $d_2 \le D_2$.

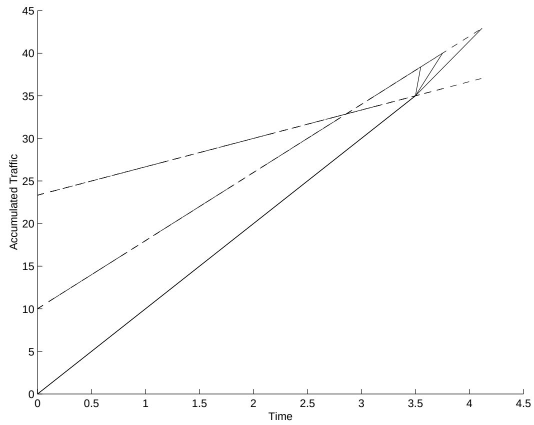


Figure 9: The service curves associated to $(n_1, n_2) = (1, 3)$, $(n_1, n_2) = (4, 2)$, and $(n_1, n_2) = (7, 1)$. The smallest service curve corresponds to $(n_1, n_2) = (7, 1)$ and the largest service curve corresponds to $(n_1, n_2) = (1, 3)$.

whole temporal extent of their support. In such cases, two or more candidate service curves intersect and the assignment of one of the candidates as the universal service curve might result in the rejection of the others as eligible service curves—a true service curve cannot cross the minimum bound set by the universal service curve. A remedy would be to use the convex hull of all candidate service curves as the universal service curve. It is important to note that in such cases, the maximum delay and/or the maximum backlog might grow slightly beyond the recommended threshold. Since in most cases the maximum delay and maximum backlog are conservatively selected, a small increase of the maximum threshold might not be destructive.

Selecting the universal service curve also relates to the issue of pricing. The network manager can select the universal curve so that the total network benefit is increased. For instance, the service curve can be selected so as to favor some traffic types that produce a higher revenue.

Another technique to select the universal service curve is to increase the total probability of call admission. In such an approach, a probability mass function is assigned to all black dots in Fig. 8. Then, the service curve is selected so as to maximize the aggregated probability.

8 Conclusion

This paper introduces a methodology for partitioning a single-node network with a GPS scheduler into several nested subnetworks. The approach is based on the worst-case service curve provisioning. In general, the service curve is a function of the scheduler, the parameters of the regulator of input flow, and the volume of the crossing traffic. We have shown that, assuming a concave subadditive upper envelope process for the input traffic and using an elaborate call admission controller, the true service curve will always be lower bounded by a deterministic universal service curve. This is achieved by adapting a suitable regulator to the input traffic and allocating appropriate scheduler parameters. The universal service curve is then used to quantify the performance of the network in terms of the maximum delay and the maximum backlog.

The universal service curve is independent of the traffic fluctuations and can also be used as a vehicle to decompose the input flows into several nested classes. Assignment of an input traffic into a certain class is based on the proximity of its normalized upper envelope process to the universal service curve.

We have constructed our technique based on the min-plus algebra. An adjoint operator has been defined and it has been shown that the maximum delay can be represented by the projection of the adjoint of the service curve onto the adjoint of the traffic upper envelope process. This observation establishes a duality between the maximum backlog and the maximum delay. The duality can be used to prescribe similar approaches to the management of the maximum delay and the maximum backlog in a network.

We have also shown that the method can be applied to aggregate the flows in a backbone network into a hierarchical structure of subnetworks. In a backbone network, the per-user QoS guarantee is not advocated. Here, all connections, belonging to a similar category, share a common — usually FIFO disciplined — queue. We have shown that if the GPS parameters are weighted with the number of connections in each queue, the relative fairness among all connections will be preserved. In other words, admission of a new connection into a queue will maintain the relative performance of all connections intact.

A Proof of Theorem 1

Let the sessions be ordered such that $T_1 \leq T_2 \leq \ldots \leq T_N$, and define $T_0 = 0$. In $[T_{k-1}, T_k], k = 1, \ldots, i$, the service curve is equal to

$$S_i(t; \underline{a}, \Phi) = \phi_i \frac{Ct - \sum_{j=1}^{k-1} a_j(t)}{\sum_{j=k}^{N} \phi_j}. \tag{83}$$

(*i*) Use the fact that if $a_j(t)$ is concave, $-a_j(t)$ will be convex, and the summation of concave functions is also concave, to conclude that $S_i(t; \underline{a}, \Phi)$ is convex for $t \in (T_{k-1}, T_k)$, $k = 1, \ldots, i$. Therefore, $S_i(t; \underline{a}, \Phi)$ is a piecewise convex function. Furthermore, $S_i(t; \underline{a}, \Phi)$ is continuous at T_k. It remains to prove that the slope of service curve at T_k^- is smaller than its slope at T_k^+.

Denote by $\dot{S}_i(t; \underline{a}, \Phi)$ the derivative of $S_i(t; \underline{a}, \Phi)$. Since the kth connection depletes at T_k, we should have

$$\frac{\dot{a}_k(T_k^+)}{\phi_k} < \dot{S}_i(T_k^+; \underline{a}, \Phi). \tag{84}$$

Take the left and the right derivatives of $S_i(t; \underline{a}, \Phi)$ at $T_k, k = 1, \ldots, i$,

$$\dot{S}_i(T_k^-; \underline{a}, \Phi) = \frac{\phi_i}{\sum_{j=k}^{N} \phi_j} \Big(C - \sum_{j=1}^{k-1} \dot{a}_j(T_k^-) \Big) \tag{85}$$

$$\dot{S}_i(T_k^+; \underline{a}, \Phi) = \frac{\phi_i}{\sum_{j=k+1}^{N} \phi_j} \Big(C - \sum_{j=1}^{k} \dot{a}_j(T_k^+) \Big) \tag{86}$$

where $\dot{a}_j(t)$ is the derivative of $a_j(t)$. Note also that since $a_j(t), j = 1, \ldots, N$, is convex, we have

$$\sum_{j=1}^{k-1} \dot{a}_j(T_k^-) \geq \sum_{j=1}^{k-1} \dot{a}_j(T_k^+) \tag{87}$$

Now use (84)-(87) and a little algebra to get

$$\dot{S}_i(T_1^-; \underline{a}, \Phi) \leq \dot{S}_i(T_1^+; \underline{a}, \Phi). \tag{88}$$

(*ii*) Use (83) and the fact that if $a_i(t)$ is u.s.c., then $-a_i(t)$ will be l.s.c., to prove that $S_i(t; \underline{a}, \Phi)$ is l.s.c.

(*iii*) Use (83) and the fact that if $a_i(t)$ is sub-additive, then $-a_i(t)$ will be super-additive, to get the result.

B Proof of Theorem 2

Sufficiency:

Assume there exists a set of time indices $0 = t_0 < t_1 < t_2 < \dots$ such that

$$S(t_j) + \dot{S}(t_j^+)(t_{j+1} - t_j) > \bar{S}(t_{j+1}), \tag{89}$$

holds for all $j = 0, 1, 2, \dots$, and $\lim_{j\to\infty} t_j \geq T$. First note that the left hand-side of (89) is a tangent line to $S(t)$ at t_j^+ and remember that $S(t)$ is convex and l.s.c. for all $t > 0$. Hence,

$$S(t_{j+1}) \geq S(t_j) + \dot{S}(t_j^+)(t_{j+1} - t_j). \tag{90}$$

Take $t_0 = 0$. We have $S(0) = \bar{S}(0) = 0$. Use the convexity and the lower semi-continuity of $S(t)$ and $\bar{S}(t)$ along with (89) and (90) for t_0 to get $S(t) \geq \dot{S}(0^+)t > \bar{S}(t)$, for $t \in [0, t_1]$. Now assume $S(t_j) > \bar{S}(t_j)$. The convexity and the lower semi-continuity of $S(t)$ and $\bar{S}(t)$ and (89) give $S(t) \geq S(t_j) + \dot{S}(t_j^+)(t - t_j) > \bar{S}(t)$, for all $t \in [t_j, t_{j+1}]$.

Necessity:
Assume $\bar{S}(t) < S(t), 0 < t < T$ and $\dot{\bar{S}}(0^+) < \dot{S}(0^+)$, with $\bar{S}(t)$ and $S(t)$ being positive, increasing, l.s.c., convex functions. Take $t_0 = 0$. Since $\dot{\bar{S}}(0^+) < \dot{S}(0^+)$, there exists a neighborhood of the origin, $(0, \epsilon)$ for which $\dot{S}(0^+)t > \bar{S}(t)$ for all $t \in (0, \epsilon)$. Take $t_1 \triangleq \sup\{\epsilon \,|\, \dot{S}(0^+)t > \bar{S}(t), t \in (0, \epsilon)\}$. Since $(0, \epsilon) \neq \emptyset$, we have that $t_1 > t_0 = 0$.

Now proceed with the same procedure and define $t_2 \triangleq \sup\{\epsilon \,|\, S(t_1) + \bar{S}(t_1^+)t > \bar{S}(t), t \in (t_1, \epsilon)\}$. Note again that $t_2 > t_1$. The procedure can be continued to produce the subsequent points.

Since $t_i < t_j$ for $i < j$, one can conclude that the sequence $\{t_n\}$ is convergent (possibly to infinity). Let $t_\infty \triangleq \lim_{n\to\infty} t_n$. With the procedure that we have generated $\{t_n\}$, if t_∞ is bounded, it should be a point for which $S(t_\infty) = \bar{S}(t_\infty)$, otherwise the process will not end at t_∞. From this observation, and the assumption that $S(t) > \bar{S}(t), 0 < t \leq T$, one can conclude that $t_\infty > T$.

References

[1] R. L. Cruz, "A calculus for network delay, part I: network elements in isolation," *IEEE Trans. Inform. Theory*, vol. 37, pp. 114–131, Jan. 1991.

[2] R. L. Cruz, "A calculus for network delay, part II: network analysis," *IEEE Trans. Inform. Theory*, vol. 37, pp. 132–141, Jan. 1991.

[3] A. K. Parekh and R. G. Gallager, "A generalized processor sharing approach to flow control in integrated services network: the single node case," *IEEE/ACM Trans. Networking*, vol. 1, pp. 334–357, June 1993.

[4] C. S. Chang, "Stability, queue length and delay of deterministic and stochastic queueing networks," *IEEE Trans. Automatic Control*, vol. 39, pp. 913–931, 1994.

[5] R. L. Cruz, "Quality of service gurantees in virtual circuit switched networks," *IEEE J., Select., Areas Commun.*, vol. 13, pp. 1048–1057, August 1995.

[6] S. Low and P. Varaiya, "A simple theory of traffic resource allocation in ATM," in *Proceeding IEEE Globecom*, pp. 1633–1637, 1991.

[7] S. Valaee, "A recursive estimator of worst-case burstiness," *IEEE/ACM Trans. Networking*, vol. 9, pp. 211–222, April 2001.

[8] J. Y. Le Boudec, "Application of network calculus to guaranteed service networks," *IEEE Trans. Inform. Theory*, vol. 44, pp. 1087–1096, May 1998.

[9] C. S. Chang, "On deterministic traffic regulation and service guarantees: a systematic approach by filtering," *IEEE Trans. Inform. Theory*, vol. 44, pp. 1097–1110, May 1998.

[10] C. S. Chang, *Performance Guarantees in Communication Networks.* Springer Verlag, 2000.

[11] J. Y. Le Boudec and P. Thiran, *Network Calculus.* Springer Verlag, 2001. Also available at: http://ica1www.epfl.ch/PS_files/NetCal.htm.

[12] F. Baccelli, G. Cohen, G. J. Olster, and J. P. Quadrat, *Synchronization and Linearity, An Algebra for Discrete Event Systems.* New York: Wiley, 1992.

[13] H. Sariowan, *A service curve approach to performance guarantees in integrated-services networks.* PhD thesis, University of California, San Diego, 1996.

[14] H. Sariowan, R. L. Cruz, and G. C. Polyzos, "SCED: A generalized scheduling policy for guaranteed quality-of-service," *IEEE/ACM Trans. Networking*, vol. 7, pp. 669–684, October 1999.

[15] S. Valaee and B. Li, "Distributed call admission control for ad hoc networks," *in Proceedings of IEEE Vehicular Technology Conference*, vol. 2, pp. 1244 –1248, Sept. 2002.

[16] I. Stoica, H. Zhang, and T. S. E. Ng, "A hierarchical fair service curve algorithm for link-sharing, real-time and priority services," in *SIGCOMM*, (Cannes, France), pp. 249–262, 1997.

[17] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An architecture for differentiated services," *Internet Engineering Task Force, RFC 2475*, 1998.

[18] C. Dovrolis and P. Ramanathan, "A case for relative differentiated services and the proportional differentiotion model," *IEEE Network*, vol. 13, pp. 26–34, Sept./Oct. 1999.

[19] A. K. Parekh and R. G. Gallager, "A generalized processor sharing approach to flow control in integrated services network: the multiple node case," *IEEE/ACM Trans. Networking*, vol. 2, pp. 137–150, April 1994.

[20] G. Cohen, P. Moller, J. P. Quadrat, and M. Viot, "Algebraic tools for the performance evaluation of discrete event systems," *Proceedings of IEEE*, vol. 77, pp. 39–58, Jan. 1989.

[21] C. S. Chang, "Deterministic traffic specification via projections under the min-plus algebra," in *Proceeding IEEE INFOCOM*, pp. 43–50, 1999.

[22] ATM Forum, *Traffic management specification.* Version 4.0, April 1996.

[23] R. Braden, D. Clark, and S. Shenker, *Integrated services in the internet architecture: An overview.* IETF RFC 1633, July 1994.

[24] S. Valaee, "Virtual subnetwork decomposition of leaky bucket regulated traffics in a generalized processor sharing scheduler," *preprint.*

[25] F. P. Kelly, "Charging and rate control for elastic traffic," *European Transactions on Telecommunications*, vol. 8, pp. 33–37, 1997. A (corrected) version is available at http://www.statslab.cam.ac.uk/~frank/elastic.html.