Available as Los Alamos Unclassified Report LA-UR 03-8012

Interference Control at Packet-level in Wireless Networks with Quality of Service Support*

Sami Ayyorgun
Research & Development in Advanced Network Technology
Los Alamos National Laboratory
P.O. Box 1663, M.S. D451
Los Alamos, New Mexico 87545
sami@lanl.gov

Rene L. Cruz
Dept. of Electrical & Computer Engineering
University of California, San Diego
9500 Gilman Drive
La Jolla, CA 92093-0407
cruz@ece.ucsd.edu

Abstract—*A dynamic approach to control interference in wireless cellular networks is presented. The goal is to support high-data-rate services with explicit Quality of Service (QoS) guarantees. Interference control is implemented by limiting the total number of packet transmissions occurring in any time-interval over a set of interfering wireless channels. The QoS guarantees are implemented by adopting a* service curve model with loss *which incorporates delay and loss guarantees. The approach is applicable to both downlink and uplink communications. To provide insight into the approach, a scheduling problem is solved. In the problem, a scheduling algorithm is sought to coordinate packet transmissions to or from mobile hosts in a cell such that the delay and loss requirements of the applications run by the mobile hosts are met and concurrently interference is controlled as indicated. The solution is obtained via a graph-theoretic approach, which provides a necessary and sufficient condition for the existence of a such scheduling. The number of basic computations required to check the correctness of the condition is shown to grow at least exponentially by time when there are more than one mobile host to support. A heuristic on-line algorithm is also proposed.*

1 Introduction

All existing wireless cellular networks have been designed originally for voice communication. With relatively minor add-on technologies, these networks have been made to offer low-data-rate data services as well. Current data services, that are commercially available in these networks, offer data rates which are only a few tens of kilobits per second. The speed (i.e., the experienced data rate) and the quality of these services at best approach to those of dial-up modems in wireline networks.

The explosion of wireline Internet services and their widespread use by the masses in the second half of the 1990's launched a demand for high-data-rate services in wireless cellular networks. The speed and the quality of these services are hoped to be similar to those of Digital Subscriber Line (DSL) and/or Cable services in wireline networks. Although some prototype high-data-rate services have been designed for long, such as Qualcomm HDR [1], they have not become commercially available yet. Technologies employed to realize these prototype services do not bring down the cost of these services to a level that they become economically viable. Some of these economic factors have been pointed out in [1].

A major factor driving up the cost of these high-data-rate services is the need to combat fluctuations in the quality of wireless channels effectively in order to provide these services with acceptable Quality of Service (QoS). Traditional approaches in communication theory to facilitate reliable communication over noisy channels have been shown to be inadequate for high-data-rate services with QoS demands, e.g., [2]. The burstiness and high data-rates (i.e., the average number of bits per second) of data flows desired to be supported by these services are the main factors that lead to the inadequateness of the traditional approaches.

*Los Alamos National Laboratory, Technical Report LA-UR-03-8012, October 2003.

Traditionally, a fixed statistical level of signal interference and channel noise is assumed over all the time in demodulating and decoding received signals from any user in a cellular network (e.g., refer to [3]). This works well, with the help of Law of Large Numbers[1], for voice and other low-data-rate applications. However for applications with bursty high-data-rate traffic, it is not suitable, since the interference caused by these traffic tends to be bursty too.

In this study, we present an dynamic approach to control signal interference in wireless cellular networks. We propose to control the level of interference in any time-interval by allowing only a predetermined total number of packet transmissions in that interval, where all the transmissions being controlled interfere with each other. The predetermined total number of packet transmissions in any interval is allowed to be time-varying.

In addition to controlling interference as explained above, the approach that we present also incorporates supporting some QoS guarantees. Each stream of packets that needs to be served by the network has some delay and loss requirements. Specifically, each packet in a stream has a deadline by which the packet has to be served; otherwise, the packet is deemed to be useless. Also, each stream requires at least a certain fraction of its packets to be served. These QoS requirements are incorporated into our approach by adopting a service model provided by Definition 4 in Section 2.

The approach that we examine in this study, with which we control interference, is applicable to both downlink (i.e., from base stations to mobile hosts) and uplink (i.e., from mobile hosts to base-stations) communications. However, it is more suitable for downlink communications. One reason for this, among other reasons, is that information rates on downlink communications tend to be much higher than those of uplink communications—though, the gap between the two may close in the future.

In a cellular network, only few number of base-stations interfere with each other when transmitting information to mobile hosts. For example in a Code Division Multiple Access (CDMA) network[2], a mobile host at one of the vertices of the imaginary hexagonal cellular structure suffers the strongest interference from two equidistant base-stations while the third equidistant base-station transmits information to the mobile host. Thus, the interference in the downlink does not enjoy the advantage of the Law of Large Numbers. Moreover, since the data being transmitted downlink on the interfering channels tend to be bursty for high-data-rate services, this further contributes to the variability of the level of interference. Assuming that interfering base-stations are almost always active, the approach that we examine controls the level of interference caused by a base-station by limiting the total number of packet transmissions emanating from that base-station in any time-interval. The assumption that an interfering base-station is almost always active is realistic, because when a base-station is planned to be established, the cell in which the base-station serves mobile hosts is determined such that the idle-time of base-station is kept close to minimum. A base-station typically serves tens of mobile hosts in its cell.

For uplink communications, our approach can become applicable as follows: Consider a cell served by a base-station in a CDMA network. Typically, again, tens of mobile hosts are served in the cell. Uplink transmissions of any mobile host causes interference to any other host in the cell. Consider a mobile host transmitting information to the base-station. In any given time, it is quite likely that at least one other mobile host is also active sending information. With our approach, we tend to control the level of interference caused by all the other mobile hosts to this particular host by limiting the total number of packet transmissions emanating from the other hosts in any time-interval. Since this particular mobile host also causes interference to the other hosts, we actually control the level of interference in uplink communications by limiting the total number of packet transmissions emanating from *all* the hosts in any time-interval.

Whether the communications is uplink or downlink, base-stations coordinate packet transmissions in our approach to control interference and to satisfy the QoS requirements.

To provide insight into our approach that is explained above, we solve a scheduling problem in this study. The problem that we solve fits into both uplink and downlink communications.

[1]Refer to, for example, [4].

[2]Refer to, for example, [5].

The rest of the paper is organized as follows: Section 2 provides a background material. Section 3 states the problem that we solve in this study. Section 4 presents a solution. Section 5 examines the complexity of the solution. Section 6 proposes an algorithm to implement our approach explained in the introduction. Section 7 provides comparison with some related works. Section 8 concludes the study.

2 Background

We adopt a discrete-time formulation for simplicity. Time is slotted into fixed-length intervals, and marked by the integers. The unit of transmission for communication is referred to as a *packet*, in this study. A *flow* is a non-decreasing function defined from the integers to the non-negative integers. The value $R(n)$ of a flow R at time n denotes the total number of packets, which belong to a stream of packets, that pass through a cross section of a communication link by time n (inclusive). Flows in this study are assumed to be bounded, i.e., $R(\infty) < \infty$. We refer to a packet stream represented by flow R also as a flow, whenever it is not ambiguous to do so. Hence, it is meaningful to refer to a packet in a flow R. A *network element* is an input-output device or a medium that accepts packets at its input, and delivers them at its output. Packets are assumed to instantaneously arrive and depart at a network element, i.e., a whole packet can arrive instantaneously at some time k and depart later at time n where $n \geqslant k$. Note that a packet can depart in the same interval in which it has arrived; this is sometimes referred to as *cut-through* operation. The capacity $c(n)$ of a network element at time n is the total number of packets that it can serve at time n. The function c is called the *instantaneous capacity rate* (or, just the *rate*) of the network element. The notation $\lceil x \rceil$ denotes the least integer greater than or equal to x, and is read as "the ceiling of x". Similarly, $\lfloor x \rfloor$ denotes the greatest integer less than or equal to x, and is read as "the floor of x". Finally, all functions are assumed to be defined from the integers to the integers, unless otherwise noted from here on.

The following definitions are utilized in this study, which are provided here for convenience. The first three were previously introduced in the literature (e.g., refer to [6, 7, 8, 9, 10, 11, 12]).

Definition 1 *A flow R is said to conform to an* arrival curve *A, and denoted as $R \sim A$, if R satisfies*

$$R(n+k) - R(k) \leqslant A(n) \quad \text{for all } k \text{ and } n,$$

where A is a non-decreasing function defined from the integers to the non-negative integers and $A(n) = 0$ for all $n \leqslant 0$.

Definition 2 *The* min-+ convolution *$f \triangledown g$ of any two functions f and g is defined as*[3]

$$(f \triangledown g)(n) = \min_{k \leqslant n}\{f(k) + g(n-k)\} \qquad \text{for all } n.$$

The convolution[4] $f \triangledown g$ is read as "f min-convolved with g", or as "the min-plus convolution of f with g".

Definition 3 *A network element is said to deliver a* service curve *S to an input flow R if the corresponding output flow G satisfies*

$$G(n) \geqslant (R \triangledown S)(n) \quad \text{for all } n,$$

where S is a non-decreasing function defined from the integers to the non-negative integers and $S(n) = 0$ for all $n \leqslant 0$.

[3]There are two other variants of the min-+ convolution, depending on how the subscript of the minimum is delimited in the definition; one for '$0 \leqslant k \leqslant n$', and one for '$k \in \mathbb{Z}$'.

[4]One reason to choose this notation for min-+ convolution over some others, for example '$*$', is that this choice of notation provides a better selection of notations for other related operators in a fitting manner. The min-+ convolution and all the related operators are employed together, for example, in [13].

A network element delivering a service curve to an input flow can be viewed as a generalization of a *work-conserving*[5] server. This becomes clear if we note that the output flow G of a work-conserving server with constant integer rate ρ and with input flow R is given[6] by

$$G(n) = \min_{k \leqslant n}\{R(k) + \rho \cdot (n-k)\} \quad \text{for all } n,$$

which is equivalently represented as

$$G(n) = (R \triangledown S)(n) \quad \text{for all } n,$$

where $S(n) = \max\{0, \rho \cdot n\}$. In other words, the work-conserving server in this case delivers the service curve $S(n) = \max\{0, \rho \cdot n\}$ to input flow R.

Recently in [16], we have introduced a generalization of the concept of service curves by incorporating a loss aspect into it, which makes the new service model more realistic. The generalization is provided by Definition 4 for which the following explanations are given first: Let R be a flow. We mark all the packets in R, which arrive at any time n arbitrarily but distinctly by the integers in $(R(n-1), R(n)]$. For any k in $[1, R(\infty)]$, the deadline D_k of packet k is defined as

$$D_k \triangleq \min\{t : (R \triangledown S)(t) \geqslant k\}\,. \tag{1}$$

Definition 4 (Service Curve Model with Loss) *A network element is said to deliver a service curve S with loss parameter $1-\alpha$ to an input flow R* if and only if *at least an α fraction of all the packets in R meet their deadlines assigned via the service curve S, where deadlines are determined by (1). The packets that are not to be delivered within their deadlines are not delivered at all, and are considered to be* dropped *(or,* lost*).*

The service model provided by Definition 4 facilitates to take advantage of applications' tolerance to loss, which is defined in precise terms, analytically. The loss aspect of the model is also motivated by some new transmission and coding schemes which are recently proposed for distribution of information over lossy packet-based networks [17, 18, 19]. These new schemes require only at least a certain fraction of all the packets in a communication session to be successfully received/decoded for reliable communication.

3 Problem Statement

Service curves together with an appropriate traffic characterization, such as deterministic ones provided by Definition 1 or [20], or a probabilistic one in [13], facilitate a systematic treatment of performance guarantees in communication networks, e.g., see [9, 10, 11, 12, 13, 20]. The facilitated systematicness is similar to that of linear systems theory. Therefore, we choose to formulate the problem that we solve to provide insight into our approach explained in the introduction in terms of service curves with loss and arrival curves. The problem is stated below, which can be interpreted for both downlink and uplink communications (as explained in the introduction).

Problem 1 *A set of flows indexed by $I = \{1, 2, \ldots, |I|\}$ is given. Each flow i in I conforms to an arrival curve A_i, needs to be transmitted over a separate wireless link with capacity $c_i(n)$, and requests a service curve S_i with loss parameter $1-\alpha_i$. A total of at most $g(n)$ packets is allowed to be transmitted at any time n over all the wireless links. Find a necessary and sufficient condition on the capacity of wireless links such that these requests can be met.*

[5] A network element is said to be *work-conserving* if it serves packets whenever it has packets to serve, unconditionally of any other criteria.

[6] This is often referred to as Reich's result [14]. See for example [15] for a simple derivation.

The capacity $c_i(n)$ of any wireless link i can be time-varying but is assumed to be non-zero at any time n, for simplicity. With this assumption, the generality is not lost, because a channel with a time-varying capacity can be modeled as a channel with a constant capacity associated with a set of fictitious streams which modulate the constant capacity to generate the real time variations; see [21] for detailed information and an example.

Clearly, a coordination among packet transmissions is needed in Problem 1 in order to deliver the service requests, if indeed the requests are feasible. Specifically, an appropriate *scheduling* algorithm is needed to deliver these requests. A *scheduling* in the context of Problem 1 is a mapping of all the packets to be transmitted into the integers where the number of packets to be transmitted over wireless link i and being mapped into n does not exceed the capacity $c_i(n)$ at time n, for all $i \in I$.

We solve Problem 1 by a graph-theoretic approach presented in the following section.

4 Solution to Problem 1

Scheduling problems could often be formulated as a *matching* problem in graph theory. Problem 1 could also be formulated as such by representing the scenario being put in place by packet and capacity "arrivals" as a *labeled graph* for any realization of Problem 1. A graph theory problem is then obtained by seeking appropriate matchings in that labeled graph.

In the following section, we present a matching theorem which we utilize in solving Problem 1.

4.1 A Matching Theorem

For mathematical clarity, we give a brief terminology; first for the *sets*, and then for *graphs*.

We denote the difference of set A from set B by $A - B$, i.e., $A - B$ includes all the elements in A but not in B. The *complement* of a set A with respect to a universal set U is the set $A - U$, which is denoted by A^c. In taking the complement of a set, the universal set is understood to be the largest possible set in the context in which the complement is taken. The *cardinality* of a set A is the total number of elements in A, denoted by $|A|$. Two sets A and B are said to be *disjoint* if they have no elements in common, i.e., $A \cap B = \emptyset$. A *partition* of a set A is a set $\{A_1, A_2, \ldots\}$ of disjoint sets whose union is A. Each set A_i in a partition $\{A_1, A_2, \ldots\}$, where $A = \bigcup_i A_i$, is called a *part* of the set A. The *power set* of a set A is the set of all the subsets of A.

Let V be a finite set, and E be a set of its 2-element subsets. The ordered pair (V, E) is called a *simple* and *undirected* ***graph*** G, and denoted usually as $G = (V, E)$. The elements of V are called the *nodes*, and those of E are called the *edges*. Two nodes composing an edge are called its *end-nodes*, and are said to be *joined* by the edge. A graph is said to be *simple* if it has no *loop* (i.e., the end-nodes of no edge are the same) and no *multi-edge* (i.e., the end-nodes of no two edges are the same), and *undirected* if none of its edges is an ordered pair—as these could also be inferred from the definition of a graph given here. Graphs are understood to be simple and undirected in this study. A node in a graph is said to be *isolated*, if it is not an end-node of an edge in the graph. A graph is called ***bipartite*** if its nodes could be partitioned into two sets M and W such that all the edges have one end-node in M and the other in W, in which case the *bipartite graph* is denoted usually as $G = (M \cup W, E)$. The sets M and W are called the *parts* of the bipartite graph G. A ***matching*** in a graph is a set of edges where no two edges have a common end-node. A matching is said to be *maximum* if no other matching in the graph has a larger cardinality. A node is said to be *matched* in a matching if it is an end-node of an edge in the matching.

The graph theory problem whose solution we employ in solving Problem 1 is given below—this problem is not set specifically to solve the Problem 1.

Problem 2 *A bipartite graph $G = (M \cup W, E)$, a partition $\{V_j : j \in J\}$ of its nodes where each part V_j is a subset of either M or W, and a mapping f of the partition into the positive integers are given. Find the cardinality of a maximum matching such that the number of nodes matched in any part of the partition does not exceed the integer to which it is mapped.*

A solution to Problem 2 is given by the following theorem proven in [21, 22]—an algorithm to find a such maximum matching in G is also suggested in the proof.

Theorem 1 *The cardinality a maximum matching in G as described Problem 2 is equal to* $\min_{X \subseteq W}\{\sigma(X) + \sigma(M - \Pi(X))\}$.

The following definitions are needed to be able to read the theorem. The first two definitions are general considering a graph $G = (V, E)$. The last one is specific to Problem 2:

1. The **image** $\Gamma(X)$ of a set of nodes X is the set of all the nodes to which a node in X is joined in G.[7] The image $\Gamma(X)$ is read as "the image of X in G".
2. The **principal** $\Pi(X)$ of a set of nodes X is the set of all the nodes whose image is a subset of X. The principal $\Pi(X)$ is read as "the principals of X in G".
3. A function σ, called the ***permission***, is defined on the power set of $M \cup W$ as
$$\sigma(X) = \sum_{j \in J} \min\{f(V_j), |X \cap V_j|\}.$$
The value $\sigma(X)$ is read as "the permission of X". The name of the function follows since $\sigma(X)$ is clearly the maximum number of nodes in X, which is allowable to be matched in G.

4.2 Utilizing Theorem 1

We can solve Problem 1 by utilizing Theorem 1. To do this, we represent the scenario being put in place by packet and capacity arrivals in Problem 1 as a labeled graph for any realization of the problem. We define the following terms for all k in $[1, R(\infty)]$ for any flow R;

$$a_k \triangleq \min\{t : R(t) \geqslant k\}, \tag{2}$$

where a_k stands for the arrival time of packet k in flow R.

A **labeled graph** for any realization of Problem 1 is constructed as follows: Represent each packet k in each flow i as a node with label k. Denote the set of nodes labeled for flow i by M_i. Represent the each capacity of transmitting a single packet at any time n over wireless link i as a node with label n; denote this set of nodes by W_i. Join every node with label k in M_i to the nodes with labels belonging to $[a_k, D_k]$ in W_i, where a_k and D_k are as defined in (2) and (1) respectively for any flow i; these make the set of edges E_i. Delete all the isolated nodes in each W_i after all the edges are obtained. View any such labeled graph as a bipartite graph $G = (M \cup W, E)$, where $M = \bigcup_{i \in I} M_i$, $W = \bigcup_{i \in I} W_i$, and $E = \bigcup_{i \in I} E_i$. Finally, for any n, let V_n be the set of all nodes whose labels are equal to n in W.

The graph-theoretic formulation of Problem 1, which is given below, is obtained by seeking an appropriately constrained matching in any given labeled graph as constructed above:

Problem 3 *Find a necessary and sufficient condition for the existence of a matching in any labeled graph G as constructed above such that at least an α_i fraction of all the nodes in M_i are matched in G, for all $i \in I$, and at most $g(n)$ nodes are matched in any V_n.*

Note that a matching as specified in Problem 3 corresponds to an appropriate scheduling which is essentially sought in Problem 1.

Surprisingly, we can solve Problem 3 by utilizing Theorem 1, even though the first constraint in Problem 3 is an "at-least" constraint, whereas the constraints in Problem 2 (which is answered by Theorem 1) are all "at-most" constraints.

[7] *Image* is also referred to as *neighborhood*, in graph theory.

We do this by first observing the fact that, in Problem 3, a such matching in G exists *if and only* if there exists a matching in G such that exactly $\lceil \alpha_i |M_i| \rceil$ many nodes are matched in each M_i and at most $g(n)$ nodes are matched in any V_n.

Secondly, we discover a necessary and sufficient condition for the existence of a such matching as just described in the previous paragraph by dressing the following instance on Problem 2: Consider any labeled graph $G = (M \cup W, E)$ in Problem 3. Let the graph in Problem 2 be this G. Furthermore, let the partition in Problem 2 be $\{M_i : i \in I\} \cup \{V_n : \text{any } n\}$, and the mapping f be given by

$$\begin{aligned} f(M_i) &= \lceil \alpha_i |M_i| \rceil && \text{for all } i \in I, \\ f(V_n) &= \min\{g(n), |V_n|\} && \text{for any } n. \end{aligned}$$

We assume without any loss of generality that $\alpha_i \leqslant 1$, for all $i \in I$, and $g(n) \geqslant 0$ for all n. The cardinality of the maximum matching satisfying the constraints in this instance of Problem 2 is given by Theorem 1 too.

Now, if we set the cardinality of a such maximum matching in this instance of Problem 2 to be equal to $\sum_{i \in I} f(M_i)$, we can discover the necessary and sufficient condition that we have mentioned in the previous paragraphs. This is explained in detail in what follows next, at the end of which the necessary and sufficient condition is found.

First of all, note that if the cardinality of a maximum matching in this instance of Problem 2 becomes equal to $\sum_{i \in I} f(M_i)$, then exactly $f(M_i)$ many nodes should be matched in each set M_i, in a such matching. Otherwise, for at least one M_i, the constraint on that M_i would be violated.

Secondly, note that the quantity $\sum_{i \in I} f(M_i)$ is equal to the permission of M (i.e., $\sigma(M)$), as shown below;

$$\begin{aligned} \sigma(M) &= \sum_{j \in J} \min\{f(V_j), |M \cap V_j|\} \qquad \text{(by definition of } \sigma) \\ &= \left(\sum_{i \in I} \min\{f(M_i), |M \cap M_i|\} \right) + \left(\sum_{n} \min\{f(V_n), |M \cap V_n|\} \right) \\ &= \left(\sum_{i \in I} \min\{f(M_i), |M_i|\} \right) + \left(\sum_{n} \min\{f(V_n), 0\} \right) \\ &= \sum_{i \in I} \min\{\lceil \alpha_i |M_i| \rceil, |M_i|\} \\ &= \sum_{i \in I} \lceil \alpha_i |M_i| \rceil \\ &= \sum_{i \in I} f(M_i). \end{aligned}$$

Thus, setting the cardinality of a maximum matching in this instance of Problem 2 as being equal to $\sum_{i \in I} f(M_i)$ is the same as setting it equal to $\sigma(M)$.

Now, note that the term $\sigma(M)$ belongs to the set of which we are taking the minimum in Theorem 1, which corresponds to setting $X = \emptyset$. Thus, if none of the other terms in the set is smaller than $\sigma(M)$, then the value of the minimum could indeed be equal to $\sigma(M)$; and, vice versa. Hence, if we write this condition down and manipulate it, we would obtain a necessary and sufficient condition for the existence of a matching in Problem 3, as we have explained it by now: All of the following statements are equivalent to each other for all $X \subseteq W$,

$$
\begin{aligned}
\sigma(M) &\leqslant \sigma(X) + \sigma(M - \Pi(X)) \\
\sigma(M) - \sigma(M - \Pi(X)) &\leqslant \sigma(X) \\
\sigma(M) - \sigma(M - \Pi(X)) &\leqslant \sigma(X) \\
\sum_{i \in I} f(M_i) - \sum_{i \in I} \min\{f(M_i), |(M - \Pi(X)) \cap M_i|\} &\leqslant \sigma(X) \\
\sum_{i \in I} f(M_i) - \sum_{i \in I} \min\left\{f(M_i), |M_i - \Pi(X)|\right\} &\leqslant \sigma(X) \\
\sum_{i \in I} f(M_i) - \sum_{i \in I} \min\left\{f(M_i), |M_i| - |\Pi_i(X)|\right\} &\leqslant \sigma(X) \quad \Pi_i(X) \triangleq \Pi(X) \cap M_i \\
\sum_{i \in I} f(M_i) - \min\left\{f(M_i), |M_i| - |\Pi_i(X)|\right\} &\leqslant \sigma(X) \\
\sum_{i \in I} \max\left\{0, f(M_i) - |M_i| + |\Pi_i(X)|\right\} &\leqslant \sigma(X) \\
\sum_{i \in I} \max\left\{0, \lceil \alpha_i |M_i| \rceil - |M_i| + |\Pi_i(X)|\right\} &\leqslant \sigma(X) \\
\sum_{i \in I} \max\left\{0, |\Pi_i(X)| - \lfloor (1 - \alpha_i) |M_i| \rfloor\right\} &\leqslant \sigma(X) \\
\max\{t_S(X) : S \subseteq I\} &\leqslant \sigma(X) \quad t_S(X) \triangleq \sum_{i \in S} |\Pi_i(X)| - \lfloor (1 - \alpha_i) |M_i| \rfloor \\
t_S(X) &\leqslant \sigma(X) \quad \text{for all } S \subseteq I.
\end{aligned}
$$

For $S = \emptyset$, $t_S(X)$ is 0 by convention for any X subset of W.

Hence, we obtain the following lemma.

Lemma 1 *The following condition,*

$$t_S(X) \leqslant \sigma(X) \qquad \textit{for all } X \subseteq W, \textit{ and for all } S \subseteq I \tag{3}$$

is the necessary and sufficient for the existence of a matching as described in Problem 3.

Proof: The proof follows by the arguments given before the lemma and after Problem 3. ∎

A matching (scheduling) that exactly (at least) α_i fraction of all the nodes (packets) in each M_i (flow i) are matched (delivered) and at most $g(n)$ nodes (packets) are matched in any V_n (at any time n) in any labeled graph as constructed at the beginning of this section, *if and only if* condition (3) holds.

4.3 The Condition In Terms Of Service Curves And Arrival Curves

In this section, we express the necessary and sufficient given by condition (3) in terms of service curves S_i's and arrival curves A_i's in Problem 1. We do this by distilling condition (3).

Consider a set X in condition (3), which includes a node w with label k but not all the nodes with that label in the W_i to which the node w belongs. Now, if we exclude all the nodes with label k from X, then the left-hand-side of the inequality (i.e., $t_S(X)$) in condition (3) does not change its value for this particular X. This holds for two reasons: (a) by noting how $t_S(X)$ is defined for any given S, observe that excluding nodes from X could change the value of $t_S(X)$ by only changing a $\Pi_i(X)$ for an i in S, (b) note in this case, however, that no $\Pi_i(X)$ could be changed, since the image of any node in any M_i includes *all* the nodes with label k in W_i if it includes a node with label k, by construction.

To state what we have just discussed above compactly, we introduce the following definition: A set of nodes in any W_i is called a ***block*** if it includes *all* the nodes in W_i with labels less than or equal to l and greater than or equal to k, if it includes a node with label k and a node with label l where $k \leqslant l$. Let us denote a block generically by B (suggesting 'Block') unless otherwise noted from here on.

By the above discussions, we have effectively proven of the following lemma.

Lemma 2 *Condition (3) is equivalent to*

$$t_S(X) \leqslant \sigma(X) \quad \text{for all } X \subseteq W,\ X \text{ is a union of blocks, and for all } S \subseteq I. \tag{4}$$

Proof: It is clear that (3) $\Rightarrow$ (4), since all sets X in (4) are also sets in (3). It is also clear that (4) $\Rightarrow$ (3), since compounding a set which is not a block to a set X in (4) could only increase the right-hand-side of the inequality, while the left-hand-side remains the same as we have explained before the lemma. ■

In Lemma 2, note that by X we are effectively considering a union of *intervals* of time over all the wireless links.

Next, we prove the following lemma.

Lemma 3 *Condition (4) is equivalent to*

$$\sum_{i \in I} \lceil \alpha_i \, |\Pi_i(X)| \rceil \leqslant \sigma(X) \quad \text{for all } X \subseteq W,\ X \text{ is a union of blocks.} \tag{5}$$

Proof: Essentially, the lemma holds since any flow i in Problem 1 conforms to arrival curve A_i.

Note that for any given X and S in Lemma 2, $t_S(X)$ will attain its maximum for some R_i's, where $R_i \sim A_i$, such that no packet arrives other than the principals of X (i.e., $\Pi_i(X)$) in flow i for any i in S. If we recall how $t_S(X)$ is defined, we would see that this holds since $|\Pi_i(X)|$ does not change its value if packets other than other than the principals of X would also arrive in flow i, while on the other hand $|M_i|$ does increase. Hence, in this case (i.e., when packets other than other than $\Pi_i(X)$ would also arrive), the value of $t_S(X)$ decreases. Thus, we see that $t_S(X)$ does indeed attain its maximum for some flows where no packet has arrived other than the principals of X as specified in condition (4).

For flows where no packet has arrived other than the principals of X, we could replace the summand in $t_S(X)$ by $\lceil \alpha_i \, |\Pi_i(X)| \rceil$, since in that case $|M_i| = |\Pi_i(X)|$. Now, since we see that the sum which yields $t_S(X)$ is non-decreasing by the cardinality of S, we could also replace the subscript '$i \in S$' of the sum by '$i \in I$'. This completes the one direction of the proof; i.e., (4) $\Rightarrow$ (5).

The other direction, (5) $\Rightarrow$ (4), follows clearly by the explanations given for the first direction and by backtracking them. ■

Note that in Lemma 3, once an X is specified, the summation is effectively carried out over the subset S of I such that X has an element in W_i for each i in S; i.e., $S = \{i : i \in I, X \cap W_i \neq \emptyset\}$.

Next, we find the maximum of the left-hand-side of the inequality in (5) for any given X. For that, it suffices to find the maximum of $|\Pi_i(X)|$, or equivalently that of $|\Pi_i(X \cap W_i)|$, for any $i \in I$. Note that, by $X \cap W_i$, we effectively consider a union of time-intervals over wireless link i, as we have indicated immediately after the proof of Lemma 2. Thus, in the following section, we find the maximum of $|\Pi_i(X \cap W_i)|$ for a given union of time-intervals $X \cap W_i$, over the set of all the flows which conform to A_i.

4.3.1 The Maximum of $|\Pi(X)|$

Fix an arrival curve A, a service curve S, and a capacity function $c(n)$ of a wireless link. Define the following set of flows;

$$\mathcal{R} \triangleq \{R : R \sim A\},$$

i.e., $\mathcal{R}$ is the set of all the flows which conform to A.

For any flow R in $\mathcal{R}$, construct the labeled graph $G' = (M' \cup W', E')$ representing the scenario where flow R is to be served service curve S over a wireless link with capacity $c(n)$, as we have constructed a labeled graph at the beginning of section 4.2.

In the labeled graph G' constructed above for any flow R, consider an X subset of W' in G', where X is a union of blocks. There is a unique *minimal* representation of X as a union of blocks. The representation $X = \bigcup_i B_i$, where all B_i's are blocks, is said to be *minimal* if the union of no two B_i's is a block. Roughly speaking, the minimal representation uses the "minimum" number of blocks to represent X. We assume without loss of generality from here on that the indices of the blocks in the minimal representation of X are all integers and such that the labels of all the nodes in B_i is less than those of B_j whenever $i < j$.

The minimal representation of X allows us to give the following lemma, where all the necessary explanations precede the lemma.

Lemma 4 *For any given X subset of W' in G', where $X = \bigcup_i B_i$ is the minimal representation of X, $\Pi(B_i)$'s form a partition of $\Pi(X)$.*

Proof: Since B_i's form the minimal representation of X, there exists at least one node w in W', that is not in X, between any two consecutive blocks B_i and B_{i+1} (i.e., the label of w is greater than the labels of all the nodes in B_i and less than those of B_{i+1}).

We first show that the principals of any two blocks B_i and B_j (in the minimal representation of X), where $i \neq j$, are disjoint; i.e., $\Pi(B_i) \cap \Pi(B_j) = \emptyset$. There can not exist a node in both $\Pi(B_i)$ and $\Pi(B_j)$, since the image of any node in M' is also a block. If a such common node were to exist in $\Pi(B_i)$ and $\Pi(B_j)$, then its image should also include all the nodes (i.e., w's) separating B_i from B_j, that we have explained in the previous paragraph. However, in this case, a such node would not be a principal of either B_i or B_j, since its image also includes a node not in $B_i \cup B_j$.

Secondly, we show that $\Pi(X) = \bigcup_i \Pi(B_i)$. It is clear that $\Pi(X) \supseteq \bigcup_i \Pi(B_i)$, since a principal of any B_i is obviously a principal of X too. To show the other direction (i.e., $\Pi(X) \subseteq \bigcup_i \Pi(B_i)$), suppose to the contrary that $\Pi(X)$ includes a node p which is not in $\bigcup_i \Pi(B_i)$. This means that the image of p has to have nodes in more than one B_i. However, since the image of p is a block, this also means that it can not be a principal of X, as its image would also include at least one node w separating B_i's. This completes the proof. ■

Now, consider the block B subset of W' in the labeled graph G' constructed for any flow R, such that the labels of the nodes in B are all greater than or equal to $k+1$ and less than or equal to $k+n$, for some k and n where $n \geqslant 1$. Note that block B specifies the time-interval $(k, n+k]$, as we have indicated after Lemma 2. The value of $|\Pi(B)|$ in any such G' is given by Lemma 6, where all the necessary explanations precede the lemma. In proving Lemma 6, we need the following lemma.

Lemma 5 *For any two non-decreasing functions f and g, the* min-+ convolution *$f \triangledown g$ is also non-decreasing.*

Proof: The proof follows immediately by the definition of min-+ convolution. For any $n_1 \leqslant n_2$, let us define the following two sets

$$T_i = \{f(k) + g(n_i - k) : k \leqslant n_i\} \quad \text{for } i \in \{1, 2\}.$$

Clearly, $(f \triangledown g)(n_i) = \min\{t_i : t_i \in T_i\}$ for $i \in \{1, 2\}$. Note that for any $k \leqslant n_1$, the term corresponding to k in T_1 is less than or equal to that in T_2, since $n_1 \leqslant n_2$ and g is non-decreasing. Also, for any k where $n_1 < k \leqslant n_2$, the term corresponding to k in T_2 is greater than or equal to the term corresponding to $k - (n_2 - n_1)$ in T_1 (note that $k - (n_2 - n_1) \leqslant n_1$); since $n_1 \leqslant n_2$ and f is non-decreasing. Thus, $(f \triangledown g)(n_1) \leqslant (f \triangledown g)(n_2)$. This concludes the proof. ■

Lemma 6 *For any given flow R, there holds $|\Pi(B)| = \max\{0, (R \triangledown S)(n + k) - R(k)\}$ in the labeled graph G' constructed for R.*

Proof: Note that any packet in R whose label is greater than $(R \triangledown S)(n + k)$ can not be among the principals of B; since a such packet would have a deadline larger than $n + k$ (this is shown below), its image would have a node not in B, hence can not belong to $\Pi(B)$. This is ensured by how deadlines are defined and by the capacity arrival(s) after $n+k$.

Observe by how deadlines are defined in (1) that for any packet l, where $l > (R\triangledown S)(n+k)$, its deadline D_l is greater than $n+k$;

$$\begin{aligned}
D_l &= \min\{t : (R\triangledown S)(t) \geqslant l\} && \text{by (1)} \\
&= \min\{t : (R\triangledown S)(t) \geqslant l > (R\triangledown S)(n+k)\} \\
&= \min\{t : t > n+k, (R\triangledown S)(t) \geqslant l\} && \text{since } R\triangledown S \text{ is non-decreasing by Lemma 5} \\
&> n+k\,.
\end{aligned}$$

Thus, at most $(R\triangledown S)(k+n)$ many packets (i.e., those with labels less than or equal to $(R\triangledown S)(k+n)$) could be among the principals of B, as we have already shown.

Similarly, any packet in R which arrives before time $k+1$ can not be among the principals of B; since the image of a such packet would have a node not in B, hence can not belong to $\Pi(B)$. This is ensured by the capacity arrival(s) before time $k+1$. Note that there are $R(k)$ many such nodes. Combining this by the observation in the previous paragraph completes the proof. ∎

The minimal representation of X specifies a union of disjoint time-intervals. In order to represent these union of time-intervals compactly, we introduce the following notation:

$$\mathbf{N} \triangleq \{\mathbf{n} : \mathbf{n} = \langle n_1, n_2, \ldots, n_{2k}\rangle,\ n_1 < n_2 < \cdots < n_{2k},\ 1 \leqslant k\}\,, \tag{6}$$

where each pair of n_{2i-1} and n_{2i} in any $\mathbf{n}$ represent the interval $(n_{2i-1}, n_{2i}]$.

Let $\mathbf{n} = \langle n_1, n_2, \ldots, n_{2k}\rangle$ stand for the union of time-intervals specified by the minimal representation of $X = \bigcup_{1\leqslant i\leqslant k} B_i$, where B_i specifies the interval $(n_{2i-1}, n_{2i}]$. It follows by Lemmas 4 and 6 that

$$|\Pi(X)| = \sum_{1\leqslant i\leqslant k} \big((R\triangledown S)(n_{2i}) - R(n_{2i-1})\big)^+, \tag{7}$$

where the notation z^+ stands for $\max\{0, z\}$.

Next, we determine the maximum of $|\Pi(X)|$ over all R in $\mathcal{R}$. For that, we define the following functions H_i's and π for $\mathbf{n} = \langle n_1, n_2, \ldots, n_{2k}\rangle$ specified by X,

$$H_1(n) = A(n - n_1) \tag{8}$$

$$\begin{aligned}
H_{i+1}(n) &= \min\Big\{\min\{H_i(n), (H_i\triangledown S)(n_{2i})\} + A(n - n_{2i+1}),\ H_i(n)\Big\} && (9)\\
&= \min\{H_i(n),\ A(n - n_{2i+1}) + (H_i\triangledown S)(n_{2i})\} \qquad \text{for } 1 \leqslant i < k && (10)
\end{aligned}$$

$$\pi(\mathbf{n}) = (H_k\triangledown S)(n_{2k})\,. \tag{11}$$

The maximum of $|\Pi(X)|$ over all R in $\mathcal{R}$ is given by Lemma 13. In proving Lemma 13, we need the following lemmas.

Lemma 7 *Let R, R_1 and R_2 be any three flows which all conform to an arrival curve A. The following statements hold;*

1. *for any constant t, $R(n-t)$ conforms to A,*
2. *for any constant $a \geqslant 0$, $R(n) + a$ conforms to A,*
3. *the flow $\min\{R_1(n), R_2(n)\}$ conforms to A.*

Proof: The proofs follow directly from Definition 1:

1. Let $F(n) \equiv R(n-t)$. It holds for all k and n that

$$\begin{aligned} F(n+k) - F(k) &= R(n+k-t) - R(k-t) \\ &= R(n+u) - R(u) \qquad\qquad u = k-t \\ &\leqslant A(n)\,. \end{aligned}$$

2. Let $F(n) \equiv R(n) + a$. It holds for all k and n that

$$\begin{aligned} F(n+k) - F(k) &= (R(n+k)+a) - (R(k)+a) \\ &= R(n+k) - R(k) \\ &\leqslant A(n)\,. \end{aligned}$$

3. Let $F(n) \equiv \min\{R_1(n), R_2(n)\}$. It is trivial to show that $F(n)$ is non-negative and non-decreasing, hence a flow. It holds for all k and n that

$$F(n+k) - F(k) = \min\{R_1(n+k), R_2(n+k)\} - \min\{R_1(k), R_2(k)\}$$

   if the first minimum is equal to $R_1(n+k)$, then we have

$$\begin{aligned} &= R_1(n+k) - \min\{R_1(k), R_2(k)\} \\ &= R_1(n+k) + \max\{-R_1(k), -R_2(k)\} \\ &= \max\{R_1(n+k) - R_1(k),\ R_1(n+k) - R_2(k)\} \\ &\leqslant \max\{R_1(n+k) - R_1(k),\ R_2(n+k) - R_2(k)\} \\ &\leqslant \max\{A(n), A(n)\} \\ &= A(n)\,. \end{aligned}$$

   The other case where the first minimum is equal to $R_2(n+k)$ follows similarly. ■

Lemma 8 *Let f_1, f_2, and g be any three functions. Define another function f as $f(n) = \min\{f_1(n), f_2(n)\}$. The following equality holds;*

$$(f \triangledown g)(n) = \min\{(f_1 \triangledown g)(n),\ (f_2 \triangledown g)(n)\} \quad \textit{for all } n.$$

Proof: The proof follows immediately from the definition of min-+ convolution and the properties of minimum, as shown below. It holds for all n that

$$\begin{aligned} (f \triangledown g)(n) &= \min_{k \leqslant n}\{f(k) + g(n-k)\} \\ &= \min_{k \leqslant n}\big\{\min\{f_1(k), f_2(k)\} + g(n-k)\big\} \\ &= \min_{k \leqslant n}\big\{\min\{f_1(k) + g(n-k),\ f_2(k) + g(n-k)\}\big\} \\ &= \min\big\{\min_{k \leqslant n}\{f_1(k) + g(n-k)\},\ \min_{k \leqslant n}\{f_2(k) + g(n-k)\}\big\} \\ &= \min\{(f_1 \triangledown g)(n),\ (f_2 \triangledown g)(n)\}\,. \end{aligned}$$

■

Lemma 9 *Let f and g be any two functions and a be an arbitrary constant. Define another function h as $h(n) = f(n) + a$. The following equality holds;*

$$(h \triangledown g)(n) = (f \triangledown g)(n) + a \quad \textit{for all } n.$$

Proof: The proof follows immediately from the definition of min-+ convolution, as shown below. It holds for all n that

$$\begin{aligned}(h \triangledown g)(n) &= \min_{k \leqslant n}\{h(k) + g(n-k)\} \\ &= \min_{k \leqslant n}\{f(k) + a + g(n-k)\} \\ &= \min_{k \leqslant n}\{f(k) + g(n-k)\} + a \\ &= (f \triangledown g)(n) + a \qquad \blacksquare\end{aligned}$$

Lemma 10 *Let f and g be any two functions and t be an arbitrary constant. Define another function h as $h(n) = f(n-t)$. The following equality holds;*

$$(h \triangledown g)(n) = (f \triangledown g)(n-t) \quad \textit{for all } n.$$

Proof: The proof follows immediately from the definition of min-+ convolution, as shown below. It holds for all n that

$$\begin{aligned}(h \triangledown g)(n) &= \min_{k \leqslant n}\{h(k) + g(n-k)\} \\ &= \min_{k \leqslant n}\{f(k-t) + g(n-k)\} \\ &= \min_{(u+t) \leqslant n}\big\{f(u) + g(n-(u+t))\big\} \\ &= \min_{u \leqslant (n-t)}\{f(u) + g(n-t-u)\} \\ &= (f \triangledown g)(n-t)\,. \qquad \blacksquare\end{aligned}$$

Lemma 11 *Let R be any flow conforming to an arrival curve A and S be a service curve. The following inequality holds;*

$$\big((R \triangledown S)(t_2) - R(t_1)\big)^+ \leqslant (A \triangledown S)(t_2 - t_1) \quad \textit{for any } t_1 \leqslant t_2.$$

Proof: It holds for any $t_1 \leqslant t_2$ that

$$\begin{aligned}\big((R \triangledown S)(t_2) - R(t_1)\big)^+ &= \max\{0, (R \triangledown S)(t_2) - R(t_1)\} \\ &= \max\big\{0, \min_{u \leqslant t_2}\{R(u) + S(t_2-u)\} - R(t_1)\big\} \\ &\leqslant \max\big\{0, \min_{t_1 \leqslant u \leqslant t_2}\{R(u) + S(t_2-u)\} - R(t_1)\big\} \\ &= \max\big\{0, \min_{t_1 \leqslant u \leqslant t_2}\{R(u) - R(t_1) + S(t_2-u)\}\big\} \\ &\leqslant \max\big\{0, \min_{t_1 \leqslant u \leqslant t_2}\{A(u-t_1) + S(t_2-u)\}\big\} \\ &= \max\big\{0, \min_{0 \leqslant u-t_1 \leqslant n}\{A(u-t_1) + S(t_2-t_1-(u-t_1))\}\big\} \\ &= \max\big\{0, \min_{0 \leqslant v \leqslant n}\{A(v) + S(t_2-t_1-v)\}\big\} \\ &= \min_{0 \leqslant v \leqslant n}\{A(v) + S(t_2-t_1-v)\}\end{aligned}$$

since $A(v) = 0$ for all $v \leqslant 0$, and since S is non-decreasing, we also have

$$= (A \triangledown S)(t_2 - t_1)\,. \qquad \blacksquare$$

Lemma 12 *For any given* $\mathbf{n} = \langle n_1, n_2, \ldots, n_{2k}\rangle$*, there holds*

$$\sum_{1\leqslant i\leqslant k} \big((H_k \triangledown S)(n_{2i}) - H_k(n_{2i-1})\big)^+ = \pi(\mathbf{n})\,,$$

where H_k *and* $\pi(\mathbf{n})$ *are defined in (8), (10), and (11).*

Proof: First, observe the following for any i where $1 \leqslant i < k$:

1. $H_{i+1}(n_{2i+1}) = (H_i \triangledown S)(n_{2i})\,,$ (12)

   which follows immediately by substituting n_{2i+1} in (10) and the simple observation that

   $$(H_i \triangledown S)(n_{2i}) \leqslant H_i(n_{2i}) \leqslant H_i(n_{2i+1})\,.$$

2. $(H_{i+1} \triangledown S)(n) = (H_i \triangledown S)(n) \quad \text{for all } n \leqslant n_{2i}\,,$ (13)

   which follows by taking the min-+ convolution of both sides in (10) with S and using Lemmas 8, 9, and 10.

3. $(H_{i+1} \triangledown S)(n_{2i+2}) - H_{i+1}(n_{2i+1}) \geqslant 0\,,$ (14)

   which follows by (12), (13), and Lemma 5 as shown below

   $$\begin{aligned}(H_{i+1} \triangledown S)(n_{2i+2}) - H_{i+1}(n_{2i+1}) &= (H_{i+1} \triangledown S)(n_{2i+2}) - H_i(n_{2i}) \\ &= (H_{i+1} \triangledown S)(n_{2i+2}) - H_{i+1}(n_{2i}) \\ &\geqslant 0\,.\end{aligned}$$

4. $H_{i+1}(n) = H_i(n) \quad \text{for all } n \leqslant n_{2i-1}\,,$ (15)

   which follows from (10) by using (14).

We now show that the equality in the lemma holds:

$$\begin{aligned}
&\sum_{1\leqslant i\leqslant k} \big((H_k \triangledown S)(n_{2i}) - H_k(n_{2i-1})\big)^+ \\
&= \big((H_k \triangledown S)(n_{2k}) - H_k(n_{2k-1})\big)^+ + \sum_{1\leqslant i\leqslant k-1} \big((H_k \triangledown S)(n_{2i}) - H_k(n_{2i-1})\big)^+ \\
&= (H_k \triangledown S)(n_{2k}) - H_k(n_{2k-1}) + \sum_{1\leqslant i\leqslant k-1} \big((H_k \triangledown S)(n_{2i}) - H_k(n_{2i-1})\big)^+ && \text{by (14)} \\
&= (H_k \triangledown S)(n_{2k}) - (H_{k-1} \triangledown S)(n_{2k-2}) + \sum_{1\leqslant i\leqslant k-1} \big((H_k \triangledown S)(n_{2i}) - H_k(n_{2i-1})\big)^+ && \text{by (12)} \\
&= (H_k \triangledown S)(n_{2k}) - (H_{k-1} \triangledown S)(n_{2k-2}) + \sum_{1\leqslant i\leqslant k-1} \big((H_{k-1} \triangledown S)(n_{2i}) - H_{k-1}(n_{2i-1})\big)^+ && \text{by (13) and (15)}
\end{aligned}$$

by repeatedly applying the same arguments to the summation in the last line above, we obtain the telescopic sum below

$$\begin{aligned}
&= \big((H_1 \triangledown S)(n_2) - (H_1 \triangledown S)(n_1)\big)^+ + \sum_{1\leqslant i< k} \big((H_{i+1} \triangledown S)(n_{2i+2}) - (H_i \triangledown S)(n_{2i})\big) \\
&= (H_1 \triangledown S)(n_2) + \sum_{1\leqslant i< k} \big((H_{i+1} \triangledown S)(n_{2i+2}) - (H_i \triangledown S)(n_{2i})\big) \\
&= (H_{2k} \triangledown S)(n_{2k}) \\
&= \pi(\mathbf{n})\,.
\end{aligned}$$

■

We now give the lemma showing the maximum of $|\Pi(X)|$ over all R in $\mathcal{R}$.

Lemma 13 $\max_{R\in\mathcal{R}}\{\,|\Pi(X)|\,\} = \pi(\mathbf{n})$, *where all the necessary explanations precede the lemma.*

Although the proof of Lemma 13 is lengthy, the idea behind it is simple: We have constructed the functions H_i's in such a way that H_k is actually a flow giving the maximum of $|\Pi(X)|$ over all R in $\mathcal{R}$. To have this property, we have first placed the maximal arrival pattern possible at n_1, effective onward from n_1; that is, $A(n-n_1)$. This flow is denoted by H_1; this is line (8). We already know by Lemma 6 that only the first $(H_1 \triangledown S)(n_2)$ many packets will be among the principals of B_1. Thus, we "turn-off" H_1 after the first $(H_1 \triangledown S)(n_2)$ many packets would arrive, in order to save the "burst-power" of the flow; that is, we do not allow any more packets to arrive in $(n_1, n_2]$. This corresponds to the minimum inside the minimum in (9). Then, at the beginning of the next interval (i.e., at time n_3), we "turn-on" the flow again in full-blast conforming to A. This corresponds to the whole line in (9). We keep turning the flow "on" and "off" in this manner until the last interval represented in $\mathbf{n}$, and obtain H_k. The value $\pi(\mathbf{n})$ gives the total number of principals of X, due to H_k, and it is the maximum over all R in $\mathcal{R}$. The proof formally shows this.

Proof of Lemma 13: It is easy to show that each function H_i defined in (8) and (10) is non-decreasing, hence can be viewed as a flow: First, H_1 is clearly non-decreasing since A is an arrival curve which by definition is non-decreasing. Second, assuming H_i is non-decreasing, H_{i+1} is also non-decreasing since it is the minimum of two non-decreasing functions—it is trivial to show this.

It is also easy to show that each H_i belongs to $\mathcal{R}$: First, H_1 belongs to $\mathcal{R}$ by Lemma 7 (item 1). Second, assuming H_i conforms to A, H_{i+1} also conforms to A by (10) and Lemma 7.

Thus, by (7) and the above remarks, we have

$$\begin{aligned}\max_{R\in\mathcal{R}}\{\,|\Pi(X)|\,\} &\geqslant \sum_{1\leqslant i\leqslant k}\big((H_k \triangledown S)(n_{2i}) - H_k(n_{2i-1})\big)^+ && \text{by (7) and the above remarks}\\ &= \pi(\mathbf{n}) && \text{by Lemma 12.}\end{aligned}$$

Next, we show the reverse inequality, i.e., $\max_{R\in\mathcal{R}}\{\,|\Pi(X)|\,\} \leqslant \pi(\mathbf{n})$, and complete the proof. For that, we need the following observations:

1. For any flow R, service curve S, and integers $t_1 < t_2 < t_3 < t_4$, there holds

$$\big((R\triangledown S)(t_4) - R(t_3)\big)^+ + \big((R\triangledown S)(t_2) - R(t_1)\big)^+ \leqslant \big((R\triangledown S)(t_4) - R(t_1)\big)^+, \tag{16}$$

which is shown below:

$$\begin{aligned}&\big((R\triangledown S)(t_4) - R(t_3)\big)^+ + \big((R\triangledown S)(t_2) - R(t_1)\big)^+\\ &= \max\{0, (R\triangledown S)(t_4) - R(t_3)\} + \max\{0, (R\triangledown S)(t_2) - R(t_1)\}\\ &= \max\big\{0,\ (R\triangledown S)(t_4) - R(t_3),\ (R\triangledown S)(t_2) - R(t_1),\ (R\triangledown S)(t_4) - R(t_3) + (R\triangledown S)(t_2) - R(t_1)\big\}\\ &\leqslant \max\big\{0,\ (R\triangledown S)(t_4) - R(t_3),\ (R\triangledown S)(t_2) - R(t_1),\ (R\triangledown S)(t_4) - R(t_3) + R(t_2) - R(t_1)\big\}\\ &\leqslant \max\big\{0,\ (R\triangledown S)(t_4) - R(t_3),\ (R\triangledown S)(t_2) - R(t_1),\ (R\triangledown S)(t_4) - R(t_1)\big\}\\ &= \big((R\triangledown S)(t_4) - R(t_1)\big)^+.\end{aligned}$$

2. For any given $\mathbf{n} = \langle n_1, n_2, \ldots, n_{2k}\rangle$, define for each i where $1 \leqslant i \leqslant k$

$$\mathbf{Z_i} \triangleq \big\{\mathbf{z} : \mathbf{z} = \langle z_1, z_2, \ldots, z_l\rangle,\ 1 = z_1 < z_2 < \cdots < z_l,\ z_j \in \{1, 3, \ldots, 2i-1\},\ 1 \leqslant l\big\}.$$

Now, for any $\mathbf{z} = \langle z_1, z_2, \ldots, z_l\rangle$ in $\mathbf{Z_k}$, the following inequalities hold: For any R in $\mathcal{R}$ and X whose minimal representation specifies $\mathbf{n} = \langle n_1, n_2, \ldots, n_{2k} \rangle$, we have

$$\begin{aligned}
|\Pi(X)| &= \sum_{1 \leqslant j \leqslant k} \big((R \triangledown S)(n_{2j}) - R(n_{2j-1})\big)^+ && \text{by (7)} \\
&\leqslant \big((R \triangledown S)(n_{2k}) - R(n_{z_l})\big)^+ + \sum_{1 \leqslant j < l} \big((R \triangledown S)(n_{z_{j+1}-1}) - R(n_{z_j})\big)^+ && \text{by repeatedly applying (16)} \\
&\leqslant (A \triangledown S)(n_{2k} - n_{z_l}) + \sum_{1 \leqslant j < l} (A \triangledown S)(n_{z_{j+1}-1} - n_{z_j}) && \text{by Lemma 11.}
\end{aligned}$$

Since the right-hand-side of the last inequality above is independent from R, we also have

$$\max_{R \in \mathcal{R}} \{ |\Pi(X)| \} \leqslant (A \triangledown S)(n_{2k} - n_{z_l}) + \sum_{1 \leqslant j < l} (A \triangledown S)(n_{z_{j+1}-1} - n_{z_j}) .$$

Furthermore, since the above inequality holds for any $\mathbf{z}$ in $\mathbf{Z_k}$, we get

$$\max_{R \in \mathcal{R}} \{ |\Pi(X)| \} \leqslant \min_{\mathbf{z} \in \mathbf{Z_k}} \left\{ (A \triangledown S)(n_{2k} - n_{z_l}) + \sum_{1 \leqslant j < l} (A \triangledown S)(n_{z_{j+1}-1} - n_{z_j}) \right\}. \tag{17}$$

We conclude the proof by showing that the right-hand-side of the inequality in (17) is equal to $\pi(\mathbf{n})$. We do this by showing a stronger claim: There holds for all i, where $1 \leqslant i \leqslant k$, that

$$(H_i \triangledown S)(n) = \min_{\mathbf{z} \in \mathbf{Z_i}} \left\{ (A \triangledown S)(n - n_{z_l}) + \sum_{1 \leqslant j < l} (A \triangledown S)(n_{z_{j+1}-1} - n_{z_j}) \right\} \quad \text{for all } n. \tag{18}$$

We show (18) by induction.

Basis: First of all, for $i = 1$, recall from (8) that $H_1(n) = A(n - n_1)$. Hence, it follows from Lemma 10 that $(H_1 \triangledown S)(n) = (A \triangledown S)(n - n_1)$. Now, note that the right-hand-side of the claimed equality in (18) is also just equal to $(A \triangledown S)(n - n_1)$ since $\mathbf{Z_1}$ only includes $\langle 1 \rangle$, hence in this case the summation inside the minimum is zero. Thus, (18) holds for $i = 1$.

Induction Step: Suppose, (18) holds for some i where $1 \leqslant i < k$. It follows from (10) by using Lemmas 8, 9, and 10 that

$$(H_{i+1} \triangledown S)(n) = \min \big\{ (H_i \triangledown S)(n),\ (A \triangledown S)(n - n_{2i+1}) + (H_i \triangledown S)(n_{2i}) \big\}. \tag{19}$$

The induction hypothesis implies that the right-hand-side of (19) is equal to the right-hand-side of the claimed equality in (18) for $i+1$. We show this by first noting that the term $(H_i \triangledown S)(n)$ inside the minimum in (19) contributes all the terms in (18) corresponding to $\mathbf{z}$'s, which do not include $2i+1$, in $\mathbf{Z_{i+1}}$ for $i+1$. All the other terms in (18) for $i+1$ (i.e., those corresponding to $\mathbf{z}$'s, which do include $2i+1$, in $\mathbf{Z_{i+1}}$) are contributed by the other term inside the minimum in (19). This completes showing that (18) holds.

Finally, substituting k and n_{2k} in place of i and n, respectively, in (18) readily shows that the right-hand-side of the inequality in (17) is indeed equal to $(H_k \triangledown S)(n_{2k})$, hence also equal to $\pi(\mathbf{n})$. This concludes the proof. ∎

4.3.2 The Answer

Now that we can determine the maximum of the left-hand-side of the inequality in Lemma 3 by using Lemma 13, we could give the necessary and sufficient condition asked in Problem 1. For that, we define the following functions:

1. For each flow j in I, define a function $\pi_j(\cdot)$ as in (8), (10), and (11) simply by replacing A with A_j and S with S_j.

2. Define a function C on the power set of $\mathbf{N}$ (defined in (6)) as

$$C(\mathbf{n}_i : i \in S) = \sum_n \min\Big\{g(n),\ \sum_{i\in S}\big(c_i(n)\cdot[n \in U_{\mathbf{n}_i}]\big)\Big\},$$

where $\mathbf{n}_i$'s belong to $\mathbf{N}$, S is a subset of I, the bracketed notation [`statement`] stands for 1 if the `statement` is true, 0 otherwise; and, for a given $\mathbf{n}_i = \langle n_1, n_2, \ldots, n_{2k}\rangle$ $U_{\mathbf{n}_i}$ is defined as

$$U_{\mathbf{n}_i} = \bigcup_{1\leqslant l\leqslant k}(n_{2l-1}, n_{2l}]\,.$$

Note that for a given S and a set of $\mathbf{n}_i$'s, the value $C(\mathbf{n}_i : i \in S)$ stands for the allowable capacity over all the wireless links in S over the time-intervals represented by $\mathbf{n}_i$'s.

The theorem below provides an answer to Problem 1.

Theorem 2 *There exists a scheduling for the delivery of the services as specified in Problem 1* if and only if *the following condition holds;*

$$\sum_{i\in S}\lceil \alpha_i\cdot\pi_i(\mathbf{n}_i)\rceil \leqslant C(\mathbf{n}_i : i\in S)\quad \textit{for all } S\subseteq I \textit{ and for all } \mathbf{n}_i\in\mathbf{N},$$

where all the necessary explanations/definitions are given before the theorem.

Proof: The proof follows by Lemmas 3 and 13. ∎

5 The Complexity of The Condition

For subsets S of I, which have only one element, i.e., $S = \{i\}$, each one of the conditions in Theorem 2 can further be reduced to

$$\lceil \alpha_i\cdot(A_i \triangledown S_i)(n)\rceil \leqslant C_i(n+k) - C_i(k)\quad \text{for all } n \text{ and } k, \tag{20}$$

where $C_i(n) \triangleq \sum_{l\leqslant n} c_i(l)$. See for example [16] for the reduction. In this case (i.e., when $|S| = 1$), the correctness of the conditions can be checked at most in $O(n^2)$ time, where n is the duration of flow i (note that n is finite since we have assumed $R_i(\infty) < \infty$). More specifically, if the capacity of wireless link i is given to be time-varying, then to check the correctness of condition (20) $n(n+1)/2$ many comparisons (i.e., '$\leqslant$') and for each comparison three basic numerical computations (a ceiling, a multiplication, and a subtraction) have to be done; i.e., a total of $2n(n+1)$ many basic computations have to be performed. Hence, the complexity of the conditions in this case is $O(n^2)$.

For subsets S of I, which have more than one element, the complexity of the condition in Theorem 2 is at least exponential. This is due to the fact that a union of disjoint time-intervals (represented by $\mathbf{n}_i$) over each wireless link i has to be considered to check the correctness of the conditions.

If the condition in Lemma 3 was able to be reduced to a condition where each $X \cap W_i$ was a block instead of a union of blocks, i.e., to

$$\sum_{i\in I}\lceil \alpha_i\,|\Pi_i(X)|\rceil \leqslant \sigma(X)\quad \text{for all } X\subseteq W, \text{ each } X\cap W_i \text{ is a block}, \tag{21}$$

then the complexity of the condition in certain cases would still be $O(n^2)$. However, it turns out that the condition in Lemma 3 can not be reduced to (21). Figure 1 provides counterexamples to show this.

The graphs in Figure 1 should be viewed by following the explanations given in constructing a labeled graph at the beginning of section 4.2. In order not to clutter the figures, the nodes are not labeled and the sets M_i's, W_i's, and V_n's are not explicitly marked. However, the nodes lie on each dashed line marked

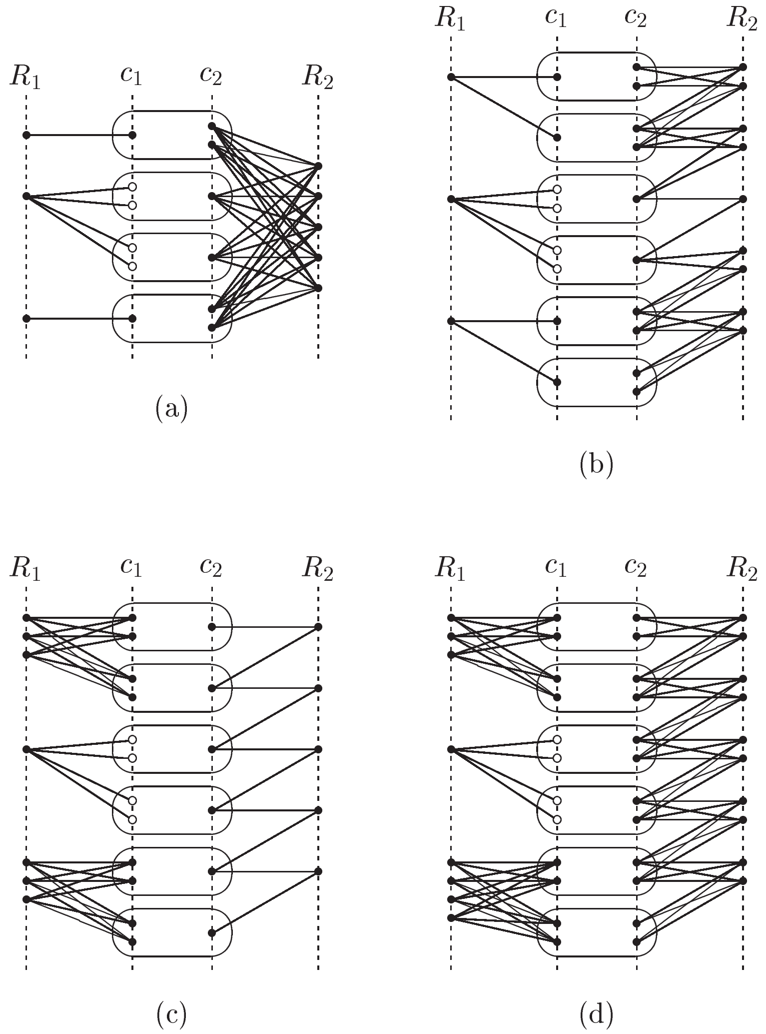

Figure 1: Counterexamples to show that the condition in Lemma 3 can not be reduced to (21). The graphs differ from one another by either the delay requirements being realized or the capacity "arrivals".

by R_i, $i \in \{1, 2\}$, stand for the packets in flow R_i (i.e., the nodes in M_i). Similarly, the nodes lie on each dashed line marked by c_i, $i \in \{1, 2\}$, stand for the capacity of transmitting a single packet in a time-slot over wireless link i (i.e., the nodes in W_i). The image of any node in any M_i is shown completely. The nodes included in each oval Venn diagram represent the capacity "arrivals" at a time-slot n over wireless links 1 and 2 (i.e., a V_n). The Venn diagrams in each graph are marked consecutively in increasing order from top to bottom. The constraint on any V_n is one less than the number of nodes included in V_n; i.e., at most $g(n) = |V_n| - 1$ nodes could be matched in any V_n. The parameters α_1 and α_2 are set equal to 1 in any graph. The graphs are constructed in such a way that checking the conditions in Lemma 3 and (21) is easy. Now, note that although condition (21) holds for any graph G, a matching does not exist in G such that all the nodes in $M = M_1 \cup M_2$ are matched. On the other hand, this is correctly concluded by the condition in Lemma 3, which holds for all but one X in any graph G. The set X for which the condition does not hold in Lemma 3 is the set which includes all the nodes in $W = W_1 \cup W_2$ except the four nodes where each is indicated by a circle '$\circ$'. Note that for this X in any graph, $X \cap W_1$ is not a block.

6 An Algorithm

We briefly describe a heuristic on-line algorithm to implement our approach explained in the introduction. The algorithm being described tries to approximate a scheduling policy (i.e., a matching) that the condition for its existence is given by Lemma 1—recall that a matching as specified in Problem 3 corresponds to an appropriate scheduling essentially sought in Problem 1. We would like to note that a scheduling mentioned in Lemma 1 is not necessarily *causal* (i.e., on-line).

We describe the algorithm in the context of downlink communications explained in the introduction. A similar algorithm could also be described for uplink communications.

Consider a base-station serving mobile hosts indexed by $I = \{1, 2, \ldots, |I|\}$. Slot time into intervals of length Δ. The fixed-length Δ can be chosen according to the *coherence time*[8] of the wireless channels between the base-station and the mobile hosts. The slotting of time in this context was previously done in the literature, e.g., [1, 2, 23, 24, 25]. The base-station estimates the capacity of the wireless channels between itself and the mobile hosts for a multiple of time-slots periodically, based on the interference control functions (i.e., g function in Problem 1) given for other base-stations interfering with this particular base-station. Channel estimation in this context had been previously used in the literature, e.g., [1, 25]. Based on the channel estimations, suppose that the base-station chooses to transmit information over any wireless channel in any time-slot at one of the constant rates which belong to $\mathrm{R} = \{r_1, r_2, \ldots, r_{|\mathrm{R}|}\}$. Hence, the capacity $c_i(n)$ of the wireless channel between the base-station and the mobile host i, for each i in I, is given as a sequence of numbers which belong to R. The cardinality of R is chosen to be small, for instance about ten. A such set of available transmission rates is utilized in Qualcomm HDR; see Table 1 in [1].

Each mobile host i in I specifies a service curve S_i with loss parameter $1 - \alpha_i$, with which it requests transmission of information from the base-station. The requested stream of packets destined to mobile host i is represented by flow R_i which is known by the base-station only up to the current time. The base-station maintains a separate queue for each flow R_i. The deadline of each packet is computed according to (1) and maintained by the base-station. The interference control function of the base-station is specified to be $g(n)$ in bits, which is chosen considering the interference that the base-station causes to other base-stations. For practical purposes, we assume that $g(n)$ is given as a sequence of integers which belong to $\mathrm{M} = \{m_1, m_2, \ldots, m_{|\mathrm{M}|}\}$.

Next, we perform discretization so as to comply with the formulation of Problem 1. For all practical purposes, we assume that both Δ and r_j's are rational numbers. The quantity $\Delta \cdot r_j$ represents the amount of bits that the base-station can transmit in any time-slot over any wireless link when it chooses to transmit information with rate r_j. Suppose that each flow R_i is composed of packets of fixed-length, where the fixed packet-length is given to be P bits. Represent the quantity $\frac{\Delta \cdot r_j}{P}$, for each j in R, as $\frac{p_j}{q_j}$ where the greatest common divisor of the integers p_j and q_j is 1. Similarly, represent the quantity $\frac{m_j}{P}$, for each j in M, as $\frac{a_j}{b_j}$; again, the greatest common divisor of the integers a_j and b_j is 1. Find the least common multiple of the integers in $\{q_1, q_2, \ldots, q_{|\mathrm{R}|}\} \cup \{b_1, b_2, \ldots, b_{|\mathrm{M}|}\}$. Let this least common multiple be denoted by q. Transform the flows, estimated capacity functions, and the interference control function for all n as

$$\begin{aligned} R'_i(n) &\triangleq q \cdot R_i(n) && \text{for all } i \in I, \\ c'_i(n) &\triangleq q \cdot \frac{\Delta \cdot c_i(n)}{P} && \text{for all } i \in I, \\ g'(n) &\triangleq q \cdot \frac{g(n)}{P}. \end{aligned}$$

In other words, we scale the variables $c_i(n)$'s and $g(n)$ so that the units of the transformed variables become packets.

[8]Refer to, for example, [5].

The deadlines of the packets in any transformed flows R'_i do not change with respect to those of R_i. In other words, for any packet k in R_i with deadline D_k, there are q packets in R'_i with the same deadline. Or, effectively, we also transform the service curves as $S'_i(n) \triangleq q \cdot S_i(n)$.

Finally, in order to have a uniform notation, let α'_i be identical to α_i for all i in I.

After completing the transformation phase as described above, the algorithm proceeds as follows.

The base-station maintains two counters $\texttt{sent}_i$ and $\texttt{dropped}_i$ for each flow R'_i. The counters hold the number of sent and dropped packets by the current time, respectively, for flow R'_i. We try to do the best we can in any time-slot n. We consider the relative deviation of $\alpha'_i = \frac{\texttt{sent}_i}{\texttt{sent}_i + \texttt{dropped}_i}$ from α_i. We try to schedule as many packets as possible over each wireless link i, starting from the packets with the earliest deadlines, but do not exceed a total of $g'(n)$. If for this initial schedule, some packets turn out to be expiring, then we try to accommodate them within the loss tolerance of the flows from which packets have been scheduled for transmission at time n. We do this considering the relative deviations indicated above.

Better algorithms can be designed by taking advantage of the available future information about the channel behavior. Similarly, by specifying a characterization for flows, future arrivals can be estimated and a better algorithm can be constructed.

We plan to measure the performance of such an on-line algorithm by comparing it to that of an off-line algorithm obtained by solving a matching problem in each channel estimation period where the arrival patterns of the on-line algorithm is given as input that the off-line algorithm.

7 Related Works

Some of the related works to this study are [1, 2, 23, 24, 25]. The interference control technique examined in this study is not considered in these works. The studies in [1, 25] and [2, 23, 24] are specific for downlink and uplink communications in CDMA networks, respectively.

In [1], a scheme to maximize throughput (i.e., the total number of packets served over all the flows) at any time n is presented. No QoS guarantees are considered. Although throughput maximization may be adequate for low-load applications (load is roughly the mean data-rate divided by the mean offered service-rate), it is not suitable for applications with moderate to high load (typically larger than 0.5) when some QoS guarantees (such as delay and loss) are desired to be delivered to these applications. It is shown in [23, 24] that throughput maximization conflicts with guaranteeing QoS for applications with moderate to high load.

The service curve model is used explicitly in [23, 24] too and implicitly in [2]. However, a loss aspect is not incorporated into the model used in [23, 24]. To account for loss resulting from the authors' approach, a deviation from the delivery of service curves has been tried to be minimized heuristically in [24]. In [23], that deviation is only observed by a plot. In [2], the packet success rate (i.e., the fraction of packets being served within their deadlines) per class of traffic is also only observed by a plot. In the end, an application's tolerance to loss has not been taken into account, and hence not utilized in [2, 23, 24]. In our approach, applications' tolerance to loss is incorporated into the service model analytically. The incorporated loss aspect allows a precise statement of the service that an application receives whatever the incurred loss might be. Furthermore, if needed, it also allows a mechanism to minimize deviation from the requested loss-guarantees, which is more meaningful to do.

Interference control is realized via power control in [1, 2] and via both power and rate control in [23, 24]. No interference control is considered in [25]. Both power and rate control schemes are performed slot by slot.

The problems examined in [2, 23, 24] are all formulated as a Linear Programming problem per time-slot (i.e., the optimization is performed slot by slot). The performance judgments of their proposed solutions are based on some simulations that they have carried out. It would be useful to compare the performances of their solutions with those of some similar Linear Programming problems with global view (i.e., with infinite time horizon) which would provide benchmark results.

The problems examined in [25] are also formulated as Linear Programming problems, but not in a slot-by-slot fashion. Interference control and loss tolerance are not considered in [25], but delay requirements are. The work in [25] mainly provides some complexity and optimality results. The optimality criterion is based on the concept of *competitiveness*.

8 Conclusion

We have presented an approach to control interference in wireless cellular networks with the goal of supporting high-data-rate services with explicit QoS guarantees. Interference control in our approach is implemented by limiting the total number of packet transmissions occurring in any time-interval over a set of interfering wireless channels. The QoS guarantees are implemented by adopting a service model provided by Definition 4 which incorporates delay and loss guarantees. The guarantees for offered service-rates are embedded into the delay-guarantees, both of which are managed by service curves. Our approach is applicable to both uplink and downlink communications, as explained in the introduction.

To provide insight into our approach, we have solved a scheduling problem (i.e., Problem 1). Problem 1 is solved by a graph-theoretic approach. The solution is given by Theorem 2 which provides a feasibility condition. We have shown that the number of basic computations required to check the correctness of this condition grows at least exponentially by n when there are more than one flow to support. Justified by this barrier, we have proposed heuristic on-line algorithm in Section 6.

Future work includes implementing the proposed algorithm. Results of the current study (specifically, Lemma 2) will be used to provide a benchmark to measure the performance of the proposed algorithm. Finding the capacity of a wireless channel for a given interference control function (i.e., g function in Problem 1) is also among the future work. Such capacity calculations will later be utilized in our approach, which will facilitate a better comparison of the approach with other approaches to support high-data-rate services.

References

[1] P. Bender, P. Black, M. Grob, R. Padovani, N. Sindhushayana, and A. Viterbi. *CDMA/HDR: A Bandwidth-Efficient High-Speed Wireless Data Service for Nomadic Users*, IEEE Communications Magazine, vol. 38, no. 7, pp. 70–77, July 2000.

[2] M. Elaoud, P. Ramanathan. *Adaptive Allocation of CDMA Resources for Network-level QoS Assurances*, Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MOBICOM), pp. 191–199, 2000.

[3] S. Haykin. *Communications Systems*, John Wiley & Sons, 4th edition, 2000.

[4] W. Feller. *An Introduction to Probability Theory and Its Applications, Volume 1*, John Wiley & Sons, 3rd edition, 1968.

[5] T. S. Rappaport. *Wireless Communications: Principles and Practice*, Prentice Hall, 2nd edition, 2001.

[6] R. L. Cruz. *A Calculus For Network Delay, Part I: Network Elements In Isolation*, IEEE Transactions on Information Theory, vol. 37, no. 1, pp. 114–131, Jan. 1991.

[7] F. Baccelli, G. Cohen, G. Oslder, and J. Quadrat, *Synchronization and Linearity: An Algebra for Discrete Event Systems*. Wiley, 1992.

[8] A. K. Parekh, R. G. Gallagher. *A generalized processor sharing approach to flow control in integrated services networks: the single-node case*, IEEE/ACM Transactions on Networking, vol. 1, pp. 344–357, 1993.

[9] R. L. Cruz. *Quality of Service Guarantees in Virtual Circuit Switched Networks*, IEEE Journal of Selected Areas in Communication, 13(6): 1048–1056, 1995.

[10] H. Sariowan. *A Service-curve Approach to Performance Guarantees in Integrated-service Networks.* Ph.D. Dissertation, Department of Electrical and Computer Engineering, University of California, San Diego, 1996.

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

[12] J.-Y. Le Boudec, P. Thiran. *Network Calculus: A Theory of Deterministic Queuing Systems for the Internet*, Lecture Notes in Computer Science-2050, Springer Verlag, Jan. 2002.

[13] S. Ayyorgun, W.-C. Feng. *A Probabilistic Definition of Burstiness Characterization: A Systematic Approach*, Los Alamos National Laboratory, Technical Report LA-UR-03-3668, May 2003.

[14] E. Reich. *On the Integrodifferential Equation of Takasc, I*, Ann. Math. Stat., vol. 29, pp. 563–570, 1958.

[15] S. Ayyorgun, W.-C. Feng. *Notes on Burstiness and Bursty Traffic Characterization*, Los Alamos National Laboratory, Technical Report.

[16] S. Ayyorgun, R. L. Cruz. *A Service Curve Model With Loss*, Los Alamos National Laboratory, Technical Report LA-UR-03-3939, June 2003.

[17] A. Albanese, J. Blomer, J. Edmonds, M. Luby, and M. Sudan. *Priority Encoding Transmission*, IEEE Transactions on Information Theory, vol. 42, no. 6, Nov. 1996.

[18] J. W. Byers, M. Luby, M. Mitzenmacher, and A. Rege. *A Digital Fountain Approach to Reliable Distribution of Bulk Data*, SIGCOMM, pp. 56–67, 1998.

[19] R. Puri, K.-W. Lee, K. Ramchandran, and V. Bharghavan. *An integrated source transcoding and congestion control paradigm for video streaming in the internet*, IEEE Transaction on Multimedia, 3(1):18–32, March 2001.

[20] S. Ayyorgun, W.-C. Feng. *A Deterministic Definition of Burstiness For Network Traffic Characterization*, Los Alamos National Laboratory, Technical Report LA-UR-03-4477.

[21] S. Ayyorgun. *Feasibility of Serving Packet Streams With Delay and Loss Requirements.* Ph.D. Dissertation, Department of Electrical and Computer Engineering, University of California, San Diego, 2001.

[22] S. Ayyorgun, R. L. Cruz, *A Bigraph Matching Theorem.* Proceedings of the 37th Annual Allerton Conference on Communication, Control, and Computing, pp. 124–126, Sept. 22–24 1999.

[23] L. Tong, P. Ramanathan, and A. Sayeed. *Service Curve Assurances versus Uplink Throughput in CDMA Networks*, ACM Wireless Mobile Multimedia Workshop (WOWMOM), pp. 74–81, July 2001.

[24] P. Ramanathan, L. Tong. *Minimizing Deviation from Service Curve in Forward Link of DS-CDMA Network*, IEEE 55th Vehicular Technology Conference (VTC), vol. 3, pp. 1155–1159, Spring 2002.

[25] M. Agarwal, A. Puri. *Base Station Scheduling of Requests With Fixed Deadlines*, IEEE INFOCOM, 2002.