Call Admission Control Schemes under the Generalized Processor Sharing Scheduling

Z. ZHANG, Z. LIU, J. KUROSE
and D. TOWSLEY

CMPSCI Technical Report 95-52

June 6, 1995

Call Admission Control Schemes under the Generalized Processor Sharing Scheduling*

Zhi-Li Zhang† Zhen Liu‡ Jim Kurose† Don Towsley†

†Department of Computer Science
University of Massachusetts
Amherst, MA 01003, USA
{zhzhang,kurose,towsley}@cs.umass.edu

‡INRIA Centre Sofia Antipolis
2004 Route des Lucioles
06560 Valboune, France
Zhen.Liu@sophia.inria.fr

June 6, 1995

Abstract

Provision of *Quality-of-Service* (QoS) guarantees is an important and challenging problem in the design of integrated-services packet networks. Call admission control is an integral part of the challenge and is closely related to other aspects of networks such as service models, scheduling disciplines, traffic characterization and QoS specification. In this paper we provide a *theoretical framework* within which call admission control schemes with multiple statistical QoS guarantees can be constructed for the Generalized Processor Sharing (GPS) scheduling discipline (also known as Weighted Fair Queueing). Using this framework, we present several admission control schemes for both session-based and class-based service models. The theoretical framework is based on recent results in statistical analyses of the GPS scheduling discipline and the theory of effective bandwidths. Both optimal schemes and suboptimal schemes requiring less computational effort are studied under these service models. The QoS metric considered is buffer loss probability.

1 Introduction

Provision of *Quality-of-Service* (QoS) guarantees is an important and challenging issue in the design of integrated-services packet networks. Call admission control is an integral part of the problem. Clearly, without call admission control, providing QoS guarantees will be impossible. The task of call admission control can be most easily illustrated by considering the following question:

> Given a new call/session that arrives to a network, can it be accepted by the network at its requested QoS, without violating existing QoS guarantees made to on-going calls?

*This work was supported in part by the NSF under grants NCR-9116183 and CCR-9119922. Part of the work was done while the first author was visting INRIA Centre Sofia Antipolis, France.

This seemingly simple question turns out to be very complicated, as the issue of call admission control is closely related to other aspects of a network, such as service models, scheduling disciplines, traffic characterization and QoS specification (see [Tow93] for a survey of these issues). Call admission control with statistical QoS guarantees [FV90] is a particularly important and challenging problem. One of the most important challenges is that of providing call admission control for a heterogeneous mixture of applications which have differing QoS requirements.

In this paper we consider the call admission control issue for a network using Generalized Processor Sharing (GPS) scheduling at its switches and supporting multiple statistical QoS guarantees. We identify several service models that are likely to be provided by future integrated services packet networks, and propose corresponding call admission schemes for these service models. Included are both optimal schemes and suboptimal schemes requiring less computational effort. The theoretical foundation for the proposed schemes are recent results in statistical analyses of GPS scheduling [YaSi94, ZTK94] and the theory of effective bandwidths (see, *e.g.*, [GAN91, GH91, Kel91, EM92a, KWC93, Whi93, Cha94, GWh94, LNT94]). The statistical QoS metric considered is buffer loss probability.

We focus on GPS (also known as *Weighted Fair Queueing*) [DKS89, PG93, PG94, Pare92] because it provides controlled sharing of bandwidth and isolation among sessions (or classes for that matter). In [CSZ92, SCZ93], GPS is recommended as a scheduling discipline where there are several different service classes which must be supported. [CSZ92, SCZ93] argue that perhaps the most important feature of GPS is its ability to isolate various service classes while, at the same time, allowing bandwidth sharing among classes. In addition, there is a rich literature analyzing GPS in a variety of settings. In [PG93, PG94, Pare92], per-session bounds on the worst-case backlog and delay are derived for both a single GPS server in isolation and a network of GPS servers under a deterministic setting. These form a basis for a heuristic call admission control algorithm proposed in [JSZC92] for the so-called *predicative service* [CSZ92]. GPS has also been studied in the stochastic setting, where per-session bounds on backlog and delay tail distributions can be derived [YaSi94, ZTK94]. These results make it possible to provide statistical QoS guarantees to applications with differing QoS requirements using GPS scheduling.

We use the theory of effective bandwidths because it provides the opportunity to place call admission control with multiple QoS requirements in a *formal and rigorous* framework. The theory of effective bandwidths has emerged recently as an elegant and promising approach to the problem of call admission control (see, e.g., [GAN91, GH91, Kel91, EM92a, KWC93, Whi93, Cha94, GWh94]). The theory has been developed in the context of a single network switch or server with a finite capacity buffer shared by many sessions, where the QoS metric in question is buffer loss probability. Under this theory, a simple *asymptotically optimal* call admission control scheme exists for a network carrying traffic with a common buffer loss probability. However, it is not sufficient for providing QoS guarantees in an integrated services packet network where applications with quite different QoS requirements must co-exist. One of the contributions of this paper is to remedy this problem.

For simplicity of exposition, the discussion will mostly focus on a single node case. Some of the schemes proposed can be extended to the end-to-end call admission control case in a fairly straightforward manner. The framework of our study is clearly theoretical. As far as the authors are aware of, this is the first formal approach to the call admission control problem with multiple statistical QoS guarantees and with various network service models accounted for. Even though many practical issues require resolution before these schemes can be applied in practice, we believe our study provides a helpful guideline under which these issues can be investigated. For example, some of our schemes present a theoretical basis for addressing the call admission control issues for the service models (*e.g.*, the predicative service model) proposed in [CSZ92, SCZ93].

As related work, besides GPS, the priority service discipline has also been proposed to provide multiple QoS guarantees for several traffic or service classes (see, *e.g* [CSZ92]). This service discipline has been studied extensively

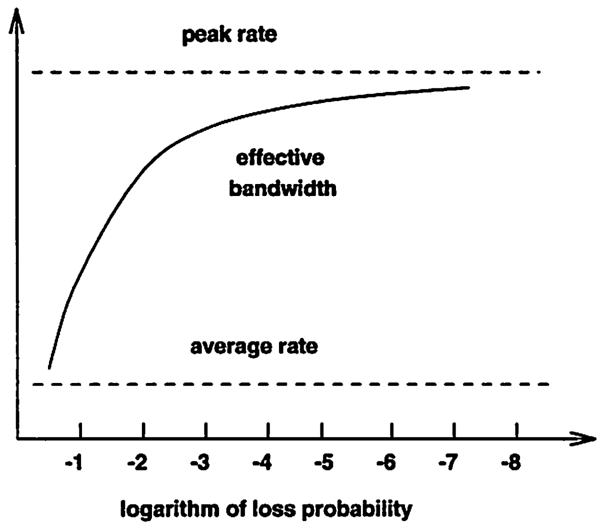

Figure 1: **Effective Bandwidth and QoS Requirement.**

by many authors in different contexts. For example, in [EM92b, EM95], queueing systems with two or more priority queues and fluid sources are studied using spectral analysis and an effective bandwidth approximation approach, and the corresponding call admission control issues are addressed. It is interesting to note that priority service discipline can be regarded as a special case of GPS scheduling.

The rest of the paper is organized as follows. Section 2 prepares the necessary background for the understanding of the paper. This includes the notion of effective bandwidth in the context of the stochastic envelope process model [Cha94] and the formal definition of GPS and some of its properties. Section 3 states a result on the upper bounds on the asymptotic decay rate of per-session backlog and delay tail distribution and describes several feasibility tests based on this result. Section 4 presents several call admission control schemes with varying time-complexity for both session-based and class-based service models. Section 5 concludes the paper.

2 Preliminaries

2.1 Effective Bandwidths and Envelope Processes

The concept of effective bandwidth was first proposed under the name *equivalent bandwidth* in [GAN91] where a call admission control scheme was described based on the result of [AMS82] for on-off fluid sources. It has thereafter been expanded and generalized to a large class of source types (see, *e.g.*, [GAN91, GH91, Kel91, EM92a, KWC93, Whi93, Cha94, GWh94]). The theory is developed with buffer loss probability as the QoS metric in the context of a single network switch or server with a single buffer (or waiting area) shared by many sessions.

Intuitively, the effective bandwidth of a session is a quantity a^* associated with its arrival process that is equivalent to the service rate required to serve the session so that its QoS requirement can be satisfied asymptotically (*i.e.*, in the region of small loss probabilities). Clearly it lies between the peak rate and the average rate of the session and increases as the QoS requirement becomes more stringent (see Figure 1).

To introduce the concept of effective bandwidths in a more formal and general basis, we use C. S. Chang's notion of stochastic envelope processes (E.P.) [Cha94] as source traffic model. The E.P. model was originally defined for discrete time, but can be easily extended to continuous time. We will describe it below in the continuous time framework.

Consider the random rate process $\{a(t), t \geq 0\}$ which describes the actual traffic being offered to the network. Here $a(t)$ is the instantaneous arrival rate at time t. We assume that $a(t)$ is bounded for all $t \geq 0$. Then $A(\tau, t) = \int_\tau^t a(s)ds$ represents the cumulative arrivals over the time interval $[\tau, t]$ to the network. A will be referred to as an arrival process with rate process $\{a(t), t \geq 0\}$. For each $\theta \geq 0$, define

$$A^*(\theta, t) = \sup_{s \geq 0} \frac{1}{\theta} \log E e^{\theta A(s, s+t)}. \tag{1}$$

$A^*(\theta, t)$ is called the minimum envelope process of A with respect to θ in [Cha94]. Associated with each $A^*(\theta, t)$ is the *minimum* envelope rate (MER) of A with respect to θ:

$$a^*(\theta) = \limsup_{t \to \infty} \frac{A^*(\theta, t)}{t}. \tag{2}$$

Given that the instantaneous arrival rate $a(t)$ is bounded for all $t \geq 0$, $a^*(\theta)$ is continuous and increasing in θ, and

$$\inf_{t \geq 0} \frac{1}{t} \sup_{s \geq 0} EA(s, s+t) = a^*(0) \leq a^*(\theta) \leq a^*(\infty) = \inf_{t \geq 0} \frac{1}{t} \sup_{s \geq 0} \|A(s, s+t)\|_\infty \tag{3}$$

where $\|X\|_\infty = \inf\{x : Pr(X > x) = 0\}$ [Cha94]. Note that $\inf_{t \geq 0} \frac{1}{t} \sup_{s \geq 0} EA(s, s+t)$ can be regarded as the long term average rate of the arrival process A and $\inf_{t \geq 0} \frac{1}{t} \sup_{s \geq 0} \|A(s, s+t)\|_\infty$ the long term peak rate of A. Hence, $a^*(\theta)$ ranges from the long term average rate to the long term peak rate. In particular, if A is stationary, then $a^*(0)$ is the average rate of the arrival process A and $a^*(\infty)$ is the peak rate.

The MER of an arrival process has the following *sub-additivity* property: let $\{a_i(t) : t \geq 0\}$, $1 \leq i \leq n$, be n *independent* rate processes, each with MER $a_i^*(\theta)$, and let $\{a(t) = \sum_{i=1}^n a_i(t) : t \geq 0\}$ be the aggregate rate process, then $a^*(\theta) \leq \sum_{i=1}^n a_i^*(\theta)$ where $a^*(\theta)$ is the MER of the aggregate rate process.

The MER is closely related to the theory of effective bandwidth mentioned earlier. Consider a network server with a single class of traffic sharing a buffer. The service rate is a constant, denoted r, and the service discipline is any work-conserving scheduling policy, say, FIFO. This corresponds to a standard $G/D/1$ queueing system. Let $a^*(\theta)$ be the MER of the (aggregate) arrival process A to the system and $\{a(t), t \geq 0\}$ be the associated (aggregate) rate process. Let $Q(t)$ denote the queue length at time $t \geq 0$. Then it can be proved [Cha94] that

$$\lim_{q \to \infty} \frac{1}{q} \log Pr\{Q(t) \geq q\} \leq -\theta \text{ if } a^*(\theta) < r \tag{4}$$

Therefore, if there are n sessions in the system, with $a_i^*(\theta)$ being the MER of the session i arrival process, then the sub-additivity property of the MER implies that for all $t \geq 0$,

$$\lim_{q \to \infty} \frac{1}{q} \log Pr\{Q(t) \geq q\} \leq -\theta \text{ if } \sum_i^n a_i^*(\theta) < r. \tag{5}$$

If the system has a buffer of size q and a loss probability requirement η, i.e., $Pr\{Q(t) \geq q\} \leq \eta$ for all $t \geq 0$, then for q large enough, the condition $\sum_{i=1}^n a_i^*(\xi) < r$, where $\xi = \frac{-\log \eta}{q}$, implies that $Pr\{Q(t) \geq q\}$ can be approximately upper bounded by $e^{-\xi q} = \eta$. *Clearly the test* $\sum_{i=1}^n a_i^*(\xi) < r$ *provides a basis for call admission control.*

The relationship in (4) can be made *tight* if stronger conditions on the rate process $\{a(t), t \geq 0\}$ is imposed. For example, assume that $\{a(t) : t \geq 0\}$ is stationary and ergodic; $a^*(\theta) = \lim_{t\to\infty} \frac{A^*(\theta,t)}{t}$ exists for all $\theta \geq 0$; and $\theta a^*(\theta)$ is strictly convex and differentiable for $\theta \geq 0$ ([Cha94]), then

$$\lim_{q\to\infty} \frac{1}{q} \log Pr\{Q \geq q\} \leq -\theta \text{ iff } a^*(\theta) < r \tag{6}$$

where Q is the stationary queue length distribution.

Hence in this case, the MER $a^*(\theta)$ is the *effective bandwidth* of the arrival process A, as it characterizes the exact bandwidth requirement for achieving the event that $Pr\{Q \geq q\} \leq e^{-\theta q}$ asymptotically. In other words, let $\theta^* = \sup\{\theta \geq 0 : a^*(\theta)\}$, (6) suggests that for q is large, we have the following *effective bandwidth approximation* to the queue length tail distribution $Pr\{Q \geq q\}$:

$$Pr\{Q \geq q\} \approx e^{-\theta^* q}. \tag{7}$$

Under these conditions, it is easy to check that $a^*(\theta)$ satisfies the *additivity* property: $a^*(\theta) = \sum_{i=1}^{n} a_i^*(\theta)$ for n independent arrival processes. Clearly, these properties of effective bandwidth makes call admission control based on the test $\sum_{i=1}^{n} a_i^*(\theta) < r$ *asymptotically optimal* (cf., (5)) for the single buffer system discussed above.

For generality, however, we will not assume any stronger conditions on any arrival process other than that the MER as defined in (2) exists. From the perspective of call admission control, the upper bound relation (4) and sub-additivity of the MER are sufficient. Stronger conditions on arrival processes are not always warranted, although an effective bandwidth result of the form (6) generally leads to provably *asymptotically optimal* call admission control scheme. In a looser sense, we will still refer to the MER $a^*(\theta)$ defined in (2) as the effective bandwidth of the arrival process A instead of using the more cryptic acronym MER.

Effective bandwidth in its most general form is not easy to compute. However, for most common arrival processes such as on-off fluid sources or MMPP, simple expressions can be derived for efficient computation. Moreover, several approximations to the effective bandwidth function using a few parameters have been proposed [CFW94, Cha93]. These parameters can be estimated either off-line or on-line.

Effective bandwidth approximation to the buffer loss probability (7) may not be good enough in many situations, for example, when the system load is low. This phenomenon has been pointed out even when the effective bandwidth approximation approach was originally proposed in [GAN91]. More recently, [CLW93] demonstrated several examples where the effective bandwidth approximation (7) perform miserably. Several possible remedies have been suggested in [CLW93]. One simple and appealing remedy is by introducing an exponential prefactor Λ in (7) as illustrated below:

$$Pr\{Q(t) \geq q\} \approx \Lambda e^{-\theta q}. \tag{8}$$

This new approximation yields considerable improvement on (7) (see, *e.g.*, [CLW93, EM95, LNT94, BD94]) in many situation. Intuitively, Λ can be viewed as an indication of the system multiplexing gains and it is a function of the system load. The call admission control schemes proposed in the paper can be modified to incorporate this new approximation.

2.2 Generalized Processor Sharing (GPS) Scheduling

Generalized Process Sharing (GPS) is a work-conserving scheduling discipline that can be regarded as the limiting form of a weighted round robin policy, where traffic from sessions is treated as an infinitely divisible fluid (hence there is no notion of "packet" in this traffic model [PG93a, Parekh92]). Assume we have n sessions sharing a GPS server with rate r. Associated with the sessions is a set of parameters $\{\phi_i\}_{1 \le i \le n}$ (called the *GPS assignment*) which determine the minimum sharing of bandwidth of each session. Each session is guaranteed a minimum service rate of $g_i = \frac{\phi_i}{\sum_{j=1}^{n} \phi_j} r$. More generally, if the set of sessions with queued data at time t is $S(t) \subseteq \{1, \ldots, n\}$, the session $i \in S(t)$ receives service at rate $\frac{\phi_i}{\sum_{j \in S(t)} \phi_j}$ at time t.

In [PG93, PG94, Pare92], Parekh and Gallager presented a thorough examination of the GPS scheduling under a deterministic setting where the source traffic of each session is characterized as a *Linear Bounded Arrival Process* (LBAP) [Cru91a]. A LBAP has two parameters, rate ρ and maximum burst size σ such that the amount of source traffic arriving over any time interval of length t is bounded above by $\rho t + \sigma$. Given that the traffic of each session conforms to a LBAP (as would be the case when a session is regulated by a leaky bucket mechanism) and that the total arrival rate of all the sessions is smaller than the service rate, it was shown that the backlog and delay of each session are bounded from above in the case of a single GPS server in isolation. In the case of a network of GPS servers, it was shown that the network is stable, *i.e.*, the end-to-end delay of each session is bounded, under a broad class of GPS assignments. These bounds are actually attainable in the worst-case scenario. Of particular interest is the so-called *Rate Proportional Processor Sharing* (RPPS) GPS networks. For RPPS GPS networks, bounds on backlog and delay for each session do not depend on the length of the route the session traverses but only on the bottleneck node on the route. In this case, simple closed form expressions for the end-to-end delay of each session can be derived.

In [YaSi94, ZTK94, ZTK95], the GPS scheduling is studied when the traffic generated by sources is modeled by an *Exponentially Bounded Burstiness* (E.B.B.) process [YaSi93]. We say an arrival process, A_i, is a $(\rho_i, \Lambda_i, \alpha_i)$-E.B.B. process, if for any τ and t such that $\tau \le t$ and for any $x \ge 0$,

$$Pr\{A_i(\tau, t) \ge \rho_i(t - \tau) + x\} \le \Lambda_i e^{-\alpha_i x} \tag{9}$$

where $A_i(\tau, t)$ denotes the amount of traffic arriving during $[\tau, t]$. Here ρ_i is called the long term *upper rate* of the arrival process, Λ_i the prefactor, and α_i the decay rate. Under the E.B.B. model, performance bounds analogous to those of the deterministic model are obtained in [ZTK94, ZTK95]. Given that the appropriate stability conditions are satisfied, upper bounds on the backlog and delay tail distributions for each session sharing a single GPS server are obtained and the departure process of each is shown to be an E.B.B process as well (the latter was also proved in [YaSi94]. For a network of GPS servers, a broad class of GPS networks with arbitrary topology is shown to be stable (see also [YaSi94]). In particular, for RPPS GPS networks, the upper bounds on the backlog and delay tail distributions for each session have simple closed form expressions.

The aforementioned results, both deterministic and statistical, are derived via an important concept introduced by Parekh and Gallager: the notion of *feasible ordering*. Given the rates ρ_i (either the rate of an LBAP or an upper rate of an E.B.B. process), $1 \le i \le n$, an ordering of the sessions, $s_1, s_2, \ldots, s_n$, is a feasible ordering with respect to $\{\rho_i\}_{1 \le i \le n}$ and $\{\phi_i\}_{1 \le i \le n}$ if for $i = 1, 2, \ldots, n$,

$$\rho_{s_i} < \frac{\phi_{s_i}}{\sum_{j=s_i}^{s_n} \phi_j} \left(r - \sum_{j=s_1}^{s_{i-1}} \rho_j\right). \tag{10}$$

Such a feasible ordering always exists as long as $\sum_{i=1}^{N} \rho_i < r$. In [ZTK94, ZTK95], the notion of *feasible partition* is introduced which generalizes the notion of feasible ordering. The feasible partition is a partition of the n sessions,

$\mathcal{H} = \{H_l\}_{1\le l\le L}$, $H_1 \cup \cdots \cup H_L = \{1, 2, \ldots, n\}$, where each H_l, $1 \le l \le L$, is defined recursively as follows:

$$i \in H_1 \text{ if } \frac{\rho_i}{\phi_i} < \frac{1}{\sum_{j=1}^{n} \phi_j} r. \tag{11}$$

and for $k \ge 1$, if $H^k := H_1 \cup \cdots \cup H_k \ne \{1, 2, \ldots, n\}$, then H_{k+1} is defined such that

$$i \in H_{k+1} \text{ if } \frac{\rho_i}{\phi_i} < \frac{1}{\sum_{j \notin H^k} \phi_j}(r - \sum_{j\in H^k} \rho_j). \tag{12}$$

A GPS assignment is an RPPS GPS assignment if $\mathcal{H} = \{H_1\}$ and $H_1 = \{1, 2, \ldots, n\}$, i.e., for $1 \le i \le n$, $\rho_i < g_i = \frac{\phi_i}{\sum_{j=1}^{n} \phi_j} r$. In particular, if $\rho_i = \phi_i$, then $\rho_i < g_i$ assuming that $\sum_{i=1}^{n} \rho_i < r$, hence the name *Rate Proportional Processor Sharing*. The notion of RPPS GPS assignment can be extended to a network of GPS servers in a straightforward manner.

Last, a packetized approximation to GPS (*Packetized* GPS, or PGPS) is defined in [PG93a] to account for the packet nature of communication. Results obtained for GPS can be applied to PGPS with appropriate modification (see [PG93a, YaSi94]), and thus it will not concern us further.

3 Bounds on Asymptotic Decay Rates and Feasibility Tests

The analytical results from [PG93, PG94, Pare92, YaSi94, ZTK94], which provide provable backlog and delay bounds, present a theoretical basis for performing call admission control using GPS scheduling. In the deterministic regime where worst-case deterministic QoS guarantees are provided, call admission control is relatively easy. On the other hand, in the stochastic regime where statistical QoS guarantees are provided, call admission control appears to be considerably more complicated. One obvious question, for example, is what is the minimum service rate (or bandwidth) a session requires in order to satisfy its QoS guarantees? Average rate or peak rate cannot be a good indicator in most cases. Here the notion of effective bandwidth proves to be a most natural indication of service or bandwidth requirement. In the following sections, we will show how the theory of effective bandwidths can be applied to address the call admission control issue under GPS scheduling.

3.1 Upper Bounds on Asymptotic Decay Rates

In order to apply the theory of effective bandwidths, we need to derive upper bounds on the asymptotic decay rate of the per-session backlog and delay tail distribution, similar in form to (4).

Suppose we have n sessions sharing a single GPS server with a given GPS assignment, $\{\phi_i\}_{1\le i\le n}$. The session arrival processes are assumed to be independent. For session i, $1 \le i \le n$, $a_i^*(\theta)$ is the effective bandwidth function of its arrival process, well-defined for $\theta \ge 0$. If r is the service rate of the GPS server, then a necessary stability condition is that $\sum_{i=1}^{n} a_i^*(0) < r$, *i.e.*, the sum of the (long term) average arrival rates of all the sessions cannot exceed the service rate. Given these assumptions and no further conditions on the session arrival processes, we are interested in identifying the best possible asymptotic decay rate of the per-session backlog or delay tail distribution for each session.

For any $\theta > 0$, let $\rho_i = a_i^*(\theta), 1 \le i \le n$. Suppose that $\sum_{i=1}^n \rho_i < r$. Then a feasible ordering of the sessions exists. Without loss of generality, assume the ordering is $1, 2, \ldots, n$. Let $Q_i(t)$ ($D_i(t)$) denote the session i backlog (delay) at time t. Then, using the techniques of [ZTK94, ZTK95], we can prove that

$$\limsup_{q\to\infty} \frac{1}{q} \log Pr\{Q_i(t) \ge q\} \le -\theta \tag{13}$$

and

$$\limsup_{d\to\infty} \frac{1}{d} \log Pr\{D_i(t) \ge d\} \le -\theta g_i \tag{14}$$

where $g_i = \frac{\phi_i}{\sum_{j=1}^n \phi_j} r$ is the guaranteed minimum service rate.

Hence if $\sum_{j=1}^n a_j^*(\theta) < r$ (thus $1, 2, \ldots, n$ is a feasible ordering), then θ is a bound on the asymptotic decay rate of the session i backlog tail distribution (and of the session i delay tail distribution when scaled by g_i), $i = 1, 2, \ldots, n$. Closer scrutiny reveals that the condition that $\sum_{j=1}^n a_j^*(\theta) < r$ (hence the existence of a feasible ordering) is actually not necessary. For example, for any i, if

$$a_j^*(\theta) < \frac{\phi_j}{\sum_{l=j}^n \phi_l}(r - \sum_{l=1}^{j-1} a_l^*(\theta)), \quad 1 \le j \le i, \tag{15}$$

then, using the same argument as in the proof of (13) and (14), it can be proved that θ is a bound on the asymptotic (backlog or delay) decay rate. Note that here we do not require that $\sum_{j=1}^n a_j^*(\theta) < r$, as a matter of fact, it can well exceed r. In other words, a feasible ordering of the n sessions with respect to $\{a_i^*(\theta)\}_{1\le i\le n}$ need not exist. This fact motivates us to generalize the notion of feasible ordering as follows.

For any subset F of the n sessions, say, $F = \{s_1, \ldots, s_i\}$, if there exists an ordering of the i sessions in F such that, after a proper renaming of the sessions (*i.e.*, the first session of the ordering is renamed session 1, the second session 2, ..., the ith session i, and the sessions not in F are arbitrarily renamed as session $i+1$ to session n.), (15) is satisfied, then such an ordering of F, a subset of the n sessions, is called a *partial feasible ordering* of the sessions. The following lemma states that the relation $a_i^*(\theta) < \frac{\phi_i}{\sum_{l=i}^n \phi_l}(r - \sum_{l=1}^{i-1} a_l^*(\theta))$ alone is sufficient to ensure that a partial feasible ordering containing i exists, the proof of which is relegated to Appendix A.

Lemma 1 *Let $N = \{1, 2, \ldots, n\}$. For any (possibly empty) set $F \subseteq N \setminus \{i\}$, if there exists $\theta > 0$ such that*

$$a_i^*(\theta) < \frac{\phi_i}{\sum_{l\notin F} \phi_l}(r - \sum_{l\in F} a_l^*(\theta)), \tag{16}$$

then there is a partial feasible ordering, $s_1, s_2, \ldots, s_k$, $1 \le k \le |F| + 1$, such that $s_k = i$ and $s_l \in F$ for $1 \le l \le k-1$.

Lemma 1 leads to the following theorem which identifies the best possible upper bound on the asymptotic (backlog or delay) decay rate for each session obtainable using the techniques of [ZTK94, ZTK95]. The proof of this theorem can also be found in Appendix A.

Theorem 2 *Under the assumptions stated at the beginning of this section, we have that for each session i,*

$$\limsup_{q\to\infty} \frac{1}{q} \log Pr\{Q_i(t) \ge q\} \le -\theta_i^* \tag{17}$$

and

$$\limsup_{d\to\infty} \frac{1}{d} \log Pr\{D_i(t) \geq d\} \leq -\theta_i^* g_i \tag{18}$$

where

$$\theta_i^* = \max_{F \subseteq N\setminus\{i\}} \sup\{\theta \geq 0 : a_i^*(\theta) < \frac{\phi_i}{\sum_{l\notin F} \phi_l}(r - \sum_{l\in F} a_l^*(\theta))\}. \tag{19}$$

3.2 Feasibility Tests

To answer the generic call admission control question in the introduction, it is sufficient to answer the following *feasibility* question:

> Given there are n sessions present in the system, each session having a given QoS guarantee, will the server be able to make the QoS guarantees for *all* the sessions?

In this section, we consider the asymptotic regime where the QoS requirements are expressed in terms of bounds on the asymptotic decay rates. For example, if $\eta_i(q)$ is the desired bound on the loss probability when the session i buffer is of size q, *i.e.*, $Pr\{Q_i(t) \geq q\} \leq \eta_i(q)$, then $\xi_i = \limsup_{q\to\infty} \frac{-\log \eta_i(q)}{q}$ is the desired bound on the asymptotic decay rate of session i backlog tail distribution. Similarly, if $\delta_i(d)$ is the desired bound on the delay probability when the tolerable delay of session i is d, *i.e.*, $Pr\{D_i(t) \geq d\} \leq \delta_i(d)$, then $\xi_i = \limsup_{d\to\infty} \frac{-\log \delta_i(d)}{dg_i}$ is the desired bound on the asymptotic decay rate of session i delay tail distribution. Hence in both cases ξ_i represents the QoS requirement of session i in our discussion.

From Theorem 2, we see that if we know the θ_i^* for each session i, then the feasibility question raised above can be answered easily: if $\xi_i \leq \theta_i^*$ for all i, then the answer is *YES*; otherwise, *NO*. Unfortunately, θ_i^* as defined in (19) cannot be computed efficiently. It generally requires a search that could take as much as exponential time in terms of n. Notwithstanding, an efficient (i.e., polynomial time in n) optimal feasibility test exists. By optimality here, we mean that if $\xi_i \leq \theta_i^*$, $1 \leq i \leq n$, then the test will output *YES*. Before we describe this test, we first introduce the notion of *partial feasible partition*, which is an extension of the notions of partial feasible ordering and feasible partition. Let $N = \{1, 2, \ldots, n\}$. A partial feasible partition, $H_1, \ldots, H_n$, of N with respect to θ is defined recursively as follows: for $1 \leq k \leq n$,

$$\beta_k = \frac{1}{\sum_{j\in N\setminus H^{k-1}} \phi_j}(r - \sum_{j\in H^{k-1}} a_j^*(\theta)) \tag{20}$$

and

$$H_k = \{j \in N \setminus H^{k-1} : a_j^*(\theta) \leq \phi_j \beta_k\}. \tag{21}$$

where $H^0 = \emptyset$ and $H^k := H_1 \cup \cdots \cup H_k$ for $k \geq 1$.

We say H_k is well-defined if $H_k \neq \emptyset$. Suppose p is such that $H_p \neq \emptyset$ but $H_{p+1} = \emptyset$. Then $\beta_{p+1} = \cdots = \beta_n$ and $H_{p+1} = \cdots = H_n = \emptyset$ if $p < n$. It is possible that $H_1 = \emptyset$, or $H_n \neq \emptyset$. In the former case, the partial feasible partition with respect to θ is null. In the latter case, all H_k's are singleton sets. Clearly, $H^p = H_1 \cup \cdots H_p \subseteq N$. The containment can be strict, hence the name *partial* feasible partition. Actually, $H^p = N$ if and only if $\sum_{j=1}^{n} a_j^*(\theta) \leq r$, in which case, $H_1, \ldots, H_p$ is a feasible partition of N.

The notion of the partial feasible partition captures in essence how the dynamic sharing of bandwidth among the sessions under GPS scheduling determines the decay rate of the backlog tail distribution of each session. Consider session i. Intuitively, if $i \in H_1 := \{j : a_j^*(\theta) \le g_j = \phi_j\beta_1\}$, *i.e.*, $a_i^*(\theta) \le g_i$, then the minimum guaranteed service rate g_i ensures that the session i backlog will decay at a rate of at least θ, independent of other sessions. On the other hand, if $i \notin H_1$ but $H_1 \neq \emptyset$, then the minimum guaranteed service rate, g_j, for $j \in H_1$, will ensure that the session j backlog will decay fast enough (at least at a rate of θ) so that its buffer will stay empty with high probability. In other words, sessions in H_1 will be idle (no queued data in their buffer) with high probability. Hence under GPS scheduling, for $k \notin H_1$, session k will receive a service rate of at least $\phi_k\beta_2 = \frac{\phi_k}{\sum_{j\in N\setminus H_1}\phi_j}(r - \sum_{j\in H_1} a_j^*(\theta))$ at times when sessions in H_1 are idle. Therefore, if $a_i^*(\theta) \le \phi_i\beta_2$, with high probability, the session i backlog will decay at a rate of at least θ. Continuing this line of argument, we see that if $a_i^*(\theta) \le \phi_i\beta_k$, *i.e.*, $i \in H_k$, for some $l, 1 \le k \le n$, then the session i backlog will decay at a rate of at least θ with high probability. This is exactly the reasoning behind the optimal feasibility test we will describe shortly.

Analogously, for any $F \subset N$ with $m = |F|$, we can also define a partial feasible partition, $F_1, F_2, \ldots, F_m$ of F with respect to θ: for $1 \le k \le m$,

$$\gamma_k = \frac{1}{\sum_{j\in N\setminus F^{k-1}}\phi_j}(r - \sum_{j\in F^{k-1}} a_j^*(\theta)) \tag{22}$$

and

$$F_k = \{j \in F \setminus F^{k-1} : a_j^*(\theta) \le \phi_j\gamma_k\}. \tag{23}$$

where $F^0 := \emptyset$ and $F^k := F_1 \cup \cdots \cup F_k$ for $k \ge 1$.

We will refer to the β_k's and γ_k's defined in (20) and (22) as the *associated delimiting numbers* for the partial feasible partitions of N and F respectively.

A partial feasible partition and its associated delimiting numbers exhibit the following monotonicity properties, the proof of which is relegated to Appendix B.

Lemma 3 *For any $F \subset N$ with $m = |F|$, let $F_1, F_2, \ldots, F_m$ be the partial feasible partition of F with respect to θ and $\gamma_1, \gamma_2, \ldots, \gamma_m$ be the associated delimiting numbers. We have*

(a) $\gamma_1 \le \gamma_2 \le \cdots \le \gamma_m$.

(b) For any $E \subseteq F$ with $l = |E|$, let $E_1, E_2, \ldots, E_m$ be the partial feasible partition of E with respect to θ and $\eta_1, \eta_2, \ldots, \eta_l$ be the associated delimiting numbers. Then $\eta_k \le \gamma_k$ and $E^k := E_1 \cup \cdots \cup E_k \subseteq F^k := F_1 \cup \cdots \cup F_k$, $1 \le k \le l$.

We now describe the optimal feasibility test: the test outputs *YES* if and only if $\xi_i \le \theta_i^*$ for all $i, 1 \le i \le n$, where ξ_i is the session i QoS requirement defined in the beginning of this section and θ_i^* is defined in (19).

For $1 \le i \le n$, let $H_{i,1}, \ldots, H_{i,n}$ be the partial feasible partition of N with respect to $\{a_j^*(\xi_i), j \in N\}$, and $\beta_{i,1}, \ldots, \beta_{i,n}$ the set of delimiting numbers used in their definition. The test is described in pseudo-code as follows:

Test 1 ***: Optimal Feasibility Test:***

for $i := 1$ to n do
    if $i \notin H_i^n := \cup_{j=1}^n H_{i,j}$
    then output NO and stop;
endfor;
output YES and stop.

To see why this test gives the correct answer, observe that, if $i \in H_i^n$, then there exists k, $1 \le k \le n$, such that $H_{i,k} \neq \emptyset$ and $i \in H_{i,k}$. Hence by Theorem 2, $\xi_i \le \theta_i^*$. We also claim that if $\xi_i \le \theta_i^*$, then $i \in H_i^n$. Therefore the test is optimal. The claim is shown as follows. From $\xi_i \le \theta_i^*$, wee see that, from Theorem 2 and the increasingness property of a^*, there exists an $F \subseteq N_i$ such that

$$a_i^*(\xi_i) < \frac{\phi_i}{\sum_{j \notin F} \phi_j}(r - \sum_{j \in F} a_j^*(\xi_i)). \tag{24}$$

Let $F' = F \cup \{i\}$ and $m = |F'|$, thus $1 \le m \le n$. Consider the partial feasible partition of F', $F_1, \ldots, F_m$, and the partial feasible partition (with respect to ξ_i) of $N = \{1, 2, \ldots, n\}$, $H_1, \ldots, H_n$. Let $\gamma_1, \ldots, \gamma_m$ be the associated delimiting numbers for the F' partition and $\beta_1, \ldots, \beta_n$ for the N partition. As $F' \subseteq N$, from Lemma 3(b), we have that $\gamma_k \le \beta_k$ and $F^k \subseteq H^k, 1 \le k \le m$.

Given that (24) holds, using an argument similar to that used in the proof of Lemma 1, it can be shown that there exists k^*, $1 \le k^* \le m$ such that $i \in F_{k^*}$, hence $i \in F^m$. But as $F^m \subseteq H^m \subseteq H^n$, therefore $i \in H^n$. This establishes the optimality of the optimal feasibility test.

Time Complexity Analysis: First we remark that for each i, $H_{i,1}, \ldots, H_{i,n}$ can be constructed efficiently by using a sorted list of $\frac{a_j^*(\xi_i)}{\phi_j}, 1 \le j \le n$. This takes $O(n \log n)$ time for each i, and $O(n^2 \log n)$ time in total. Once $H_{i,1}, \ldots, H_{i,n}$ is given, the rest of the test can be carried out in $O(n^2)$ time. Hence, the optimal feasible test takes $O(n^2 \log n)$ time.

In many circumstances, a faster and simpler feasibility test is more desirable, although such a test may be "sub-optimal" in the sense that it rejects calls that may be admitted otherwise. Since the complexity of the optimal feasibility test mostly lies in the construction of the partial feasible partitions, simpler but suboptimal tests can be derived by only using part of this information. We provide two such examples.

Test 2 ***RPPS Feasibility Test:***

for $i := 1$ *to* n *do*
    if $a_i^*(\xi_i) > g_i = \beta_{i,1}$
    then output NO and stop;
endfor;
output YES and stop.

This test is equivalent to checking whether $i \in H_{i,1}$ for all i and does not require that the partial feasibility partition be constructed at all. Clearly, this test takes only $O(n)$ time. Note that if we regard $\rho_i = a_i^*(\xi_i)$ as the "rates" of the session, since $\rho_i \le g_i$, the sessions admitted are scheduled according to a RPPS-like GPS policy: each session is guaranteed a minimum bandwidth independent of other sessions. Hence we call this the RPPS feasibility test.

A slightly more sophisticated test which subsumes the RPPS feasibility test is described below. For each i, let $I_i = \{j \neq i : a_j^*(\xi_i) \le \phi_j \beta_{i,1}\}$ and define $\nu_i = \frac{1}{\sum_{j \in N \setminus I_i} \phi_j}(r - \sum_{j \in I} a_j^*(\xi_i))$.

Test 3 ***Idle-Set Feasibility Test:***

for $i := 1$ *to* n *do*
    if $a_i^*(\xi_i) > \phi_i \nu_i$

then output NO and stop;
endfor;
output YES and stop.

Note that since $\nu_i \geq \frac{1}{\sum_{j \in \mathcal{N} \setminus I_i} \phi_j}(r - \sum_{j \in I} \phi_j \beta_{i,1}) = \beta_{i,1}$, $a_i^*(\xi_i) \leq \phi_i \nu_i$ if and only if $i \in H_i^2 = H_{i,1} \cup H_{i,2}$. Hence this test uses the first two sets of the partial feasible partitions. The test is called the *Idle-Set Feasibility Test* because I_i contains the sessions that will likely be idle as $a_j^*(\xi_i) \leq \phi_j \beta_{i,1} = g_j$ for $j \in I_i$, hence session i will receive a service rate of at least $\phi_i \nu_i$ most of the time. The sets $I_1, \ldots, I_n$ can be constructed in $O(n \log n)$ time by observing that $I_j \subseteq I_i$ if $\xi_j \leq \xi_i$, hence sorting $\xi_1, \ldots, \xi_n$ in increasing order, and for each j, a binary search of $\phi_j \beta_{i,1} = g_j$ in the sorted list $a_j^*(\xi_1), \ldots, a_j^*(\xi_n)$ will decide which I_i to place j. The process takes $O(n \log n)$ time as claimed. Once $I_1, \ldots, I_n$ is known, the rest of the test can be done in linear time.

As a comparison, we look at another simple test which is not unique to the GPS scheduling.

Test 4 ***The Aggregate Feasibility Test:***

$\hat{\xi} := \max_{1 \leq i \leq n} \xi_i$;
if $\sum_{j=1}^{n} a_j^*(\hat{\xi}) \leq r$
then output YES and stop;
else output NO and stop.

The above condition $\sum_{j=1}^{n} a_j^*(\hat{\xi}) < r$ is equivalent to $\sum_{j=1}^{n} a_j^*(\xi_i) \leq r$ for all i. The condition ensures that ξ_i is a bound on the asymptotic decay rate of the aggregate backlog distribution of all the queues, hence it is also a bound on each individual queue. In other words, this test is oblivious of the queue scheduling policy as long as it is work-conserving, so it is applicable to such queue policies as GPS, priority-queue and head-of-line scheduling. Clearly, the test only takes $O(n)$ time.

It is not difficult to construct scenarios where Test 2 or Test 3 returns *YES* and Test 4 does not, or Test 4 returns *YES* but Test 2 and Test 3 return *NO*.

3.3 A Numerical Example

In this section we present a simple numerical example to help illustrate the feasibility tests described in the previous section. We consider a GPS server with two classes of traffic, each with its own buffer. The rate of the server is r. For class i, $i = 1, 2$, the GPS assignment is ϕ_i, with $\phi_1 + \phi_2 = 1$, the buffer size i is q_i, and the buffer loss probability requirement is η_i, *i.e.,*, the probability of loss due to buffer i overflow should be bounded from above by η_i. Sources belonging to the same class are identical and are modeled by the standard on-off fluid source model. For class i, the on-off fluid source is described by three parameters. When the source is in the on-state, it generates traffic in a constant rate λ_i; when it is in the off-rate, it generates no traffic. The rate at which the source changes from the off-state to the on-state is α_i, the rate from the on-state to the off-state is β_i. Hence the peak rate of the source is λ_i and the average rate, $\bar{\lambda}_i$, is $\frac{\lambda_i \alpha_i}{\alpha_i + \beta_i}$. The effective bandwidth function for a single class i source is given by the following expression [AMS82]:

$$a_i^*(\theta) = \frac{\theta\lambda_i - \alpha_i - \beta_i + \sqrt{(\theta\lambda_i - \beta_i + \alpha_i)^2 + 4\alpha_i\beta_i}}{2\theta}. \tag{25}$$

Class	α_i	β_i	λ_i	$\bar{\lambda}_i$	q_i	η_i
1	0.025	0.045	1	0.357	100	10^{-3}
2	0.5	0.5	2	1	10	10^{-9}

Table 1: System and On-Off Fluid Source Parameters for Both Classes

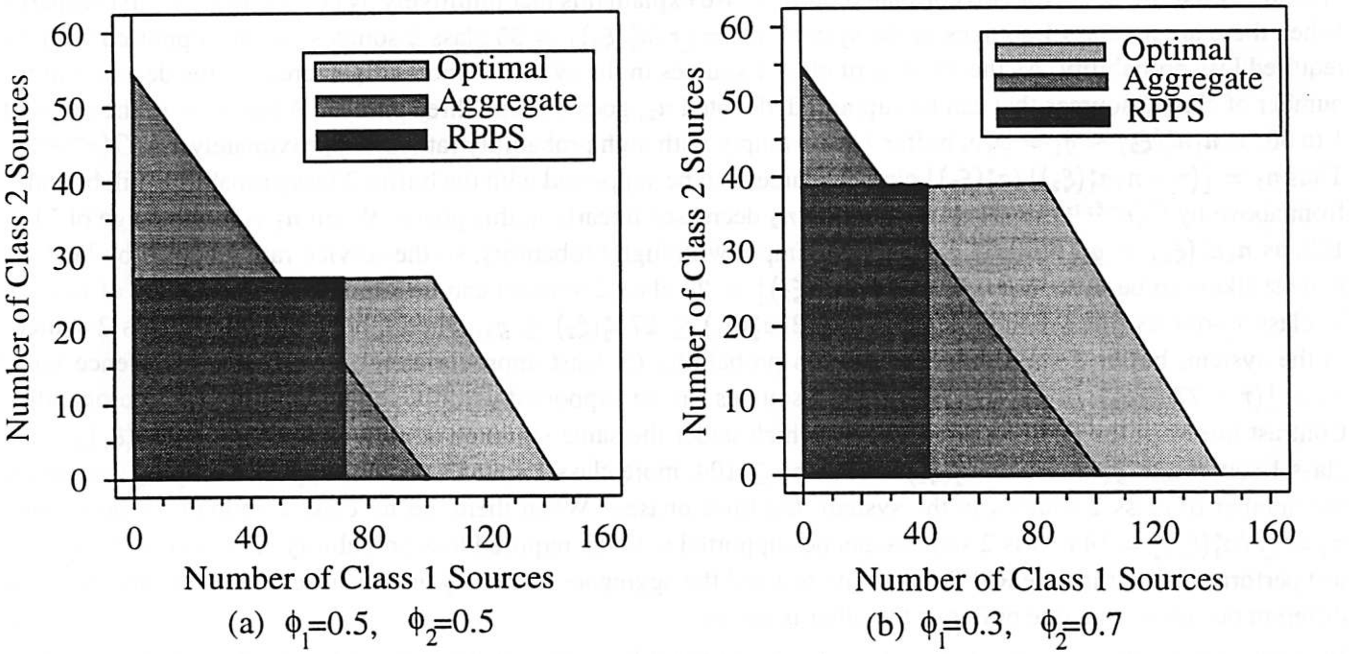

Figure 2: **Feasible Regions of the Feasibility Tests.**

By the additivity of effective bandwidth function, if there are n_i sources in class i, then the aggregate effective bandwidth function for the class is $n_i a_i^*(\theta)$.

We are interested in comparing the feasible regions under the feasibility tests described in the previous section. In other words, we look at the number of class 1 and class 2 sources that can be admitted into the system under those tests such that the designated QoS requirements for both buffers are satisfied. Note that, since we have only two classes, the idle-set feasibility test is identical to the optimal feasibility test.

The source and system parameters for both classes are listed in Table 1. The service rate $r = 100$. From the data we see that a class 1 source is burstier and has a smaller average rate than class 2 source. Moreover, class 1 also has a less stringent QoS requirement, since $\xi_1 = \frac{-\log \eta_1}{q_1} \approx 0.069 < \xi_2 = \frac{-\log \eta_2}{q_2} \approx 2.07$.

We look at two scenarios: first one with a GPS assignment $\phi_1 = \phi_2 = 0.5$; second one with a GPS assignment $\phi_1 = 0.3$ and $\phi_2 = 0.7$. The results are shown in Figure 2.

The feasible region of the aggregate feasibility test covers the same triangle area independent of the GPS assignment, since the test is oblivious of the underlying scheduling policy and always uses the most stringent QoS requirement (here ξ_2) as the QoS requirement for both classes. Because RPPS scheduling treats the two-class system as if it were two independent one-class systems, each with a server of service rate $g_i = \phi_i r$ and a buffer of size q_i, the feasible region of the RPPS feasibility test is a rectangle. The RPPS feasible test performs better when both classes have many sources, whereas the aggregate feasibility test performs better when one class has a dominant number of sources.

Under both scenarios, the feasible region of the optimal feasible test (thus also the idle-set feasible test) contains those of the RPPS feasibility test and the aggregate feasibility test. Notice that, unlike many known results, the feasible region of the optimal feasibility test is not a convex set. This is due to the fact that we have two different loss probability requirements for two correlated buffers. We explain this fact intuitively by considering the first scenario. When there are no class 1 sources in the system, $n_2 = \lfloor r/a_2^*(\xi_2)\rfloor = 55$ class 2 sources can be supported with the required loss probability. As the number of class 1 sources in the system, denoted n_1, increases, the decrease in the number of class 2 sources that can be supported, denoted n_2, goes through three phases. When n_1 is in the range of 1 to 50, as $n_1 a_1^*(\xi_2) \leq g_1 = \phi_1 r$, buffer 1 stays empty with high probability (at least approximately $1 - O(e^{-\xi_2 q_2})$). Thus $n_2 = \lfloor (r - n_1 a_1^*(\xi_2))/a_2^*(\xi_2)\rfloor$ class 2 sources can be supported with the buffer 2 loss probability still bounded from above by $O(e^{-\xi_2 q_2}) = O(\eta_2)$. Therefore n_2 decreases linearly in this phase. When n_1 is in the range of 51 to 103, as $n_1 a_1^*(\xi_2) > g_1$, buffer 1 is no longer empty with high probability, so the service rate received by buffer 2 is most likely to be $g_2 = \phi_2 r$. $n_2 = \lfloor g_2/a_2^*(\xi_2)\rfloor = 27$ class 2 sources can be supported independent of number of class 1 sources (the second phase). Since $27a_2^*(\xi_1) \leq 27a_2^*(\xi_2) \leq g_2$, when there are only 27 class 2 sources in the system, buffer 2 stays empty with high probability (at least approximately $1 - O(e^{-\xi_1 q_1})$). Hence up to $n_1 = \lfloor (r - 27a_2^*(\xi_1))/a_1^*(\xi_1)\rfloor = 103$ class 1 sources can be supported with the required buffer 1 loss probability. Contrast this with the RPPS feasibility test, which under the same situation admits only $n_1 = \lfloor g_1/a_1^*(\xi_1)\rfloor = 27$ class 1 sources, as $g_1 < r - 27a_2^*(\xi_1)$. When $n_1 \geq 104$, more class 1 sources can be supported only by decreasing the number of class 2 sources in the system (the third phase). When there are no class 2 sources in the system, $n_1 = \lfloor r/a_1^*(\xi_1)\rfloor = 144$ class 2 sources can be supported with the required loss probability. The optimal feasibility test performs better than the RPPS feasibility test and the aggregate feasibility test because it takes advantage of the different occasions that one buffer or the other is empty.

By varying the GPS assignment (*i.e.*, ϕ_1 and ϕ_2), the feasible regions of the optimal feasibility test and the RPPS feasibility test also changes, favoring one class or the other. By proper choice of the GPS assignment, more sources of one class or the other or both can be admitted. For example, from Figure 2, we see that under the RPPS feasibility test, increasing ϕ_2 admits more class 2 sources (which are less bursty and have a higher average rate) but fewer class 1 sources. But under the optimal feasibility test, the feasible region becomes larger as ϕ_2 increases. This suggests that $\phi_2 = 1, \phi_1 = 0$ (*i.e.*, a strict priority policy) would be the best choice. Optimal choice of GPS assignment is an issue that needs to be addressed in the design of the call admission control schemes and network service models.

4 Call Admission Control Schemes for Various Services

In this section we present several call admission control schemes under GPS scheduling, based on the feasibility tests described in the previous section.

Given the scheduling discipline and the QoS metric, the design of call admission control schemes will still depend on the services provided by the network. A number of studies (see, e.g., [CSZ92, SCZ93, Flo93]) have investigated the kinds of network services that should be provided to support real-time applications with widely varying QoS requirements. For example, applications have been categorized into two types of services, *rigid* and *tolerant*, according to the stringency of the QoS requirement and the nature of the applications. For *rigid* applications, the QoS guarantees must be strictly fulfilled, be they deterministic or statistical; otherwise the applications will fail. On the other hand, for tolerant applications, their QoS requirement will invariably be statistical; furthermore, such statistical requirements do not need to be satisfied *ad verbatim*. As long as the received performance is within a "reasonable" range, the applications will be able to adapt to minor fluctuations in the quality of the network service. Thus it has been suggested that for tolerant applications that can adapt, it may be a good idea to group the applications with similar traffic type or QoS requirement together, so that greater multiplexing gains can be exploited and higher bandwidth utilization achieved. For rigid applications, their QoS guarantees should be supported on a per-session basis, at the expense of possibly low bandwidth utilization.

In light of the above discussion, it will be appropriate for the network to provide two types of network services in terms of QoS guarantees to applications: for rigid applications, QoS guarantees will be made on a per-session basis, whereas for tolerant applications, QoS guarantees will be made on a per-class basis. Corresponding to these two types of services, call admission control mechanisms need to be provided at two different levels: session-based and class-based call admission control schemes. We now examine this distinction between session-based and class-based network services further.

Session-Based Network Services

A session is an instance of an application, say, a session of audio or video application, a rlogin session or a ftp session. It consists a logical stream of data from one end to another, *i.e.*, it is a *flow* in the sense of [Zha90]. Each session has its intrinsic performance or QoS requirement under which it can run properly. For applications using the session-based network service, the QoS requirement of each session is supported explicitly. Call admission control decisions will be made based on whether sufficient network resources (*e.g.*, bandwidth and buffer space) can be allocated to establish a connection for the session and satisfy its QoS requirement. We call such a connection with its associated network resources a *channel*, borrowing telephony terminology. In the context of GPS scheduling, the network resource committed to a session at a network switch is a guaranteed share of bandwidth (in terms of ϕ_i's) and a fixed-size buffer.

We further divide the session-based network service into two types of services: fixed-channel and demand-channel services (see Figure 3), as they capture different requirements of potential applications and the need of the network to effectively manage resources and provide fast call set-up. In the fixed-channel service (which corresponds in principle with the service provided by the permanent VCI channels in ATM [ATM93]), a number of channels are established beforehand, each with a given QoS specification and pre-allocated share of bandwidth. In this way, frequently requested applications can be supported quickly without an explicit end-to-end call set-up and per-hop allocation of resources. In the demand-channel service, however, channels are established on an as-needed basis. It can be used to cater to less frequently used applications, or applications with special needs.

Class-Based Network Services

A class is simply a collection of sessions grouped together for some purpose. A class can be defined for the purpose of isolating different traffic types, sharing network resources and providing a common QoS guarantee, or for the purpose of accounting, resource management, or dynamic link sharing among various organizations, institutions or protocol families ([CSZ92, SCZ93, Flo93]).

We identify two possible types of class-based network services: fixed-class service and demand-class service (see Figure 4), similar to the session-based network services. Analogous to the fixed-channel service, the fixed-class service provides a number of pre-defined traffic/service classes with pre-specified QoS guarantees for each class. We assume that the classes are serviced using GPS scheduling, with each class having a guaranteed service rate and a

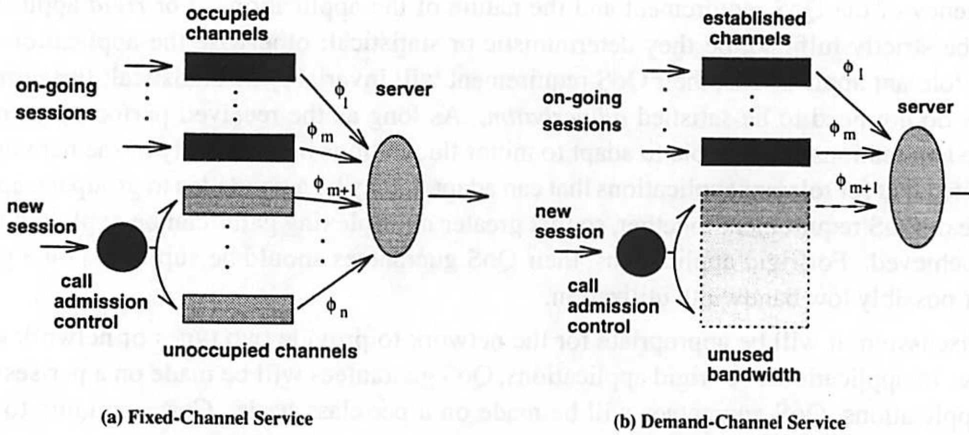

Figure 3: **Session-Based Network Services.**

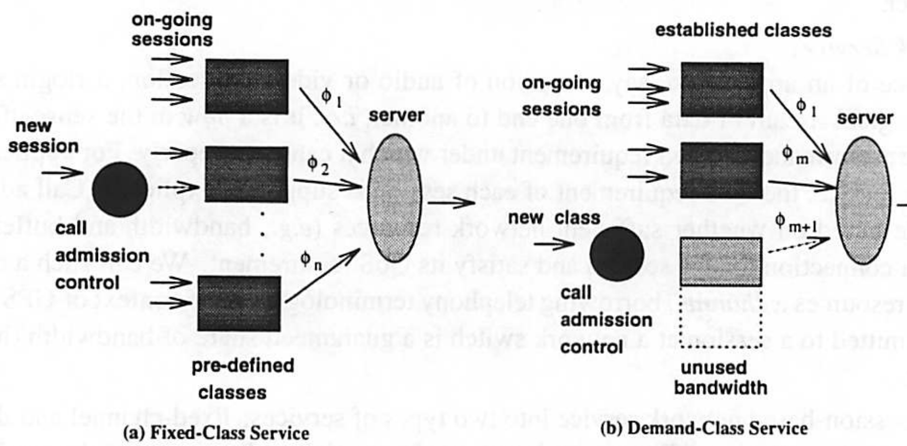

Figure 4: **Class-Based Network Services.**

fixed-size buffer associated with it. Sessions from the same classes are multiplexed together and are scheduled using some work-conserving best-effort or jitter-control scheduling discipline. Applications using this service are assumed to be tolerant so that they can adapt to minor jitter in the network service. As in the case of the demand-channel service, the demand-class service allows the users of the network to define their own service classes by explicitly requesting a class of user-defined QoS guarantees. Such a service also provides a mechanism for dynamic link sharing among various organizations, institutions or protocol families.

We will illustrate the various call admission control schemes for these service models using mainly buffer loss probability as the concerned QoS metric. For simplicity in exposition, we will focus on the case of a single GPS server in isolation, with a constant service rate r. To make the definition of the QoS requirement more precise, we consider the asymptotic regime, i.e., the loss probability is represented by a bound ξ_i on the asymptotic decay rate of loss tail probability for a session i, or class, namely, the following relationship holds

$$\limsup_{q\to\infty} \frac{1}{q} \log Pr\{Q_i(t) \geq q\} \leq -\xi_i. \tag{26}$$

Call admission control schemes using delay probability as the concerned QoS metric can also be designed, although in many cases it is somewhat more complex, since the effective bandwidth function also depends on the minimum guaranteed service rate for a session or a class. Some of the issues are addressed in Section 4.3.

The end-to-end call admission control issue is briefly discussed in Section 4.4.

4.1 Session-Based Call Admission Control

4.1.1 Fixed-Channel Call Admission Control

In the fixed-channel service model, a session is admitted into the network if an unoccupied channel is available that can support its QoS requirement. Suppose we have n pre-established channels serviced by a GPS server. The n channels share the server in an *a priori* determined manner: channel i is guaranteed a minimum service rate of $g_i = \phi_i r$, where $\sum_{i=1}^{n} \phi_i = 1$. Assume that we already have m sessions present in the system which occupy, without loss of generality, channels $1, \ldots, m$. For each session i, $1 \leq i \leq m$, the effective bandwidth function is a_i^*, and the supported QoS guarantee is ξ_i.

Consider now a new session which arrives with a QoS requirement ξ_{new} and an effective bandwidth function a_{new}^*. The call admission control needs to decide whether it can admit this new session into the system, namely, it needs to determine whether one of the unoccupied channels can sustain the QoS requirement of the new session without affecting the QoS of any existing session.

Let $F = \{1, 2, \ldots, m\}$, construct the partial feasible partition of F with respect to $\{a_i^*(\xi_{new}), i \in F\}$, $F_1, \ldots, F_m$, and let $\gamma_1, \ldots, \gamma_m$ be the set of the associated delimiting numbers (see the definition in Section 3.2). The call admission control scheme based on the optimal feasibility test is described below:

Scheme 5 ***Optimal Call Admission Control Scheme: Fixed-Channel***

for $i := m + 1$ *to* n *do*
    if $a_{new}^*(\xi_{new}) \leq \phi_i \gamma_m$
    then accept the new session;
endfor;
reject the new session.

If the new session is accepted and can be supported by more than one channel, the channel i with the smallest ϕ_i among them will be allocated to the new session. This allocation method is chosen so as to enhance system bandwidth utilization, since the larger ϕ_i is, the more bandwidth is committed to the channel when it is busy.

Time Complexity Analysis: The computation of $\gamma_1, \ldots, \gamma_m$ takes at most $O(m \log m)$ time and the test in the above procedure can be done in $O(\log(n - m))$ time. Hence the worst-case time complexity is $O(n \log n)$.

Note that in the scheme above, we do not check explicitly whether the admission of the new session will affect the QoS of any existing session (i.e., sessions 1 to m). The following lemma states that this is not necessary, the proof of which is relegated to Appendix C.

Lemma 4 *Any session admitted using the call admission control scheme based on the optimal feasibility test will not affect the QoS of any existing session.*

If we use γ_2 in place of γ_k in the above procedure, we have a call admission control scheme based on the idle-set feasibility test. As γ_2 can be computed in $O(m)$ time, the call admission can be performed in linear time. Furthermore, a call admission control scheme based on the RPPS feasibility test will be even faster.

Scheme 6 ***RPPS Call Admission Control Scheme: Fixed-Channel***

for $i := m+1$ *to* n *do*
  if $a^*_{new}(\xi_{new}) < \phi_i r$
  then accept the new session;
endfor;
reject the new session.

Time Complexity Analysis: This procedure can be implemented in $O(\log n)$ time using a binary search on the pre-sorted list of $\phi_i r$'s.

The aggregate feasibility test can also be used here. Let $\hat{\xi}_m = \max_{1\le i\le m} \xi_i$ and $\hat{\xi}_{m+1} = \max\{\hat{\xi}_m, \xi_{new}\}$.

Scheme 7 ***Aggregate Call Admission Control Scheme: Fixed-Channel***

if $\sum_{j=1}^{m} a^*_j(\hat{\xi}_{m+1}) + a^*_{new}(\hat{\xi}_{m+1}) \le r$
then accept the new session; else reject the new session.

The scheme takes $O(n)$ time in the worst-case when $\xi_{new} = \hat{\xi}_{m+1}$. Note that under this scheme, if the new session is admitted, as ξ_{new} is a bound on the aggregate queue length decay rate, channel allocation for the new session is need-blind, i.e., independent of $a^*_{new}(\xi_{new})$. Any remaining channel, even a channel with $\phi_i < a^*_{new}(\xi_{new})$ or $\phi_i < a^*_{new}(0)$ (note tha $a^*_{new}(0)$ is the long-term average rate of the new session), can be allocated to the new session. This is a further indication that the aggregate feasibility test is likely to be too conservative.

4.1.2 Demand-Channel Call Admission Control

In the demand-channel service model, when a new session arrives, a new channel will be set up for the session if its QoS requirement can be satisfied and the QoS guarantee of any existing session not violated. Once admitted, the call admission control must decide what fraction of the server bandwidth (ϕ_{new}) it can or wants to commit to this new session so as to satisfy its QoS requirements without affecting those of the existing sessions.

As in Section 4.1.1, we assume that there are m sessions already present in the system, labelled $1, \ldots, m$. For each of the pre-existing sessions i, $1 \le i \le m$, its QoS requirement is ξ_i, its effective bandwidth function is a^*_i and $\phi_i r$ is its minimum guaranteed bandwidth where $0 < \phi_i < 1$. Moreover we assume $\sum_{i=1}^{m} \phi_i < 1$. Define $\phi_0 = 1 - \sum_{i=1}^{n} \phi_i$ to be the unallocated fraction of the bandwidth. For simplicity of discussion, we imagine that we have a fictitious session 0 with a guaranteed ϕ_0 sharing of the bandwidth. However, this session never has traffic to transmit and thus its allocated bandwidth is shared by the other m existing sessions via the GPS scheduling mechanism. Note that each of the m sessions has an actual available bandwidth of $\frac{\phi_i}{\sum_{j=1}^{m}\phi_j} r$.

Consider now a new session which arrives with a QoS requirement ξ_{new} and an effective bandwidth function a^*_{new}. The call admission control mechanism checks whether it can subtract a new fraction of bandwidth, ϕ_{new}, for the new session from the unallocated fraction of bandwidth, ϕ_0, to support the new session's QoS requirement. If so, it is desirable to make ϕ_{new} in order to avoid over-allocating resources.

As in Section 4.1.1, let $N = \{0, 1, 2, \ldots, m\}$ and $F = \{1, 2, \ldots, m\}$, construct the partial feasible partition of F with respect to $\{a_i^*(\xi_{new}), i \in F\}$, $F_1, \ldots, F_m$, and let $\gamma_1, \ldots, \gamma_m$ be the associated delimiting numbers. For the demand-channel service, we have the following call admission control scheme based on the optimal feasibility test.

Scheme 8 ***Optimal Call Admission Control Scheme: Demand-Channel***

if $\sum_{i=1}^{m} \phi_i + a_{new}^*(\xi_{new})/\gamma_m \le 1$
then accept the new session with $\phi_{new} := a_{new}^*(\xi_{new})/\gamma_m$
else reject the new session.

To see why this scheme works correctly, suppose that the new session is admitted with the designated share of bandwidth in the scheme. We now have $m+2$ sessions in the system (including the fictitious session 0 with its ϕ_0 decreased to $1 - \sum_{i=1}^{m+1} \phi_i$). Observe that since the sum of ϕ_{new} and the new ϕ_0 equals the old ϕ_0, this does not change the partial feasible partition of $F = \{1, 2, \ldots, m\}$ and $\gamma_1, \ldots, \gamma_m$ with respect to $\{a_i^*(\xi_{new}), i \in F\}$. As $a_{new}^*(\xi_{new}) \le \phi_{new}\gamma_m$, Lemma 1 and Theorem 2 implies that the new session's QoS can be supported.

The computational requirement for this scheme is $O(n \log n)$ in the worst case as in the case for the fixed-channel service. Likewise, if we use γ_2 in place of γ_k in the above procedure, we have a linear-time call admission control scheme based on the idle-set feasibility test. Furthermore, a call admission control scheme based on the RPPS feasibility test takes only constant time.

Scheme 9 ***RPPS Call Admission Control Scheme: Demand-Channel***

if $\sum_{i=1}^{m} \phi_i + a_{new}^*(\xi_{new})/r \le 1$
then accept the new session with $\phi_{new} := a_{new}^*(\xi_{new})/r$
else reject the new session.

As in the case for the fixed-channel service, we do not need to check whether the admission of the new session will affect the QoS of any existing session explicitly in the above schemes. The penalty associated with this simplification is that this test could be somewhat conservative, as if $\phi_{new} < 1 - \sum_{i=1}^{m} \phi_i$, then the actual guaranteed bandwidth for session $m+1$ (when there are only $m+1$ active sessions in the system) is $\frac{\phi_{new}}{\sum_{i=1}^{m} \phi_i + \phi_{new}} > \phi_{new}$. Note, however, that the actual guaranteed bandwidth for session i, $1 \le i \le m$, decreases from $\frac{\phi_i}{\sum_{i=1}^{m} \phi_i}$ to $\frac{\phi_i}{\sum_{i=1}^{m} \phi_i + \phi_{new}}$. Another possible approach is that whenever a new session arrives, a new set of ϕ_i's is chosen for all the existing sessions including the new session so that the QoS requirements of all the sessions are satisfied. This approach could lead to more efficient use of the system bandwidth. However, it makes call admission control more complex. For the RPPS scheme this approach is still workable, but for Scheme 8 it will involve solving a complex optimization problem, since the partial feasible partition depends on the choice of ϕ_i's.

A call admission control scheme based on the aggregate feasibility test can also be designed. Since it is based on the guarantee on the asymptotic decay rate of the aggregate backlog tail distribution, the choice of the ϕ_i's is immaterial.

Before we leave this section, it is worth pointing out here that in certain circumstances, the minimum bandwidth requirement can be part of the new session's QoS requirement in addition to ξ_{new}. In other words, the new session may explicitly require a share ϕ_{new} of the total bandwidth r. The above call admission control schemes can be easily modified to incorporate this requirement. For example, Scheme 8 becomes: the new session is accepted if $\sum_{i=1}^{m} \phi_i + \phi_{new} \leq 1$ and $a^*_{new}(\xi_{new}) \leq \phi_{new}\gamma_k$. Similarly, Scheme 9 becomes: the new session is accepted if $\sum_{i=1}^{m} \phi_i + \phi_{new} \leq 1$ and $a^*_{new}(\xi_{new}) \leq \phi_{new} r$.

4.2 Class-Based Call Admission Control

4.2.1 Fixed-Class Call Admission Control

In the fixed-class service model, a session chooses to join a pre-defined class and it is admitted if it does not affect the network service provided to any of the classes.

Assume that we have n fixed service classes scheduled using GPS, where each class i is guaranteed a fraction ϕ_i of the total service rate r and $\sum_{i=1}^{n} \phi_i = 1$. Moreover, each class is associated with a QoS guarantee ξ_i; hence the asymptotic decay rate of the aggregate backlog tail distribution of the sessions in class i is bounded by ξ_i. For simplicity, we assume the sessions within a class are scheduled under the FIFO policy.

Suppose there are m_i calls present in each class i, $1 \leq i \leq n$. Let a^*_{ij} be the effective bandwidth function for session j of class i, $1 \leq j \leq m_i$, and define $a^*_i = \sum_{j=1}^{m_i} a^*_{ij}$. We call a^*_i the aggregate class effective bandwidth function. For $1 \leq i \leq n$, let $H_{i,1}, \ldots, H_{i,n}$ be the partial feasible partition of the n classes (with respect to $\{a^*_k(\xi_i), 1 \leq k \leq n\}$) and $\beta_{i,1}, \ldots, \beta_{i,n}$ the corresponding delimiting numbers.

Consider the arrival of a new session, wanting to join class k. Let a^*_{k,m_k+1} be the effective bandwidth function for this new session. The following call admission control schemes decide whether the new session can be admitted into class k.

Scheme 10 ***Optimal Call Admission Control Scheme: Fixed-Class***

for $i := 1$ *to* n *do*
    if $k \in H_i^n := \cup_{l=1}^{n} H_{i,l}$
    then begin
        reconstruct $H_{i,1}, \ldots, H_{i,n}$
        with $(a^*_k + a^*_{k,m_k+1})(\xi_i)$ *in place of* $a^*_k(\xi_i)$;
        if $a^*_i(\xi_i) \notin H_i^n$
        then reject the new session and stop;
      end;
endfor;
accept the new session and stop.

The computational requirement for implementing this scheme is $O(n^2)$, as the sorted list of $a^*_1(\xi_i), \ldots, a^*_n(\xi_i)$ can be updated and thus $H_{i,1}, \ldots, H_{i,n}$ can be constructed in linear time for each i. A simpler version of this scheme based on the idle-set feasibility test leads to a linear time scheme as follows:

Scheme 11 ***Idle-Set Call Admission Control Scheme: Fixed-Class***

for $i := 1$ *to* n *and* $i \neq k$ *do*
    if $k \in I_i := \{j \neq i : a^*_j(\xi_i) \leq \phi_j r\}$
    then begin

$a_k^*(\xi_i) := (a_k^* + a_{k,m_k+1}^*)(\xi_i) > \phi_k r$
if $a_k^*(\xi_i) > \phi_k r$
then $I_i := I_i \setminus \{k\}$;
$\nu_i := \frac{1}{\sum_{j \notin I_i} \phi_j}(r - \sum_{j \in I_i} a_j^*(\xi_i)$;
if $a_i^*(\xi_i) > \phi_i \nu_i$
then reject the new session and stop;
end;
endfor;
if $a_k^*(\xi_k) \leq \phi_i \nu_k$
then reject the new session and stop;
else accept the new session and stop.

The RPPS call admission control scheme takes only constant time.

Scheme 12 ***RPPS Call Admission Control Scheme: Fixed-Class***

if $(a_k^* + a_{k,m_k+1}^*)(\xi_k) \leq \phi_k r$
then accept the new session and stop
else reject the new session and stop.

4.2.2 Demand-Class Call Admission Control

In the demand-class service model, a user requests a new service class by explicitly requesting a desired share of bandwidth for the new class. Unlike the demand-channel service, the class-level QoS guarantee is not explicitly supported. This is left at the discretion of the user. In terms of call admission control, there are two levels of call admission control. One is the network inter-class call admission control, i.e., when a user is requesting a new service class with a guaranteed minimum bandwidth, the network has to decide whether it is able to allocate sufficient bandwidth at this time or not. This is very similar to, but simpler than, call admission control for the demand-channel service, where an explicit minimum bandwidth sharing requirement (i.e., ϕ_{new}) is the QoS requirement. Therefore, with suitable modification, the call admission control schemes for the demand-channel service can be used for this purpose.

Another level of call admission control is the intra-class call admission control, by which we mean QoS guarantees for sessions within the class. The network may be required to provide a specific scheduling policy for scheduling sessions within the class and administer the call admission control for the class, or it may provide a mechanism to support user-provided call admission control algorithms.

4.3 Delay as QoS Requirement

We have discussed various call admission control schemes for the service models using loss probability as the concerned QoS metric. In this section, we turn our attention to the issue of call admission control using delay probability as the concerned QoS metric. For each session i, let $\delta_i(d)$ be the desired bound on the probability that the session i delay exceeds d, i.e., $Pr\{D_i(t) \geq d\} \leq \delta_i(d)$. In the asymptotic regime, define $\zeta_i = \limsup_{d \to \infty} \frac{-\log \delta_i(d)}{d}$.

Then

$$\limsup_{d\to\infty} \frac{1}{d} \log Pr\{D_i(t) \geq d\} \leq -\zeta_i. \tag{27}$$

Hence ζ_i represents the session i QoS requirement in the asymptotic regime. From Theorem 2, we see that it is sufficient to ensure that $\zeta_i \leq g_i\theta_i^*$ in order to support a session with a requested delay QoS of ζ_i. In contrast to the case where loss probability is the concerned QoS, we notice that the effective bandwidth function for the session is not parametrized directly by ζ_i, but by $\xi_i = \zeta_i g_i^{-1}$ where $g_i = \frac{\phi_i}{\sum_{j\in N}\phi_j} r$ and N is the set of the sessions in the system. Note that in some cases, the guaranteed share of bandwidth, ϕ_{new}, for a new session is exactly the parameter that needs to be determined at the time of call admission, hence the minimum guaranteed service rate for the new session, g_{new}, is unknown, so is the effective bandwidth parameter, ξ_i. This makes the issue of call admission control using delay probability as the concerned QoS metric more complex than using loss rate as the concerned QoS metric. In the rest of this section we will address this issue in the context of the four services discussed above.

For class-based call admission control, this turns out not to be an issue. For example, for the fixed-class service, as the classes are defined *a priori* with a pre-determined share of bandwidth (in terms of ϕ_i') and the delay QoS guarantee ζ_i, the minimum guaranteed service rate g_i for each class i is known. Hence for class i, the effective bandwidth parameter $\xi_i = \zeta_i g_i^{-1}$ is given and call admission for this type of service can proceed in exactly the same way as discussed in Section 4.2.1. Similarly, for the demand-class service, as we assume that a request for setting up a new class always comes with an explicit request for a guaranteed minimum service rate for the class, ϕ_{new} is an input parameter to the call admission control procedure and thus network-level call admission control decision can be made in the same way as well. The intra-class level call admission control will depend on what session-level service and scheduling policy are used, and is thus application-dependent, as in the case of loss probability.

For the fixed-channel service, the problem becomes slightly more complicated, but still has an easy solution. When deciding which channel can support the new session with the requested QoS ζ_i, the effective bandwidth parameter ξ_{new} will depend on the channel that is under investigation. For instant, $\xi_{new} = \frac{\zeta_i}{\phi_i r}$ when channel i is currently being considered. This implies that the partial feasible partition $F_1, \ldots, F_n$ and the associated delimiting numbers $\gamma_1, \ldots, \gamma_n$ in the optimal admission control scheme for the fixed class service (cf., Scheme 5) need to be constructed or computed for each channel i. Therefore, the scheme will take $O(n^2 \log n)$ time instead of $O(n \log n)$ time. However, for the RPPS call admission control scheme (cf., Scheme 6), since no partial feasible partition needs to be constructed, the scheme is still linear time.

For the demand-channel service, the issue is most challenging. Since admission control needs to determine whether a minimum guaranteed share of bandwidth, ϕ_{new}, can be allocated for a new session with a given QoS requirement ζ_{new}, the effective bandwidth $a^*_{new}(\xi_{new}) = a^*_{new}(\frac{\zeta_{new}}{\phi_{new} r})$ is a function of ϕ_{new}. Note that the larger ϕ_{new} is, the smaller the effective bandwidth a^*_{new} is. A call admission control scheme based on the optimal feasible test (cf., Scheme 8) involves solving a set of fairly complex equations parametrized by ϕ_i's. Hence an optimal solution may not be easy to find. For the RPPS call admission control scheme (cf., Scheme 9), the problem is tractable:

Scheme 13 ***RPPS Call Admission Control Scheme using Delay QoS: Demand-Channel***

solve $a^*_{new}(\frac{\zeta_{new}}{\phi_{new} r}) = \phi_{new} r$;
if the solution $\phi_{new} \leq 1 - \sum_{i=1}^{m} \phi_i$
then accept the new session
else reject the new session.

Observe that as ϕ_{new} decreases, $\phi_{new}r$ decreases, and since a^*_{new} is an increasing function of its paramter, $a^*_{new}(\frac{\zeta_{new}}{\phi_{new}r})$ increases. Hence there is a point ϕ_{new} at which $\phi_{new}r$ and $a^*_{new}(\frac{\zeta_{new}}{\phi_{new}r})$ intersect, i.e., the solution ϕ_{new} to the equation $a^*_{new}(\frac{\zeta_{new}}{\phi_{new}r}) = \phi_{new}r$ always exists and it is the smallest ϕ_{new} that satisfies $a^*_{new}(\frac{\zeta_{new}}{\phi_{new}r}) < \phi r$. Hence, if $\phi_{new} \leq 1 - \sum_{i=1}^{m} \phi_i$, then it is the smallest possible choice of ϕ_{new} that supports the new session's QoS ζ_{new}.

We can also use empirical methods to choose ϕ_{new}. For example, if the traffic of the new session is very bursty, we may want to choose a larger ϕ_{new}, say, 75% of its peak rate; and if the traffic is less bursty, we can choose a smaller ϕ_{new} so that the guaranteed service rate is close to its mean rate. In the case that the new session has either a loose QoS requirement or it is of low priority, we may want to choose a ϕ_{new} so that its guaranteed service rate is below its mean rate, its QoS guarantee can still be met due to the idle period of other bursty sessions.

4.4 End-to-End Call Admission Control Schemes

In this section we briefly discuss the end-to-end call admission control issue under GPS scheduling.

From the perspective of an application, only the QoS received by the end user at the destination matters. Hence, the QoS specified by a session is almost inevitably the so-called *end-to-end* QoS requirement, *e.g.*, the total loss probability (i.e., fraction of packets lost over the entire route of the session) as opposed to the loss probability at any particular node along the route. From the perspective of the network, however, whether the QoS of a session can be supported depends on the availability of the network resources at each node along the route. These two seemingly incompatible factors makes the end-to-end call admission control much more difficult: how can we decide whether the end-to-end QoS requirement of a session can be satisfied by examining the status of each node along the route the session is supposed to traverse?

One approach is to divide the end-to-end QoS requirement of a session into a set of *nodal* QoS requirements, one for each node along the route. For this apparently simple approach to work correctly and effectively, many technical issues need to be resolved. One obvious problem is how to divide the end-to-end QoS requirement into the nodal QoS requirements so that fulfillment of the nodal QoS guarantees ensures the fulfillment of the end-to-end QoS guarantee as well, while minimizing the resources committed to the session at each node. For example, an end-to-end loss probability requirement can be divided evenly into a set of nodal loss probability requirements. This even division policy ensures the end-to-end QoS guarantees will always be met. However, it may be too pessimistic: the loss may occur mostly at a bottleneck node on the route, so the even division policy imposes too stringent a QoS requirement on each node (including the bottleneck node!). An interesting study of this issue can be found in [NKT93], where several QoS division policies are compared with the *optimal* division policy under different QoS metrics.

Under GPS scheduling, another approach to the end-to-end call admission control problem is possible and perhaps more appealing. The approach is based on RPPS GPS scheduling. Consider a session with an effective bandwidth function $a^*_i(\theta)$ and an end-to-end loss probability requirement η_i. Let q be the smallest buffer size at the nodes along the route of the session, define $\xi_i = -\frac{\log \eta_i}{q}$. If for every node m along the route, $a^*_i(\xi_i) < g^m_i$, the minimum guaranteed service rate for session i at node m, then the result for RPPS GPS scheduling implies that the end-to-end QoS of the session can be satisfied asymptotically. The session behaves as if it has a dedicated channel with a guaranteed service rate of $g_i = \min_m g^m_i$, independent of the other sessions in the network.

For session-based network services, the RPPS call admission control schemes for the single node case can be extended straightforwardly to the end-to-end case: a new session is admitted into the network if a channel with a guaranteed share of the bandwidth, ϕ^m_{new}, can be allocated or established at each node, m, along the route such that $a^*_{new}(\xi) < g^m_{new}$. Note that under such a scheme, each node only allocates enough bandwidth so that the same loss probability as the end-to-end loss probability is supported at every node, unlike the even-division policy, where a more stringent loss probability is supported at each node.

For class-based network services, if a class is defined and established on an end-to-end basis, with an end-to end QoS guarantee, then the class-based RPPS call admission control schemes for the single node case can also be extended to the end-to-end case in a similar fashion as in the session-based network service. On the other hand, if the class is defined on per-node basis, *e.g.*, at each node of the network, a number of service classes with pre-determined minimum bandwidth sharing and pre-specified nodal QoS guarantees are supported, then it is up to the user or the network to decide into which class to place the session at each node so that the nodal QoS guarantees are sufficient to support the user's end-to-end QoS requirement. This decision can be made based on an end-to-end QoS division policy, or using an RPPS-like approach, where the end-to-end QoS is used as a reference to decide which service class to join at each node, and intra-class scheduling is used to help sustain the desired end-to-end QoS. This approach is particularly suitable to the predicative service classes proposed in [CSZ92] where FIFO+ is suggested as the intra-class scheduling policy.

5 Conclusion

In this paper we have proposed several call admission control schemes for the case in which the network must support multiple statistical QoS guarantees. We have done so under various session-based and class-based service models, where the sessions or the classes are scheduled by the Generalized Processor Sharing (GPS) Scheduling service disciplines. The proposed schemes include both optimal schemes and suboptimal schemes requiring less computational effort. The schemes are based on bounds on the asymptotic decay rates of the per-session backlog and delay tail distributions, derived from recent statistical analyses of the GPS scheduling discipline. The call admission control schemes have been discussed in the context of the asymptotic regime in which the theory of effective bandwidth provides a convenient tool to address the call admission control issue. The statistical QoS metrics considered are buffer loss probability and delay probability. For simplicity of exposition, we have mostly focused on a single node case. The end-to-end call admission control issue is much harder than the single node case. Fortunately, the proposed RPPS schemes can be extended to the end-to-end call admission control for a network of GPS servers in a fairly straightforward manner. However, Problems such as the end-to-end QoS division problem and the network stability problem are among many technical issues that need to be resolved for a complete satisfactory answer to the end-to-end call admission control issue. Besides these theoretical issues, many practical issues still need to be resolved before these schemes can be applied in practice. For example, issues of on-line estimation of the effective bandwidth function, buffer loss probability and their relations to system load and source traffic characteristics must be addressed. Another important area of research which we did not touch on in the paper is the call admission control issues for multicast communications [FT95]. This will be a subject of the future research as well.

References

[AMS82] D. Anick, D. Mitra and M. M. Sondhi, Stochastic Theory of a Data Handling System with Multiple Sources, *Bell Syst. Tech. J.*, Vol. 61, pp. 1871-1894, 19982.

[ATM93] ATM Forum, *ATM User-Network Interface Specification*, Version 3.0, Prentice Hall, Englewood Cliffs, New Jersey, 1993.

[BD94] E. Buffet and N. G. Duffield, Exponential Upper Bounds via Martingales for Multiplexers with Markovian Arrivals, *J. Appl. Probability.*, 1994. To appear.

[Cha93] C. S. Chang, Approximations of ATM Networks: Effective Bandwidths and Traffic Descriptors. *IBM RC 18586*, 1993.

[Cha94] C. S. Chang, Stability, Queue Length and Delay of Deterministic and Stochastic Queueing Networks, *IEEE Transaction on Automatic Control*, May 1994.

[CFW94] C. Courcoubetis, G. Fouskas and R. Weber, On the Performance of an Effective Bandwidths Formula, *ITC 14*, J. Labetoulle and J. W. Roberts Ed., 1994, pp. 201-212.

[CLW93] G. L. Choudhury, D. M. Lucantoni and W. Whitt, Squeezing the Most out of ATM, Preprint, 1993. An abridged version with the title "On the Effectiveness of Effective Bandwidths for Admission Control in ATM Networks" appeared in *ITC 14*, J. Labetoulle and J. W. Roberts Ed., 1994, pp. 411-420.

[Cru91a] R. L. Cruz, A Calculus for Network Delay, Part I: Network Elements in Isolation, *IEEE Transaction on Information Theory*, Vol. 37, No. 1, Jan. 1991, pp. 114-131.

[Cru91b] R. L. Cruz, A Calculus for Network Delay, Part II: Network Analysis, *IEEE Transaction on Information Theory*, Vol. 37, No. 1, Jan. 1991, pp. 132-141.

[CSZ92] D. Clark, S. Shenker and L. Zhang, Supporting Real-Time Applications in an Integrated Service Packet Network: Architecture and Mechanism. In *Proceedings of ACM SIGCOMM'92*, pp. 14-26, 1992.

[DKS89] A. Demers, S. Keshav and S. Shenker, Analysis and Simulation of a Fair Queueing Algorithm, *Journal of Internetworking: Research and Experience*, 1, pp. 3-26, 1990. Also in *Proceedings of ACM SIGCOMM '89*, pp. 3-12.

[EM92a] A. Elwalid and D. Mitra, Effective Bandwidth of General Markovian Traffic Sources and Admission Control of High Speed Networks, *IEEE/ACM Trans. on Networking*, Vol. 1, No. 3, pp. 329-357, June 1993.

[EM92b] A. Elwalid and D. Mitra, Fluid Models for the Analysis and Design of Statisticl Multiplexing with Loss Priorities on Multiple Classes of Bursty Traffic, *IEEE INFOCOM'92*, pp. 3C.4.1-3C.4.11, 1922.

[EM95] A. Elwalid and D. Mitra, Analysis, Approximations and Admission Control of A Multi-Service Multiplexing System with Priorities, INFOCOM'95, 1995.

[Flo93] S. Floyd, Link-Sharing and Resource Management Models for Packet Networks, *Preprint*, 1993.

[FV90] D. Ferrari and D. Verma, A Scheme for Real-Time, Channel Establishment in Wide-Area Networks, *IEEE J. Select. Areas Commun.*, Vol. 8, No. 3, April 1990, pp. 368-379.

[FT95] V. Firoiu and D. Towsley, Call Admission and Resource Reservation for Multicast Sessions, *Technical Report TR 95-17*, Computer Science Department, University of Massachusetts, 1995.

[GAN91] R. Guérin, H. Ahmadi and M. Naghshineh, Equivalent Capacity and Its Application to Bandwidth Allocation in High-Speed Networks, *IEEE Journal on Selected Areas in Communications*, Vol. 9, No. 7, Sept. 1991, pp. 968-981.

[GH91] R. J. Gibbens and P. J. Hunt, Effective Bandwidth for Multi-Type UAS Channel, *QUESTA*, No. 9, 1991, pp. 17-28.

[GWh94] P. W. Glynn and W. Whitt, Lorarithmic Asymptotics for Steady-State Tail Probabilities in a Single Server Queue, *J. Appl. Prob.*, Vol. 31, 1994.

[JSZC92] S. Jamin, S. Shenker, L. Zhang and D. D. Clark, An Admission Control Algorithm for Predicative Real-Time Service (Extended Abstract), In *Third International Workshop on Network and Operating System Support for Digital Audio and Video*, Nov. 1992, pp. 308-315.

[Kel91] F. P. Kelly, Effective Bandwidths for Multi-Class Queues, *QUESTA*, Vol. 9, 1991, pp. 5-16.

[KWC93] G. Kesidis, J. Walrand and C. S. Chang, Effective Bandwidths for Multiclass Markov Fluids and Other ATM Sources, *IEEE/ACM Trans. Networking*, Aug. 1993.

[LNT94] Z. Liu, P. Nain and D. Towsley, Exponential Bounds for a Class of Stochastic Processes with Application to Call Admission Control in Networks, In *Proceedings of 33rd Conference on Decision and Control (CDC'93)*, Feb. 1994.

[NKT93] R. Nagarajan, J. F. Kurose and D. Towsley, Local Allocation of End-to-End Quality-of-Service in High-Speed Networks, In *Proceedings of 1993 IFIP Workshop on Perf. Anal. of ATM*, North Holland, 1993, pp. 99-118.

[Pare92] A. K. Parekh, *A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks*, Ph.D Dissertation, Department of Electrical Engineering and Computer Science, MIT, February 1992.

[PG93] A. K. Parekh and R. G. Gallager, A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Single Node Case, *IEEE/ACM Transaction on Networking*, Vol. 1, No. 3, June 1993, pp. 344-357.

[PG94] A. K. Parekh and R. G. Gallager, A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Multiple Node Case, *IEEE/ACM Transaction on Networking*, No. 2, Vol. 2, pp. 137-150, April 1994.

[SCZ93] S. Shenker, D. Clark and L. Zhang, A Scheduling Service Model and a Scheduling Architecture for an Integrated Services Packet Network, Preprint, 1993.

[Tow93] D. Towsley, Providing Quality of Service in Broadband Integrated Services Digital Networks, *Performance Evaluation of Computer and Communication Systems*, Sringer-Verlag, 1993, pp. 560-586.

[Whi93] W. Whitt, Tail Probabilities with Statistical Multiplexing and Effective Bandwidths in Multi-Class Queues, *Telecommunication Systems*, No. 2, 1993, pp. 71-107.

[YaSi93] O. Yaron and M. Sidi, Performance and Stability of Communication Networks via Robust Exponential Bounds, *IEEE/ACM Trans. on Networking*, Vol. 1, No. 3, 1993, pp. 372-385.

[YaSi94] O. Yaron and M. Sidi, Generalized Processor Sharing Networks with Exponentially Bounded Burstiness Arrivals, In *Proceedings of IEEE INFOCOM '94*, June 1994.

[Zha90] L. Zhang, Virtual Clock: A New Traffic Control Algorithm for Packet Switching Networks, In *Proceedings of ACM SIGCOMM '90*, Sept. 1990, pp. 19-29.

[ZTK94] Z.-L. Zhang, D. Towsley and J. Kurose, Statistical Analysis of Generalized Processor Sharing Scheduling Discipline, In *Proceedings of ACM SIGCOMM '94*, August, 1994.

[ZTK95] Z.-L. Zhang, D. Towsley and J. Kurose, Statistical Analysis of Generalized Processor Sharing Scheduling Discipline, *Technical Report TR 95-10*, Computer Science Department, University of Massachusetts, Febuary 1995.

A Proofs of Lemma 1 and Theorem 2

In this appendix, we prove Lemma 1 and Theorem 2 of Section 3.1.

Proof of Lemma 1: We first claim that there exists a $j \in F \cup \{i\}$ such that

$$a_j^*(\theta) < \frac{\phi_j}{\sum_{l=1}^n \phi_l} r. \tag{28}$$

Suppose the opposite is true, then for any $j \in F$,

$$a_j^*(\theta) \geq \frac{\phi_j}{\sum_{l=1}^n \phi_l} r. \tag{29}$$

which implies

$$\sum_{j \in F} a_j^*(\theta) \geq \frac{\sum_{j \in F} \phi_j}{\sum_{l=1}^n \phi_l} r. \tag{30}$$

Hence,

$$r - \sum_{j \in F} a_j^*(\theta) \leq \frac{\sum_{l \notin F} \phi_l}{\sum_{l=1}^n \phi_l} r. \tag{31}$$

Therefore,

$$\frac{\phi_i}{\sum_{l \notin F} \phi_l} (r - \sum_{j \in F} a_j^*(\theta)) \leq \frac{\phi_i}{\sum_{l=1}^n \phi_l} r \leq a_i^*(\theta). \tag{32}$$

The last step follows from the supposition. However, by the hypothesis (16) of the lemma, the left hand side of (32) is bigger than $a_i^*(\theta)$, thus we obtain a contradiction. We conclude that there exists a $j \in F \cup \{i\}$ such that (28) holds. Set $s_1 = j$. If $s_1 = i$, then we are done, s_1 is a partial feasible ordering involving only session i itself. Otherwise, we proceed by induction. Suppose we have found a partial feasible ordering $s_1, s_2, \ldots, s_k$ such that $s_j \in F$, thus

$s_j \neq i, 1 \leq j \leq k$. If $k = |F|$, then (16) implies that $s_1, \ldots, s_k, s_{k+1}$ with $s_{k+1} = i$ is a partial feasible ordering and we are done. If $k \neq |F|$, let $K = \{s_1, \ldots, s_k\}$, we show that there exists $s_{k+1} = j \in (F \setminus K) \cup \{i\}$ such that

$$a_j^*(\theta) < \frac{\phi_j}{\sum_{l \notin K} \phi_l}(r - \sum_{l \in K} a_l^*(\theta)). \tag{33}$$

Suppose again the opposite is true. Then for any $j \in F \setminus K$,

$$a_j^*(\theta) \geq \frac{\phi_j}{\sum_{l \notin K} \phi_l}(r - \sum_{l \in K} a_l^*(\theta)). \tag{34}$$

which implies

$$\sum_{j \in F \setminus K} a_j^*(\theta) \geq \frac{\sum_{j \in F \setminus K} \phi_j}{\sum_{l \notin K} \phi_l}(r - \sum_{l \in K} a_l^*(\theta)). \tag{35}$$

Thus,

$$r - \sum_{l \in F} a_l^*(\theta) = r - \sum_{l \in K} a_l^*(\theta) - \sum_{j \in F \setminus K} a_j^*(\theta) \leq \frac{\sum_{l \notin F} \phi_l}{\sum_{l \notin K} \phi_l}(r - \sum_{l \in K} a_l^*(\theta)). \tag{36}$$

Therefore,

$$\frac{\phi_i}{\sum_{l \notin F} \phi_l}(r - \sum_{j \in F} a_j^*(\theta)) \leq \frac{\phi_i}{\sum_{l \notin K} \phi_l}(r - \sum_{l \in K} a_l^*(\theta)) \leq a_i^*(\theta) \tag{37}$$

where the last step follows from the supposition. This contradicts the hypothesis (16). Hence we prove that $s_{k+1} = j \in (F \setminus K) \cup \{i\}$ exists such that (33) holds. If $s_{k+1} = i$, we are done. Otherwise, the process is repeated until either $k = |F|$ or $s_{k+1} = i$.

∎

Proof of Theorem 2: We prove the theorem for session i. Let F_i be the set that attains the supremum in (19), i.e.,

$$a_i^*(\theta_i^*) = \frac{\phi_i}{\sum_{j \notin F_i} \phi_j}(r - \sum_{j \in F_i} a_j^*(\theta_i^*)). \tag{38}$$

Since $a_j^*(\theta)$ is continuous and increasing, it follows from (38) that

$$a_i^*(\theta) < \frac{\phi_i}{\sum_{j \notin F_i} \phi_j}(r - \sum_{j \in F_i} a_j^*(\theta)). \tag{39}$$

where $\theta = \theta_i^* - \frac{1}{2}\epsilon$ for any ϵ, $0 < \epsilon \leq \theta_i^*$.

From Lemma 1, a partial feasible ordering over $F_i \cup \{i\}$ exists. Without loss of generality, we assume that the partial feasible ordering is $1, \ldots, i-1, i$. For each j, $1 \leq j \leq i$, define $\delta_j^\theta(t) = \sup_{\tau \leq t}\{A_j(\tau, t) - a_j^*(\theta)(t - \tau)\}$. Then from Lemma 1 of [ZTK95], we have that

$$Q_i(t) \leq \sum_{j=1}^{i} \delta_j^\theta(t) \tag{40}$$

Hence, by Chernoff's bound, for $0 < \theta' < \theta$, say, $\theta' = \theta - \epsilon$, and for any $q \geq 0$,

$$Pr\{Q_i(t) \geq q\} \leq e^{-\theta' q} E[\exp\{\theta'(\sum_{j=1}^{i} \delta_j^\theta(t))\}]. \tag{41}$$

By the independence of the session arrival processes, $E[\exp\{\theta'(\sum_{j=1}^{i} \delta_j^\theta(t))\}] = \prod_{j=1}^{i} E[\exp \theta' \delta_j^\theta(t)]$. For each $j, 1 \leq j \leq i$, $E[\exp \theta' \delta_j^\theta(t)]$ can be shown to be finite (cf. Lemma 5 of [ZTK95] and Lemma 3.7 of [Cha94]). Therefore, from (41), we have

$$\limsup_{q \to \infty} \frac{1}{q} \log Pr\{Q_i(t) \geq q\} \leq -\theta' \tag{42}$$

Let $\epsilon \to 0$, we have (17). (18) can be similarly proved by noting that session i has a backlog clearing rate of g_i. ∎

B Monotonicity of Partial Feasible Partition

In this appendix, we prove Lemma 3.

Proof of Lemma 3:
(a) For any k, $1 \leq k \leq m - 1$, note that

$$\frac{1}{\sum_{j \in N \setminus F^{k-1}} \phi_j} = \frac{1}{\sum_{j \in N \setminus F^k} \phi_j}\left(1 - \frac{\sum_{j \in F_k} \phi_j}{\sum_{j \in N \setminus F^{k-1}} \phi_j}\right). \tag{43}$$

Hence

$$\begin{aligned}
\gamma_k &= \frac{1}{\sum_{j \in N \setminus F^k} \phi_j}\left(1 - \frac{\sum_{j \in F_k} \phi_j}{\sum_{j \in N \setminus F^{k-1}} \phi_j}\right)\left(r - \sum_{j \in F^{k-1}} a_j^*(\theta)\right) & (44)\\
&= \frac{1}{\sum_{j \in N \setminus F^k} \phi_j}\left(r - \sum_{j \in F^{k-1}} a_j^*(\theta) - \frac{\sum_{j \in E_k} \phi_j}{\sum_{j \in N \setminus F^{k-1}} \phi_j}\left(r - \sum_{j \in F^{k-1}} a_j^*(\theta)\right)\right) & (45)\\
&\leq \gamma_{k+1}. & (46)
\end{aligned}$$

The last inequality follows as $a_j^*(\theta) \leq \phi_j \gamma_k$, for $j \in F_k$.
(b) First observe that $E^k \subseteq F^k$ follows from $\eta_k \leq \gamma_k$, $1 \leq k \leq l$. We now show the latter by induction on k.
For $k = 1$, we have $\eta_1 = \gamma_1 = \frac{1}{\sum_{j=1}^{n} \phi_j} r$, hence $E_1 \subseteq F_1$.

Now for $1 \leq k \leq l - 1$, assume that $\eta_{k'} \leq \gamma_{k'}$ and thus $E^{k'} \subseteq F^{k'}$ is true for $1 \leq k' \leq k$, we show it is also true for $k + 1$.

$$\begin{aligned}
\gamma_{k+1} &= \frac{1}{\sum_{j \in N \setminus E^k} \phi_j}\left(1 + \frac{\sum_{j \in F^k \setminus E^k} \phi_j}{\sum_{j \in N \setminus F^k} \phi_j}\right)\left(r - \sum_{j \in F^k} a_j^*(\theta)\right) & (47)\\
&= \frac{1}{\sum_{j \in N \setminus E^k} \phi_j}\left(r - \sum_{j \in F^k} a_j^*(\theta) + \frac{\sum_{j \in F^k \setminus E^k} \phi_j}{\sum_{j \in N \setminus F^k} \phi_j}\left(r - \sum_{j \in F^k} a_j^*(\theta)\right)\right) & (48)\\
&= \frac{1}{\sum_{j \in N \setminus E^k} \phi_j}\left(r - \sum_{j \in E^k} a_j^*(\theta) - \sum_{j \in F^k \setminus E^k} a_j^*(\theta) + \sum_{j \in F^k \setminus E^k} \phi_j \gamma_{k+1}\right). & (49)
\end{aligned}$$

Since for any $j \in F^k \setminus E^k$, there exists $k', 1 \leq k' \leq k$ such that $j \in F_{k'}$. Hence $a_j^*(\theta) \leq \phi_j \gamma_{k'} \leq \phi_j \gamma_{k+1}$. Therefore $\sum_{j \in F^k \setminus E^k} a_j^*(\theta) \leq \sum_{j \in F^k \setminus E^k} \phi_j \gamma_{k+1}$. Combining this fact with (49) yields that $\gamma_{k+1} \geq \eta_{k+1}$.

■

C Proof of Lemma 4

Proof of Lemma 4: The proof is by induction.

When the first session arrives, since there is no session before it, it is only necessary to check whether the QoS of this session can be satisfied. Scheme 5 makes sure that it is true when the session is admitted. In particular, we have $a_1^*(\xi_1) \leq \gamma_{1,1} = \phi_1 r$, here we assume, without loss of generality, session 1 is allotted channel 1.

In general, suppose we have m sessions in the system, occupying channels 1 through m respectively. Let $F = \{1, 2, \ldots, m\}$ and $F_{i,1}, \ldots, F_{i,m}$ be the partial feasible partition of F with respect to $\{a_j^*(\xi_i), 1 \leq j \leq m\}, 1 \leq i \leq n$, $\gamma_{i,1}, \ldots, \gamma_{i,m}$ be the associated delimiting numbers. The induction hypothesis states that $a_i^*(\xi_i) \leq \gamma_{i,m}$ or $i \in \cup_{l=1}^m F_{i,l}$ for $1 \leq i \leq m$.

Consider now a new session, session $m+1$, that is admitted into the system by Scheme 5 and is allotted channel $m+1$. We have $F' = \{1, \ldots, m, m+1\}$. For $1 \leq i \leq m+1$, let $F'_{i,1}, \ldots, F'_{i,m}, F'_{i,m+1}$ be the partial feasible partition of F' with respect to $\{a_j^*(\xi_i), 1 \leq j \leq m+1\}$ and $\gamma'_{i,1}, \ldots, \gamma'_{i,m}, \gamma'_{i,m+1}$ be the associated delimiting numbers. Scheme 5 ensures that $a_{m+1}^*(\xi_{m+1}) \leq \gamma_{m+1,m+1}$ or $m+1 \in \cup_{l=1}^{m+1} F'_{m+1,l}$, i.e., the QoS of session $m+1$ will be satisfied. To see that the QoS of session $i, 1 \leq i \leq m$, is still satisfied after the admission of session $m+1$, it is sufficient to prove that $\gamma_{i,j} \leq \gamma'_{i,j}, 1 \leq j \leq m$, since then $i \in \cup_{l=1}^m F_{i,l} \subseteq \cup_{l=1}^m F'_{i,l} \subseteq \cup_{l=1}^{m+1} F'_{i,l}$. But as $F \subset F'$, from Lemma 3(b), we have $\gamma_{i,j} \leq \gamma'_{i,j}, 1 \leq j \leq m$. Thus the lemma follows. ■

The version of Lemma 4 for the demand-channel service model holds as well and can be proved in a similar manner. The only difference is that in the fixed channel service model the set N of channels (or sessions) in the system is fixed, whereas in the demand-channel service model, this is not true. But more careful observation shows that this difference is superfluous and does not affect the proof very much.