Delivering QoS Requirements to Traffic with Diverse Delay Tolerances in a TDMA Environment

Jeffrey M. Capone, *Member, IEEE*, and Ioannis Stavrakakis, *Senior Member, IEEE*

***Abstract*— The focus of this paper is on determining the call admission region and scheduling policies for a time-division multiple-access (wireless) system supporting *heterogeneous* real-time variable bit rate applications with distinct quality of service (QoS) requirements and traffic characteristics. The QoS is defined in terms of a maximum tolerable packet delay and dropping probability. A packet is dropped if it experiences excess delay. The call admission region is established for policies that are *work-conserving* (WC) and that satisfy the *earliest due date* (EDD) service criterion (WC-EDD policies). Such policies are known to optimize the overall system performance. In addition to the determination of the call admission region, this study leads also to the construction of scheduling policies that deliver any performance in the region established for WC-EDD policies. Finally, an upper bound on the call admission region that can be achieved under *any* policy (not limited to the WC-EDD policies) is determined.**

***Index Terms*— Integrated services, QoS, scheduling, TDMA, wireless.**

I. Introduction

IN A WIRELESS network, many users communicate over a shared channel. In this paper, time-division multiple access (TDMA) is employed to coordinate the sharing of the uplink channel. In TDMA, time is divided into periodic frames and each frame contains a number of time slots. Each time slot is the time required for the transmission of a packet (plus some guard-time). A base station (or central access point) coordinates the usage of the time slots which allows for service diversification. The transmissions on the uplink (terminal-to-base) are distributed among the geographically dispersed users and the coordination typically takes place in two phases: call establishment and channel access.

At call establishment, users typically request access through a control channel with request packets. During call establishment, an amount of bandwidth (measured in time slots per frame) is requested for servicing the call. If the call is requesting constant bit rate (CBR) service (or circuit switch), then the user is allocated (scheduled) a fixed number of slots per frame (if available) for the duration of the call. If the call is requesting variable bit rate (VBR) service, where the bandwidth needed to service the call may vary over time, then it must be determined if, with the addition of the new call, there exist a scheduling policy that can deliver the QoS to all the supported applications. Once it is determined that the call can be admitted, channel access is allowed based on an appropriate transmission scheduling policy designed to deliver the target QoS to each application in the network.

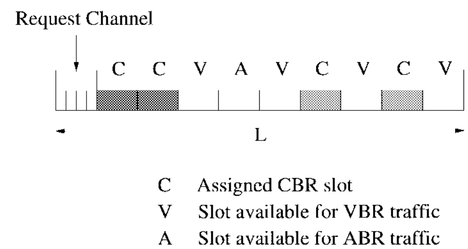


Fig. 1. A typical uplink TDMA frame structure supporting CBR, VBR, and ABR traffic classes.

In this paper, the call admission region and transmission scheduling policies are determined for a TDMA system, where service requests are processed at frame boundaries, see Fig. 1.

This resource structure has been widely considered in both cellular systems [1], [2] and wireless LAN's [3], as well as in recent work toward the development of wireless asynchronous transfer mode (ATM) networks [4], [5]. A call admission rule and transmission scheduling policies for this environment were developed in [6]–[8] to accommodate heterogeneous VBR applications with diverse packet dropping tolerances. In those works, packets which are not serviced in the frame following their arrival are considered to have excess delay and are dropped. That is, all packets assume a common maximum tolerable delay of one frame. In this paper, the packets are considered to have diverse maximum delay as well as dropping probability tolerances. As becomes apparent, the added diversity regarding the maximum delay tolerance creates a number of issues. Some of these issues are addressed by considering a specific family of policies, as discussed in the main body of this paper.

In the next section, the system model considered in this work is described. In Section III, the region of achievable QoS vectors is established for policies that are *work-conserving* (WC) and satisfy the *earliest due date* (EDD) service criterion

Manuscript received April 11, 1997; revised June 1, 1998; approved by IEEE/ACM Transactions on Networking Editor D. Raychaudhuri. This work was supported in part by the Advanced Research Project Agency under Grant F49620-93-1-0564 monitored by the Air Force Office of Scientific Research and by the National Science Foundation under Grant NCR 9628116.

J. M. Capone is with the Department of Electrical Engineering, Arizona State University, Tempe, AZ 85287-7206 USA (e-mail: jcapone@asu.edu).

I. Stavrakakis is with the Department of Electronic and Computer Engineering, Northeastern University, Boston, MA 02115 USA (e-mail: ioannis@cdsp.neu.edu).

Publisher Item Identifier S 1063-6692(99)02467-X.

(WC-EDD policies). Determining this region is central to the call admission control problem as well as in the design of effective transmission scheduling algorithms. In Section IV, the probability distribution of the residual traffic process is derived. Section V examines a class of policies that can deliver any achievable WC-EDD performance. An upper bound on the region of QoS vectors that can be achieved under *any* policy (not limited to the WC-EDD policies) is determined in Section VI. The region of acceptable QoS vectors (call admission region) is examined in Section VII. Numerical examples are presented in Section VIII and the conclusions of this work are contained in Section IX.

II. System Model

In this paper, the problem of sharing T resources by N heterogeneous VBR sources is considered. The source packet arrival process is described in terms of a general arrival process embedded at the boundaries of fixed length intervals called service cycles (or frames). No additional assumptions for the packet arrival process are necessary at this point. Up to T packets from the VBR traffic may be transmitted (served) during each service cycle. Depending on the QoS requirements, packets which cannot be transmitted over the service cycle following their arrival may be dropped (due to delay violation) or may be delayed to compete for service in the next frame. A TDMA system in which arrivals are considered at frame boundaries may be modeled in terms of a discrete time system in which packet delays are measured in frames (1 frame $= L$ time units, see Fig. 1).

In this work, a two-class heterogeneous environment is considered, where real-time sources have diverse delay tolerances. This development is a considerable extension to the case where all sources have a homogeneous 1 frame delay tolerance [6], [8]. By assuming that some of the sources have a greater frame delay tolerance (of 2 frames instead of 1, leading to a (1,2)-delay system), the region of achievable QoS will be enlarged, compared to that under the homogeneous case. Thus, a larger collection of sources can be accommodated under such an (heterogeneous) environment. It should be noted that the (1,2)-delay system is the most resource demanding of all heterogeneous (1,x)-delay systems. Similarly, the (1,∞)-delay system is the least resource-demanding such system, which corresponds to a system where one class supports real-time traffic while the other class supports delay-tolerant traffic that receives "best effort" service. Such two-class systems have been studied extensively in the past (for instance in the context of integrating voice and data applications) and are not considered in this paper.

In addition to the interesting issues addressed in the process of developing and studying the (1,2)-delay system—such as those related to the residual process, the EDD policies, etc.—the results under the (1,2)-delay system can provide valuable information regarding the behavior of a (1,x)-delay system for $x \geq 3$ without getting into the more complex study of such systems. For instance, if the desirable QoS vector for a (1,x)-delay system for $x \geq 3$ is included in the achievable region for the (1,2)-delay system, then such QoS is possible to deliver. Thus, the bound under the (1,2)-delay system can be employed as a lower bound on the region of achievable QoS for a (1, x)-delay system, for $x \geq 3$. Practically, it is expected that if x is greater than 3, then the class with the larger delay tolerance may be considered nonreal time VBR and not compete for resources with the class with delay tolerance of 1 frame (real-time VBR).

In the (1,2)-delay system, the N VBR sources are partitioned into two classes, $S_1 = \{1, 2, \cdots, K\}$ and $S_2 = \{K+1, K+2, \cdots, N\}$. Packets generated from sources in S_1 have a common maximum delay tolerance of L time units (1 frame) and packets generated from sources in S_2 have a common maximum delay tolerance of $2L$ time units (2 frames). This environment could be used to model real time VBR (rt-VBR) traffic where the delay is critical, such as in interactive video conferencing. If more than a two frame delay can be tolerated, the traffic could be characterized as nonreal time VBR (nrt-VBR) and therefore, treated as a separate traffic class.

Let $\lambda_i(n), i \in \{S_1 \cup S_2\}$ denote the number of newly generated packets requesting service from sources in S_1 and S_2 at the nth frame boundary. The aggregate traffic from sources in S_1 is given by $\lambda_{S_1}(n) = \Sigma_{i \in S_1} \lambda_i(n)$ and has a delay tolerance of L time units, thus, must be either serviced or dropped over frame n. Newly generated requests from sources in $S_2, \lambda_{S_2}(n) = \Sigma_{i \in S_2} \lambda_i(n)$ may be either serviced or delayed to the next frame to compete for service. Packets from S_2 that have been delayed (and must be serviced or dropped in the current frame) are denoted as $\lambda_i^{r,f}(n)$; the superscript r is used to indicate residual traffic (those packets from $\lambda_i(n-1), i \in S_2$ not serviced) and the superscript f indicates its dependency on the service policy f. Thus, the total residual traffic from sources in class S_2 requiring service in frame n is given by $\lambda_{S_2}^{r,f}(n) = \Sigma_{i \in S_2} \lambda_i^{r,f}(n)$.

The number of packets from source i that are dropped under policy f during frame n are given by

$$d_i^f(n) = \lambda_i(n) - a_i^f(n), \quad i \in S_1 = \{1, 2, \cdots, K\} \tag{1}$$

$$d_i^f(n) = \lambda_i^{r,f}(n) - a_{i,1}^f(n), \quad i \in S_2 = \{K+1, \cdots, N\}. \tag{2}$$

$a_i^f(n)$ denotes the number of packets from source i serviced during frame n under policy $f; a_{i,1}^f(n), i \in S_2$ denotes the amount of service (in slots) provided to the residual traffic associated with source i during frame n. Since in any frame two types of packet from a source in S_2 may be present (new arrivals and residual traffic), the total amount of resources devoted to source $i \in S_2$ in frame n, is given by

$$a_i^f(n) = a_{i,1}^f(n) + a_{i,2}^f(n), \qquad i \in S_2 \tag{3}$$

where $a_{i,2}^f(n)$ is the amount of available resources allocated to the new arrivals from S_2 in frame n under some policy f. Since the residual traffic in frame n consist of the new arrivals from sources in S_2 which did not receive service in frame $(n-1)$, the residual traffic in frame n under policy f

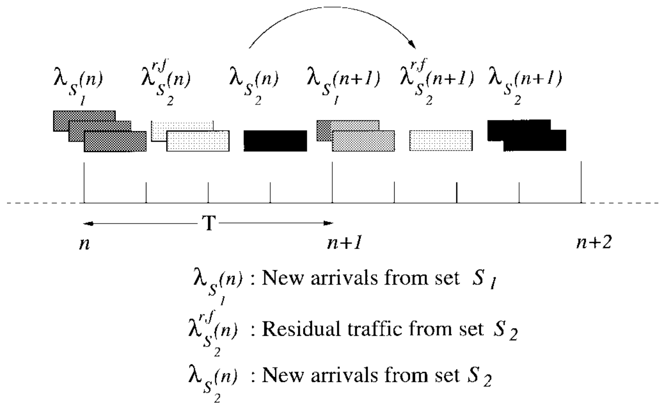


Fig. 2. Realization of residual traffic.

is given by

$$\lambda_i^{r,f}(n) = \lambda_i(n-1) - a_{i,2}^f(n-1), \qquad i \in S_2. \tag{4}$$

Let $d_i^f = E[d_i^f(n)], a_i^f = E[a_i^f(n)]$ and $\lambda_i = E[\lambda_i(n)]$ be the (assumed time invariant) expected values of the associated quantities. The residual traffic is illustrated in the realization depicted in Fig. 2.

If a WC-EDD policy is employed, then four out of the five packets with a service deadline (or due date) in the current frame will be served, the remaining one will be dropped, and the new arrival from class S_2 will form the residual traffic in the next frame [frame $(n+1)$]. According to the WC-EDD policy, all three packets with service deadline in frame $(n+1)$ will be served during the frame, as well as one of the two packets with service deadline in frame $(n+2)$.

From the above example, it is evident that the employed EDD policy imposes restrictions on the level of QoS diversification that could be achieved otherwise. For instance, new arrivals from class S_2 cannot be serviced in the presence of packets from class S_1, imposing a limit on the minimum dropping rate for sources in class S_2. This limit is higher than the dropping rate achieved if, for instance, all packets (new and residual) from sources in class S_2 had service priority over those in S_1.

In most of this paper, the class of WC-EDD policies is considered for the following reasons. First, the WC-EDD policies are known to minimize the *system* packet dropping (delay violation) probability [9], resulting in throughput maximization. Unlike a more general case, in which the service deadlines or due dates would form a continuum or may be drawn from a large collection of values, only two service deadlines are considered in the TDMA environment in this work. As a consequence, a potentially large number of packets from different sources will have identical service deadlines or due dates (one of two values) and, thus, significant room for dropping rate diversification may be possible without departing from the WC-EDD policies. In addition, it is possible to determine the region of achievable QoS vectors under *any* WC-EDD policy, as well as scheduling policies delivering any QoS vector in this region.

If the QoS vector is not in the region of achievable QoS vectors under the WC-EDD policies, it can be concluded that such level of QoS diversification *may* be achieved only at the expense of system throughput [10]. This may suggest that the sharing of the resources by such diverse applications may need to be restricted by allowing for resource sharing by less diverse applications. In any case, by deriving an upper bound on the region of achievable QoS vectors under *any* WC policy, it can be determined whether a given QoS vector is achievable.

In this work, the QoS requirements of application i is described in terms of a maximum tolerable delay and a maximum dropping probability p_i; this is the probability that a packet from source i experiences a delay greater than its maximum tolerable delay and, thus, is dropped. The corresponding packet dropping rate or delay violation rate, d_i (measured in expected number of dropped packets per frame) is easily determined by $d_i = p_i\lambda_i, 0 \le i \le N$. In the rest of this paper the QoS vector associated with the supported applications will be described in terms of the dropping rates, with the understanding that these rates are induced due to violation of diverse delay tolerances. The QoS vector associated with the supported applications can be defined in terms of the (performance) packet dropping rate vector $\boldsymbol{d}$

$$\boldsymbol{d} = (d_1, d_2, \cdots, K, K+1, \cdots, d_N). \tag{5}$$

The first question addressed in the sequel (Section III) is whether a given QoS vector $\boldsymbol{d}$ is achievable under any WC-EDD policy f. Necessary and sufficient conditions are derived in order for the QoS vector to be achievable under these policies, leading to the precise determination of the region of achievable QoS vectors $\boldsymbol{d}$.

III. Region of Achievable QoS Vectors Under WC-EDD Policies

The main question investigated in this section is whether or not a scheduling policy exists that can deliver a given QoS vector $\boldsymbol{d}$. To answer this question, the region of achievable QoS vectors is established. It is based on a set of inequalities and an equality constraint derived by employing work-conserving (nonidling) arguments.

The region of achievable QoS is the set of points (performance vectors) that can be delivered under some policy. The determination of the region of achievable QoS leads to the development of a call admission rule. For example, if with the addition of the new source, the new multidimensional target QoS vector is in the region of achievable QoS vectors, then the call can be admitted since there exists some policy that can deliver the target service to each application. If the call cannot be admitted and more resources can be made available, a precisely defined region of achievable QoS can also be used to determine the minimum additional resources required in order for the new call to be admitted.

A. Conservation Law and Inequality Constraints

A formal definition of a WC scheduling policy for the system described in the previous section is given first.

Definition 3.1: A scheduling policy f is WC if it satisfies the following conditions:

$$\sum_{i=1}^{K} d_i^f(n) + \sum_{i=K+1}^{N} \lambda_i^{r,f}(n+1) = 0, \quad \text{if } \lambda_{S_1}(n) + \lambda_{S_2}^{r,f}(n) + \lambda_{S_2}(n) \le T \tag{6}$$

$$\sum_{i=1}^{N} a_i^f(n) = T, \quad \text{if } \lambda_{S_1}(n) + \lambda_{S_2}^{r,f}(n) + \lambda_{S_2}(n) > T. \tag{7}$$

□

A WC policy does not waste resources (slots) as long as there is work to perform (packets to transmit). Let $S = \{S_1 \cup S_2\}$ be the set of all sources and d_S^f denote the average system packet dropping rate under scheduling policy f, given by

$$d_S^f \triangleq E\left[\sum_{i=1}^{N} d_i^f(n)\right] = \sum_{i=1}^{N} E[d_i^f(n)] = \sum_{i=1}^{N} d_i^f. \tag{8}$$

□

Definition 3.2: Let $\mathcal{F}$ be the family of *WC-EDD* policies.

Definition 3.3: Frame n is said to be *under-loaded* when $\lambda_{S_1}(n) + \lambda_{S_2}^{r,f}(n) \le T$ and *overloaded* when $\lambda_{S_1}(n) + \lambda_{S_2}^{r,f}(n) > T$. □

Notice that during an overloaded frame, packets will be dropped and none of the $\lambda_{S_2}(n)$ packets will receive service under any policy $f \in \mathcal{F}$. Moreover, for any $f \in \mathcal{F}$ the following hold:

$$d_i^f(n) = \begin{cases} = 0, & \text{if } \lambda_{S_1}(n) + \lambda_{S_2}^{r,f}(n) \le T \\ \ge 0, & \text{if } \lambda_{S_1}(n) + \lambda_{S_2}^{r,f}(n) > T \end{cases} \quad \forall i \in S \tag{9}$$

and

$$\lambda_i^{r,f}(n+1) = \begin{cases} \lambda_i(n) + \lambda_i^{r,f}(n) - a_i^f(n), & \text{if } \lambda_{S_1}(n) + \lambda_{S_2}^{r,f}(n) \le T \\ \lambda_i(n), & \text{if } \lambda_{S_1}(n) + \lambda_{S_2}^{r,f}(n) > T \end{cases} \quad \forall i \in S_2. \tag{10}$$

Theorem 3.1: The system dropping rate, d_S^f, is conserved under any $f \in \mathcal{F}$ and is a lower bound on the system dropping rate induced under any policy. More specifically

$$\begin{aligned} d_S^f &\triangleq \{E[\lambda_{S_1}(n) + \lambda_{S_2}^{r,f}(n) | \lambda_{S_1}(n) + \lambda_{S_2}^{r,f}(n) > T] - T\} \\ &\quad \cdot P(\lambda_{S_1}(n) + \lambda_{S_2}^{r,f}(n) > T) \\ &\triangleq b_S \quad \forall f \in \mathcal{F}. \end{aligned} \tag{11}$$

□

Proof: First, the first equality in (11) is proved. Summing (1) and (2) over all $i \in S$, the following is obtained:

$$\sum_{i \in S} d_i^f(n) = \sum_{i \in S_1} \lambda_i(n) + \sum_{i \in S_2} \lambda_i^{r,f}(n) - \sum_{i \in S_1} a_i^f(n) - \sum_{i \in S_2} a_{i,1}^f(n). \tag{12}$$

Since f is a WC-EDD policy it satisfies (6) and (7) of Definition 3.1 and (9) and (10) of Definition 3.2. Thus, the above expression becomes

$$\sum_{i \in S} d_i^f(n) = \begin{cases} 0, & \lambda_{S_1}(n) + \lambda_{S_2}^{r,f}(n) \le T \\ \lambda_{S_1}(n) + \lambda_{S_2}^{r,f}(n) - T, & \lambda_{S_1}(n) + \lambda_{S_2}^{r,f}(n) > T. \end{cases} \tag{13}$$

By applying the expectation operator in (13), the first equality is obtained. To prove the second equality in (11)—and thus, show that d_S^f is conserved—it suffices to show that $\lambda_{S_2}^{r,f}(n)$ is independent of the policy $f \in \mathcal{F}$. For any $f \in \mathcal{F}$

$$\lambda_{S_2}^{r,f}(n+1) = \begin{cases} \max\{0, \lambda_{S_2}(n) + \lambda_{S_1}(n) + \lambda_{S_2}^{r,f}(n) - T\}, & \text{if } \lambda_{S_1}(n) + \lambda_{S_2}^{r,f}(n) \le T \\ \lambda_{S_2}(n), & \text{if } \lambda_{S_1}(n) + \lambda_{S_2}^{r,f}(n) > T. \end{cases} \tag{14}$$

Therefore, $\lambda_{S_2}^{r,f}(n)$ depends on $\lambda_{S_2}^{r,f}(n-1)$, which may in turn be dependent on f and is otherwise independent from f. By induction, $\lambda_{S_2}^{r,f}(n)$ is only dependent on the initial condition $\lambda_{S_2}^{r}(0)$, which is independent of the policy and equal to zero. Thus, the *total* residual traffic in frame n is independent of the policy $f \in \mathcal{F}$ and therefore can be denoted as $\lambda_{S_2}^{r}(n)$. That is

$$\lambda_{S_2}^{r,f}(n) = \sum_{i \in S_2} \lambda_i^{r,f}(n) \triangleq \lambda_{S_2}^{r}(n) \quad \forall n; \forall f \in \mathcal{F} \tag{15}$$

and therefore proving the second equality. Finally, it is a well-known result that d_S^f is the minimum dropping rate, since it is induced by a WC-EDD policy [9]. □

Since the total residual traffic in frame $n, \lambda_{S_2}^{r}(n)$ is independent of the policy $f \in \mathcal{F}$, the following corollary is self-evident in view of the previous theorem.

Corollary 3.1: The number of service opportunities for the new arrivals from S_2 in frame n is independent of the selected $f \in \mathcal{F}$. Therefore, the residual traffic for any source i in frame $(n+1), \lambda_i^{r,f}(n+1)$ is only dependent on the policy f chosen in the *present frame* n and *not in past frames*.

Corollary 3.1 is employed in the proof of Theorem 3.5 and in the development of a class of policies in Section V.

Let $\lambda_g(n) = \Sigma_{i \in g} \lambda_i(n)$ be the aggregate arrivals from sources in subset g in frame n. Let d_g^f denote the aggregate packet dropping rate associated with sources in group g only, under policy f. All N sources in S are assumed to be present and served under the policy f. d_g^f is defined by

$$d_g^f = E\left[\sum_{i \in g} d_i^f(n)\right] = \sum_{i \in g} E[d_i^f(n)] = \sum_{i \in g} d_i^f, \qquad g \subseteq S. \tag{16}$$

The following lemma will be used in the proof of the theorem that follows.

Lemma 3.1: Let $g \subseteq S_2$ and let $\lambda_g^{r,f}(n) = \Sigma_{i \in g} \lambda_i^{r,f}(n)$ be the aggregate residual traffic form subset g under some policy f. Then

$$\lambda_g^{r,f}(n) \geq \lambda_g^r(n) \quad \forall n \forall f \in \mathcal{F} \tag{17}$$

where $\lambda_g^r(n)$ is given by

$$\lambda_g^r(n+1) = \begin{cases} \max\{0, \lambda_g(n) + \lambda_{S_1}(n) + \lambda_{S_2}^r(n) - T\}, \\ \qquad \text{if } \lambda_{S_1}(n) + \lambda_{S_2}^r(n) \leq T \\ \lambda_g(n), \quad \text{if } \lambda_{S_1}(n) + \lambda_{S_2}^{r,f}(n) > T. \end{cases} \tag{18}$$

$\lambda_g^r(n)$ is called the *minimum residual traffic process* for sources in $g \in S_2$, for a system that is served under a WC-EDD policy. □

Proof: The inequality (17) is self-evident since $\lambda_g^r(n)$ corresponds to the residual traffic where new arrivals from sources in g are given all excess resources.[1] □

Theorem 3.2: Let b_g denote the lower bound for the aggregate packet dropping rate for sources in set $g, g \subset S$ under any policy $f \in \mathcal{F}$. Then this bound is given by

$$b_g = \{E[\lambda_g^r(n) | \lambda_g^r(n) > T] - T\} P(\lambda_g^r(n) > T), \qquad g \subseteq S_2 \tag{19}$$

$$b_g = \{E[\lambda_g(n) | \lambda_g(n) > T] - T\} P(\lambda_g(n) > T), \qquad g \subseteq S_1 \tag{20}$$

$$\begin{aligned} b_g = &\{E[\lambda_x(n) + \lambda_y^r(n) | \lambda_x(n) + \lambda_y^r(n) > T] - T\} \\ &\cdot P(\lambda_x(n) + \lambda_y^r(n) > T), \qquad g \subseteq S \\ x = &\{g \cap S_1\} \neq \emptyset \quad \text{and} \quad y = \{g \cap S_2\} \neq \emptyset. \end{aligned} \tag{21}$$

Proof: If $g \subseteq S_2$, the bound b_g is achieved when $\lambda_g^r(n)$ packets do not see any interference from packets in S_1 and $\{S_2 - g\}$, or equivalently, all T resources are available to service the $\lambda_g^r(n)$ packets. Therefore, by employing $\lambda_g^r(n)$ in (11), the proof follows. The probability distribution of the minimum residual traffic $\lambda_g^r(n)$ is calculated from (18), as in Section IV. If $g \subseteq S_1$, the proof can be derived as in [6]. If $g \subset S$, where $x = \{g \cap S_1\}$ and $y = \{g \cap S_2\}$ are nonempty, b_g is achieved when the traffic $\lambda_x(n) + \lambda_y^r(n)$ does not see interference from traffic from sources in $\{S - g\}$. By generalizing (19), the proof follows. □

The following class of policies is needed for the establishment of the region of achievable QoS vectors under the WC-EDD policies, as well as for the development of policies which deliver any achievable QoS vector for this region.

Definition 3.4: A deadline-sensitive ordered head-of-line (DSO-HoL) priority service policy is defined to be the policy which first separates packets into two sets: packets having a service deadline in the current frame and packets having a service deadline in the next frame. The former are serviced according to a priority service policy $\boldsymbol{\pi} = (\pi_1, \pi_2, \cdots, \pi_N); \pi_i \in \{1, 2, \cdots, N\}, \pi_i \neq \pi_j, 1 \leq i, j \leq N$. The index of π_i indicates the order of the priority given to the π_i source to service packets having a service deadline in the current frame. None of the π_j sources $j > i$ may be served as long as packets with current service deadline from sources $\pi_k, k \leq i$, are present. After servicing the packets having a current deadline, the same service policy $\boldsymbol{\pi} = (\pi_1, \pi_2, \cdots, \pi_N)$ is followed for packets from sources that are present and do not have a current deadline. □

Theorem 3.3: The following constraints associated with the induced packet dropping rate vector $\boldsymbol{d}^f = (d_1^f, d_2^f, \cdots, d_N^f)$ are satisfied by any scheduling policy $f \in \mathcal{F}$

$$d_g^f \geq b_g \quad \forall g \subset S; \forall f \in \mathcal{F} \tag{22}$$

where d_g^f and b_g are given in (16) and (19)–(21). □

Proof: Let $g \subset S$ and $x = g \cap S_1$ and $y = g \cap S_2$. Suppose that the packets from sources in g are served under a scheduling policy $f_o \in \mathcal{F}$ according to which they are given DS-HoL priority over the packets from sources in $\{S - g\}$. That is, during overload conditions no packet from sources in $\{S - g\}$ is served unless no packets with a current service deadline from sources in g are present. During underload conditions, no packets from sources in $\{S_2 - y\}$ are serviced while packets from sources in $y = \{g \cap S_2\}$ are present, and the minimum residual traffic is achieved for set y. Thus

$$\sum_{i \in g} d_i^{f_o}(n) = \begin{cases} 0, & \text{if } \lambda_x(n) + \lambda_y^r(n) \leq T \\ \lambda_x(n) + \lambda_y^r(n) - T, & \text{if } \lambda_x(n) + \lambda_y^r(n) > T. \end{cases} \tag{23}$$

From (21) and (23) it is clear that

$$d_g^{f_o} = b_g \tag{24}$$

for all policies $f_o \in \mathcal{F}$ that provide DS-HoL priority to the packets from sources in g. Since no other policy $f \in \mathcal{F}$ can provide better service (that is, lower aggregate packet dropping rate) to sources in g than f_o, it is evident that

$$d_g^f \geq b_g \quad \forall f \in \mathcal{F}; g \subset S. \tag{25}$$

□

B. Region of Achievable QoS Vectors

The main result of this section is the determination of the region $\mathcal{D}$ of the achievable QoS vectors $\boldsymbol{d}$ under WC-EDD policies. The following corollary provides a set of necessary conditions in order for a QoS vector $\boldsymbol{d}$ to be achievable, followed by a corollary regarding an upper bound on the achievable region for QoS vectors $\boldsymbol{d}$ under WC-EDD policies. Their proofs are self-evident in view of Theorems 3.1 and 3.3.

Corollary 3.2: A necessary condition in order for a QoS vector $\boldsymbol{d} = (d_1, d_2, \cdots, d_N)$ to be achieved by a policy $f \in \mathcal{F}$ is that its components satisfy the following constraints

$$d_g \geq b_g \quad \forall g \subset S \tag{26}$$

$$d_S = b_S. \tag{27}$$

□

[1] As state in Corollary 3.1, the number of service opportunities (or excess resources) available to the new arrivals, $\lambda_i(n) i \in S_2$, is independent of the policy $f \in \mathcal{F}$.

Corollary 3.3: Let $\mathcal{D}^u$ denote the collection of all vectors $\boldsymbol{d}$ satisfying (26) and (27). Then $\mathcal{D}^u$ is an upper bound on the region $\mathcal{D}$ of achievable QoS vectors $\boldsymbol{d}$. That is

$$\mathcal{D} \subseteq \mathcal{D}^u. \tag{28}$$

□

The following theorem describes the vectors contained in $\mathcal{D}^u$.

Theorem 3.4: Any vector in the set $\mathcal{D}^u$ can be expressed as a convex combination of extreme points (vertices) of $\mathcal{D}^u$; that is, $\mathcal{D}^u$ may be expressed as the convex hull of its extreme points $\mathcal{D}^u = \mathrm{conv}[\mathrm{ext}(\mathcal{D}^u)]$. □

Proof: The proof follows from the fact that $\mathcal{D}^u$ is a bounded set defined by a finite intersection of closed half-spaces [see (26) and (27)]. Then, by definition, $\mathcal{D}^u$ is a polytope [11] and Theorem 3.4 follows directly from properties of polytopes [11]. □

The following theorem establishes a relationship between scheduling policies and the vertices of $\mathcal{D}^u$.

Theorem 3.5: $\boldsymbol{d}^*$ is a vertex (extreme point) of the set $\mathcal{D}^u$ if $\boldsymbol{d}^*$ is a dropping rate vector resulting from a DSO-HoL priority service policy $\boldsymbol{\pi} = (\pi_1, \pi_2, \cdots, \pi_N)$. □

Proof: Assume that $\boldsymbol{d}^*$ is a vertex of $\mathcal{D}^u$. Then $\boldsymbol{d}^*$ must lie at the intersection of N hyper-planes and its coordinates must satisfy N simultaneous, linearly independent equations (for definition of a polytope vertex, see [11]), given by

$$\sum_{i \in g_j} d_i^* = b_{g_j}, \qquad j = 1, 2, \cdots, N \tag{29}$$

where one of the g_j's is the set $S = \{1, 2, \cdots, N\}$ and the remaining $(N-1)$ are proper, nonempty, and different subsets of S. Lemma 10.3 (see appendix) establishes that the subsets g_i's are strictly included in eachother. Therefore, by adopting the order $g_1 \subset g_2 \subset \cdots \subset g_{N-1} \subset g_N \equiv S$, the g_i's are given by

$$\begin{aligned} g_1 &= \{\pi_1\} \\ g_1 \subset g_2 &= \{\pi_1, \pi_2\} \\ g_2 \subset g_3 &= \{\pi_1, \pi_2, \pi_3\} \\ &\vdots \\ g_{N-1} \subset g_N &= \{\pi_1, \pi_2, \cdots, \pi_N\} = \{1, 2, \cdots, N\} \end{aligned} \tag{30}$$

where $\pi_i \in \{1, 2, \cdots, N\}, \pi_i \neq \pi_j$ for $i \neq j, 1 \leq i, j \leq N$. By using (21), the following can be obtained from (29)

$$\begin{aligned} d^*_{\{\pi_1\}} &= b_{\{\pi_1\}} \\ d^*_{\{\pi_2\}} &= b_{\{\pi_1,\pi_2\}} - b_{\{\pi_1\}} \\ d^*_{\{\pi_3\}} &= b_{\{\pi_1,\pi_2,\pi_3\}} - b_{\{\pi_1,\pi_2\}} \\ &\vdots \\ d^*_{\{\pi_N\}} &= b_{\{\pi_1,\pi_2,\cdots,\pi_N\}} - b_{\{\pi_1,\pi_2,\cdots,\pi_{N-1}\}} \end{aligned} \tag{31}$$

which is precisely the dropping rate vector induced by the DSO-HoL priority policy $\boldsymbol{\pi} = (\pi_1, \pi_2, \cdots, \pi_N)$. Thus, for any vertex (extreme point) there exists a DSO-HoL priority policy that induces it. Since it is easy to see that the dropping rate vector resulting from a DSO-HoL priority policy must satisfy (31), it is evident that these dropping rate vectors will be the vertices of $\mathcal{D}^u$. □

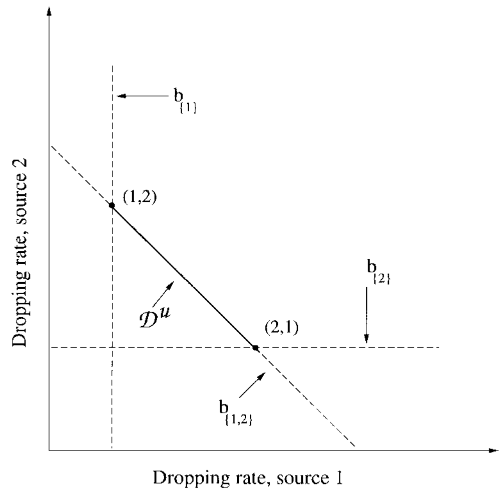


Fig. 3. The region (polytope) $\mathcal{D}^u$ for a system with two sources.

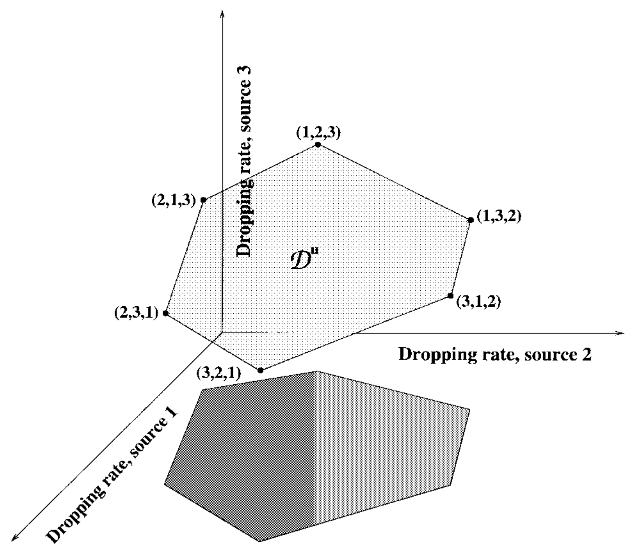


Fig. 4. The region (polytope) $\mathcal{D}^u$ for a system with three sources.

Figs. 3 and 4 provide a graphical illustration of theregion $\mathcal{D}^u$ for the case of $N = 2$ and $N = 3$ sources, respectively. The extreme points $\boldsymbol{d}_{\mathrm{ext}-i}$'s correspond to QoS vectors induced by the $(N!)$ DSO-HoL priority policies $\boldsymbol{\pi}^i = (\pi_1, \pi_2, \cdots, \pi_N), 1 \leq i \leq N!$, as shown in Theorem 3.5. Referring to Fig. 3, it may be observed that the policy $(\pi_1, \pi_2) = (1, 2)$ corresponds to the intersection of the line for the lower bound on the packet dropping rate line for source one, $b_{\{1\}}$, with the system dropping rate line $b_{\{1,2\}}$. Similarly, the second extreme point induced by the policy $(\pi_1, \pi_2) = (2, 1)$ is the intersection of the lower bound on the packet dropping rate for source two, $b_{\{2\}}$ and $b_{\{1,2\}}$. Similar observations can be made for the region $\mathcal{D}^u$ for a system of $N = 3$ sources, shown in Fig. 4.

Let $\mathcal{D}, \mathcal{D} \subseteq \mathcal{D}^u$, be the achievable region of QoS vectors $\boldsymbol{d}$. The following theorem establishes its convexity.

Theorem 3.6: Let $\boldsymbol{d}_1, \boldsymbol{d}_2 \in \mathcal{D}$, then $\boldsymbol{d}_3$, where

$$\boldsymbol{d}_3 = \alpha \boldsymbol{d}_1 + (1-\alpha)\boldsymbol{d}_2, \qquad \alpha \geq 0; \alpha \leq 1 \tag{32}$$

is also in $\mathcal{D}$. That is $\mathcal{D}$ is convex. □

Proof: Let $\boldsymbol{d}_1(\boldsymbol{d}_2)$ be the dropping rate vector induced by the policy $f_1(f_2) \in \mathcal{F}$; that is, $\boldsymbol{d}_1, \boldsymbol{d}_2 \in \mathcal{D}$. Consider a scheduling policy f_3 that at each underloaded frame decides to follow the scheduling rule of policy f_1 with probability α and policy f_2 with probability $(1-\alpha)$; the decisions over consecutive underloaded frames are independent. Notice that no dropping occurs during underloaded frames, and thus, any policy $f \in \mathcal{F}$ could be selected. In fact, any policy $f \in \mathcal{F}$ can be selected during an underloaded frame. As shown in Corollary 3.1, the residual traffic for source i in frame $n, \lambda_i^{r,f_1(f_2)}(n)$ is only dependent on the policy $f_1(f_2)$ selected in frame $(n-1)$ and not in earlier frames, and therefore, decisions over the first overloaded frame following an underloaded one, follow the policy selected for the previous frame—ensuring that the $\lambda_i^{r,f_1(f_2)}(n)$ packets are serviced under policy $f_1(f_2)$.[2] Over overloaded frames other than the first of an overloaded period, policy $f_1(f_2)$ is selected with probability $\alpha(1-\alpha)$ independently from the selection in the previous frame. Note that the selected policy during an overloaded frame does *not* affect the residual traffic in the next frame; refer to (18). It should also be noted that the aggregate traffic $\lambda_{S_1}(n) + \lambda_{S_2}^r(n)$ is independent of the policy $f \in \mathcal{F}$ (Theorem 3.1). This implies that the occurrence of overloaded periods is independent of the policy. In view of the above, it is easy to establish that the dropping rate performance of policy $f_1(f_2)$ is induced with probability $\alpha(1-\alpha)$. Thus, the packet dropping rate induced by policy S_3 is given by

$$\boldsymbol{d}_3 = \alpha \boldsymbol{d}_1 + (1-\alpha)\boldsymbol{d}_2. \tag{33}$$

Since $\boldsymbol{d}_3$ is achieved by a policy in $\mathcal{F}$, $\boldsymbol{d}_3 \in \mathcal{D}$, establishing the convexity of $\mathcal{D}$. □

The next theorem establishes the region $\mathcal{D}$ of achievable QoS vectors.

Theorem 3.7: $\mathcal{D} \equiv \mathcal{D}^u$. □

Proof: Since $\mathcal{D} \subseteq \mathcal{D}^u$ (Corollary 3.3), it suffices to establish that $\mathcal{D}^u \subseteq \mathcal{D}$ to complete the proof. Notice that $\mathcal{D}^u = \mathrm{conv}[\mathrm{ext}(\mathcal{D}^u)]$ (Theorem 3.4) and that $\mathcal{D}$ is convex (Theorem 3.6) and, thus, if $\{\mathrm{ext}(\mathcal{D}^u)\} \subseteq \mathcal{D}$ then $\mathcal{D}^u \subseteq \mathcal{D}$. The latter holds true since the extreme points of $\mathcal{D}^u$ are induced by the DSO-HoL priority policies (Theorem 3.5) and thus, these points are in the region $\mathcal{D}$ of achievable QoS vectors. □

IV. Analysis of Residual Traffic

In this section, the probability distribution of the minimum residualtraffic process $\lambda_g^r(n)$ is derived. The main developments in this paper (presented in the previous sections) are valid for any arrival process and can lead to numerical results, provided that the distribution on the total residual traffic process $\lambda_{S_2}^r(n)$ is characterized. In this section, the residual traffic distribution is determined under arrival processes $\lambda_i(n), i \in S$ that are mutually independent and have independent increments. The residual traffic process can also be determined under other arrival processes, such as Markov-modulated processes, in which case it would result in a 2-dependent chain. Such an arrival process would only increase the numerical complexity of the analysis for the residual traffic, but would not impact either the feasibility of the approach (for a reasonable range of parameters) or the fundamental developments of this work. Therefore, the residual traffic is derived in this section under an arrival process with independent increments.

As seen in (14), future evolution of the total residual traffic process $\lambda_{S_2}^r(n+1)$ depends only on the present values of $\lambda_{S_2}^r(n), \lambda_{S_1}(n)$, and $\lambda_{S_2}(n)$. Therefore, under the independent increment assumption for the arrival processes, $\lambda_{S_2}^r(n)$ is Markovian with transition probabilities

$$P(\lambda_{S_2}^r(n+1) = i | \lambda_{S_2}^r(n) = j) = \begin{cases} P(\lambda_{S_2}(n) + \min(j + \lambda_{S_1}(n), T) - T = i), & i > 0 \\ P(\lambda_{S_2}(n) + \min(j + \lambda_{S_1}(n), T) \le T), & i = 0. \end{cases} \tag{34}$$

Assuming the stationary distribution of $\lambda_{S_2}^r(n)$ exists, the distribution vector $\boldsymbol{\lambda}_{S_2}^r$ is easily computed as

$$\boldsymbol{\lambda}_{S_2}^r P = \boldsymbol{\lambda}_{S_2}^r \tag{35}$$

where P is the probability transition matrix with elements $p_{j,i}$ given in (34).

The distribution of the minimum residual traffic process for subset g in (18) is found by conditioning on $\lambda_{S_2}^r(n)$. As shown in Theorem 3.1, no matter what the subset g under consideration, the quantity $\lambda_{S_2}^r(n)$ is conserved. The stationary distribution for $\lambda_g^r(n)$ is found as

$$(\lambda_g^r(n) = i) = \sum_j P(\lambda_g^r(n) = i | \lambda_{S_2}^r(n-1) = j) \cdot P(\lambda_{S_2}^r(n-1) = j). \tag{36}$$

From the independent increment assumption

$$P(\lambda_g^r(n+1) = i | \lambda_{S_2}^r(n) = j) = \begin{cases} P(\lambda_g(n) + \min(j + \lambda_{S_1}(n), T) - T = i), & i > 0 \\ P(\lambda_g(n) + \min(j + \lambda_{S_1}(n), T) \le T), & i = 0. \end{cases} \tag{37}$$

V. A Class of Policies

Let $C_{\text{DSO-HoL}}$ denote the class of DSO-HoL service policies $\boldsymbol{\pi}^i$ introduced in Section III-A.

Definition 5.1: In each *underloaded* frame, a *Mixing DSO-HoL Policy* f_m decides to follow the DSO-HoL policy $\boldsymbol{\pi}^i$ with probability $\alpha_i, \alpha_i \ge 0, 1 \le i \le N!, \Sigma_{i=1}^{N!} \alpha_i = 1$; decisions over consecutive under-loaded or overloaded frames are independent. The DSO-HoL policy from the previous under-loaded frame is chosen in the first overloaded frame. Clearly, f_m is completely determined by the $N!$ dimensional vector $\boldsymbol{\alpha}, \boldsymbol{\alpha} \ge \boldsymbol{0}, \boldsymbol{1} \cdot \boldsymbol{\alpha} = 1$. Let $M_{\text{DSO-HoL}}$ denote the class of such policies. □

The proof of the following theorem follows directly from the proof of Theorem 3.6.

[2] Otherwise, packets dropped in the overloaded frame under policy f_1 would be associated with the residual traffic generated in the previous (underloaded) frame under policy f_2; the resulting performance would be that of neither f_1 nor f_2.

Theorem 5.1: The dropping rate vector induced by a Mixing DSO-HoL policy is given by

$$\boldsymbol{d}^{f_m} = \sum_{i=1}^{N!} \boldsymbol{d}^{f_i} \alpha_i. \tag{38}$$

The following theorem establishes the main results of this section.

Theorem 5.2: For each packet dropping rate vector $\boldsymbol{d} \in \mathcal{D}$ there exist a policy $f_m \in M_{\text{DSO-HoL}}$ that induces $\boldsymbol{d}$. □

Proof: Let $\boldsymbol{d} \in \mathcal{D}$, then $\boldsymbol{d} = \Sigma_{i=1}^{N!} \alpha_i \boldsymbol{d}_{\text{ext-}i}$ for some $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \cdots, \alpha_{N!})$ where $\alpha_i \geq 0, 1 \leq i \leq N!, \Sigma_{i=1}^{N!} \alpha_i = 1$, since any point in $\mathcal{D}$ can be written as a convex combination of the extreme points (vertices) $\boldsymbol{d}_{\text{ext-}i}$ of $\mathcal{D}(= \mathcal{D}^u)$; each $\boldsymbol{d}_{\text{ext-}i}$ is induced by some policy in $C_{\text{DSO-HoL}}$ (Theorem 3.5).

Let f_m be the mixing policy which selects the HoL priority $\boldsymbol{\pi}^i$ (that induces $\boldsymbol{d}_{\text{ext-}i}$) with probability α_i. The packet dropping rate vector $\boldsymbol{d}^{f_m}$ induced by f_m is given by

$$\boldsymbol{d}^{f_m} = \sum_{i=1}^{N!} \alpha_i \boldsymbol{d}_{\text{ext-}i} \tag{39}$$

and thus f_m induces $\boldsymbol{d}$. □

The following corollary is obvious in view of the above theorem.

Corollary 5.1: Let $\boldsymbol{d} \in \mathcal{D}$ be a target packet dropping rate vector. The mixing policy $f_m \equiv \boldsymbol{\alpha}$ induces $\boldsymbol{d}$, where $\boldsymbol{\alpha}$ is such that

$$E_{\boldsymbol{\alpha}}[\boldsymbol{d}_{\text{ext}}] \triangleq \sum_{i=1}^{N!} \alpha_i \boldsymbol{d}_{\text{ext-}i} = \boldsymbol{d} \tag{40}$$

$$\boldsymbol{\alpha} \geq \boldsymbol{0} \tag{41}$$

$$\boldsymbol{1} \cdot \boldsymbol{\alpha} = 1 \tag{42}$$

where $E_{\boldsymbol{\alpha}}[\cdot]$ is weighted average of the set of extreme points, $\boldsymbol{d}_{\text{ext}}$, of $\mathcal{D}$ with respect to the probability mass function $\boldsymbol{\alpha}$. □

VI. Upper Bound on the Region of Achievable QoS Vectors Under Any Service Policy

In the previous section the region of achievable QoS vectors induced by WC-EDD policies (denoted in this section by $\mathcal{D}^{\text{EDD}}$) was determined and described in terms of conditions (26) and (27), restated (using superscript EDD) as

$$\begin{aligned} d_g &\geq b_g^{\text{EDD}} \qquad \forall g \subset S \\ d_S &= b_S^{\text{EDD}} \end{aligned} \tag{43}$$

$b_S^{\text{EDD}\cdot}$ and b_g^{EDD} are lower bounds on the performance of WC-EDD policies as given in (11) and (21), respectively. It is well known that the WC-EDD policies optimize the system (S) performance by minimizing the (S) dropping rate. Therefore, any policy that attempts to improve (decrease) the dropping rate for a subset of sources g beyond the lower bound shown in (18)–(21) by relaxing the EDD condition will result in increased (S) dropping rate. That is, the lower bounds on the dropping rates achieved by the WC-EDD policies and the class of policies which do not necessarily satisfy the EDD condition

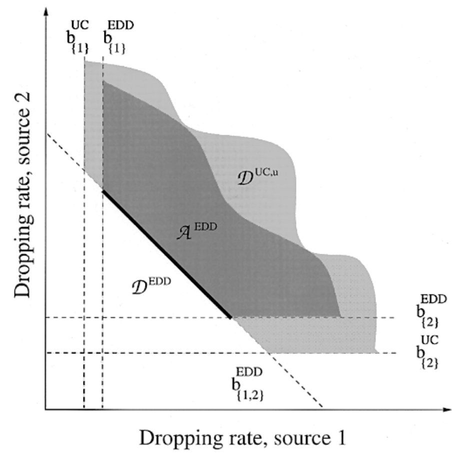


Fig. 5. Necessary performance bounds under any policy and the sufficient bounds for the WC-EDD family of policies.

(denoted here as an unconstrained (UC) policy) satisfy the following conditions:

$$\begin{aligned} b_g^{\text{EDD}} &\geq b_g^{\text{UC}} \qquad \forall g \subset S \\ b_S^{\text{UC}} &\geq b_S^{\text{EDD}}. \end{aligned} \tag{44}$$

b_g^{UC} is the *unconstrained* lower bound on the dropping rate for the sources in set g. This bound is achieved by considering that packets from sources in set g *only* are present and serviced under a WC EDD policy; sources in $\{S - g\}$ are considered to be absent. b_g^{UC} is calculated by applying (11) and (14) to the set g only. That is, replacing S_1 and S_2 with $x = \{g \cap S_1\}$ and $y = \{g \cap S_2\}$ in (11) and (14), respectively.

Since no policy can do better for sources in set g than b_g^{UC} when all sources in S are present, the following necessary conditions must be satisfied by any QoS vector $\boldsymbol{d}$ which is achieved under some policy:

$$\begin{aligned} d_g &\geq b_g^{\text{UC}} \qquad \forall g \subset S \\ d_S &\geq b_S^{\text{EDD}}. \end{aligned} \tag{45}$$

The following proposition is self-evident in view of the above discussion.

Proposition 6.1: An upper bound on the region of QoS vectors achieved under *any* policy $\mathcal{D}^{\text{UC},u}$ is given by (45). □

Fig. 5 depicts the region $\mathcal{D}^{\text{EDD}}$ and $\mathcal{D}^{\text{UC},u}$ for the case of two sources.

VII. Region of Acceptable QoS Vectors

Definition 7.1: The region of acceptable QoS vectors, $\mathcal{A}(\mathcal{D})$, associated with the region of achievable QoS vectors $\mathcal{D}$, is established by relaxing the equality condition on the S performance. It is defined to be the region of vectors $\boldsymbol{d}$, satisfying

$$\begin{aligned} d_g &\geq b_g \qquad \forall g \subset S \\ d_S &\geq b_S. \end{aligned} \tag{46}$$

□

TABLE I
LOWER BOUNDS ON DROPPING RATES UNDER WC-EDD POLICIES

b_1^{EDD}	b_2^{EDD}	$b_{\{1,2\}}^{EDD}$
0.100	0.160	1.856

Proposition 7.1: If $\boldsymbol{d} \in \mathcal{A}(\mathcal{D})$ then there exists a vector $\boldsymbol{d}' \in \mathcal{D}$ which is such that $d_i' \leq d_i \forall i \in S$. □

The proof of the above proposition may be found in [6]. Proposition 7.1 implies that if the required QoS vector $\boldsymbol{d}$ is in $\mathcal{A}(\mathcal{D})$, there exist a policy which can deliver $\boldsymbol{d}$ or *better* (that is, less than the dropping rate required by any source) and thus, the call can be admitted into the network. Fig. 5 depicts the acceptable region $\mathcal{A}(\mathcal{D}^{\mathrm{EDD}})$ and the upper bound on the region of achievable QoS vectors $\mathcal{D}^{\mathrm{UC},u}$.

$\mathcal{A}(\mathcal{D}^{\mathrm{EDD}})$ contains all the QoS vectors which can either be delivered exactly, or a better QoS vector can be delivered under some WC-EDD policy and, therefore, completely describes the call admission region. $\mathcal{D}^{\mathrm{UC},u}$ is an upper bound on the QoS vectors which can be delivered under any policy.

VIII. NUMERICAL EXAMPLE AND VERIFICATION THROUGH SIMULATION

In this section, two examples are presented for a system with two and three sources, respectively, competing for T slots in a TDMA frame. The source packet arrival processes are assumed to be mutually independent. Each process is described in terms of a sequences of independent and identically distributed random variables embedded at the frame boundaries. Let $E^k = \{0, 1, \cdots, M^k - 1\}$ denote the state space of the arrival process associated with source k.

Consider a system in which there are five $(T = 5)$ slots per frame to service two sources, $k = 2$. In this example, $M^1 = 7$ and $M^2 = 7$ and when in state i, a source generates i packets in the current frame. Source one has a maximum arrival rate of 6 packets per frame with an average per frame arrival rate of 3.6. Source two has an average arrival rate of 3.2 packets per frame with a maximum arrival rate of 6 packets per frame. Sources one and two have state probability distributions of (0.0, 0.1, 0.1, 0.3, 0.2, 0.2, 0.1) and (0.1, 0.0, 0.4, 0.0, 0.3, 0.0, 0.2), respectively. Source one has a maximum delay tolerance of 1 frame, while sources two has a maximum delay tolerance of 2 frames.

The region of achievable QoS vectors under WC-EDD policies for this system is calculated from (11), (14), (18), and (21); the bounds on the dropping rates are given in Table I. The unconstrained lower bounds were evaluated by applying (11) and (14) to each source k only, and the results are presented in Table II. Notice that the unconstrained lower bound for source one b_1^{UC} cannot be decreased beyond the WC-EDD lower bound b_1^{EDD} since every packet from source one has a maximum delay tolerance of one frame. The WC-EDD bound for source two can be decreased by relaxing the EDD condition on the system; that is, if the policy allows service to new arrivals from source two before packets from source one, a lower dropping rate will be delivered to source two. The decrease in the dropping rate beyond the WC-EDD bound for source two is achieved at the expense of increasing the system dropping rate. This result is verified through simulations and displayed in Fig. 6.

TABLE II
LOWER BOUNDS ON DROPPING RATES UNDER ANY POLICIES

b_1^{UC}	b_2^{UC}
0.100	0.001

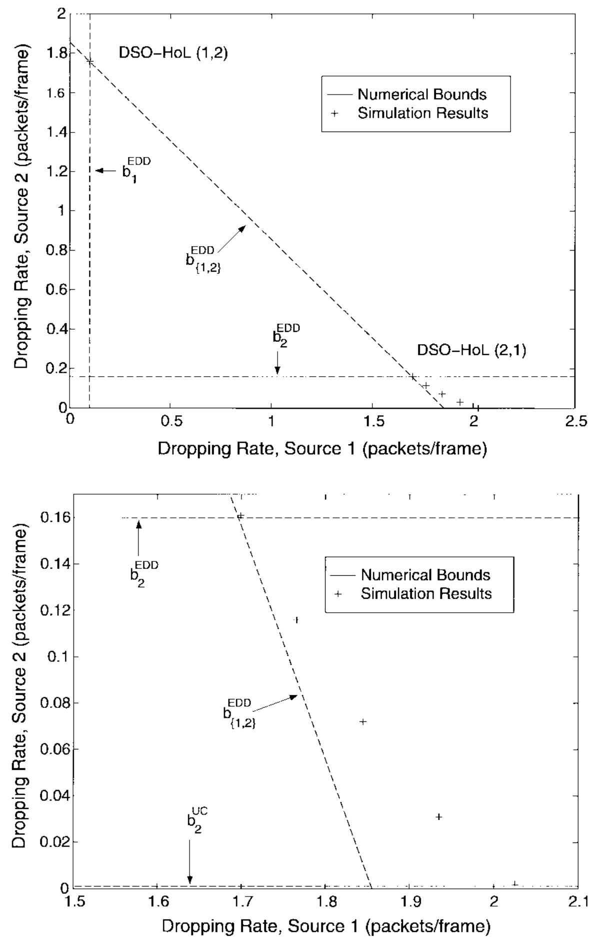


Fig. 6. Evaluation of a numerical example with supporting simulation results.

In the simulation results, policies[3] were generated that ranged from a policy that gives *all* packets from source two service priority over packets from source one, to the DSO-HoL service policy $\boldsymbol{\pi} = (2, 1)$ in which only the packets with a current service deadline from source two were given priority over packet from source one. Under the former policy, the UC lower bound $(b_2 = 0.001)$ is achieved for source two, but the system dropping rate is increased from its minimum of 1.859 to 2.025 packets per frame. The three policies which induce the other points shown in Fig. 6 were obtained by varying the frequency at which source two was allowed to violated the EDD condition. As is clearly observed in Fig. 6, for each of the four policies that violate the EDD condition, the resulting system dropping rate is increased compared to the conserved system dropping rate induced by WC-EDD policies.

[3] The performance of the DSO-HoL policy $\pi = (1, 2)$ was also verified through simulation and is displayed in Fig. 6.

As it was shown in Section III-B, satisfying the condition on the system performance, given by (27), is only necessary and not sufficient to guarantee that the target QoS vector is achievable.[4] To illustrate this concept, consider the system given in the previous example and let $\boldsymbol{d} = (0.099, 1.760)$ be the target QoS vector. $\boldsymbol{d}$ satisfies the condition on the system dropping rate, that is $d_S = b_S^{\mathrm{EDD}} = 1.859$. Although $d_1 < b_1^{UC} = 0.100$ and thus, the target QoS vector cannot be achieved under any policy. For this system, the best possible performance source one can receive is if it is given absolute service priority, resulting in dropping rates of 0.100 and 1.759 for sources one and two, respectively. The overall system performance is satisfied, but source one is experiencing poorer service than what is desired, while sources two is experiencing improved performance.

To illustrate the design of WC-EDD scheduling policies, consider the following example consisting of three VBR sources competing for 7 ($T = 7$) slots of a TDMA frame. In this example $M^1 = 3, M^2 = 5$, and $M^3 = 7$ and when in state i, a source generates i packets in the current frame. Each source is modeled by a sequence of i.i.d. random variables with a maximum arrival rate of 2, 4, and 6 packets per frame, respectively. The probability distributions for each source are (0.200,0,0.405,0.295,0,2.00), (0.600,0,0.400,0,0,0,0) and (0.360,0,0.480,0,0.160,0,0), for source one, two, and three, respectively.

Packets from source one have a maximum delay tolerance of 1 frame, while packets from sources two and three can tolerate delays of up to 2 frames. The maximum dropping probabilities (resulting from delay violations) acceptable for sources one, two, and three are $p_1 = 0.02, p_2 = 0.01$, and $p_3 = 0.02$. The QoS vector in this case is $\boldsymbol{d} = (0.064, 0.008, 0.032)$.

By using (46), it can be determined that $\boldsymbol{d}$ is in the region of acceptable QoS vectors for WC-EDD policies. Furthermore, it can be shown that $\boldsymbol{d} \in \mathcal{D}^{\mathrm{EDD}}$, and thus (as shown in Section V) there exists a mixing DSO-HoL policy $f_m \equiv \boldsymbol{\alpha}$, delivering it. Any $\boldsymbol{\alpha}$ satisfying (40)–(42) of Corollary 5.1 may be chosen. For this example, the following $\boldsymbol{\alpha}_o$ was chosen by employing linear programming techniques

$$\boldsymbol{\alpha}_o = \begin{bmatrix} \alpha_1 = 0 \\ \alpha_2 = 0 \\ \alpha_3 = 0.3367 \\ \alpha_4 = 0.5386 \\ \alpha_5 = 0 \\ \alpha_6 = 0.1248 \end{bmatrix}. \tag{47}$$

In Fig. 7, the time-averaged performance for each source under the selected mixing policy, f_m, is displayed. It can be seen that the target dropping rates, d_i, (and therefore dropping probabilities) are achieved for each source $i = 1, 2, 3$.

As is expected from the formulation of the constraints, more than one solution may be found. This allows for the incorporation of additional constraints representing other desirable qualities of the policies. Functions of interest may be minimized subject to the constraints presented in this paper to guarantee the achievement of the resulting policies. For instance, among all mixing policies inducing $\boldsymbol{d}$, the one which minimizes the variance in the service provided to certain sources may be identified. Such additional objectives will be pursued in the future.

[4] In the special case of a homogeneous system, such as a cellular voice system, satisfying (27) is sufficient to guarantee that the target QoS vector is achievable. This result has been established in [6].

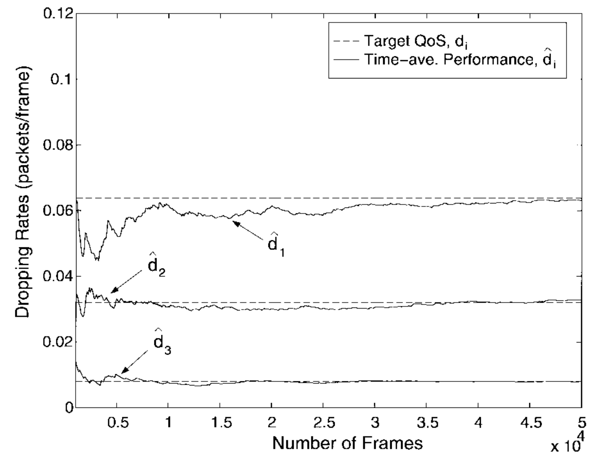


Fig. 7. Time-averaged dropping rate for each source under the policy f_m.

IX. Conclusion

In this work, the call admission region was precisely determined for a system of heterogeneous VBR sources serviced under policies that are WC and that satisfy the EDD service criterion (WC-EDD). The QoS requirements for each application were defined in terms of a maximum tolerable packet delay and dropping probability. In addition to determining the call admission region, a class of scheduling policies was developed which deliver (or better) any performance in the call admission region established for WC-EDD policies. The effectiveness of these policies in delivering the QoS vectors was verified through simulation. Also, an upper bound on the region of QoS vectors that can be achieved under any policy was determined. Numerical examples were presented with simulation results verifying the theoretical results presented in this paper.

Appendix

Lemma 10.1: If λ_A (λ_B) are generic random variables representing the minimum residual traffic[5] from sources in set $A(B) \subseteq S$ in frame n (as defined in Lemma 3.1), then $\lambda_A(\lambda_B)$ satisfy the following :

$$\begin{aligned} \lambda_A + \lambda_B &= \lambda_x(n) + \lambda_y^r(n) + \lambda_v(n) + \lambda_w^r(n) \\ &\le \lambda_{\{x \cup v\}}(n) + \lambda_{\{y \cup w\}}^r(n) + \lambda_{\{x \cap v\}}(n) \\ &\quad + \lambda_{\{y \cap w\}}^r(n) \\ &= \lambda_{\{A \cup B\}} + \lambda_{\{A \cap B\}} \quad \forall A, B \subseteq S \end{aligned} \tag{48}$$

where $x = \{A \cap S_1\}, y = \{A \cap S_2\}, \{v = B \cap S_1\}$, and $\{w = B \cap S_2\}$. □

[5] Residual traffic from sources in S_1 refers to the traffic that has a transmission deadline in the current frame. Thus, the minimum residual traffic for sources in set S_1 is just new arrivals from sources in S_1.

Proof: Since $x, v \subseteq S_1, \lambda_x(n) + \lambda_v(n) = \lambda_{\{x\cup v\}}(n) + \lambda_{\{x\cap v\}}(n)$ and therefore (48) holds if

$$\lambda_y^r(n) + \lambda_w^r(n) \leq \lambda_{\{y\cup w\}}^r(n) + \lambda_{\{y\cap w\}}^r(n) \quad \forall y, w \subseteq S_2. \tag{49}$$

As stated in Corollary 3.1, the number of service opportunities for the new arrivals $\lambda_i(n), i \in S_2$ (or excess resources) is independent of the policy $f \in \mathcal{F}$. Let X represent the number of such opportunities in any given frame. Therefore, by definition in (18), (49) is rewritten as

$$\begin{aligned}\max\{0, \lambda_y(n-1) - X\} &+ \max\{0, \lambda_w(n-1) - X\} \\ &\leq \max\{0, \lambda_{\{y\cup w\}}(n-1) - X\} \\ &\quad + \max\{0, \lambda_{\{y\cap w\}}(n-1) - X\} \quad \forall y, w \subseteq S_2.\end{aligned} \tag{50}$$

As can be seen from (18), equality holds in (50) when $\lambda_{S_1}(n) + \lambda_{S_2}^r(n) > T$, since $X = 0$. When $\lambda_{S_1}(n) + \lambda_{S_2}^r(n) \leq T$ and

Case i: If $\lambda_{y\cap w}(n-1) \leq X$ and $\lambda_y(n-1)$ and/or $\lambda_w(n-1) \leq X$, then (50) holds.[6]

Case ii: If $\lambda_{y\cap w}(n-1) > X \implies \lambda_y(n-1), \lambda_w(n-1) > X$, then equality holds in (50).

Case iii: If $\lambda_{y\cap w}(n-1) \leq X$ and $\lambda_y(n-1), \lambda_w(n-1) > X$, then

$$\begin{aligned}\lambda_y(n-1) &- X + \lambda_w(n-1) - X \\ &= \lambda_y(n-1) + \lambda_w(n-1) - 2X \\ &= \lambda_{\{y\cup w\}}(n-1) - X + (\lambda_{\{y\cap w\}}(n-1) - X) \\ &\leq \lambda_{\{y\cup w\}}(n-1) - X + \max\{0, \lambda_{\{y\cap w\}}(n-1) - X\}\end{aligned} \tag{51}$$

and thus proving Lemma 10.1. □

Lemma 10.2: b_A [defined by (21)]: is a super-modular set function; that is

$$b_A + b_B \leq b_{\{A\cup B\}} + b_{\{A\cap B\}} \tag{52}$$

and equality holds only if and only if $A \subset B$ or $B \subset A$. □

Proof: Let λ_A (λ_B) be generic random variables representing the minimum residual traffic from sources in set $A(B) \subseteq S$ in frame n. Using Lemma 10.1, the following can be established:

$$\begin{aligned}\{E[\lambda_A + \lambda_B | \lambda_A + \lambda_B > T] &- T\}P(\lambda_A + \lambda_B > T) \\ &\leq \{E[\lambda_{\{A\cup B\}} + \lambda_{\{A\cap B\}} | \lambda_{\{A\cup B\}} + \lambda_{\{A\cap B\}} > T] - T\} \\ &\quad \cdot P(\lambda_{\{A\cup B\}} + \lambda_{\{A\cap B\}} > T), \quad \forall A, B \subseteq S.\end{aligned} \tag{53}$$

Case (I): Assume A and B are disjoint ($A \cap B = \emptyset$), then (53) is written as

$$\{E[\lambda_A + \lambda_B | \lambda_A + \lambda_B > T] - T\}P(\lambda_A + \lambda_B > T) \leq b_{\{A\cup B\}}. \tag{54}$$

The following implication is self-evident:

$$\lambda_A + \lambda_B > T \Longrightarrow \begin{cases}\lambda_A > T, \lambda_B \leq T \\ \lambda_A \leq T, \lambda_B > T \\ \lambda_A > T, \lambda_B > T \\ \lambda_A \leq T, \lambda_B \leq T, \lambda_A + \lambda_B > T.\end{cases}$$

Assuming that all sources are independent, then λ_A and λ_B are independent since sets A and B are disjoint, thus,

$$\begin{aligned}P(\lambda_A + \lambda_B > T) &\\ = P(\lambda_A > T)&P(\lambda_B \leq T) + P(\lambda_A \leq T)P(\lambda_B > T) \\ &+ P(\lambda_A \leq T, \lambda_B \leq T, \lambda_A + \lambda_B > T) \\ &+ P(\lambda_A > T)P(\lambda_B > T)\end{aligned} \tag{55}$$

and therefore (54) may be written as

$$\begin{aligned}b_{\{A\cup B\}} \geq &\{E[\lambda_A + \lambda_B | \lambda_A > T, \lambda_B \leq T] - T\} \\ &\cdot P(\lambda_A > T)P(\lambda_B \leq T) \\ &+ \{E[\lambda_A + \lambda_B | \lambda_A \leq T, \lambda_B > T] - T\} \\ &\cdot P(\lambda_A \leq T)P(\lambda_B > T) \\ &+ \{E[\lambda_A + \lambda_B | \lambda_A > T, \lambda_B > T] - T\} \\ &\cdot P(\lambda_A > T)P(\lambda_B > T) \\ &+ \{E[\lambda_A + \lambda_B | \lambda_A \leq T, \lambda_B \leq T, \\ &\quad \lambda_A + \lambda_B > T] - T\} \\ &\cdot P(\lambda_A \leq T, \lambda_B \leq T, \lambda_A + \lambda_B > T).\end{aligned} \tag{56}$$

Expanding the expected value operator over λ_A and λ_B and rearranging terms, (56) can be written as

$$\begin{aligned}b_{\{A\cup B\}} \geq &\{E[\lambda_A | \lambda_A > T] - T\}P(\lambda_A > T) \\ &+ \{E[\lambda_B | \lambda_B > T] - T\}P(\lambda_B > T) \\ &+ E[\lambda_A | \lambda_A \leq T]P(\lambda_A \leq T)P(\lambda_B > T) \\ &+ E[\lambda_B | \lambda_B \leq T] \\ &\cdot P(\lambda_A > T)P(\lambda_B \leq T) \\ &+ TP(\lambda_A > T)P(\lambda_B > T) \\ &+ \{E[\lambda_A + \lambda_B | \lambda_A \leq T, \lambda_B \leq T, \\ &\quad \lambda_A + \lambda_B > T] - T\} \\ &\cdot P(\lambda_A \leq T, \lambda_B \leq T, \lambda_A + \lambda_B > T) \\ = &b_{\{A\}} + b_{\{B\}} + \mathcal{K}\end{aligned} \tag{57}$$

where

$$\begin{aligned}\mathcal{K} = &E[\lambda_A | \lambda_A \leq T]P(\lambda_A \leq T)P(\lambda_B > T) + E[\lambda_B | \lambda_B \leq T] \\ &\cdot P(\lambda_A > T)P(\lambda_B \leq T) + TP(\lambda_A > T)P(\lambda_B > T) \\ &+ \{E[\lambda_A + \lambda_B | \lambda_A \leq T, \lambda_B \leq T, \\ &\quad \lambda_A + \lambda_B > T] - T\} \\ &\cdot P(\lambda_A \leq T, \lambda_B \leq T, \lambda_A + \lambda_B > T).\end{aligned} \tag{58}$$

Notice that $\mathcal{K} > 0$ (for the cases of interest corresponding to $T > 0$). Thus, from (57), and since $b_{\{A\cap B\}} = 0$ for $\{A \cap B\} = \emptyset$

$$b_A + b_B < b_{\{A\cup B\}} + b_{\{A\cap B\}} \tag{59}$$

or

$$b_A + b_B = b_{\{A\cup B\}} + b_{\{A\cap B\}} - C^{A,B} \tag{60}$$

where $C^{A,B} > 0$ is the additional expected number of dropped packets as a result of the competition for resources between sources in set A and B when $\{A \cap B\} = \emptyset$. The following property regarding $C^{A,B}$ is used to complete the proof.

[6] Under Case i, strict inequality holds if $\lambda_y(n-1), \lambda_w(n-1) \leq X$ and $\lambda_{\{y\cup w\}}(n-1) > X$.

Property 10.1:

$$C^{\{A\cup x\},B} - C^{A,B} > 0 \tag{61}$$

when $\{A\cap B\} = \emptyset, x \notin \{A\cup B\}$ and $x \in S$. □

Proof: By definition of $C^{A,B}$ in (60) and by (21), (61) may be expressed as

$$\begin{aligned} E[&\max(\lambda_{\{A\cup B\cup x\}} - T, 0) - \max(\lambda_{\{A\cup x\}} - T, 0) \\ &- \max(\lambda_B - T, 0) - \max(\lambda_{\{A\cup B\}} - T, 0) \\ &+ \max(\lambda_A - T, 0) + \max(\lambda_B - T, 0)] > 0. \end{aligned} \tag{62}$$

Consider the following cases for each realization.

Case i: If $\lambda_A \geq T$ then by Lemma 10.1, (62) holds.

Case ii: If $\lambda_{\{A\cup x\}} \geq T$ and $\lambda_A \leq T$, then using Lemma 10.1

$$\begin{aligned} \lambda_{\{A\cup B\cup x\}} - \lambda_{\{A\cup B\}} - \lambda_{\{A\cup x\}} + T &\geq \lambda_x - \lambda_{\{A\cup x\}} \\ &+ T \geq -\lambda_A + T > 0. \end{aligned} \tag{63}$$

Case iii: If $\lambda_{\{A\cup B\}} \geq T$ and $\lambda_{\{A\cup x\}} < T$, then using Lemma 10.1

$$\lambda_{\{A\cup B\cup x\}} - \lambda_{\{A\cup B\}} \geq 0. \tag{64}$$

Case iv: If $\lambda_{\{A\cup B\cup x\}} > T$ and $\lambda_{\{A\cup B\}} \leq T$, then

$$\lambda_{\{A\cup B\cup x\}} - T > 0. \tag{65}$$

Therefore, the random variable operated on by the expected value operator in (62) is nonnegative. Provided any event that causes this random variable to have a positive value (as the ones indicated) has a nonzero probability (which is true for nondegenerate cases[7]), (62) is satisfied, thus proving Property 10.1. □

Case (II): Assume the A and B are not disjoint $(A\cap B \neq \emptyset)$. By writing $\{A\cup B\}$ as the union of the disjoint sets $\{A\}$ and $\{B-A\}$ and using the results from Case I (60), the following can be written:

$$b_{\{A\cup B\}} = b_{\{A\cup\{B-A\}} = b_A + b_{\{B-A\}} + C^{\{A\},\{B-A\}}. \tag{66}$$

Similarly, by writing set B as the union of the disjoint sets $\{A\cap B\}$ and $\{B-A\}$, the following can be written:

$$b_B = b_{\{A\cap B\}} + b_{\{B-A\}} + C^{\{A\cap B\},\{B-A\}} \tag{67}$$

or

$$b_{\{B-A\}} = b_B - b_{\{A\cap B\}} - C^{\{A\cap B\},\{B-A\}}. \tag{68}$$

Finally, by combining (66) and (68), the following is obtained:

$$b_{\{A\cup B\}} + b_{\{A\cap B\}} = b_A + b_B + C^{\{A\},\{B-A\}} - C^{\{A\cap B\},\{B-A\}}. \tag{69}$$

If $A \subset B$ then

$$\begin{aligned} C^{\{A\},\{B-A\}} - C^{\{A\cap B\},\{B-A\}} &= C^{\{A\},\{B-A\}} - C^{\{A\},\{B-A\}} \\ &= 0. \end{aligned} \tag{70}$$

Similarly if $B \subset A$. If $A \not\subset B$, then $\{A\cap B\} \subset A$ and by Property 10.1

$$C^{\{A\},\{B-A\}} - C^{\{A\cap B\},\{B-A\}} > 0 \tag{71}$$

completing the proof of the lemma. □

[7] A degenerate case is when there is no packet dropping in the system, that is $b_A = b_B = b_{\{A\cup B\}} = 0$.

Lemma 10.3: The sets $g_j, j = 1, 2, \cdots, N$, that satisfy (29), defining the vertices of the polytope $\mathcal{D}^u$, must be strictly included in each other; that is

$$\text{either } h_j = g_j - (g_j \cap g_k) = \emptyset \quad \text{or} \quad h_k = g_k - (g_k \cap g_j) = \emptyset \tag{72}$$

for all g_j and g_k in (29). □

Proof: Let g_1 and g_2 be two sets satisfying (29). By adding the corresponding equations the following is obtained:

$$b_{\{g_1\}} + b_{\{g_2\}} = \sum_{i\in g_1} d_i^* + \sum_{i\in g_2} d_i^* = \sum_{i\in\{g_1\cup g_2\}} d_i^* + \sum_{i\in\{g_1\cap g_2\}} d_i^* \tag{73}$$

or

$$\sum_{i\in\{g_1\cup g_2\}} d_i^* = b_{\{g_1\}} + b_{\{g_2\}} - \sum_{i\in\{g_1\cap g_2\}} d_i^*. \tag{74}$$

Since for any subset of S, (26) must be satisfied, then for $\{g_1\cap g_2\}, \Sigma_{i\in\{g_1\cap g_2\}} d_i^* \geq b_{\{g_1\cap g_2\}}$, the following is obtained from (74):

$$\sum_{i\in\{g_1\cup g_2\}} d_i^* \leq b_{\{g_1\}} + b_{\{g_2\}} - b_{\{g_1\cap g_2\}}. \tag{75}$$

Suppose now that g_1 and g_2 are not strictly included in each other. That is $h_1 = g_1 - (g_1\cap g_2)$ and $h_2 = g_2 - (g_1 \cap g_2)$ are both nonempty. Then, Lemma 10.2 implies that

$$b_{\{g_1\}} + b_{\{g_2\}} < b_{\{g_1\cup g_2\}} + b_{\{g_1\cap g_2\}} \tag{76}$$

or

$$b_{\{g_1\}} + b_{\{g_2\}} - b_{\{g_1\cap g_2\}} < b_{\{g_1\cup g_2\}}. \tag{77}$$

Therefore,

$$\sum_{i\in\{g_1\cup g_2\}} d_i^* \leq b_{\{g_1\}} + b_{\{g_2\}} - b_{\{g_1\cap g_2\}} < b_{\{g_1\cup g_2\}} \tag{78}$$

or

$$\sum_{i\in\{g_1\cup g_2\}} d_i^* < b_{\{g_1\cup g_2\}} \tag{79}$$

which implies $\boldsymbol{d}^* \notin \mathcal{D}^u$, this is a contradiction since $\boldsymbol{d}^*$ is a vertex of $\mathcal{D}^u$. Thus the assumption that g_i are strictly included in each other follows. □

REFERENCES

[1] D. J. Goodman, R. A. Valenzeula *et al.*, "Packet reservation multiple access for local wireless communication," *IEEE Trans. Commun.*, vol. 37, pp. 885–890, Aug. 1989.

[2] N. D. Newman, R. Ganesh, K. Joseph, and D. Raychaudhuri, "Packet CDMA versus dynamic TDMA for multiple access in an integrated voice/data PCN," *IEEE J. Select. Areas Commun.*, vol. 11, pp. 870–884, Aug. 1993.

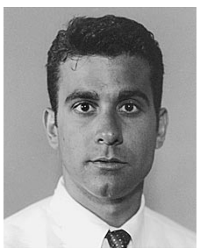
[3] D. Bantz and F. Baucho. "Wireless LAN design alternatives," *IEEE Network Mag.*, vol. 8, pp. 2, pp. 43–45, Mar.-Apr. 1994.

[4] D. Raychaudhuri *et al.*, "WATMnet: A prototype wireless ATM system for multimedia personal communications," *IEEE J. Select. Areas Commun.*, vol. 15, pp. 83–95, Jan. 1997.

[5] S. K. Biswas, D. Reininger, and D. Raychaudhuri, "User parameter control based bandwidth allocation for VBR video in wireless ATM links," *Proc. ICC*, Phoenix, AZ, June 1997.

[6] J. Capone and I. Stavrakakis, "Achievable QoS and scheduling policies in integrated services wireless networks," *Perform. Eval.*, vols. 26, 27, no. 1, Oct. 1996.

[7] ——, "Determining the call admission region for real-time heterogeneous applications in wireless TDMA networks," *IEEE Network Mag.*, vol. 12, pp. 38–47, Mar.–Apr. 1998.
[8] D. Mitra and J. Morrison, "Erlang capacity and uniform approximations for shared unbuffered resources," *IEEE/ACM Trans. Networking*, vol. 2, pp. 558–570, Dec. 1994.
[9] S. S. Panwar, D. Towsley, and J. K. Wolf, "Optimal scheduling policies for a class of queues with customer deadlines to the beginning of service," *J. Ass. Comput. Mach.*, , pp. 832–844, 1988.
[10] G. Chen and I. Stavrakakis, "ATM traffic management with diversified loss and delay requirements," in *Proc. IEEE INFOCOM*, San Francisco, CA, Mar. 1996, pp. 1037–1044.
[11] A. Brøndsted, *An Introduction to Convex Polytopes*. New York: Springer-Verlag, 1983.
[12] E. G. Coffman and I. Mitrani, "A characterization of waiting time performance realizable by single-server queues," *Oper. Res.*, vol. 28, no. 3, pp. 810–82, May–June 1980.
[13] W. W. Sharkey, "Cooperative games with large cores," *Int. J. Game Theory*, vol. 11, pp. 175–182, 1982.
[14] D. J. A. Welsh, *Matroid Theory*. New York: Academic, 1976.



Jeffrey M. Capone (S'92–M'98) received the B.S.E.E. degree from the University of Vermont, Burlington, in 1992, and the M.S.E.E. and Ph.D. degrees from Northeastern University, Boston, MA, in 1995 and 1997, respectively.

Since 1997 he has been with the Electrical Engineering Department, Arizona State University, Tempe. His primary research interest is in the design and analysis of controlling policies for bandwidth management in integrated service wireless communication networks, multi-hop packet radio networks, and large-scale network models.

Dr. Capone is a member of the Technical Committee on Computer Communications and has served on the Technical Committee for GLOBECOM, and the program committee for MobiCom and ICASSP, as well as having organized and chaired sessions for ICC and ICUPC. He is a member of the IEEE Communications Society.

Ioannis Stavrakakis (S'84–M'88–SM'93) received the Diploma in electrical engineering from the Aristotelian University of Thessaloniki, Thessaloniki, Greece, in 1983, and the Ph.D. degree in electrical engineering from the University of Virginia, Charlottesville, in 1988.

In 1988, he joined the faculty of Computer Science and Electrical Engineering, University of Vermont, Burlington, first as Assistant and then as Associate Professor. Since 1994, he has been an Associate Professor of Electrical and Computer Engineering at Northeastern University, Boston, MA. His research interests are in stochastic system modeling, teletraffic analysis, and discrete-time queueing theory, with primary focus on the design and performance evaluation of broadband integrated services digital networks (B-ISDN).

Dr. Stavrakakis is a member of the IEEE Communications Society, and the Technical Committee on Computer Communications. He has organized and chaired sessions, and has been a technical committee member for conferences, such as GLOBECOM, ICC, and INFOCOM.