October, 1991

LIDS-P 2074

Research Supported By:

NSF grant NCR 8802991
ARO grant DAAL03-86-K-0171
Vinton Hayes Fellowship
CICS Fellowship

A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Multiple Node Case*

Parekh, A.K.J.
Gallager, R.G.

A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Multiple Node Case. [1]

Abhay K. Parekh [2] [3] and Robert G. Gallager [4]
Laboratory for Information and Decision Systems
Massachusetts Institute of Technology

Abstract

Worst-case bounds on delay and backlog are derived for leaky bucket constrained sessions in arbitrary topology networks of Generalized Processor Sharing (GPS) [6] servers. The inherent flexibility of the service discipline is exploited to analyze broad classes of networks. When only a subset of the sessions are leaky bucket constrained, we give succinct per-session bounds that are independent of the behavior of the other sessions and also of the network topology. However, these bounds are only shown to hold for each session that is guaranteed a *backlog clearing rate* that exceeds the token arrival rate of its leaky bucket.

A much broader class of networks, called Consistent Relative Session Treatment (CRST) networks is analyzed for the case in which **all** of the sessions are leaky bucket constrained. First, an algorithm is presented that characterizes the internal traffic in terms of average rate and burstiness, and it is shown that all CRST networks are stable. Next, a method is presented that yields bounds on session delay and backlog given this internal traffic characterization: The session i route is treated as a whole, yielding tighter bounds than those that result from adding the worst-case delays (backlogs) at each of the servers in the route. The bounds on delay and backlog for each session are efficiently computed from a *universal service curve*, and it is shown that these bounds are achieved by "staggered" greedy regimes when an *independent sessions relaxation* holds. Propagation delay is also incorporated into the model.

Finally, the analysis of arbitrary topology GPS networks is related Packet GPS networks (PGPS). For small packet sizes, the behavior of the two schemes is seen to be virtually identical, and the effectiveness of PGPS in guaranteeing worst-case session delay is demonstrated under Rate Proportional Processor Sharing assignments.

---

[1] An abridged version of the results in this paper has been submitted to INFOCOM '93.

[2] Research partly funded by a Vinton Hayes Fellowship and a Center for Intelligent Control Systems Fellowship.

[3] Current Address: IBM, T.J. Watson Research Center, Attn. Dr. Roch Guerin, P.O. Box 704, Yorktown Heights, New York 10598.

[4] Research funded by the National Science Foundation under 8802991-NCR and by the Army Research Office under DAAL03-86-K-0171.

1 Introduction

We extend our analysis in [6] of a single node Generalized Processor Sharing (GPS) system to arbitrary topology networks of GPS servers. These results are then related to networks in which the nodes follow a packet-based service discipline, packet GPS (PGPS) discussed extensively in [6].

A GPS server that serves N sessions on a link is characterized by N positive real numbers, $\phi_1, \phi_2, ..., \phi_N$. These numbers denote the relative amount of service to each session in the sense that if $S_i(\tau, t)$ is defined as the amount of session i traffic served during an interval $[\tau, t]$, then

$$\frac{S_i(\tau,t)}{S_j(\tau,t)} \geq \frac{\phi_i}{\phi_j}, \quad j = 1, 2, ..., N \tag{1}$$

for any session i that is backlogged in the interval $[\tau, t]$. Thus (1) is satisfied with equality for two sessions i and j that are both backlogged during the interval $[\tau, t]$.

Note from (1) that whenever session i is backlogged it is guaranteed a service rate of

$$g_i = \frac{\phi_i}{\sum_{j=1}^{N} \phi_j} r, \tag{2}$$

where r is the rate of the link. This rate is called the *session i backlog clearing rate* since a session i backlog of size q is served in at most $\frac{q}{g_i}$ time units.

We assume a virtual circuit, connection-based packet network, and analyze the performance of leaky bucket constrained sessions. The session i leaky bucket is characterized by a token bucket of size σ_i and a token arrival rate of ρ_i. The amount of session i traffic entering the network during any interval $(\tau, t]$ is defined to be $A_i(\tau, t)$; if session i is leaky bucket constrained, then

$$A_i(\tau, t) \leq \sigma_i + \rho_i(t - \tau), \quad \forall t \geq \tau \geq 0. \tag{3}$$

As in [6], we say that A_i conforms to (σ_i, ρ_i), or $A_i \sim (\sigma_i, \rho_i)$. For details on how to accommodate peak rate constraints as well, see [5]. The constraint (3) is identical to the one suggested by Cruz [1].

The main question we address in this paper is the following: Given a network with the values of the server parameters fixed and a set of leaky bucket constrained sessions, what is the worst-case session delay and backlog for each of the sessions in this set?

In Section 2 we set up our model of the network and specify notation. Then the notions of network backlog and delay are discussed and graphically interpreted. Section 4 contains succinct per-session bounds for the leaky bucket constrained sessions of a network, which are independent of the topology and of the behavior of other sessions. Next, we treat the case when all of the sessions are leaky bucket constrained. An important tool for the analysis, the All-Greedy bound, is presented in Section 6. In Section 7, an algorithm is derived that enables a characterization of internal traffic in terms of burstiness, average and peak rates for a broad class of server allocations called Consistent Relative Session Treatment (CRST) assignments. This class of assignments is flexible enough to accommodate a wide variety of session delay constraints. In Section 8, we show that worst-case session delay and backlog can be bounded from an easily computable universal service curve. This is accomplished even though *different* worst-case regimes may maximize delay and backlog for a given session. The bounds are shown to be tight under an independent relaxation assumption, when the the traffic follows a *staggered greedy* regime. Propagation delay is included in Section 10, and the analysis extended to GPS networks in which packets are not served at node until the last bit has arrived. Having analyzed GPS networks, we turn our attention to PGPS networks in Section 11 and show how our results extend to this case. Conclusions are in Section 12.

Note that all of our bounds can be applied to networks of arbitrary topology.

2 The Network Model

The network is modeled as a directed graph in which nodes represent switches and arcs represent links. A route is a path in the graph, and the path taken by session i is defined as $P(i)$. Let $P(i,k)$ be the k^{th} node in $P(i)$, and K_i be the total number of nodes in $P(i)$. The rate of the server at node m is r^m.

The amount of session i traffic that enters the network in the interval $[\tau, t]$ is given by $A_i(\tau, t)$. Let $S_i^{(k)}(\tau, t)$, $k = 1, \ldots, K_i$, be the amount of session i traffic served by node $P(i,k)$ in the interval $[\tau, t]$. Thus, $S_i^{(K_i)}$ is the traffic that leaves the network. We characterize the service function by "pseudo" leaky bucket parameters $\sigma_i^{(k)}$ and ρ_i so that

$$S_i^{(k)}(\tau, t) \leq \sigma_i^{(k)} + \rho_i(t - \tau), \quad \forall t \geq \tau \geq 0, \tag{4}$$

i.e., $S_i^{(k)} \sim (\sigma_i^{(k)}, \rho_i)$.

Often, we will analyze what happens at a particular server, m. In this case the notation described above becomes overly cumbersome. Define $I(m)$ to be the set of sessions that are served by server m. For every session $i \in I(m)$, let the arrival function into that node be described by $A_i^m \sim (\sigma_i^m, \rho_i)$ and the departure function be described by $S_i^m \sim (\sigma_i^{m,out}, \rho_i)$. For example, at server 0 in Figure 1: $A_0^0 = A_0$, $A_2^0 = S_2^{(2)}$, and $A_3^0 = S_3^{(1)}$. Thus when $k = P(i,j)$ for a particular session, i, the functions $S_i^{(j)}$ and S_i^k are identical. The rate of the link associated with server m is denoted by r^m. The value of ϕ_i at m is denoted by ϕ_i^m for all $i \in I(m)$. Finally, let g_i^m be the session i backlog clearing rate from (2) at node m, i.e.,

$$g_i^m = \frac{\phi_i^m}{\sum_{j \in I(m)} \phi_j^m} r^m. \quad (5)$$

3 Network Delay, Backlog and Stability

In this section we extend the notions of session i delay and backlog introduced in [6] to the multiserver case. Given a set of arrival functions for every session in the network, define $Q_i^{(k)}(t)$ to be the session i backlog at node $P(i,k)$ at time t. Similarly, let $Q_i^m(t)$ be the session i backlog at node $m \in P(i)$. Thus, if $m = P(i,k)$, then

$$Q_i^{(k)}(t) = Q_i^m(t) = A_i^m(0,t) - S_i^m(0,t). \quad (6)$$

Define the total session i backlog at time t to be

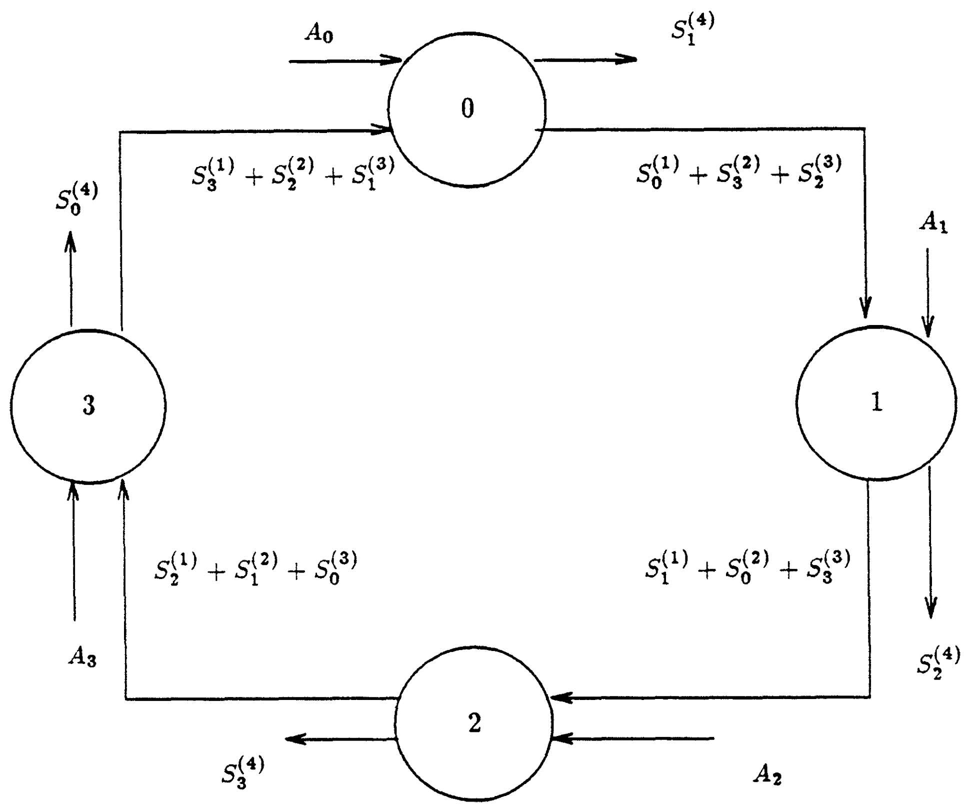
$$Q_i(t) = \sum_{k=1}^{K_i} Q_i^{(k)}(t). \quad (7)$$

Thus, $Q_i(t)$ is the amount of session i traffic buffered in the network at time t. By assumption,

$$Q_i(t) = 0, \quad \forall t \leq 0$$

for every session i. Also, let $D_i(t)$ be the time spent in the network by a session i bit that arrives at time t. Figure 2 shows how to represent the notions of backlog and delay graphically. We see that $D_i(\tau)$ is the horizontal distance between the curves $A_i(0,t)$ and $S_i^{(K_i)}(0,t)$ at the ordinate value of $A_i(0,\tau)$. Clearly, $D_i(\tau)$ depends on the arrival functions $A_1, ..., A_N$, where N is the total number of sessions in the network). We are interested in computing the maximum delay over all time, and over all arrival functions that are consistent with (3). Let D_i^* be the maximum delay for session i. Then

$$D_i^* = \max_{(A_1,\ldots,A_N)} \max_{\tau \geq 0} D_i(\tau).$$



Server 0

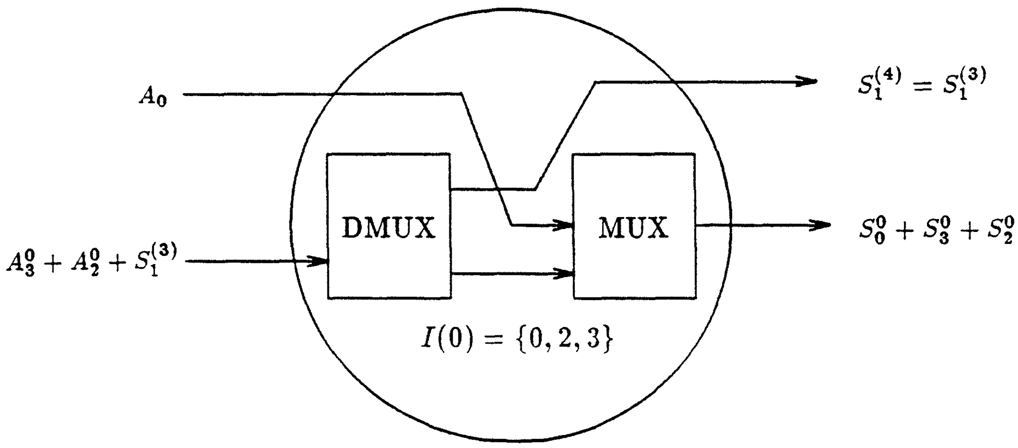
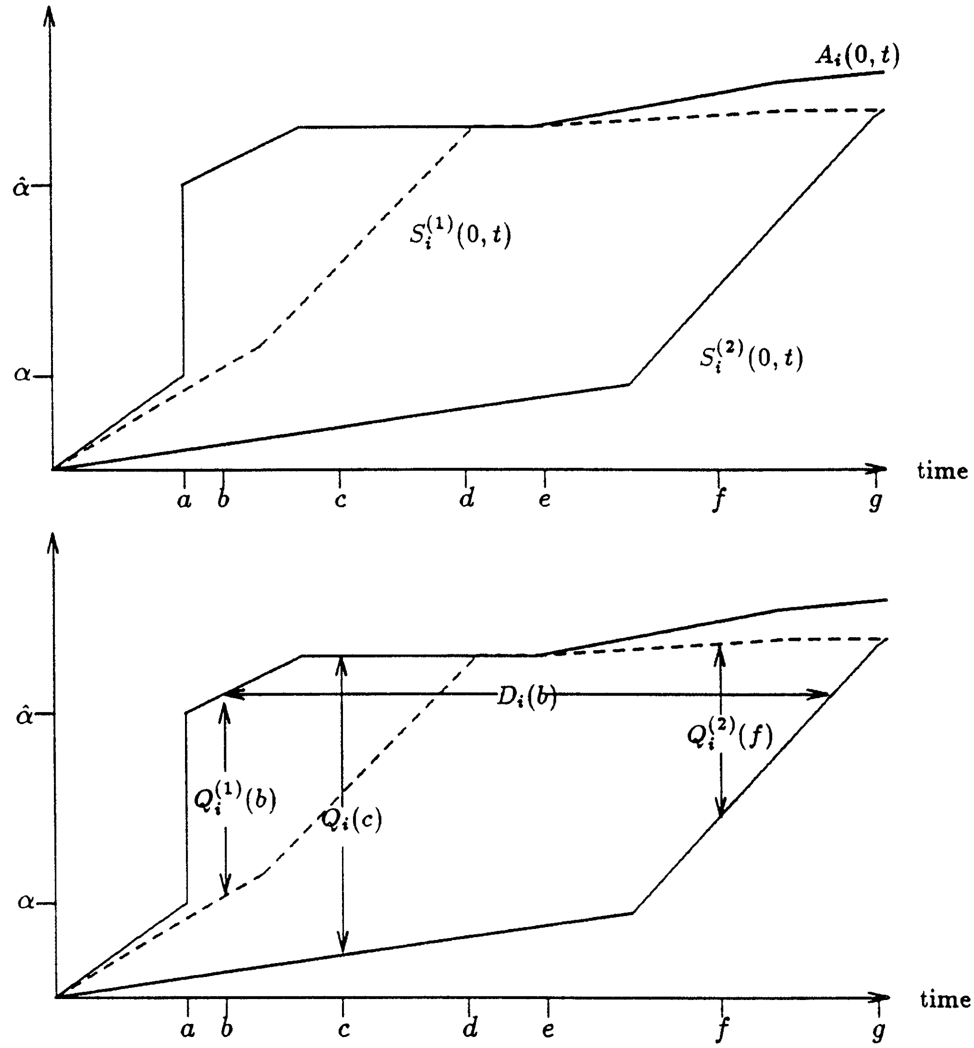


Figure 1: A four server network. The demultiplexer works instantaneously.



The first figure shows how session i traffic progresses through the nodes of its route. Notice that the arrival function to node 2 is the session i service function of node 1. The second figure shows how the backlog and delay can be measured and illustrates the definitions of Section 3.

Figure 2: An Example of Session i flow when $K_i = 2$.

The backlogs at every node in $P(i)$ can be determined from Figure 2 as shown. Define the maximum backlog for session i, Q_i^*:

$$Q_i^* = \max_{(A_1,\ldots,A_N)} \max_{\tau \geq 0} Q_i(\tau).$$

Note that A_i contains an impulse at time a; As in [6], we adopt the convention that the arrival functions are continuous from the left, so that $A_i(0, a) = \alpha$ and $A_i(0, a^+) = \hat{\alpha}$.

Define the utilization of server m to be

$$u^m = \frac{\sum_{j \in I(m)} \rho_j}{r^m}. \tag{8}$$

A network is defined to be *stable* if $D_i^* < \infty$ for all sessions i. In most of our analysis we will show stability under the assumption that $u^m < 1$ at every server m. Allowing utilizations of greater than 1 would permit backlogs and delays to build up unboundedly, and we have shown elsewhere ([5]) that permitting $u^m = 1$ at each server m can result in problems as well.

The minimum session i backlog clearing rate along its route is

$$g_i = \min_{m \in P(i)} g_i^m. \tag{9}$$

When $g_i > \rho_i$ we define session i to be *locally stable.* Note that if $\phi_i^m = \rho_i$ for all i and $m \in P(i)$ then each session i is locally stable.

Finally, the definitions of system and session busy periods given in [6] for a single node are extended to the multiple node case. A network system (session i) busy period is defined to be the maximal interval B (B_i) such that for every $\tau \in B$ ($\tau \in B_i$), there is at least one server in the network that is in a system (session i) busy period at time τ.

4 Bounds for Locally Stable Sessions

While every route in a data network is acyclic, the union of several routes may result in cycles being induced in the network topology. The presence of these cycles can complicate the analysis of delay considerably, but more importantly, it can lead to feedback effects that drive the system towards instability. This phenomenon has been noticed by researchers from fields as diverse as manufacturing systems [7, 4], communication systems [2] and VLSI circuit simulation [3]. Consider the four node example in Figure 1 (which is identical to Example 2 of Cruz [2]). Suppose the service discipline is FCFS. As an illustration of virtual feedback, notice that $S_0^{(1)}$ depends on the traffic from sessions $2, 3, \ldots, K-1$, but the form of this traffic is not independent of $S_0^{(1)}$.

In this section we will show that for a locally stable session, i, these virtual feedback effects are completely absent even when the other sessions are not leaky bucket constrained. For notational convenience let $P(i) = (1, 2, ..., K_i)$. The following useful Lemma is straightforward and stated without proof—to see that it is true, recall that we are ignoring propagation delays:

Lemma 1 *For every interval* $[\tau, t]$ *that is contained in a single session* i *network busy period:*

$$S_i^{(K_i)}(\tau, t) \geq g_i \ (t - \tau).$$

The Lemma leads us to the main result of this section:

Theorem 1 *If* $g_i \geq \rho_i$ *for session* i*:*

$$Q_i^* \leq \sigma_i,$$

$$D_i^* \leq \frac{\sigma_i}{g_i}.$$

Note that the delay bound in Theorem 1 is independent of the topology of the network and number of links in the route taken by the session. Also, it is independent of the σ_j, $j \neq i$.

Proof. Suppose Q_i^* is achieved at time t, and let τ be the first time before t when there are no session i bits backlogged in the network. Then by Lemma 1, $S_i^{(K_i)}(\tau, t) \geq \rho_i(t - \tau)$. Consequently,

$$Q_i^* \leq (\sigma_i + \rho_i(t - \tau)) - \rho_i(t - \tau) = \sigma_i.$$

An arriving session i bit will be served after at most Q_i^* session i bits have been served. Using Lemma 1 again, these backlogged bits are served at a rate of at least g_i. Therefore:

$$D_i^* \leq \frac{Q_i^*}{g_i} \leq \frac{\sigma_i}{g_i}.$$

□

Notice that the bounds are independent of K_i, the number of hops in the session i route. The naive bound on delay arrived at by adding the worst-case delays at each node is $D_i^* \leq \sigma_i \sum_{m=1}^{K_i} \frac{1}{g_i^m}$, illustrating the fact that much better bounds result from analyzing the session i route as a whole. When all of the sessions are leaky bucket constrained and $\phi_i^m = \rho_i$ at all m and $i \in I(m)$:

$$Q_i^* \leq \sigma_i, \tag{10}$$

and

$$D_i^* \leq \frac{\sigma_i}{\rho_i}. \tag{11}$$

However, note that given a locally stable session i, the result of Theorem 1 is valid for any GPS assignment for the other sessions. In fact, the other sessions need not be leaky bucket constrained, nor need the system be stable.

5 The Importance of Sessions that are not Locally Stable

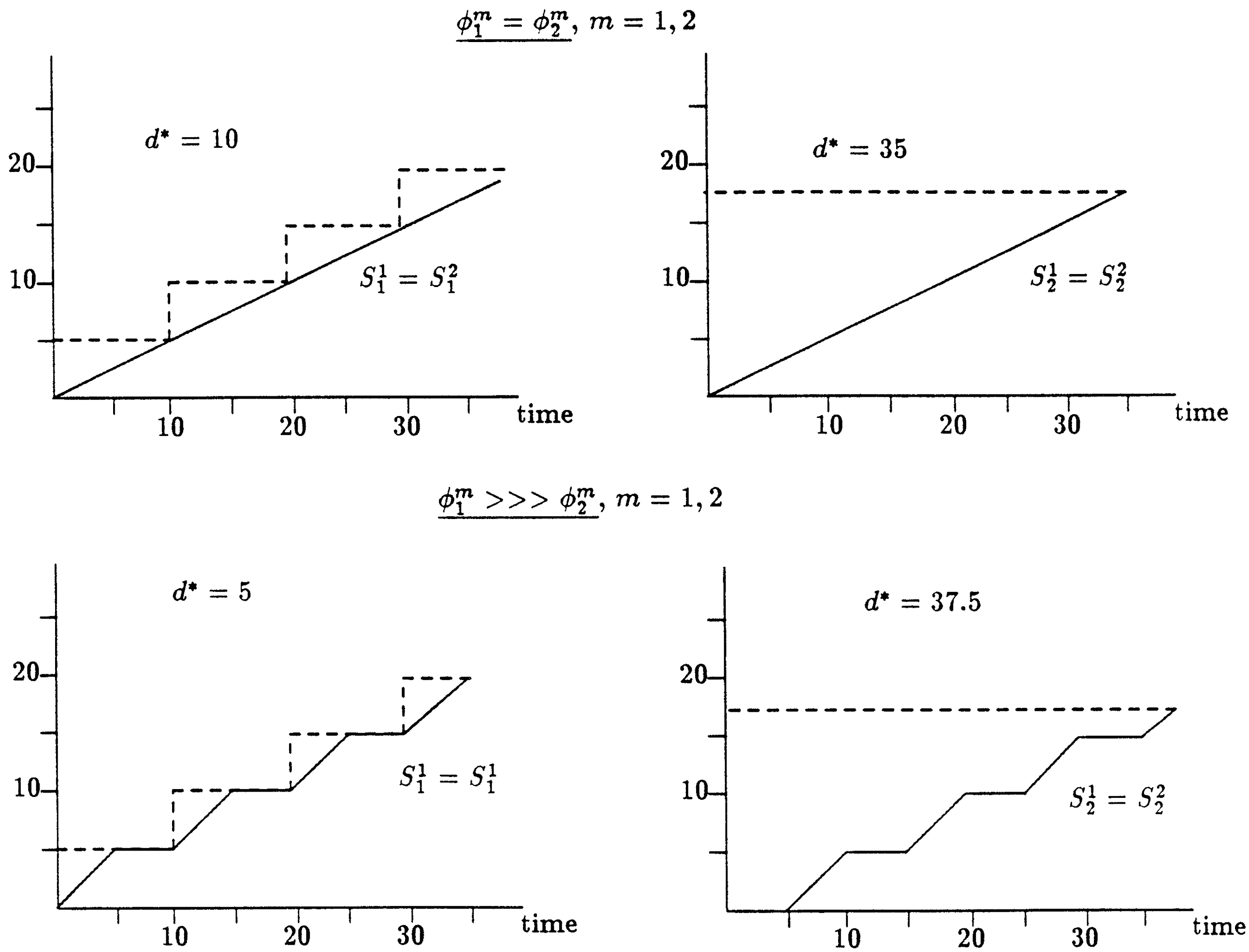
When **all** of the sessions are leaky bucket constrained, it is possible to guarantee finite delay even for the sessions that are not locally stable. This is because GPS is work conserving and the token arrival rates are assigned such that $\sum_{j \in I(m)} \rho_j < r^m$ at all nodes m. Thus we may allow g_i to be less than ρ_i for sessions that are not delay sensitive, and much greater than ρ_i for delay sensitive sessions.

To see why such assignments are important, consider the following example: There are two sessions in the network, and $P(i) = (1, 2)$ for $i = 1, 2$. As illustrated in Figure 3, Session 1 is more steady than Session 2. Notice that by giving session 1 a very large backlog clearing rate we can minimize its delay while degrading session 2 delay only slightly. Thus even when the backlog clearing rate for session 2 is much smaller than ρ_2 at each node, the session does not suffer much in terms of delay.

6 The All-Greedy Bound for a single node

The presence of sessions that are not locally stable complicates our analysis considerably; yet after performing the analysis we will see that the computation of per-session delay and backlog remains efficient and intuitive. There are two steps to providing worst case bounds on delay and backlog: The first consists of characterizing the internal traffic of the network so that at each node, m and $j \in I(m)$ we have σ_j^m such that $A_j^m \sim (\sigma_j^m, \rho_j)$. In the second step, the internal characterization is used to analyze the session i route for delay and backlog.

Central to our analytical technique is the concept of the all-greedy bound: We calculate *upper bounds* on the minimum value $\sigma_i^{m,out}$ such that $S_i^m \sim (\sigma_i^{m,out}, \rho_i)$. These upper bounds will be shown to be quite good for a wide variety of networks. Consider a particular node m. Suppose that for every $j \in I(m)$, we are given that $A_j^m \sim (\sigma_j^m, \rho_j)$. In [6] it was shown that the worst-case delay and backlog for session i (at node m) is each achieved when all the sessions $j \in I(m)$ are simultaneously greedy from time zero, the beginning of a system busy period. However, if two sessions j and p are both served by the same node, n, just before



The backlog clearing rate for the Session 2 is infinitesimal when $\phi_1^m >> \phi_2^m$, $m = 1, 2$. However, its delay is not increased significantly. The service curves S_j^1 and S_j^2 coincide for $j = 1, 2$ in this example since there are no other sessions in the network. Also note that we the traffic is perfectly pipelined and propagation delays have been ignored.

Figure 3: Giving delay sensitive, steady sessions large values of ϕ.

they contend for node m, then it may not be possible for both of them to be simultaneously greedy, as is required in the all-greedy regime. Thus, the achievable worst-case delay and backlog at node m may be less (but never more) than that calculated under the all-greedy regime.

In the rest of this paper we will make frequent use of the all-greedy bound, in order to simplify procedures for estimating D_i^* and Q_i^*. The following notation is useful in this regard:

We are given σ_j^m, ρ_j for each $j \in I(m)$, such that $\sum_{j \in I(m)} \rho_j < r^m$. Consider a fictitious system in which no traffic enters node m before time zero, and all the sessions at m are greedy starting at time zero. Denote $\hat{A}_i^m$ as the resulting session i arrival function for all $i \in I(m)$. Also denote $\hat{S}_i^m$ as the service function at node m. Recall from [6], that for $t > 0$, as long as $Q_i^m(t) > 0$, the function $\hat{S}_i^m(0,t)$ is piecewise linear and convex-$\cup$ in t. By using the techniques of [6] we can find the smallest value $\hat{\sigma}_i^{m,out}$ such that $\hat{S}_i^m \sim (\hat{\sigma}_i^{m,out}, \rho_i)$. From the discussion above,

$$\hat{\sigma}_i^{m,out} \geq \sigma_i^{m,out}. \tag{12}$$

Thus, we may bound the burstiness of S_i^m by $\hat{\sigma}_i^{m,out}$.

7 Non-Acyclic GPS networks under Consistent Relative Session Treatment

We begin by the following useful definition:

Definition. *Session j is said to impede a session i at a node m if*

$$\frac{\phi_i^m}{\phi_j^m} < \frac{\rho_i}{\rho_j}.$$

Note that for any two sessions, i and j, that contend for a node m, either session i impedes session j or vice-verse, unless $\frac{\phi_i^m}{\phi_j^m} = \frac{\rho_i}{\rho_j}$, in which case neither session impedes the other.

A Consistent Relative Session Treatment GPS assignment (CRST) is one for which there exists a strict ordering of the sessions such that for any two sessions i, j, if session i is less than session j in the ordering, then session i does not impede session j at any node of the network.

The class of assignments that are CRST is quite broad: For example, consider the special case of a CRST system for which

$$\phi_{ij} = \frac{\phi_i^m}{\phi_j^m}, \quad \forall m \text{ s.t. } i,j \in I(m). \tag{13}$$

Thus, whenever sessions i and j contend for service at a link, they are given the same relative treatment. Note that $\phi_{ij} = \frac{\phi_{ip}}{\phi_{jp}}$, where session p is in $I(i) \cap I(j)$. Such CRST systems are called Uniform Relative Session Treatment (URST) systems. Note that

- By normalizing the values of the ϕ_i^m's at each node m, we may equivalently define a URST system to be one in which for every session i, and node m that is on the session i route: $\phi_i = \phi_i^m$.
- Suppose $\phi_i = \rho_i$ for every session i. Then from (9) each session is locally stable. We call this special case of a URST system, Rate Proportional Processor Sharing (RPPS).

We will show that a CRST system is stable if $u^m < 1$ at each node, is stable, and will also provide an algorithm for characterizing the internal traffic for every session in a CRST system.

The sessions of any network with a CRST assignment can be partitioned into non-empty classes $H_1, ..., H_L$, such that the sessions in H_k are impeded only by those in H_l, $l < k$. If two sessions i, j, are in the same class their routes are either edge disjoint or

$$\frac{\phi_i^m}{\phi_j^m} = \frac{\rho_i}{\rho_j}$$

at every node, m, that is common to the routes of sessions i and j. Clearly, the sessions in H_1 are not impeded by *any* other session.

Lemma 2 *For any session $j \in H_1$:*

$$\rho_j < \frac{\phi_j^m}{\sum_{p \in I(m)} \phi_p^m} r^m \tag{14}$$

for all nodes $m \in P(j)$.

Proof. Consider a session $j \in H_1$, and suppose that its route includes the node m. Since $\sum_{j \in I(m)} \rho_j < r^m$, there must exist at least one session i, such that

$$\rho_i < \frac{\phi_i^m}{\sum_{p \in I(m)} \phi_p^m} r^m.$$

By definition, i cannot impede session j. Therefore:

$$\frac{\phi_j^m}{\phi_i^m} \geq \frac{\rho_j}{\rho_i} > \frac{\rho_j \sum_{p \in I(m)} \phi_p^m}{\phi_i^m r^m}$$

$$\Rightarrow \phi_j^m r^m > \rho_j \sum_{p \in I(m)} \phi_p^m.$$

Now the claim is proven by rearranging the terms. □

For $j \in H_1$, (14) shows that j's guaranteed backlog clearing rate exceeds ρ_j so that

$$\hat{\sigma}_j^{m,out} = \sigma_j.$$

Thus from (12):

$$\sigma_j^{m,out} \leq \sigma_j. \tag{15}$$

Lemma 2 enables us to upper bound the internal traffic of all the sessions in H_1. The following Lemma will be crucial to us in continuing the process to the sessions belonging to the higher indexed classes:

Lemma 3 *Suppose sessions i and j contend for a link m, and that session j does not impede session i. Then the value of $\hat{\sigma}_i^{m,out}$ is independent of the value of σ_j^m.*

Proof. We will first consider the case $\sigma_j^m = 0$; the case $\sigma_j^m > 0$ will follow from this easily: As we explained in [6], under an all-greedy regime, the service function, $\hat{S}_i^m$ is a continuous piece-wise linear convex-∪ functions, with break points corresponding to the times that individual session backlogs clear at node m under the all-greedy regime. Let this feasible ordering be $\mathcal{F}$ when $\sigma_j^m = 0$. Let q_i be the (least) time at which maximum backlog is achieved for session i under the all greedy regime, and let e_i^0 be the time that the session i backlog is cleared when $\sigma_j^m = 0$. Notice that $e_i^0 \geq q_i$. Similarly, let session j terminate its busy period at time e_j^0. (It is clear that if $\sigma_j^m > 0$ then the time at which this busy period would be terminated must be $\geq e_j^0$.) Now suppose that session i is less than session j in the feasible ordering, $\mathcal{F}$. Then

$$q_i < e_i^0 < e_j^0.$$

Since e_j^0 can only increase for positive values of σ_j^m, it follows that time time q_i is the same for all non-negative values of σ_j^m. Now recalling from Lemma 12 of [6] that

$$\hat{\sigma}_i^{m,out} = Q_i^* = Q_i^m(q_i), \tag{16}$$

we note that the value of σ_j^m does not influence the value of $\hat{\sigma}_i^{m,out}$. Thus the Lemma holds.

Now suppose that session j is less than session i under $\mathcal{F}$: Then

$$\hat{S}_j^m(0, e_j^0) = 0 + \rho_j e_j^0 \geq \frac{\rho_i \phi_j^m}{\phi_i^m} e_j^0.$$

Thus,

$$\hat{S}_i^m(0, e_j^0) = \frac{\phi_i^m \hat{S}_j^m(0, e_j^0)}{\phi_j^m} \geq \rho_i e_j^0, \tag{17}$$

and

$$Q_i^m(e_j^0) \leq \sigma_i.$$

Now since $Q_i^* \geq \sigma_i$, and since $Q_i(t)$ strictly increases in the interval $[0, q_i]$, it follows that $q_i \leq e_j^0$. Since the time at which the session j busy period terminates can only be greater than e_j^0 for arbitrary values of σ_j^m, we see that the maximum session i backlog is always achieved in the all-greedy regime before the session j busy period terminates. Since session j is in a busy period in the interval $[0, q_i]$, the value of q_i is independent of σ_j^m. Now from (16) we are done. □

Lemma 4 *Suppose session i is in $I(m)$ for some node k, and that for every session $j \in I(m)$ that can impede i, σ_j^m is bounded. Then $\sigma_i^{m,out}$ must be bounded as well.*

Proof. From Lemma 3 it follows that $\hat{\sigma}_i^{m,out}$ is bounded. Now from (12) we are done. □

Lemmas 3 and 4 can be used to sequentially characterize the internal traffic of the sessions in classes $H_2, H_3, ..., H_L$. The following procedure specifies the method.

- Compute $H_1, ..., H_L$.
- $k = 1$

  While $k \leq L$, for each session $i \in H_k$

  For $p = 1$ to K_i

  $m = P(i, p)$

  Compute $\hat{\sigma}_i^{m,out}$ given:

  $\sigma_j^m = \hat{\sigma}_j^m$ for all sessions j that impede i at m (computed in earlier steps).

  σ_i^m as computed earlier.

  $\sigma_j^m = 0$ for all sessions j that do not impede i at m.

  Set $\sigma_j^{(p)} = \hat{\sigma}_i^{m,out}$.

  $k := k + 1$

Now from (12) we have upper bounds to σ_i^m for every session i and node $m \in P(i)$. This procedure enables us to show that

Theorem 2 *A CRST GPS network is stable if $u^m < 1$ at each node m.*

8 Computing Delay and Backlog for Stable Systems with Known Internal Burstiness

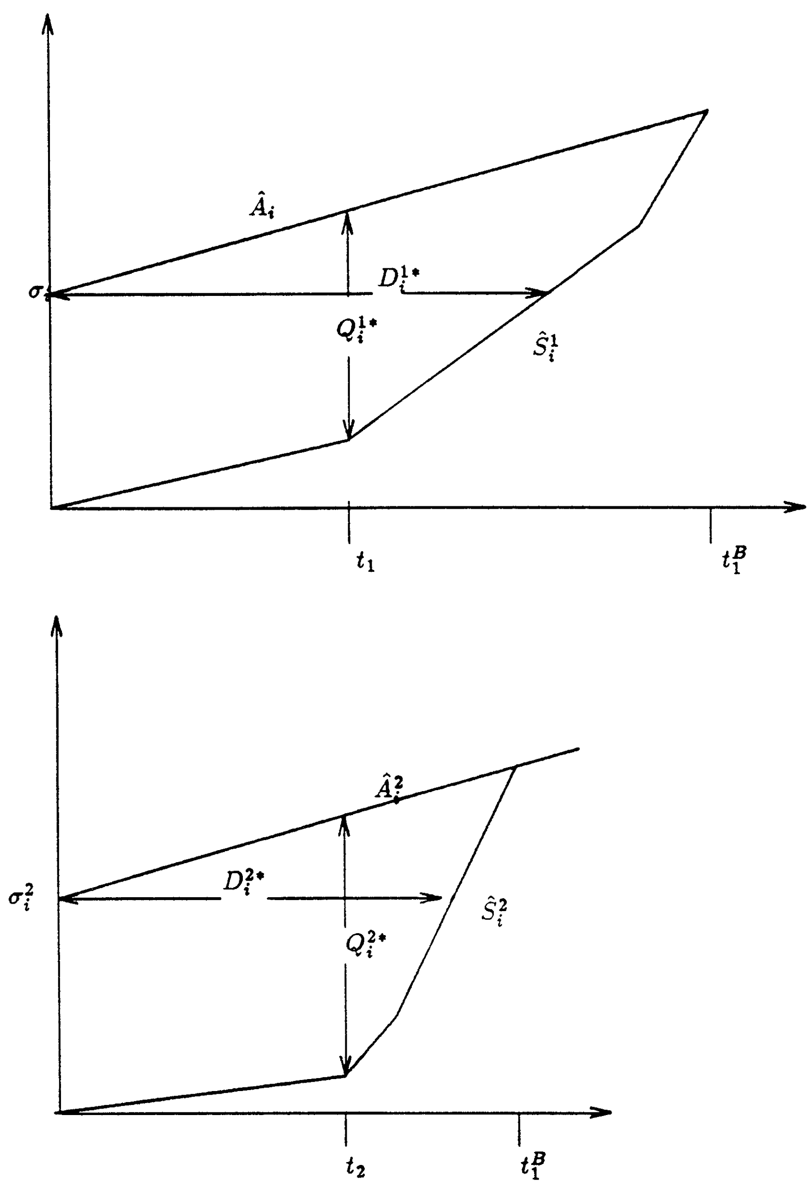
Suppose that we are given a stable GPS system in which the sessions are leaky bucket constrained as in (3), i.e., for every session j and node m such that $j \in I(m)$, we are given a value σ_j^m, such that $A_j^m \sim (\sigma_j^m, \rho_j)$. As we discussed in Section 6, worst case delay (backlog) at a single node of the network can be upper bounded by applying the techniques of [6] when the traffic characterization of sessions sharing that node is known. Under the Additive Method due to [2], we add the worst case bounds on delay (backlog) for session i at each of the nodes $m \in P(i)$ considered in isolation. While this approach works for any server discipline for which the single node can be analyzed, it may yield very loose bounds. For example, when applied to an RPPS system (defined in Section 7 we get $D_i^* \leq K_i \frac{\sigma_i}{\rho_i}$, rather than $D_i^* \leq \frac{\sigma_i}{\rho_i}$. The problem, of course, is that we are ignoring strong dependencies among the queueing systems at the nodes in $P(i)$. Figure 4 illustrates the Additive Method. In order to imporve the bounds the session route as a whole is treated as a whole. For notational simplicity we focus on a particular session, i, that follows the route $1, 2, ..., K$. Figure 5 illustrates the system to be analyzed. We will assume that:

1. The sessions $j \in I(m) - \{i\}$ (for $m = 1, 2, ..., K$) are free to send traffic in any manner as long as $A_j^m \sim (\sigma_j^m, \rho_j)$. Thus it is appropriate to call the sessions in $I(m) - \{i\}$, the *independent sessions* at node m ($m = 1, 2, .., K$).

2. Session i traffic is constrained to flow along its route so that

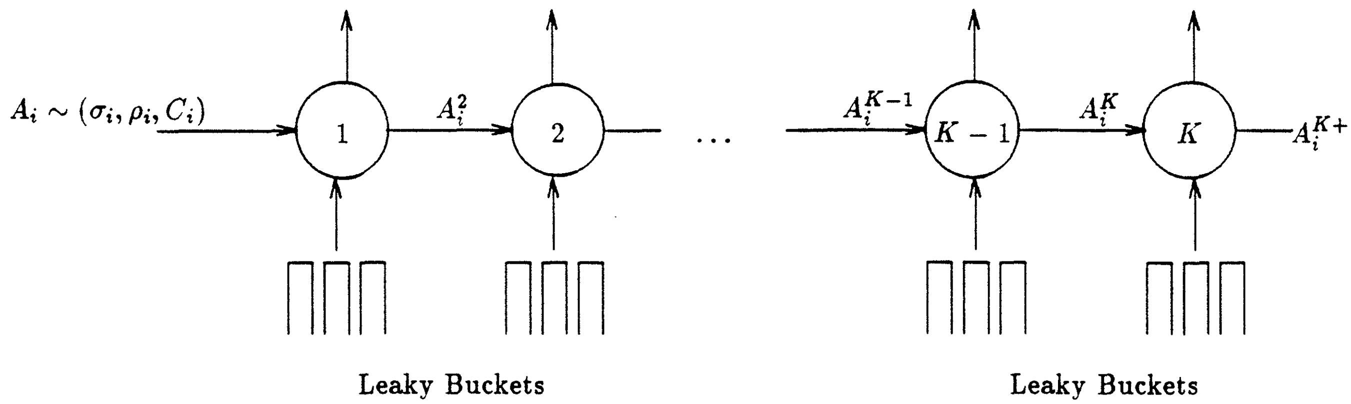
$$A_i^m = S_i^{m-1} \quad m = 2, 3, ..., K.$$

Assumptions 1 and 2 are collectively known as the <u>independent sessions relaxation</u>. This is because while the network topology may preclude certain arrival functions of A_j^k that are consistent with (σ_j^k, ρ_j), these functions are included under the independent sessions relaxation. On the other hand, every arrival function allowable in the network, is allowed under the independent sessions relaxation. Thus, the values of D_i^* and Q_i^* that hold under the independent sessions relaxation, must be upper bounds on the true values of these



Both the figures can be determined independently. The Additive Method yields bounds $Q_i^* \leq Q_i^{1*} + Q_i^{2*}$ and $D_i^* \leq D_i^{1*} + D_i^{2*}$.

Figure 4: The Additive Method for session i when $P(i) = \{1, 2\}$.



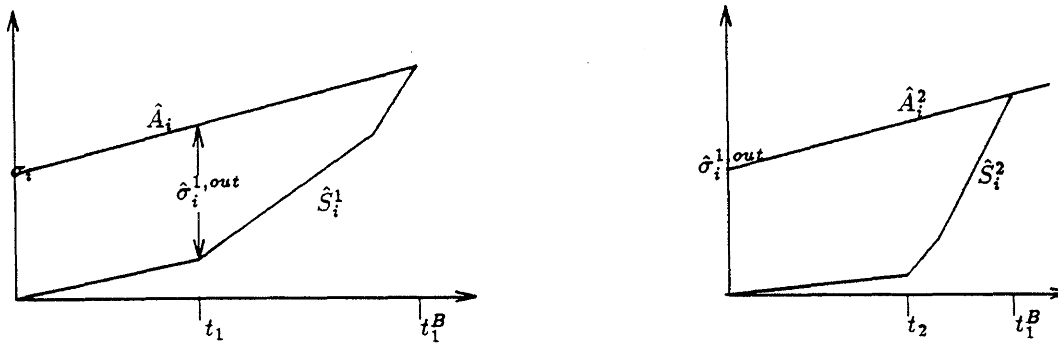
Session i traffic enters the network so that it is consistent with (σ_i, ρ_i), and $A_i^k = S_i^{k-1}$ for $k = 2, 3, ..., K$. The independent sessions at node k are free to send traffic in any manner as long as $A_j^k \sim (\sigma_j^k, \rho_j)$ for every session $j \in I(k) - \{i\}$, $k = 1, 2, ..., K$.

Figure 5: Analyzing the Session i route as a whole, under the Independent Sessions Relaxation.

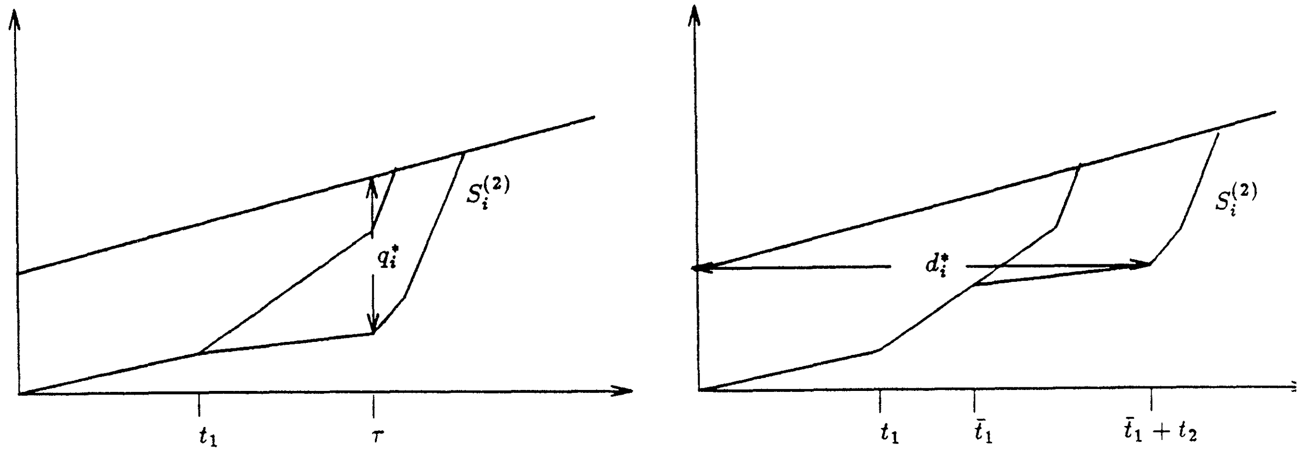
quantities. The use of all-greedy bounds enables us to compute D_i^* and Q_i^* exactly under the independence relaxation.

In view of our results for the single node case, it would be satisfying if maximum delay (and backlog) were achieved when all the sessions of the network are greedy starting at time zero (the beginning of a system busy period). However, this is generally not true. It turns out that what is required is that the sessions at a particular node j become greedy simultaneously, but only after the sessions at node $j-1$ become greedy. We call this pattern of arrivals a staggered greedy regime. The instants of time at which the sessions become greedy depend on the session for which maximum delay and backlog is being estimated. We will also find that Q_i^* and D_i^* may not both be achieved for the *same* staggered greedy regime. This important point is illustrated in Figure 6. The possibility of D_i^* and Q_i^* being achieved under different staggered greedy regimes is discouraging from a practical standpoint, especially if computing either one of these quantities involves solving a complicated optimization problem. It would be much more desirable to have a single function from which both delay and backlog can be bounded. (In the single-node case this curve is just $\hat{S}_i$, i.e. Lemma 10 of [6].)

In Section 8.1 we describe such a function, which we call the session i universal curve, $U_i(t)$. This curve is constructed without computing any staggered greedy regimes, and both D_i^* and Q_i^* can be determined efficiently and exactly from it (under the independent sessions relaxation). In addition, the staggered greedy regimes that achieve these worst-case values can also be efficiently determined from $U_i(t)$. In Section 8.2, we prove that these worst-case



(a)



(b)

The curves $\hat{S}_i^1$ and $\hat{S}_i^2$ are shown in (a). Note that $\sigma_i^2 = \hat{\sigma}_i^{1,out}$, and so $\hat{S}_i^1$ and $\hat{S}_i^2$ cannot be determined independently.

Figure (b) shows two staggered greedy regimes. In the first, the sessions in $I(2)-\{i\}$ become greedy at time t_1, which yields a maximum backlog of q_i^* at time τ. In the second staggered greedy regime, the sessions at $I(2) = \{i\}$ wait until time $\bar{t}_1$ to become greedy—this results in a maximum delay of d_i^* for session i at time zero.

Figure 6: Two Staggered Greedy Regimes when $P(i) = \{1, 2\}$

staggered greedy regime achieve the same bounds on D_i^* and Q_i^*, as computed from $U_i(t)$.

8.1 The Session i Universal Service Curve

For notational simplicity, we will focus on a session i such that $P(i) = (1, 2, ..., K)$. The functions $\hat{S}_i^1, ..., \hat{S}_i^K$ can be computed using the internal traffic characterization of Section 7 by using the independent sessions relaxation. Recall that for each node $m = 1, 2, ..., K$, $\hat{S}_i^m$ is continuous, piece-wise linear and is convex-$\cup$ in the range $[0, t_m^B]$, where t_m^B is the duration of the session i busy period at m under the all-greedy regime. Also $\hat{S}_i^m(0) = 0$. Thus it can be specified (in the range $[0, t_m^B]$) by a list of pairs:

$$(s_1^m, d_1^m), (s_2^m, d_2^m), \ldots, (s_{n_m}^m, d_{n_m}^m),$$

where s_j^m is the slope of the j^{th} line segment and d_j^m is its duration. Here

$$s_1^m < s_2^m < \ldots < s_{n_m}^m, \tag{18}$$

and

$$\sum_{j=1}^{n_m} d_j^m = t_m^B. \tag{19}$$

We first describe how to construct U_i from $\hat{S}_i^1, ..., \hat{S}_i^K$, and then define the curve analytically. Finally, we establish the relationship between U_i and the session i departures from the network, $S_i^{(K)}$:

Let E_i^k be the collection of all the pairs (s_j^m, d_j^m) for $m = 1, 2, ..., k$—i.e.

$$E_i^k = \bigcup_{m=1}^{k} \bigcup_{j=1}^{n_m} \{(s_j^m, d_j^m)\}.$$

The session i universal service curve, U_i is defined as:

$$U_i(t) = \min\{G_i^K(t), \hat{A}_i(0, t)\},$$

where the curve G_i^k (for $k = 1, 2, ..., K$) is a continuous curve constructed from the elements of E_i^k as follows:

1. Set $G_i^k(0) = 0$, Remaining-in-E $= E_i^k$; $Glist = \phi$; $u = 0$; $t = 0$.
2. Order the elements of E_i^k in increasing order of slope. Remove from Remaining-in-E an element of smallest slope: $e^{new} = (s^{new}, d^{new})$.

Append Glist with e^{new}. If Remaining-in-E is not empty then repeat step 2.

3. G_i^k is a piece-wise linear convex-∪ defined in the range $[0, \sum_{m=1}^{k} t_m^B]$ by the elements of $Glist$[5].
   For $t \geq \sum_{m=1}^{k} t_m^B$ set

$$G_i^k(t) = G_i^k(\sum_{m=1}^{k} t_m^B) + \hat{A}_i(\sum_{m=1}^{k} t_m^B, t). \tag{20}$$

Figure 8.1 illustrates the construction of U_i for a simple two node example. Note that:

- G_i^k is defined for $k = 1, 2, ..., K$, but U_i is defined in terms of G_i^K.
- For each m, the relative of order of the elements from $\hat{S}_i^m$ is preserved in $Glist$.
- We still have to show that for any network, the curve G_i^K always meets $\hat{A}_i$—this is established in Lemma 5.

Describing the construction of G_i^K is useful in understanding its form, but we need an analytical definition of the curve in order to prove things about it. The following is a useful, notationally compact definition for times t in the range $[0, \sum_{m=1}^{k} t_m^B]$:

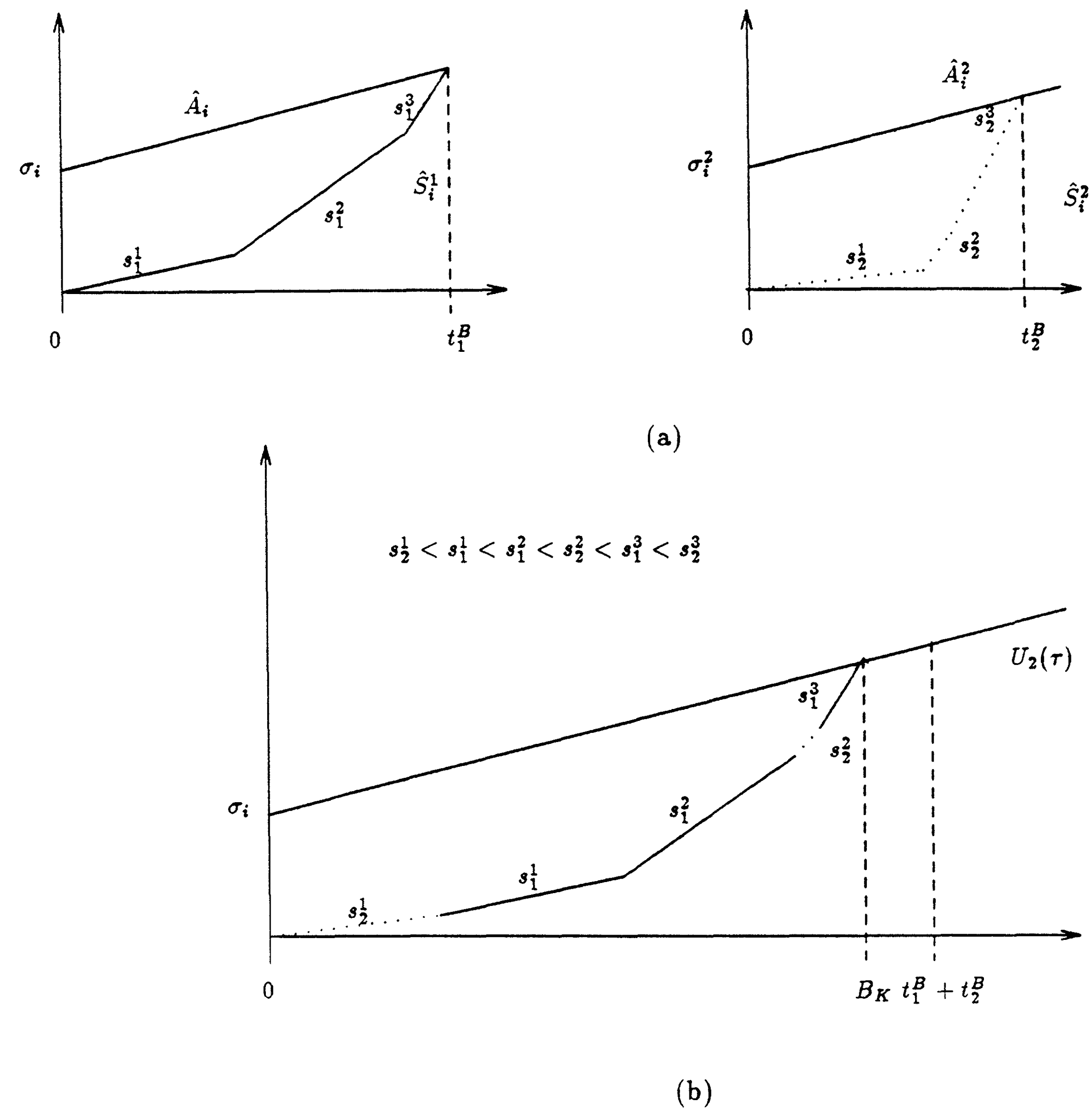
$$G_i^k(t) = \begin{cases} \hat{S}_i^1(0,t), & \text{for } k = 1 \\ \min_{\tau \in [0,t]}\{G_i^{k-1}(\tau) + \hat{S}_i^k(0, t-\tau)\}, \quad t - \tau \leq t_k^B, & \text{for } k \geq 2. \end{cases} \tag{21}$$

To see how (21) corresponds to the algorithm given earlier, expand the recursion in terms of $\tau_1, ..., \tau_k$ where τ_m, corresponds to the minimizing value for node m. Clearly, $\tau_1 = 0$, and define $\tau_{k+1} = t$. Then $\tau_{m+1} - \tau_m \leq t_m^B$ for each $m = 1, 2, .., k$ and

$$\begin{aligned} G_i^k(t) &= \min_{\tau_k \in [0,t]} \min_{\tau_{k-1} \in [0,\tau_k]} \cdots \min_{\tau_2 \in [0,\tau_3]} \left\{ \sum_{m=1}^{k} \hat{S}_i^m(0, \tau_{m+1} - \tau_m) \right\} \\ &= \min_{0 \leq \tau_2 \leq \tau_3 \leq \cdots \tau_K \leq t} \sum_{m=1}^{k} \hat{S}_i^k(0, \tau_{m+1} - \tau_m). \end{aligned} \tag{22}$$

For each m, the quantity $\tau_{m+1} - \tau_m$ corresponds to the total duration of the elements picked for $Glist$ from the list describing $\hat{S}_i^k$. Suppose we are given $G_i^1 = \hat{S}_i^1(0,t)$, and wish to compute $G_i^2(t)$ for some $t \in [0, \sum_{m=1}^{2} t_m^B]$. Applying the algorithm to the construction of G_i^2, we determine $\hat{\tau}$, the duration of the elements picked from the list describing $\hat{S}_i^1$. Then $\hat{\tau}$ corresponds to the minimizing value of τ in (21). Thus $G_i^k(t)$ is the curve described by $Glist$. Note that the minimizing values of $\tau_2, ..., \tau_k$ are functions of t.

[5] In the same manner as $\hat{S}_i^m$ was specified earlier.



The two service curves in (a) show $\hat{S}_i^1$ and $\hat{S}_i^2$. In (b) the line segments that make up these curves are concatenated to make a piece-wise linear convex curve that meets $\hat{A}_i$ at time B_K. Thus

$$U_i(t) = \begin{cases} G_i^K(t) & t \leq B_K \\ \hat{A}_i(0,t) & t > B_K. \end{cases}$$

Note that the line segment with slope s_2^3 is never used in the construction of U_2, i.e. $T_2 < t_1^B + t_2^B$.

Figure 7: An example of how U_i is constructed for $K = 2$.

In the next Lemma we show that $G_i^k(t)$ must meet $\hat{A}_i(0,t)$ at some time before $\sum_{m=1}^{k} t_m^B$:

Lemma 5

$$G_i^k(\sum_{m=1}^{k} t_m^B) \geq \hat{A}_i(0, \sum_{m=1}^{k} t_m^B).$$

Proof. Let $\tau_1, ..., \tau_{k+1}$ be the minimizing values of (22.)

By definition:

$$\tau_{m+1} - \tau_m \leq t_m^B.$$

for each $m = 1, 2, ..., k$. For $t = \sum_{m=1}^{k} t_m^B$ we must have equality in each of these K inequalities. Thus

$$\hat{S}_i^m(0, \tau_{m+1} - \tau_m) = \hat{S}_i^m(0, t_m^B) = \hat{A}_i(0, t_m^B)$$

(where the second equality follows from the definition of t_m^B), and

$$G_i^k(\sum_{m=1}^{k} t_m^B) = \sum_{m=1}^{k} \hat{S}_i^m(0, t_m^B) = \sum_{m=1}^{k} \hat{A}_i(0, t_m^B) \geq \hat{A}_i(0, \sum_{m=1}^{k} t_m^B).$$

□

Now observe from (20) that for any $t \geq \sum_{m=1}^{k} t_m^B$ we must have:

$$G_i^k(t) \geq \hat{A}_i(0,t). \tag{23}$$

Then there exists $B_k \leq \sum_{m=1}^{K} t_m^B$ such that

$$\begin{aligned} G_i^k(t) &< \hat{A}_i(0,t), \quad t < B_k \\ &= \hat{A}_i(0,t), \quad t = B_k \\ &\geq \hat{A}_i(0,t), \quad t \geq B_k. \end{aligned} \tag{24}$$

Thus,

$$U_i(t) = \begin{cases} G_i^K(t) & t \leq B_K \\ \hat{A}_i(0,t) & t > B_K. \end{cases} \tag{25}$$

Having defined U_i, we now relate it to the session i departures from the network. First, we state two important results that are crucial to the analysis that follows. Lemma 6 establishes that if the independent sessions at a node m are greedy from time zero, then as long as session i remains busy in an interval $[0, \tau]$, the function S_i^m will be identical to $\hat{S}_i^m$ in this interval. Thus session i does not have to be greedy, just busy during the interval. Lemma 7 states that if the independent sessions at a node m are quiet during the interval $[0, \tau]$ and then are greedy starting at τ, then this behavior minimizes $S_i^m(\tau, t)$, the amount of service received by session i at node m from time τ on. The proofs of these lemmas follow almost directly from our work in [6].

Lemma 6 *Suppose the independent sessions relaxation holds, and that t is contained in a session i busy period at node m that begins at time 0. Also, suppose that none of the independent sessions have sent any traffic before time 0, and that each is greedy starting at time zero. Then S_i^m is identical to $\hat{S}_i^m$ in the range $[0, t]$.*

Lemma 7 *Suppose the independent sessions relaxation holds, and that time t is contained in a session i busy period at server m that starts at time $\tau \leq t$. Then for all $t \geq \tau$, $S_i^m(\tau, t)$ is minimized over all arrival functions when for every independent session p at node m:*

1. $A_p^m(0, \tau) = 0$.

2. *Session p is greedy from time τ.*

Proof. When the independent sessions behave according to conditions 1 and 2 of the Lemma:

$$S_i^m(\tau, t) = \hat{S}_i^m(0, t - \tau)$$

from Lemma 6. Now using Lemma 10 of [6] we are done. □

In the next Lemma we establish the relationship between S_i^m and G_i^m:

Lemma 8 *Consider a given arrival function, A_i, and a given time τ such that $Q_i(\tau) = 0$. Then for each m, $1 \leq m \leq K$, each $t > \tau$:*

$$S_i^m(\tau, t) \geq \min_{V \in [\tau, t]} \{A_i(\tau, V) + G_i^m(t - V)\}. \tag{26}$$

Proof. See Appendix A. □

In the next section we are going to show that $G_i^m(t)$ is the amount of service given to session i under a specific staggered greedy regime called the (m, t)-staggered greedy regime. Thus Lemma 8 shows that the service to session i is minimized when a such a staggered greedy regime is delayed by an appropriate amount, which is the minimizing value of V. Equation (26) facilitates the following bounds on delay and backlog:

Theorem 3 *For every session i:*

$$Q_i^* \leq \max_{\tau \geq 0} \{\hat{A}_i(0, \tau) - G_i^K(\tau)\}, \tag{27}$$

and

$$D_i^* \leq \max_{\tau \geq 0} \left\{ \min\{t : G_i^k(t) = \hat{A}_i(0, \tau)\} - \tau \right\}. \tag{28}$$

Proof. See Appendix A. □

The inequalities (27) and (28) illustrate the importance of the universal curve. To find the bound on D_i^* compute the maximum horizontal distance between the curves $A_i(0,t)$ and $U_i(t)$ at the ordinate value of $\hat{A}_i(0,t)$. Similarly, Q_i^* is bounded by the maximum vertical distance between the two curves. In the next section, we will show that these bounds are *achieved* for (K,t)-staggered greedy regimes under the independent sessions relaxation.

8.2 The (K,t)-Staggered Greedy Regime

In this section we make clear the relationship between staggered greedy regimes and the session i universal curve U_i. As in the previous sections, we will focus on staggered greedy regimes with respect to a session i and assume that $P(i) = \{1,2,...,K\}$.

Any staggered greedy regime can be characterized by a vector

$$(T_1,...,T_K), \quad T_1 \leq T_2 \leq \ldots \leq T_K$$

such that all the sessions at node 1 are simultaneously greedy starting at time T_1, and the independent sessions at node j do not send any traffic in the interval $[T_1, T_j)$, but are simultaneously greedy starting at time T_j. Observe that the first staggered greedy regime in Figure 6(b) can be characterized by $(0, t_1)$ and the second by $(0, \bar{t}_1)$.

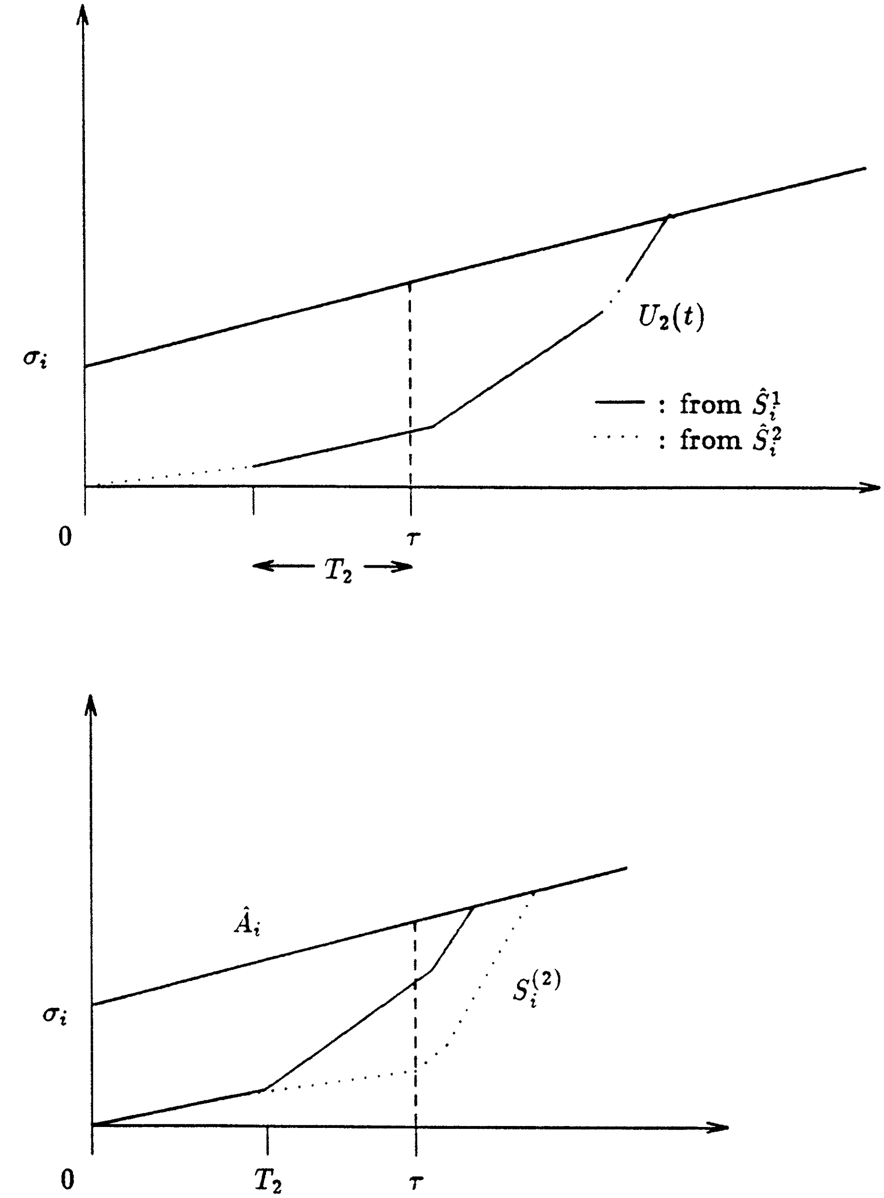
A $\underline{(K,t)\text{-staggered greedy regime}}$, $t \leq B_K$, is the staggered greedy regime characterized by $(0, T_2, ..., T_K)$ such that

$$\sum_{k=1}^{K} \hat{S}_i^k(0, T_{k+1} - T_k) = G_i^K(t) \tag{29}$$

where $T_1 = 0$, $T_{K+1} = t$ and $T_{k+1} - T_k \leq t_k^B$ for $k = 1,2,...,K$.
Note that

- Since $t \leq B_K$, $G_i^K(t) = U_i(t)$.
- For each $k = 1,2,...,K-1$ the staggered greedy regime defined by $(0, T_2, ..., T_k)$ describes a (k, T_{k+1})-staggered greedy regime.

Comparing (29) with (22) it is clear that $(T_2, ..., T_K)$ is a minimizing vector in (22). Thus, the universal service curve can be used to to determine $T_2, ..., T_K$. This is illustrated in Figure 8 for the simple case of $K = 2$. Notice from the figure that in the range $[0, T_2]$, S_i^2 is comprised of the line segments belonging to $\hat{S}_i^1$ that make up the universal curve in the



The top figure the curve U_2 that was constructed from $\hat{S}_i^1$ and $\hat{S}_i^2$. In order to find the $(2, \tau)$-staggered greedy regime, add the durations of the line segments taken from $\hat{S}_1$ that are in $U_2(t)$, $t \leq \tau$. This sum is T_2, the time that the independent sessions at node 2 become greedy. This characterizes the staggered greedy regime which is shown in the bottom figure.

Figure 8: Computing a (k, t)-Staggered Greedy Regime when $P(i) = \{1, 2\}$

range $[0, t]$. Also,

$$S_i^2(0, \tau) = U_i(\tau).$$

It turns out that this is true in general:

Theorem 4 *For any (K, t)-staggered greedy regime:*

$$S_i^{(K)}(0, t) = G_i^K(t).$$

Proof. See Appendix B. □

Figure 9 shows how to construct the staggered greedy regimes that maximize backlog and delay. From Theorems 3 and 4 we have the main theorem of this section:

Theorem 5 *Under the independent sessions relaxation, D_i^* and Q_i^* are each achieved under (K, t)-staggered greedy regimes.*

Now since the values of D_i^* and Q_i^* achieved under the independent sessions relaxation are upper bounds to the actual values of these quantities, we have shown how to find upper bounds on session backlog and delay. Also, since an infinite capacity link can always simulate a finite capacity link, worst case session i backlog and delay calculated under this relaxation must upper bound the values of these quantities for finite capacity links.

9 Propagation Delay

It is easy to incorporate deterministic propagation delays into our network framework: Suppose that every bit transmitted on link (i, j), incurs a delay of $d_{l,m}$ time units. Then each link acts as a constant delay element, and the characterization of internal traffic (using the method of Section 7) remains the same. A natural modification of the independent sessions relaxation allows us to bound end-to-end delay as well: Consider a session i such that $P(i) = 1, 2, ..., K$: Also, let $d_{0,1}$ be the propagation delay on the access link. Then

1. The independent sessions at node m, $j \in I(m) - \{i\}$ (for $m = 1, 2, ..., K$) are free to send traffic in any manner as long as $A_j^m \sim (\sigma_j^m, \rho_j)$.

2. Session i traffic is constrained to flow along its route so that

$$A_i^m(\tau, t) = S_i^{m-1}(\tau - d_{m-1,m}, t - d_{m-1,m}) \qquad m = 2, 3, ..., K.$$

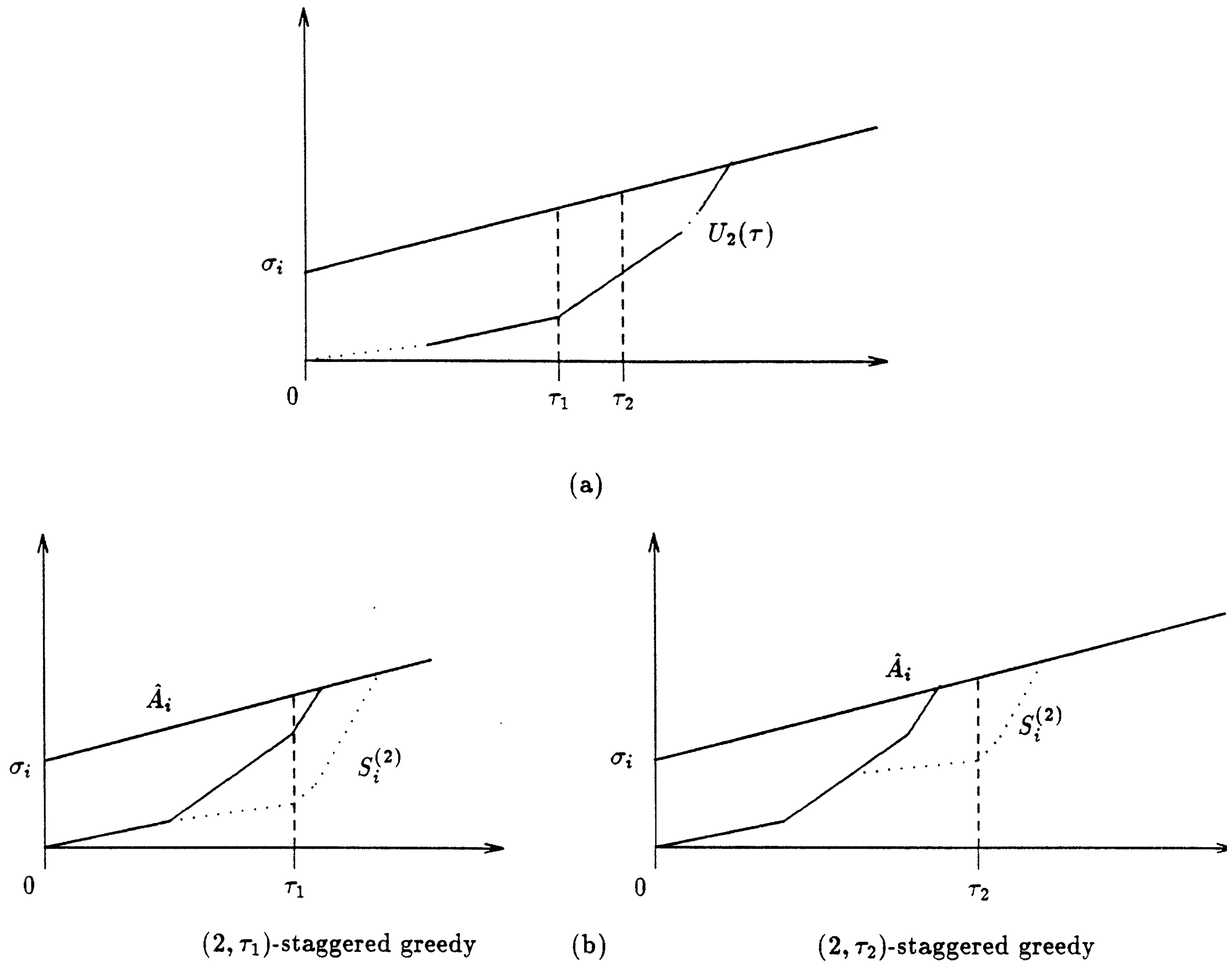


Figure (a) shows the session i universal curve. Notice that for this curve "backlog" is maximized at time τ_1 and "delay" is maximized at time τ_2. Figure (b) shows the two staggered greedy regimes corresponding to these times. Notice that the backlog at time τ_1 in the first regime is exactly equal to the "backlog" at time τ_1 in (a), and similarly the delay at time τ_2 in the second regime is exactly equal to the "delay" at that time in (b).

Figure 9: The staggered greedy regimes that maximize backlog and delay under the independent sessions relaxation.

In view of the analysis of Section 8:

$$D_i^* \leq \sum_{m=1}^{K} d_{m-1,m} + D_i^{*,\text{noprop}},$$

where $D_i^{*,\text{noprop}}$ is the the worst-case session i delay computed for the same characterization of internal traffic when propagation delays are zero. The number of bits in "flight" on a link (l,m) is at most

$$q_{l,m} = r_l d_{l,m}. \tag{30}$$

Thus

$$Q_i^* \leq \sum_{m=1}^{K} q_{m-1,m} + Q_i^{*,\text{noprop}}.$$

10 GPS Networks with Non-negligible Packet Sizes

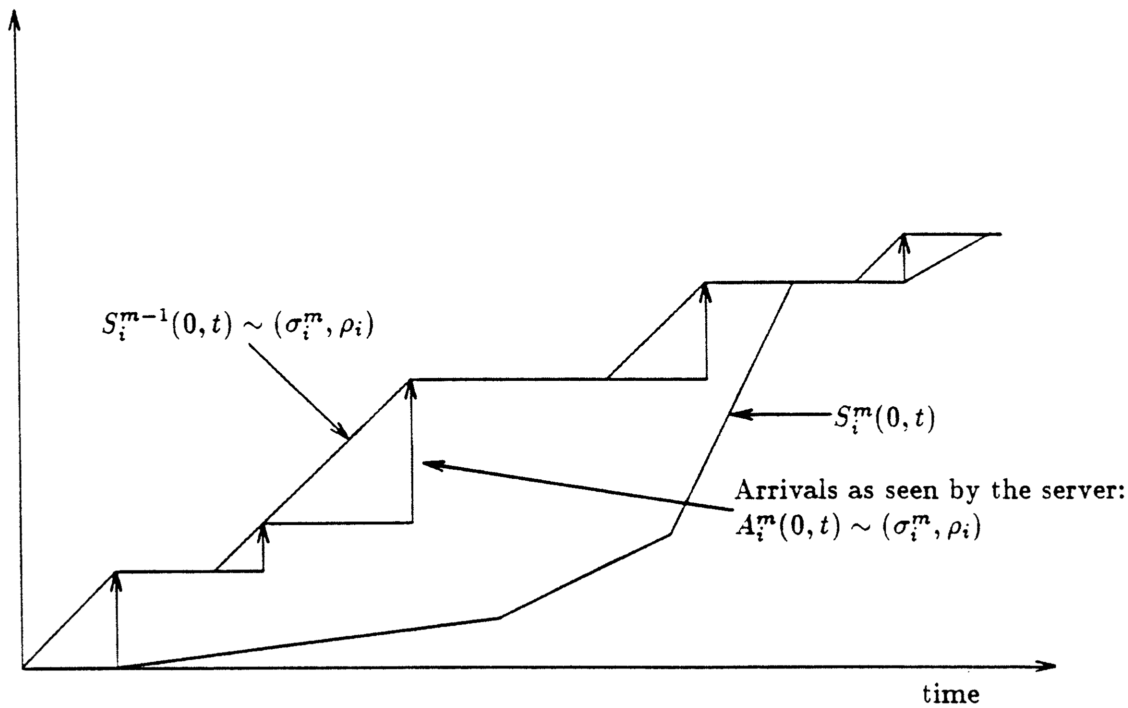
The analysis in Section 8 dealt with GPS networks in which the packet lengths are negligible,i.e., the traffic is assumed to be perfectly pipelined throughout the network. However, for most networks, particularly those with heterogeneous link speeds, packets are not transmitted until they have *completely arrived.* Thus, if $m-1$ and m are successive nodes on a session i's route, we cannot assume, as we did in Section 8, that $S_i^{m-1} = A_i^m$. In fact, for $P(i) = \{1, 2, ..., K_i\}$:

$$S_i^{m-1}(0,t) \geq A_i^m(0,t) \geq S_i^{m-1}(0,t) - L_i, \quad m = 2, ..., K_i,$$

where $L_i \leq \sigma_i$ is the maximum packet size for each session i. This important difference notwithstanding, we will still find the results for $L_i = 0$, to be very useful in the more general case of non-negligible packet sizes.

Since the GPS server does not begin serving a packet until its last bit has arrived, it "sees" the arrivals as a series of impulses, such that the height of each impulse is at most L_i. However, since we are not assuming any peak rate constraint in the input characterizations, A_i^m, is still consistent with (σ_i^m, ρ_i). (see Figure 10). Similarly, the arrivals seen by the server from every other session j at node m are consistent with (σ_j^m, ρ_j). Thus, the results of [6] can be applied to bound worst-case delay and backlog.

To analyze networks of such GPS servers, we follow the same steps as we did in Sections 7 and 8—we first characterize the internal traffic in terms of leaky bucket parameters, and then bound the worst-case delay and backlog for each session by analyzing its route as a whole.



$A_i^m(0,t)$ represents the cumulative arrivals seen by server, m. The length of each impulse of $\mathcal{A}_i(0,t)$ is bounded by L_i, the maximum packet size for session i. Since $L_i \leq \sigma_i^m$, it can be seen from the figure that $A_i^m \sim (\sigma_i^m, \rho_i)$.

Figure 10: A GPS Server when the packet sizes are non-negligible.

To incorporate the effects of finite packet lengths we stipulate that for $P(i) = \{1, 2, ..., K_i\}$,

$$S_i^{m-1}(\tau, t) \geq A_i^m(\tau, t) \geq S_i^{m-1}(\tau, t) - L_i, \quad m = 2, ..., K_i, \quad \tau < t. \tag{31}$$

Consider a GPS network with CRST assignments. The internal traffic can be characterized using the same procedure as in 7 to compute the all-greedy bounds.

To analyze the session i route given internal characterization of the traffic, we proceed as follows: Define $\hat{S}_i^m$ to be the session i output at node m under the all-greedy regime. Then the session i universal service curve is computed as it was in Section 8. Note that Lemma 6 also holds.

However, Lemma 8 and Theorem 3 must be modified in order to incorporate (31).

In what follows we assume (for notational simplicity) that $P(i) = \{1, 2, ..., K\}$:

Lemma 9 *Consider some time τ such that $Q_i(\tau) = 0$. Then for each m, $1 \leq m \leq K$, each $t > \tau$:*

$$S_i^m(\tau, t) \geq \min_{V \in [\tau, t]} \{A_i(\tau, V) + G_i^m(0, t - V)\} - mL_i. \tag{32}$$

Proof. See Appendix C. □

Theorem 6 *For every session i:*

$$Q_i^* \leq \max_{\tau \geq 0} \{\hat{A}_i(0, \tau) - G_i^K(\tau)\} + KL_i. \tag{33}$$

Proof. See Appendix C. □

Having bounded the worst-case backlog, we turn to delay. Here we need a slight refinement of Lemma 8, the details of which are in [5]. Essentially, we restrict our values of t to be such that a session i packet, p_m, departs node m at time t. Given such a time t, let the corresponding packet arrive at time a_m, i.e.,

$$A_i(0, a_m) = S_i^m(0, t). \tag{34}$$

Note that a_m is also the time that packet arrives at node 1, i.e.,

$$A_i^1(0, a_m) = A_i(0, a_m) \tag{35}$$

We show in [5] that for each node m, $1 \leq m \leq K$:

$$S_i^m(\tau, t) \geq \min_{V \in [\tau, a_m]} \{A_i(\tau, V) + G_i^m(0, t - V)\} - (m - 1)L_i. \tag{36}$$

Upon establishing (36), we can prove the following:

Theorem 7 *For every session i, define D_i^* to be the maximum session i packet delay. Then*

$$D_i^* \leq \max_{\tau \geq 0} \left\{ \min\{t : G_i(t) = \hat{A}_i(0,\tau) + (K-1)L_i\} - \tau \right\}. \tag{37}$$

Proof. See Appendix C. □

Theorems 6 and 7 allow us to bound D_i^* and Q_i^* in terms of the universal service curve.

11 PGPS networks

When packet sizes are small so that maximum packet transmission time at any link of the network is negligible, we may conclude from Theorem 2 of [6], that the behavior of GPS and PGPS are (essentially) identical. Thus in this case, all of the bounds for GPS networks in Sections 7 and 8 apply to PGPS networks as well.

We now consider the more general case in which packet sizes are not negligible, and outline how the results of Section 10 can be extended to this case:

11.1 Characterizing the Internal Traffic

Suppose we are given a network of PGPS servers such that the assignments of the ϕ_i's meet the CRST requirements of Section 7. Recall that a CRST assignment ensures a partition the sessions into classes $H_1, H_2, \ldots$ such that a no session in class c may only be impeded by sessions belonging to classes indexed lower than c. Consider a session $j \in H(1)$ and let $P(j) = \{1, 2, \ldots, K_j\}$. We know from Corollary 1 of [6] that

$$\hat{Q}_j^1(\tau) - Q_j^1(\tau) \leq L_{\max}$$

for all τ where $\hat{Q}_j^m$, and Q_j^m represent the session i backlogs at node m, under PGPS and GPS respectively. Thus

$$\hat{Q}_j^{1,*} \leq Q_j^{1,*} + L_{\max}.$$

Also, from Lemma 12 of [6]:

$$\sigma_j^{\text{out}} = Q_j^*.$$

Since $S_j^1 \sim (\sigma_j, \rho_j)$ under GPS, it follows that $\sigma_j^{out,1} \leq \sigma_j + L_{\max}$ under PGPS. Similarly, we can apply a simple modification of the procedure in Section 7 to characterize the internal traffic at each node in $P(j)$:

- Compute $H_1, ..., H_L$.
- $k = 1$

  While $k \leq L$, for every session $i \in H_k$

  For $p = 1$ to K_i

  $m = P(i,p)$

  Compute $\hat{\sigma}_i^{m,out}$ using the all-greedy bound given:

  $\sigma_j^m = \hat{\sigma}_j^m$ for all sessions j that impede i at m (computed in earlier steps).

  σ_i^m as computed earlier.

  $\sigma_j^m = 0$ for all sessions j that do not impede i at m.

  Set $\sigma_i^{(p)} = \hat{\sigma}_i^{m,out} + L_{\max}$.

  $k := k + 1$

11.2 Analyzing Delay along the Session i Route

In Appendix D we prove the following Theorem that allows us to relate worst-case session delay in a PGPS network to the universal service curve, and consequently to GPS networks.

Theorem 8 *For each session i:*

$$D_i^{*,\mathrm{PGPS}} \leq \max_{\tau \geq 0} \left\{ \min\{t : G_i^K(t) = \hat{A}_i(0,\tau) + (K-1)L_{\max}\} - \tau \right\} + \sum_{m=1}^{K} \frac{L_{\max}}{r^m}. \tag{38}$$

where the universal service curve, G_i^K is computed using the algorithm of Section 11.1.

Also note that as the link speeds become faster, i.e., as $r^m \to \infty$,

$$D_i^{*,\mathrm{PGPS}} = D_i^{*,\mathrm{GPS}}.$$

11.3 Rate Proportional Processor Sharing Networks

In this section we will interpret the results of the previous section for a special CRST assignment. Under RPPS Networks $\phi_i^m = \rho_i$ for every session i and $m \in I(m)$. Recall that in Section 4 we analyzed RPPS networks when the packet sizes are negligible, and derived the bounds (10) and (11) for delay and backlog respectively. Here the corresponding bounds for PGPS service are derived.

Applying the fact that the slope of G_i^K is never less than ρ_i for each session i to (38), we have:

$$D_i^{*,\mathrm{PGPS}} \leq \frac{\sigma_i + (K-1)L_{\max}}{\rho_i} + \sum_{m=1}^{K} \frac{L_{\max}}{r^m}. \tag{39}$$

The first term on the RHS is likely to dominate in most instances. In particular, in high speed networks we assume that $r^m \to \infty$, and we have

$$D_i^{*,\text{PGPS}} \leq \frac{\sigma_i + (K-1)L_{\max}}{\rho_i}. \tag{40}$$

Also, as $L_{\max} \to 0$, we get (11).

The extra delay of $\frac{(K-1)L_{\max}}{\rho_i}$ in (39) does not diminish with increasing link speed. However, as the following example shows, this term is not superfluous, but is a consequence of the PGPS service discipline:

Consider a PGPS network with a large number of identically characterized sessions—i.e. $A_j \sim (\sigma, \rho, r)$, $\phi_j = 1$ for each session j, and all the packets have the same length, L. Every link operates at rate r, is shared by N sessions, and

$$N\rho = r - \epsilon, \quad \epsilon > 0, \ \epsilon \approx 0. \tag{41}$$

We focus on a session i route that consists of nodes $1, 2, ..., K$, and follow the progress of a session i packet, p, along this route. If p arrives at a node 1 at time t_1, then assume that every other session contending for service at that node sends a packet at time t_1^-. Under PGPS, all $N-1$ packets will be served before p at node 1. Similarly, letting t_m be the time at which p arrives at node m, $2 \leq m \leq K$, we stipulate that for every other session contending for service at that node a packet arrives at time t_m^-. The delay incurred by p from these packets at node m is $\frac{(N-1)L}{r}$, which is $\approx \frac{L}{\rho}$ for large N. Thus, over all nodes in the route, this delay is $\approx \frac{KL}{\rho}$ for large N. Now letting r and L approach ∞ together, we observe that the delay term is unchanged as long as (41) continues to hold. If $L = \sigma$, the worst-case packet delay for session i will be at least $\frac{KL}{\rho}$ for large N, which corresponds to (40).

This example and (40) strongly indicate that small packet lengths should be chosen in RPPS networks so that the term $\frac{L_i}{\rho_i}$ is small. For ATM networks, in which the packets are about 400 bits long, this holds for most kinds of applications. Finally, note that the phenomenon described in our example occurs in other non-preemptive service disciplines such as FCFS as well.

12 Conclusions

Per-session bounds were derived for the leaky bucket constrained sessions of arbitrary topology GPS and PGPS networks. With this analysis, we have provided framework for rate-based flow control in which real-time guarantees can be made to a wide variety of co-existing session types. An important part of any flow control scheme, and one that is missing from this paper is call-admission. Another area for future research is the incorporation of traffic types that require real-time performance but that cannot predict the exact values of their leaky bucket parameters at session set-up time.

Appendix A

Proof of Lemma 8: For $m = 1$, (26) states that

$$S_i^1(\tau,t) \geq \min_{V\in[\tau,t]}\{A_i(\tau,V) + \hat{S}_i^1(0,t-V)\}.$$

Choosing V to be last time in the interval $[\tau, t]$ that session i begins a busy period at node 1:

$$\begin{aligned} S_i^1(\tau,t) &\geq A_i(\tau,V) + \hat{S}_i(0,t-V) \\ &\geq \min_{V\in[\tau,t]}\{A_i(\tau,V) + \hat{S}_i^1(0,t-V)\}. \end{aligned} \tag{42}$$

Now assume the result for nodes $1, 2, ..., m-1$. Then, letting t_m be the last time in the interval $[\tau, t]$ that session i is in a busy period at node m:

$$S_i^m(\tau,t) = S_i^{m-1}(\tau,t_m) + S_i^m(t_m,t) \tag{43}$$

By the induction hypothesis:

$$S_i^{m-1}(\tau,t_m) \geq \min_{V\in[\tau,t_m]}\{A_i(\tau,V) + G_i^{m-1}(t_m-V)\}. \tag{44}$$

Also, from Lemma 7:

$$S_i^m(t_m,t) \geq \hat{S}_i^m(0,t-t_m). \tag{45}$$

Substituting (44) and (45) into (43):

$$S_i^m(\tau,t) \geq \min_{V\in[\tau,t_m]}\{A_i(\tau,V) + G_i^{m-1}(t_m-V)\} + \hat{S}_i^m(0,t-t_m) \tag{46}$$

$$\geq \min_{V\in[\tau,t_m]}\{A_i(\tau,V) + G_i^{m-1}(t_m-V) + \hat{S}_i^m(0,t-t_m)\} \tag{47}$$

$$\geq \min_{V\in[\tau,t_m]}\{A_i(\tau,V) + G_i^m(t-V)\} \tag{48}$$

$$\geq \min_{V\in[\tau,t]}\{A_i(\tau,V) + G_i^m(t-V),\} \tag{49}$$

where the inequality in (48) follows from the definition of G_i^m in (21). □

Proof of Theorem 3: We first show (27): For some given set of arrival functions $A_1, ..., A_N$:

$$Q_i(t) = A_i(0,t) - S_i^K(0,t).$$

From Lemma 8,

$$
\begin{aligned}
Q_i(t) &\leq A_i(0,t) - \min_{V \in [0,t]} \{A_i(0,V) + G_i^K(t-V)\} && (50)\\
&= A_i(0,t) - A_i(0,V_{\min}) + G_i^K(t - V_{\min}) && (51)
\end{aligned}
$$

where $V_{\min}$ is the minimizing value of V. Thus

$$
\begin{aligned}
Q_i(t) &\leq A_i(V_{\min}, t) - G_i^K(t - V_{\min}) && (52)\\
&\leq \hat{A}_i(0, t - V_{\min}) - G_i^K(t - V_{\min}) && (53)\\
&\leq \max_{\tau \geq 0} \{\hat{A}_i(0,\tau) - G_i^K(\tau)\}, && (54)
\end{aligned}
$$

and (27) follows.

Next we show (28): For a given set of arrival functions, $A_1, ..., A_N$ and $t \geq 0$, we have from Lemma 8:

$$
S_i^K(0,t) \geq \min_{V \in [0,t]} \{A_i(0,V) + G_i^K(t-V)\}.
$$

Thus, for all $\hat{t} \geq 0$:

$$
\begin{aligned}
D_i(\hat{t}) &= \min\left\{t : S_i^K(0,t) = A_i(0,\hat{t})\right\} - \hat{t} && (55)\\
&\leq \min\left\{t : \min_{V \in [0,t]} \{A_i(0,V) + G_i^K(t-V)\} = A_i(0,\hat{t})\right\} - \hat{t} && (56)\\
&= \min\left\{t : A_i(0,V_{\min}) + G_i^K(t - V_{\min}) = A_i(0,\hat{t})\right\} - \hat{t} && (57)\\
&= \min\left\{t : G_i^K(t - V_{\min}) = A_i(V_{\min},\hat{t})\right\} - \hat{t} && (58)\\
&\leq \min\left\{t : G_i^K(t) = A_i(V_{\min},\hat{t})\right\} + V_{\min} - \hat{t} && (59)\\
&\leq \min\left\{t : G_i^K(t) = \hat{A}_i(0,\hat{t} - V_{\min})\right\} + V_{\min} - \hat{t} && (60)\\
&\leq \min\left\{t : G_i^K(t) = \hat{A}_i(0,\hat{t} - V_{\min})\right\} - (\hat{t} - V_{\min}) && (61)\\
&\leq \max_{\tau \geq 0}\left\{\min\{t : G_i^K(t) = \hat{A}_i(0,\tau)\} - \tau\right\}. && (62)
\end{aligned}
$$

In (57) we choose the *smallest* minimizing value of V. Then $V_{\min} \leq \hat{t}$, since $G_i^K(t - V_{\min}) \geq 0$. □

Appendix B

The following Lemma establishes (among other things) that for a (K, t)-staggered greedy regime, backlogs are not built up at node m prior to time T_m.

Lemma 10 *Suppose we are given a (K, t)-staggered greedy regime characterized by $(0, T_2, ..., T_K)$, $t \leq B_K$, and a node $k \in \{1, 2, ..., K\}$.*
For each $j = 1, 2, ..., k-1$, and $\tau \in [T_j, T_{j+1}]$:

$$S_i^k(0, \tau) = (\sum_{m=1}^{j-1} \hat{S}_i^m(0, T_{m+1} - T_m)) + \hat{S}_i^j(0, \tau - T_j), \tag{63}$$

and for $\tau > T_k$:

$$S_i^k(0, \tau) = \min\{\hat{A}_i(0, \tau), \sum_{m=1}^{k-1} \hat{S}_i^m(0, T_{m+1} - T_m) + \hat{S}_i^k(0, \tau - T_k)\}. \tag{64}$$

Proof. We proceed by induction on k: For $k = 1$ only (64) applies. Since $S_i^1 = \hat{S}_i^1$ the basis step is shown. Now assume the result for nodes $1, 2, ..., k-1$. We will prove it for node k by contradiction using a somewhat intricate argument.
Observe that $(0, T_2, ..., T_k)$ is a (k, T_k)-staggered greedy regime. Then by induction hypothesis, the function A_i^k is given by (64) for all times $\tau \geq T_k$. I.e.,

$$S_i^{k-1}(0, \tau) = \min\{\hat{A}_i(0, \tau), \sum_{m=1}^{k-2} \hat{S}_i^m(0, T_{m+1} - T_m) + \hat{S}_i^{k-1}(0, \tau - T_k)\}. \tag{65}$$

Now if $Q_i^k(\tau) = 0$ for all $\tau \leq T_k$ we are done by (66) and the induction hypothesis. So let us assume that $Q_i^k(\tau) > 0$ for some $\tau \leq T_k$. Then

$$S_i^k(0, \tau) = \min\{\hat{A}_i(0, \tau), \sum_{m=1}^{k-1} \hat{S}_i^m(0, T_{m+1} - T_m) + \hat{S}_i^k(0, \tau - T_k)\} - Q_i^k(T_k). \tag{66}$$

Since the independent sessions at k are quiet during the interval $[0, T_k]$ it follows that there is at least one interval before T_k during which $S_i^{(k-1)}$ has slope greater than r_k (where r_k is the rate of k). But the slope of $\hat{S}_i^k(0, t)$ is never greater than r_k for $t \in [0, t_k^B]$. Since $T_1, T_2, ..., T_k$ are derived from the the minimization of (22), it follows that $T_{k+1} - T_k = t_k^B$. Now we have already shown in [6] that no node k busy period can be longer than t_k^B time units, so it follows that $Q_i^k(T_{k+1}) = 0$. Thus

$$S_i^k(0, T_{k+1}) = \hat{A}_i(0, T_{k+1}) = G_i^k(T_{k+1}) - Q_i^k(T_k),$$

where the first equality is from the induction hypothesis. Then $G_i^k(T_{k+1}) \geq \hat{A}_i(0, T_{k+1})$, and

$$T_{k+1} = B_k.$$

Now, let $[a, a + \Delta]$, such that $\Delta > 0$ and $a + \Delta \leq T_k - a$, be an interval during which $S_i^{(k-1)}$ has largest slope, and such that this slope belong to a single node, $j < k$. As we have already argued, the slope of $S_i^{(k-1)}$ during this interval must be greater than r_k, since $Q_i^k(\tau) > 0$ for some $\tau < T_k$. Then the staggered greedy regime characterized by

$$\hat{T} = (0, T_2, ..., T_j - \Delta, T_{j+1} - \Delta, ..., T_k - \Delta)$$

is a $(k, T_{k+1} - \Delta)$-staggered greedy regime. I.e.,

$$\sum_{m=1}^{k} \hat{S}_i^m(0, \hat{T}_{m+1} - \hat{T}_m) = G_i^k(t - \Delta), \tag{67}$$

where $\hat{T}_{k+1} = T_{k+1} - \Delta$. Now since $\hat{T}_{k+1} - \hat{T}_k = t_k^B$, it follows from similar reasoning as above that under $\hat{T}$, session i is not backlogged at k at time $\hat{T}_{k+1}$, and that therefore

$$\hat{T}_{k+1} = B_k.$$

Thus

$$\hat{T}_{k+1} = T_{k+1} - \Delta = B_k.$$

But this implies that $\Delta = 0$, which is a contradiction. □

To show Theorem 4, pick $\tau = t > T_K$ in Lemma 10. Then (64) applies, and since $t \leq B_K$ the result follows.

Appendix C

Proof of Lemma 9 For $m = 1$, (32) states that

$$S_i^1(\tau, t) \geq \min_{V \in [\tau, t]} \{A_i(\tau, V) + \hat{S}_i^1(0, t - V)\} - L_i.$$

Choosing V to be last time in the interval $[\tau, t]$ that session i begins a busy period at node 1:

$$\begin{aligned} S_i^1(\tau, t) &\geq A_i^1(\tau, V) + \hat{S}_i^1(0, t - V) & (68) \\ &\geq A_i(\tau, V) - L_i + \hat{S}_i^1(0, t - V) \\ &\geq \min_{V \in [\tau, t]} \{A_i(\tau, V) + \hat{S}_i^1(0, t - V) - L_i\}. & (69) \end{aligned}$$

Now assume the result for nodes $1, 2, ..., m - 1$. Then, letting t_m be the last time in the interval $[\tau, t]$ that session i begins a busy period at node m:

$$S_i^m(\tau, t) = A_i^m(\tau, t_m) + S_i^m(t_m, t).$$

From (31):

$$S_i^m(\tau, t) \geq S_i^{m-1}(\tau, t_m) - L_i + S_i^m(t_m, t). \tag{70}$$

By the induction hypothesis:

$$S_i^{m-1}(\tau, t_m) \geq \min_{V \in [\tau, t_m]} \{A_i(\tau, V) + G_i^{m-1}(t_m - V) - (m - 1)L_i\}. \tag{71}$$

Also, from Lemma 6:

$$S_i^m(t_m, t) \geq \hat{S}_i^m(0, t - t_m). \tag{72}$$

Substituting (71) and (72) into (70):

$$\begin{aligned} S_i^m(\tau, t) + mL_i &\geq \min_{V \in [\tau, t_m]} \{A_i(\tau, V) + G_i^{m-1}(t_m - V)\} + \hat{S}_i^m(0, t - t_m) & (73) \\ &\geq \min_{V \in [\tau, t_m]} \{A_i(\tau, V) + G_i^{m-1}(t_m - V) + \hat{S}_i^m(0, t - t_m)\} & (74) \\ &\geq \min_{V \in [\tau, t_m]} \{A_i(\tau, V) + G_i^m(t - V)\} & (75) \\ &\geq \min_{V \in [\tau, t]} \{A_i(\tau, V) + G_i^m(t - V), \} & (76) \end{aligned}$$

where the inequality in (75) follows from the definition of G_i^m in (21). □

Proof of Theorem 6 For some given set of arrival functions $A_1, ..., A_N$:

$$Q_i(t) = A_i(0,t) - S_i^K(0,t).$$

From Lemma 9,

$$\begin{aligned} Q_i(t) - KL_i &\leq A_i(0,t) - \min_{V\in[0,t]}\{A_i(0,V) + G_i^K(t-V)\} && (77) \\ &\leq A_i(0,t) - A_i(0,V_{\min}) + G_i^K(t - V_{\min}) && (78) \end{aligned}$$

where $V_{\min}$ is the minimizing value of V. Thus

$$\begin{aligned} Q_i(t) - KL_i &\leq A_i(V_{\min}, t) - G_i^K(t - V_{\min}) && (79) \\ &\leq \hat{A}_i(0, t - V_{\min}) - G_i^K(t - V_{\min}) && (80) \\ &\leq \max_{\tau\geq 0}\{\hat{A}_i(0,\tau) - G_i^K(\tau)\}, && (81) \end{aligned}$$

and (33) follows. □

Proof of Theorem 7 For a given set of arrival functions, $A_1, ..., A_N$ and t such that a packet departs node K at time $t \geq 0$, we have from (36):

$$S_i^K(0,t) \geq \min_{V\in[0,\hat{t}]}\{A_i(0,V) + G_i^K(t-V)\} - (K-1)L_i,$$

where the packet departing at time t arrived at time $\hat{t}$. Thus, for all packet arrival times $\hat{t} \geq 0$:

$$\begin{aligned} D_i(\hat{t}) &= \min\left\{t : S_i^{(K)}(0,t) = A_i(0,\hat{t})\right\} - \hat{t} \\ &\leq \min\left\{t : \min_{V\in[0,\hat{t}]}\{A_i(0,V) + G_i^K(t-V) - (K-1)L_i\} = A_i(0,\hat{t})\right\} - \hat{t} \\ &\leq \min\left\{t : A_i(0,V_{\min}) + G_i^K(t-V_{\min}) = A_i(0,\hat{t}) + (K-1)L_i\right\} - \hat{t} \\ &\leq \min\left\{t : G_i^K(t - V_{\min}) = A_i(V_{\min},\hat{t}) + (K-1)L_i\right\} - \hat{t} \\ &\leq \min\left\{t : G_i^K(t) = A_i(V_{\min},\hat{t}) + (K-1)L_i\right\} + V_{\min} - \hat{t} \\ &\leq \min\left\{t : G_i^K(t) = \hat{A}_i(0,\hat{t} - V_{\min}) + (K-1)L_i\right\} + V_{\min} - \hat{t} \\ &\leq \min\left\{t : G_i^K(t) = \hat{A}_i(0,\hat{t} - V_{\min}) + (K-1)L_i\right\} - (\hat{t} - V_{\min}) \\ &\leq \max_{\tau\geq 0}\left\{\min\{t : G_i^K(t) = \hat{A}_i(0,\tau) + (K-1)L_i\} - \tau\right\}. && (82) \end{aligned}$$

□

Appendix D

Proof of Theorem 8: The next two Lemmas are useful:

Lemma 11 *Given the same set of arrival functions at a node, m. Defining $S_i^{\mathrm{GPS}}(0,t) = 0$ for $t \leq 0$:*

$$S_i^{\mathrm{PGPS}}(0,t) \geq S_i^{\mathrm{GPS}}(0, t - \frac{L_{\max}}{r^m}) \tag{83}$$

for each session i and time t.

Proof. From Theorem 2 of [6] the completion of a packet arrival time under PGPS is delayed by at most $\frac{L_{\max}}{r^m}$ more under PGPS than under GPS. Then the service curve is translated in time at most that amount. The result follows. □

Lemma 12 *Suppose we are given arrival functions $A_1, ..., A_N$ at a single GPS server, such that for a particular session i, the k^{th} session i packet has length $l_k < L_i$. Replace A_i, with $\bar{A}_i$ such that $\bar{A}_i(0,t) \geq A_i(0,t)$ for all t and the k^{th} packet still has length l_k. Then*

$$\bar{S}_i(0,t) \geq S_i(0,t)$$

for all t.

Proof. The following proposition can be shown to be true for all time t by induction on k: Suppose a session i bit, belonging to the k^{th} packet, arrives at time t under $\bar{A}$, and is served at time t'. Then no bit that was served after it under A, can be served before it under $\bar{A}$ in the interval (t, t').

Thus $\bar{S}_i(0,t) \geq S_i(0,t)$ for all t. □

Now suppose we are given a PGPS network with arrival functions $A_1, ..., A_N$, and we would like to bound delay for a particular session, i. Without loss of generality, assume that $P(i) = \{1, 2, ..., K\}$. The arrival functions A_j^m, for each m, $1 \leq m \leq K$, and $j \in I(m)$ are completely determined and are assumed to be known.

Now construct a GPS network consisting of nodes $1, 2, ..., K$ connected in a line, i.e. the links are given by $\{(e, e+1) : e = 1, 2, ..., K-1\}$. The rates of the links are the same as the corresponding links in the PGPS network, but the link leaving node m has a fixed propagation delay of $\frac{L_{\max}}{r^m}$. The GPS network supports a session i, with route $1, 2, ..., K$ and arrival function given by A_i, i.e. the route and arrival functions are identical to those in in the PGPS network. The other sessions on the GPS network have a route of exactly one hop and are defined as follows: At each node m, for every session j in the PGPS network define a session j' such that

$$A_{j'}^m(0,t) = A_j^m(0,t).$$

Now for each node, m, let $S_i^{m,\mathrm{PGPS}}$ describe the session i departures from node m in the PGPS network, and let $S_i^{m,\mathrm{GPS}}$ describe the session i departures from node m in the corresponding GPS network. Then

Lemma 13 *For $k = 1, 2, ..., K$*

$$S_i^{k,\mathrm{GPS}}(0, t - \frac{L_{\max}}{r^k}) \leq S_i^{k,\mathrm{PGPS}}(0, t)$$

for all t.

Proof. By induction on k: For $k = 1$ we are done from Lemma 11. Assume the result at nodes $1, 2, ..., k-1$ and show it at node $k \leq K$:
First, consider the service function $\bar{S}_i^{k,\mathrm{GPS}}$ that results at node m if the session i arrivals at node k in the GPS network are identical to the session i arrivals at node k in the PGPS network. Then from Lemma 11:

$$\bar{S}_i^{k,\mathrm{GPS}}(0, t - \frac{L_{\max}}{r^k}) \leq S_i^{k,\mathrm{PGPS}}(0, t). \tag{84}$$

By the induction hypothesis:

$$S_i^{k-1,\mathrm{GPS}}(0, t - \frac{L_{\max}}{r^{k-1}}) \leq S_i^{k-1,\mathrm{PGPS}}(0, t) \tag{85}$$

for all t. The LHS of (85) describes the traffic that has traversed the link $(k-1, k)$ in the GPS network in the interval $[0, t]$. Thus, every session i packet arrives at node k *earlier* in the PGPS network, than it does in the GPS network. From Lemma 12:

$$S_i^{k,\mathrm{GPS}}(0, t - \frac{L_{\max}}{r^k}) \leq \bar{S}_i^{k,\mathrm{GPS}}(0, t - \frac{L_{\max}}{r^k}) \tag{86}$$

for all t. From (86) and (84):

$$S_i^{k,\mathrm{GPS}}(0, t - \frac{L_{\max}}{r^k}) \leq S_i^{k,\mathrm{PGPS}}(0, t). \tag{87}$$

□

Now assume a fixed network topology with no propagation delay. Also assume a fixed internal characterization for all the sessions. Let $D_i^{*,GPS}$ be the worst-case session i delay when the nodes have GPS servers. and let $D_i^{*,PGPS}$ be the worst-case session i delay when the nodes have PGPS servers. Then a direct consequence of Lemma 13 is that

$$D_i^{*,\text{PGPS}} \leq D_i^{*,\text{GPS}} + \sum_{m=1}^{K} \frac{L_{\max}}{r^m}. \tag{88}$$

Note that the GPS network being considered here has internal characterization identical to the PGPS network—thus the traffic is burstier than it would be if the procedure of Section 7 had been used.

Now using the bounds in Theorems 6 and 7:

$$D_i^{*,\text{PGPS}} \leq \max_{\tau \geq 0} \left\{ \min\{t : G_i^K(t) = \hat{A}_i(0,\tau) + (K-1)L_{\max}\} - \tau \right\} + \sum_{m=1}^{K} \frac{L_{\max}}{r^m}. \tag{89}$$

□

References

[1] R. L. Cruz, *A calculus for network delay, Part I: Network elements in isolation*, IEEE Transactions on Information Theory, 37 (1991), pp. 114–131.

[2] ——, *A calculus for network delay, Part II: Network analysis*, IEEE Transactions on Information Theory, 37 (1991), pp. 132–141.

[3] R. Kolla and B. Serf, *The virtual feedback problem in hierarchical representations of combinatorial circuits*, Acta Informatica, 28 (1991), pp. 463–476.

[4] C. Lu and P. R. Kumar, *Distributed scheduling based on due dates and buffer prioritization*, tech. rep., University of Illinois Technical Report, 1990.

[5] A. Parekh, *A Generalized Processor Sharing Aproach to Flow Control in Integrated Services Networks*, PhD thesis, Department of Electrical Engineering and Computer Science, MIT, February 1992.

[6] A. K. Parekh and R. G. Gallager, *A Generalized Processor Sharing approach to flow control—The Single Node Case*, Tech. Rep. 2040, Laboratory for Information and Decision Systems, MIT, 1991.

[7] J. R. Perkins and P. R. Kumar, *Stable distributed real-time scheduling of flexible manufacturing systems*, IEEE Transactions on Automatic Control, AC-34 (1989), pp. 139–148.