Achievable QoS and Scheduling Policies for Integrated Services Wireless Networks*

Jeffrey M. Capone   Ioannis Stavrakakis

Electrical & Computer Engineering, Northeastern University
Boston, MA 02115, USA

Abstract

Networks are expected to support applications with diverse traffic characteristics and Quality of Service (QoS) requirements, such as voice, data, video and multi-media. In this environment it is important to determine whether a given QoS vector is achievable (for call admission control), and if it is, design efficient resource allocation policies that deliver it.

In this paper a system of T resources (wireless channels) shared by a number of un-buffered applications is considered. Packets which are not allocated a channel within the current assignment cycle are dropped. The QoS vector is equivalently described in terms of (diverse) packet dropping probabilities (p_i), or (diverse) packet dropping rates (d_i). The region of achievable QoS vectors is precisely determined and easy to implement scheduling policies that deliver the achievable QoS vectors are derived. Some applications to call admission control and resource management are presented along with some numerical examples.

*Research supported in part by the Advanced Research Project Agency under Grants F49620-93-1-0564 monitored by the Air Force Office of Scientific Research (AFOSR) and MDA972-93-1-0023 monitored by the Army Research Office (ARO) and a Grant from GTE Corporation.

1 Introduction

The problem of allocating shared resources is an interesting and continuously revisited one. The primary resources to be shared in a networking environment are memory and transmission bandwidth. As technology advances and resources increase their capabilities, existing services are enhanced and new services are provided, driving the need to develop resource allocation technologies that integrate diverse services.

Networking resources are typically managed at two largely different time scales, determined by the service request inter-arrival and duration time scale and a smaller time scale over which certain amount of resources need to be allocated. Management of resources at the larger time scale is the responsibility of the call admission controller. Management of resources at the smaller time scale is the task of the buffer/bandwidth managers.

Typically, both call admission and buffer/bandwidth management functions need to be employed to ensure (sometimes in a probabilistic sense) the delivery of a target Quality of Service (QoS). The more stringent and diverse the target QoS of the supported applications, the more critical is the exercise of the aforementioned management and the more sophisticated their functions become. Typically, the more stringent the target QoS, the greater the importance and need for admission management. This is the case in the telephone network supporting voice services with stringent QoS requirements. In traditional computer data networks where the QoS is not considered to be stringent, the role of call admission management is minimal (if any). While "uniform" allocation of buffer/bandwidth capacities would result in delivering a target QoS - provided that it is achievable - which is common to identical supported applications, such allocation will be ineffective when diverse QoS is to be delivered to applications with diverse traffic characteristics. Sophisticated buffer/bandwidth management schemes are needed to provide for the desired diversification, such as space priority for buffer management and priority scheduling for bandwidth allocation [1].

The work presented in this paper investigates the above mentioned problem of resource management at both time scales when the supported applications are diverse in terms of both the required QoS and traffic characteristics. In addition, the QoS requirement is assumed not to allow for packet (information unit) buffering beyond the present service cycle and, thus, buffer management is not considered to be an issue. Packets that do not receive service over the cycle following their arrival time are considered to have excess delay and are dropped. Applications with such QoS requirements are considered to be un-buffered and, thus, the problem is concentrated on the management of the available transmission resources at both time scales.

A wireless networking environment supporting real-time applications has characteristics of a shared un-buffered environment in which the management of the transmission resource is most critical due to its scarcity. Since the associated QoS requirements and the mobility of the supported users do not seem to allow for buffer management schemes, the work in this paper is presented in the context of wireless networking to provide for some reasonable motivation. The investigated problem of the achievable QoS and the associated scheduling schemes do maintain their general appeal throughout the paper.

Two major wireless networking technologies have recently emerged: cellular systems and wireless Local Area Networks (LAN). Cellular systems - originally designed to support voice applications - are currently facing the challenge of accommodating data and video services. Similarly, wireless LANs - developed to support computer data applications - are currently being redesigned to be capable of providing service to image, video and possibly voice applications. The convergence of the services supported by these two technologies drives a demand for a single network architecture capable of supporting all aforementioned services in an integrated fashion. In addition, this new architecture should be compatible with the prevalent network architecture for integrated services over fiber/copper based channels, the Asynchronous Transfer Mode (ATM). The development of an integrated services wireless network technology generates new transmission resource allocation problems that are more challenging than in the "specialized services" cellular voice systems or wireless LANs.

Consider for instance the transmission resource management problem in a cellular (voice) system. Since the supported (voice) applications are identical - in terms of traffic characteristics and QoS requirements - the transmission resource management problem at the large time scale is relatively easily addressed. Given a certain amount of available resources, the maximum number of voice applications that can be supported is such that the induced (*common*) system packet dropping *probability*, p_S, be less than a threshold, typically set at $p_S \leq 0.01$ [2]. Equivalently, this number may be determined as the one inducing a system packet dropping *rate*, $d_S \leq (0.01)(\lambda_S)$; $d_S(\lambda_S)$ is defined to be the average number of packets dropped (arrived) per service cycle. This simple rule also addresses the problem of determining the number of required transmission resources in order to support (simultaneously) a target number of applications. The uniformity in the QoS requirements and traffic characteristics of the supported voice applications simplifies substantially the transmission resource problem at the smaller time scale as well. A uniform, non-discriminating scheduling policy would induce the same (system) packet dropping probability to all supported applications.

The above discussion on resource management for cellular voice systems can be extended to that for wireless LANs. The QoS requirements of the supported computer applications suggest that resource management at the smaller time scale is of primary importance. Again, due to the uniformity of the QoS requirements, the resource management at the smaller time scale is not nearly as challenging as that in a integrated services network.

In this paper, the shared transmission resources are defined to be the slots (packet transmission times) of a TDMA frame. This resource structure has been widely considered in both cellular systems [3] and wireless LANs [4], as well as in recent work toward the development of wireless

ATM networks [5, 6]. *Variable bit rate* (VBR) applications with distinct QoS requirements and traffic characteristics are considered. The region of achievable multi-dimensional (non-degenerate) QoS is precisely determined and simple scheduling policies that deliver the achievable QoS are developed.

In the next section, the system model considered in this work is described. A set of equality and inequality constraints are presented in section 3.1 that are employed in the establishment of the region of the achievable QoS vectors in section 3.2. The derivations of section 3 are employed in source call admission control and resource allocation applications and presented in section 4. The special case of a homogeneous system is also presented in section 4. Based on the developments of section 3, a class of simple scheduling policies that achieve *any* achievable QoS vector is developed in section 5. Some simple numerical examples are presented in section 6, followed by conclusion of this work in section 7.

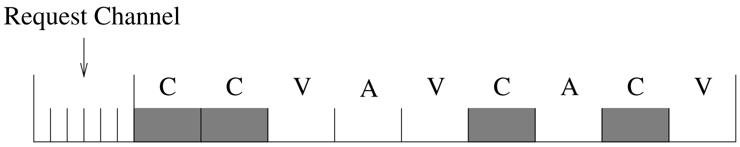
2 Description of the System Model

In this paper, the problem of sharing T resources by N diverse Variable Bit Rate (VBR) source applications 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 may be transmitted (served) during each service cycle. Packets which cannot be transmitted over the service cycle following their arrival are considered to have excess delay and are dropped at the source.

The above environment could model the sharing of the up-link TDMA frame of a wireless network by N applications; T slots of this frame are available to these applications. Fig. 1 illustrates a typical up-link TDMA frame whose slots are allocated to various classes of services, such as Continuous Bit Rate (CBR), Available Bit Rate (ABR) and Variable Bit Rate (VBR). In this figure $T = 3$ slots are assumed to be available to the VBR applications. Let $\lambda_i(n),\ 0 \leq i \leq N$, denote the number of source i packets generated at the n^{th} frame boundary and requiring service over this frame. If $\sum_{i=1}^{N} \lambda_i(n) \leq T$ frame n is said to be *underloaded*; it is said to be *overloaded* if $\sum_{i=1}^{N} \lambda_i(n) > T$. Let $a_i^f(n),\ 0 \leq i \leq N$, denote the number of source i packets serviced during the n^{th} frame under some scheduling policy f. The number of source i packets dropped during frame n under policy f is given by $d_i^f(n)$, where,

$$d_i^f(n) = \lambda_i(n) - a_i^f(n) \begin{cases} = 0 & \text{if} \quad \sum_{i=1}^{N} \lambda_i(n) \leq T \\ \geq 0 & \text{if} \quad \sum_{i=1}^{N} \lambda_i(n) > T \end{cases} ,0 \leq i \leq N. \tag{1}$$

Let $d_i^f = E\left[d_i^f(n)\right]$, $a_i^f = E\left[a_i^f(n)\right]$ and $\lambda_i = E\left[\lambda_i(n)\right]$ be the (assumed time invariant) expected values of the associated quantities.



C Assigned CBR slot

V Slot available for VBR traffic

A Slot available for ABR traffic

Figure 1: TDMA frame supporting CBR, VBR and ABR traffic classes.

Suppose that the QoS requirement of application i is defined in terms of a maximum tolerable per frame packet dropping rate d_i, $0 \leq i \leq N$. Then the QoS vector associated with the supported applications can be defined in terms of the (performance) packet dropping rate vector $\mathbf{d}$,

$$\mathbf{d} = (d_1, d_2, \ldots, d_N). \tag{2}$$

When the QoS requirement of application i is defined in terms of a maximum tolerable packet dropping probability p_i, the corresponding packet dropping rate d_i is easily determined by $d_i = \lambda_i \, p_i$.

The first question addressed in the sequel (section 3) is whether a given QoS vectors $\mathbf{d}$ is achievable under any policy f. Necessary and sufficient conditions are derived in order for the QoS vector to be achievable, leading to the precise determination of the region of achievable QoS vectors $\mathbf{d}$. The second question, addressed in section 4, is concerned with the design of scheduling policies which deliver an achievable QoS vector $\mathbf{d}$.

3 Determination of the Region of Achievable QoS Vectors

In section 3.1, a set of equality and inequality constraints associated with the packet dropping rate vector $\mathbf{d}^f$ induced by scheduling policies f are presented. These constraints lead to the derivation of the region of the achievable QoS vectors, presented in section 3.2.

3.1 Conservation Law and Inequality Constraints

Definition 3.1 *A scheduling policy f, is work-conserving iff it satisfies the following conditions:*

$$\sum_{i=1}^{N} d_i^f(n) = 0 \quad when \quad \sum_{i=1}^{N} \lambda_i(n) \leq T \quad (underloaded\ frame) \tag{3}$$

$$\sum_{i=1}^{N} a_i^f(n) = T \quad when \quad \sum_{i=1}^{N} \lambda_i(n) > T \quad (overloaded\ frame). \quad \square \tag{4}$$

A work-conserving policy does not waste resources (slots) as long as there is work to perform (packets to transmit). Let $S = \{1, 2, \ldots, N\}$ be the set of all sources. Let $d^f_{\{S\}}$ denote the average system packet dropping rate under scheduling policy$f$, given by,

$$d^f_{\{S\}} \triangleq E\left[\sum_{i=1}^{N} d_i^f(n)\right] = \sum_{i=1}^{N} E\left[d_i^f(n)\right] = \sum_{i=1}^{N} d_i^f. \tag{5}$$

The following theorem provides for a well know result for work-conserving policies [7].

Theorem 3.1 (Conservation Law) *The system dropping rate, $d^f_{\{S\}}$, is conserved under any work-conserving policy f; that is,*

$$d^f_{\{S\}} \triangleq \left\{E\left[\sum_{i\in S} \lambda_i(n) \;\Big|\; \sum_{i\in S} \lambda_i(n) > T\right] - T\right\} P\left(\sum_{i\in S} \lambda_i(n) > T\right) \triangleq b_{\{S\}} \quad \forall f. \tag{6}$$

Notice that the system dropping rate is independent of the policy f and only depends on the arrival process and the amount of resources T. □

Proof. The number of source i packets dropped during frame n under policy f is given by,

$$d_i^f(n) = \lambda_i(n) - a_i^f(n). \tag{7}$$

Summing (7) over all sources i in the system S yields,

$$\sum_{i\in S} d_i^f(n) = \sum_{i\in S} \left(\lambda_i(n) - a_i^f(n)\right) = \sum_{i\in S} \lambda_i(n) - \sum_{i\in S} a_i^f(n). \tag{8}$$

Assuming f is work-conserving and using equation (3) and (4) of Definition 3.1, the previous equation becomes,

$$\sum_{i\in S} d_i^f(n) = \begin{cases} 0 & \sum_{i\in S} \lambda_i(n) \le T \\ \sum_{i\in S} \lambda_i(n) - T & \sum_{i\in S} \lambda_i(n) > T. \end{cases} \tag{9}$$

By considering the expected value of (9) the proof of the theorem is completed. □

Let $d^f_{\{g\}}$ denote the average subsystem g packet dropping rate under policy f, defined by,

$$d^f_{\{g\}} = E\left[\sum_{i\in g} d_i^f(n)\right] = \sum_{i\in g} E\left[d_i^f(n)\right] = \sum_{i\in g} d_i^f, \quad g \subseteq S = \{1, 2, \ldots, N\}. \tag{10}$$

That is, $d^f_{\{g\}}$ is equal to the cumulative 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 policy f.

Let $b_{\{g\}}$ denote the lower bound for the cumulative packet dropping rate for sources in g, determined to be equal to the packet dropping rate of a system in which only sources in g are present and served under a work-conserving policy. Sources in set $\{S - g\}$ are considered to be removed. In view of Theorem 3.1 and by applying equation (6) for the subsystem g, the following is derived,

$$b_{\{g\}} = \left\{ E\left[\sum_{i\in g}\lambda_i(n) \;\Big|\; \sum_{i\in g}\lambda_i(n) > T\right] - T\right\} Pr\left(\sum_{i\in g}\lambda_i(n) > T\right), \; g \subseteq S. \tag{11}$$

The following theorem provides for a set of inequality constraints associated with scheduling policies.

Theorem 3.2 *The following constraints associated with the induced packet dropping rate vector* $\mathbf{d}^f = (d_1^f, d_2^f, \ldots, d_N^f)$ *are satisfied by any scheduling policy:*

$$d_{\{g\}}^f \geq b_{\{g\}}, \quad \forall g \subset S = \{1, 2, ..., N\}, \tag{12}$$

where $d_{\{g\}}^f$ *and* $b_{\{g\}}$ *are given in (10) and (11).* $\Box$

Proof. Suppose that the packets from sources in g are served under a work-conserving scheduling policy f_o in which they are given Head of the Line (HoL) priority over packets from sources in $\{S-g\}$. That is, no packet from sources in $\{S-g\}$ is served unless no packets from sources in g are present. As a consequence, packets from sources in g do not experience interference from the traffic delivered by sources in $\{S-g\}$ and thus,

$$\sum_{i\in g} d_i^{f_o}(n) \;=\; 0 \;\text{ when }\; \sum_{i\in g}\lambda_i(n) \leq T \tag{13}$$

$$\sum_{i\in g} a_i^{f_o}(n) \;=\; T \;\text{ when }\; \sum_{i\in g}\lambda_i(n) > T. \tag{14}$$

From (10) and (11) it is clear that,

$$d_{\{g\}}^{f_o} = b_{\{g\}}, \tag{15}$$

for all work-conserving policies that provide HoL priority to the packets from sources in g. Since no other policy f can provide better service (that is lower packet dropping rates) to sources in g than f_o, it is evident that,

$$d_{\{g\}}^f \geq b_{\{g\}}, \quad \forall f, \quad g \subset S. \;\Box \tag{16}$$

3.2 Achievable Region for QoS Vectors

The main result of this section is the determination of the region $\mathcal{D}$ of the achievable QoS vectors $\mathbf{d}$. The following corollary provides a set of necessary conditions in order for a QoS vector $\mathbf{d}$ to be achievable, followed by a corollary regarding an upper bound on the achievable region for QoS vectors $\mathbf{d}$. Their proofs are self-evident in view of Theorems 3.1 and 3.2.

Corollary 3.1 *A necessary condition in order for a QoS vector* $\mathbf{d} = (d_1, d_2, \ldots, d_N)$ *to be achieved by a work-conserving scheduling policy* f *is that its components satisfy the following constraints,*

$$d_{\{g\}} \;\geq\; b_{\{g\}} \quad \forall g \subset S \tag{17}$$

$$d_{\{S\}} \;=\; b_{\{S\}}. \;\Box \tag{18}$$

Corollary 3.2 *Let $\mathcal{D}^u$ denote the collection of all vectors* $\mathbf{d}$ *satisfying (17) and (18). Then $\mathcal{D}^u$ is an upper bound on the region $\mathcal{D}$ of achievable QoS vectors* $\mathbf{d}$. *That is,*

$$\mathcal{D} \subseteq \mathcal{D}^u. \ \Box \tag{19}$$

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

Theorem 3.3 *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, $\mathcal{D}^u$=conv[ext($\mathcal{D}^u$)].* $\Box$

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 (17) and (18)). Then by definition, $\mathcal{D}^u$ is a polytope [8] and Theorem 3.3 follows directly from properties of polytopes [8]. $\Box$

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

Theorem 3.4 $\mathbf{d}^*$ *is a vertex of the set $\mathcal{D}^u$ iff* $\mathbf{d}^*$ *is a dropping rate vector resulting from an Ordered HoL (O-HoL) priority service policy, $\boldsymbol{\pi}= (\pi_1, \pi_2, \ldots, \pi_N)$; $\pi_i \in \{1, 2, \ldots, N\}$, $\pi_i \neq \pi_j$, $1 \leq i, \ j \leq N$. The index of π_i indicates the order of the priority given to the π_i source. None of the π_j sources, $j > i$, may be served as long as packets from sources π_k, $k \leq i$, are present.* $\Box$

Proof. Assume that $\mathbf{d}^*$ is a vertex of $\mathcal{D}^u$. Then $\mathbf{d}^*$ must lie at the intersection of N hyperplanes and its coordinates must satisfy N simultaneous, linear independent equations (definition of a polytope vertex, see [8]), given by,

$$\sum_{i \in g_j} d_i^* = b_{\{g_j\}}, \qquad j = 1, 2, \ldots, N, \tag{20}$$

where one of the g_j's is the set $S = \{1, 2, \ldots, N\}$ and the remaining $(N-1)$ are proper, non-empty and different subsets of S. Lemma 8.2 (see appendix) establishes that the subsets g_i's are strictly included in each other. Therefore, by adopting the order $g_1 \subset g_2 \subset \ldots \subset g_{N-1} \subset g_N \equiv S$, the g_i's are given by,

$$\begin{aligned} g_1 &= \{\pi_1\} \\ g_1 \subset g_2 &= \{\pi_1, \pi_2\} \\ g_2 \subset g_3 &= \{\pi_1, \pi_2, \pi_3\} \\ &\vdots \\ g_{N-1} \subset g_N &= \{\pi_1, \pi_2, \ldots, \pi_N\} = \{1, 2, \ldots, N\}, \end{aligned} \tag{21}$$

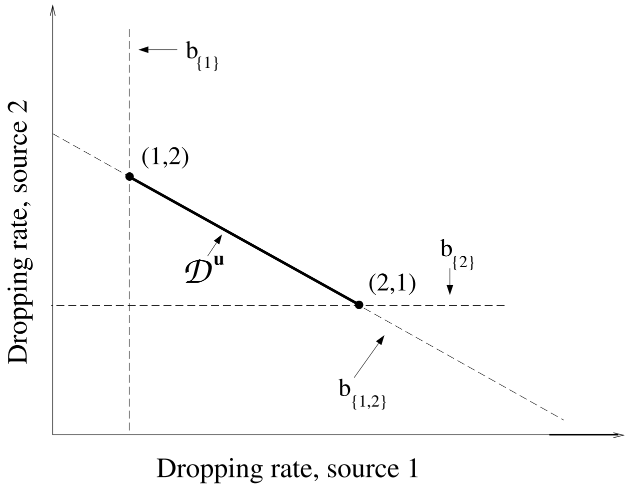


Figure 2: The region (polytope) $\mathcal{D}^u$ for a system with two sources.

where $\pi_i \in \{1, 2, \ldots, N\}$, $\pi_i \neq \pi_j$, for $i \neq j$, $1 \leq i$, $j \leq N$. Substituting into (20) yields,

$$\begin{aligned}
d^*_{\{\pi_1\}} &= b_{\{\pi_1\}} \\
d^*_{\{\pi_2\}} &= b_{\{\pi_1,\pi_2\}} - b_{\{\pi_1\}} \\
d^*_{\{\pi_3\}} &= b_{\{\pi_1,\pi_2,\pi_3\}} - b_{\{\pi_1,\pi_2\}} \\
&\vdots \\
d^*_{\{\pi_N\}} &= b_{\{\pi_1,\pi_2,\ldots,\pi_N\}} - b_{\{\pi_1,\pi_2,\ldots,\pi_{N-1}\}},
\end{aligned} \tag{22}$$

which is precisely the dropping rate vector induced by the ordered HoL priority policy $\boldsymbol{\pi} = (\pi_1, \pi_2, \ldots, \pi_N)$. Thus for any vertex there exists an ordered HoL priority policy that induces it. Since it is easy to see that the dropping rate vector resulting from an ordered HoL priority policy must satisfy (22), it is evident that these dropping rate vectors will be vertices of $\mathcal{D}^u$. □

Fig. 2 and Fig. 3 provide a graphical illustration of the region $\mathcal{D}^u$ for the case of $N = 2$ and $N = 3$ sources, respectively. The extreme points $\mathbf{d}_{ext-i}$'s correspond to QoS vectors induced by the $N!$ ordered HoL priority policies $\boldsymbol{\pi}^i = (\pi_1, \pi_2, \ldots, \pi_N)$, $1 \leq i \leq N!$, as shown in Theorem 3.4. Referring to Fig. 2, 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 1, $b_{\{1\}}$, 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 2, $b_{\{2\}}$ 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. 3.

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

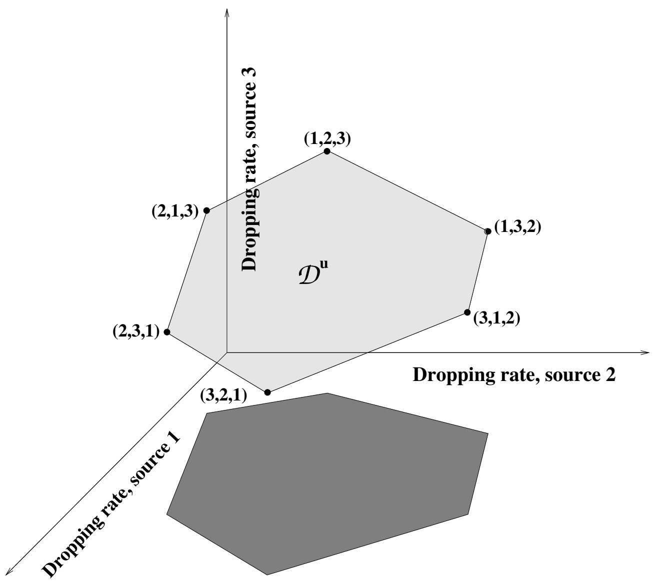


Figure 3: The region (polytope) $\mathcal{D}^u$ for a system with three sources.

Theorem 3.5 *Let $\mathbf{d}_1, \mathbf{d}_2 \in \mathcal{D}$, then $\mathbf{d}_3$, where*

$$\mathbf{d}_3 = \alpha\mathbf{d}_1 + (1-\alpha)\mathbf{d}_2, \qquad \alpha \geq 0, \quad \alpha \leq 1, \tag{23}$$

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

Proof. Let $\mathbf{d}_1$ ($\mathbf{d}_2$) be the dropping rate vector induced the achievable policy S_1 (S_2); that is, $\mathbf{d}_1, \mathbf{d}_2 \in \mathcal{D}$. Consider a scheduling policy S_3 that at each frame decides to follow the scheduling rule of policy S_1 with probability α and policy S_2 with probability $(1-\alpha)$; the decisions over consecutive frames are independent. The packet dropping rate induced by policy S_3 is given by,

$$\mathbf{d}_3 = \alpha\mathbf{d}_1 + (1-\alpha)\mathbf{d}_2. \tag{24}$$

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

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

Theorem 3.6 $\mathcal{D} \equiv \mathcal{D}^u$. $\Box$

Proof. Since $\mathcal{D} \subseteq \mathcal{D}^u$ (Corollary 3.2), it suffices to establish that $\mathcal{D}^u \subseteq \mathcal{D}$ to complete the proof. Notice that $\mathcal{D}^u = conv[ext(\mathcal{D}^u)]$ (Theorem 3.3) and that $\mathcal{D}$ is convex (Theorem 3.5) and, thus, if $\{ext(\mathcal{D}^u)\} \subseteq \mathcal{D}$ then $\mathcal{D}^u \subseteq \mathcal{D}$. The latter holds true since the extreme points of $\mathcal{D}^u$ are induced by the ordered HoL priority policies (Theorem 3.4) and, thus, these points are in the region $\mathcal{D}$ of achievable QoS vectors. $\Box$

4 Applications

In the proceeding sections a system of N sources sharing T resources has been considered and the region $\mathcal{D}$ of achievable QoS vectors $\mathbf{d}$ induced by any work-conserving policy has been established.

If the amount of available resources T is fixed, it is important to determine whether all N diverse applications can be supported. If the required resources can be made available, it is important to determine the minimum number of required resources. These two questions are addressed in this section.

4.1 Call Admission Control

The following definitions facilitates the discussion of this section.

Definition 4.1 *The worst-tolerable QoS-vector* $\hat{\mathbf{d}} = (\hat{d}_1, \hat{d}_2, \ldots, \hat{d}_N)$ *associated with the* N *applications is defined to be the one whose coordinates are equal to the largest packet dropping rate acceptable by the corresponding application.* $\square$

Definition 4.2 $\mathbf{d}^1 = (d_1^1, d_2^1, \ldots, d_N^1)$ *is said to be better than* $\mathbf{d}^2 = (d_1^2, d_2^2, \ldots, d_N^2)$ *- denoted by* $\mathbf{d}^1 \leq \mathbf{d}^2$ *- iff*

$$d_i^1 \leq d_i^2, \quad 1 \leq i \leq N. \square \tag{25}$$

Definition 4.3 *The QoS of* N *applications is said to be satisfied if there exists a policy inducing a packet dropping rate vector* $\mathbf{d}$ *(that is* $\mathbf{d} \in \mathcal{D}$*) which is better than the worst-tolerable QoS vector associated with these applications.* $\square$

Definition 4.4 *Let* $b_{\{g\}}$*,* $g \subseteq S = \{1, 2, \ldots, N\}$ *be the lower bound on the cumulative packet dropping rates associated with subsets of* N *given application and* T *available resources (* $b_{\{g\}}$ *is defined precisely earlier); let* $\mathcal{D}$ *be the associated region of achievable QoS vectors. The call admission region* $\mathcal{D}^+$ *is defined to be the set of dropping rate vectors* $\mathbf{d} = (d_1, d_2, \ldots, d_N)$ *satisfying,*

$$d_{\{g\}} \geq b_{\{g\}}, \quad \forall g \subseteq S. \square \tag{26}$$

Notice that for $g \equiv S$, (26) is satisfied with strict equality for the dropping rate vectors in $\mathcal{D}$. Thus, $\mathcal{D}^+$ is a polyhedral having $\mathcal{D}$ as one of its faces [8]. A graphical representation of the call acceptance region $\mathcal{D}^+$ is shown in Fig. 4 and Fig. 5 for a system of $N = 2$ and $N = 3$ sources, respectively. The following theorem establishes a call admission rule.

Theorem 4.1 *Let* $\hat{\mathbf{d}} = (\hat{d}_1, \hat{d}_2, \ldots, \hat{d}_N)$ *be the worst-tolerable QoS vector associated with* N *given applications. Then if* $\hat{\mathbf{d}}$ *is in the call-acceptance region* $\mathcal{D}^+$*, then the QoS of the* N *applications can be satisfied.* $\square$

Proof. The proof is self-evident provided that there exists a policy that induces a dropping rate vector $\mathbf{d}'$, such that $\mathbf{d}' \in \mathcal{D}$ and $\mathbf{d}' \leq \hat{\mathbf{d}}$. The latter holds true if,

$$\begin{aligned} \hat{d}_{\{g\}} &\geq d'_{\{g\}} \geq b_{\{g\}} \quad \forall g \subset S, \\ \hat{d}_{\{S\}} &\geq d'_{\{S\}} = b_{\{S\}}, \end{aligned} \tag{27}$$

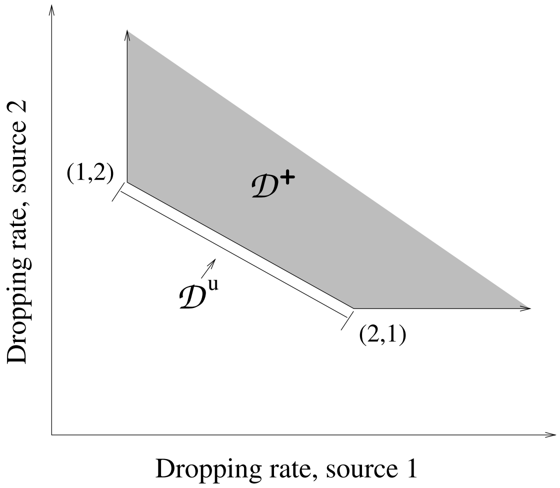


Figure 4: Achievable region $\mathcal{D}$ and call-admission region (polyhedral) $\mathcal{D}^+$ for a system with two sources.

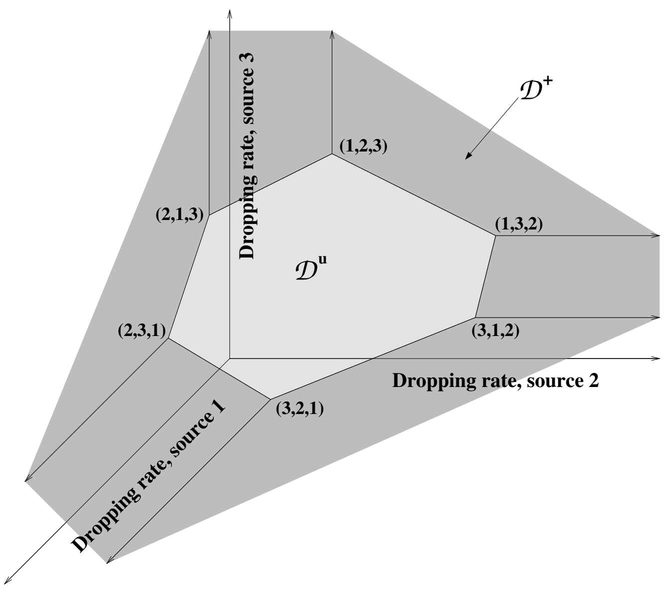


Figure 5: Achievable region $\mathcal{D}$ and call-admission region (polyhedral) $\mathcal{D}^+$ for a system with three sources.

which is established in [9] by denoting $\mathcal{D}$ to be the core of the game (S, b). According to [9] the core of game (S, b) consists of all vectors $\mathbf{d}$ which satisfy $d_{\{S\}} = b_{\{S\}}$ and $d_{\{g\}} \geq b_{\{g\}}$ for all $g \subset S$ where $d_{\{g\}} = \sum_{i \in g} d_i$; the core of the game is defined to be large if the conditions in (27) are satisfied for every $\hat{\mathbf{d}} \in \mathcal{D}^+$. A sufficient condition for the core of the game to be large is for the set function $b_{\{g\}}$ to be *super modular*, see Theorem 4 of [9]; this result has been established in the appendix, see Lemma 8.1. □

4.2 Resource Allocation

As implied earlier in the discussion associated with Fig. 1, applications in integrated services networks are not expected to be allowed to access *all* available resources. Some broad groups (or classes) are expected to be defined and only a proportion of the resources will typically be available to each group. If it is desirable that a certain number of N applications be supported, it is important that the minimum amount of required resources to be alloted to that group be determined. This problem is discussed below.

To emphasize the dependence of the quantities $\mathcal{D}, \mathcal{D}^+$ and $b_{\{g\}}$ for all $g \subseteq S$, on T, they will be written explicitly as functions of T and be written as $\mathcal{D}(T)$, $\mathcal{D}^+(T)$ and $b_{\{g\}}(T)$, respectively.

Theorem 4.2 *Let $\hat{\mathbf{d}} = (\hat{d}_1, \hat{d}_2, \ldots, \hat{d}_N)$ be the worst-tolerable QoS vector associated with N applications. The minimum amount of resources - denoted by T_{min} - required in order for the QoS of these applications to be satisfied is given by,*

$$T_{min} = \Big\{T,\ T \geq 1 : \Big[\Big(\hat{d}_{\{g\}} \geq b_{\{g\}}(T), \forall g \subseteq S\Big) \text{ and } \Big(\hat{d}_{\{g_o\}} < b_{\{g_o\}}(T-1) \text{ for some } g_o \subseteq S\Big)\Big]\Big\}. \tag{28}$$

Proof. Since $\hat{\mathbf{d}} \in \mathcal{D}^+(T)$, where $\mathcal{D}^+(T)$ is the call acceptance region determined by T and $\{b_{\{g\}},\ g \subseteq S\}$, the QoS of the N applications will be satisfied (Theorem 4.1). To establish that T_{min} determined by (28) is indeed the minimum it suffices to show that $b_{\{g\}}(T)$ are non-decreasing functions of T. This is obvious since the lower bound on the dropping rates for a group g, $b_{\{g\}}$, cannot increase as the number of service opportunities fixed service cycle increase. □

4.3 The Special Case of Homogeneous Applications

In this section the theory developed earlier is applied to a system supporting identical applications. That is, both the packet arrival process and the largest acceptable packet dropping rates are identical. In this case, the worst-tolerable QoS vector will have identical coordinates; that is $\hat{\mathbf{d}} = (\hat{d}_v, \hat{d}_v, \ldots, \hat{d}_v)$. The following theorems establish a (simplified) call admission rule for a homogeneous system.

Theorem 4.3 *Let* $\hat{\mathbf{d}} = (\hat{d}_v, \hat{d}_v, \ldots, \hat{d}_v)$ *be the worst-tolerable QoS vectors associated with the* N *homogeneous applications. The vector* $\hat{\mathbf{d}}$ *is in the call-acceptance region* $\mathcal{D}^+$ *(and, thus, the QoS of all* N *applications is satisfied) if*

$$\hat{d}_S \geq b_S \text{ or } \hat{d}_v \geq \frac{b_{\{S\}}}{N}. \ \Box \tag{29}$$

Proof. The definition of $\mathcal{D}^+$ (Definition 4.4) and the call-admission rule (Theorem 4.1) establish that (29) is a necessary and sufficient condition provided that it implies that,

$$d_{\{g\}} \geq b_{\{g\}} \text{ or } \hat{d}_v \geq \frac{b_{\{g\}}}{|g|}, \quad \forall g \subseteq S, \tag{30}$$

where $|g|$ denotes the number of elements (sources) in set g. In view of (29), (30) will hold if,

$$\frac{b_{\{S\}}}{N} \geq \frac{b_{\{g\}}}{|g|}, \quad \forall g \subseteq S. \tag{31}$$

Since all the sources are statistically identical, set functions $b_{\{g\}}$ associated with sets g having the same number of sources are equal. That is,

$$b_{\{a_1,\ldots,a_m\}} = b_{\{1,\ldots,m\}} \ ,\forall\{a_1,\ldots,a_m\}, \ m = 1,2,\ldots,N, \ a_i \neq a_j, \ a_i, a_j \in S. \tag{32}$$

Using (32) and Lemma 8.1,

$$\begin{aligned} 2b_{\{a_1\}} &= b_{\{a_1\}} + b_{\{a_2\}} < b_{\{a_1,a_2\}} \\ 3b_{\{a_1,a_2\}} &= b_{\{a_1,a_2\}} + b_{\{a_1,a_3\}} + b_{\{a_2,a_3\}} < b_{\{a_1,a_2,a_3\}} + b_{\{a_1\}} + b_{\{a_2,a_3\}} < 2b_{\{a_1,a_2,a_3\}} \\ &\vdots \\ (|g|+1)b_{\{g\}} &< |g|b_{\{g\cup a\}} \qquad a \notin \{g\}. \end{aligned} \tag{33}$$

Equivalently,

$$\frac{b_{\{g\cup a\}}}{|g|+1} > \frac{b_{\{g\}}}{|g|} \qquad a \notin \{g\}, \tag{34}$$

and (31) immediately follows from this result.

5 A Class of Policies Inducing All QoS Vector in $\mathcal{D}$

Let C_{O-HoL} denote the class of ordered HoL priority service policies $\boldsymbol{\pi}^i$'s introduced in section 3.2.

Definition 5.1 *A mixing ordered HoL priority service policy* f_m *is defined to be one that at each frame decides to follow the ordered HoL priority policy* $\boldsymbol{\pi}^i = (\pi_1^i, \pi_2^i, \ldots, \pi_N^i)$ *with probability* $\alpha_i, \alpha_i \geq 0, \ 1 \leq i \leq N!, \ \sum_{i=1}^{N!} \alpha_i = 1$*; decisions over consecutive frames are independent. Clearly,* f_m *is completely determined by the* $N!$ *dimensional vector* $\boldsymbol{\alpha}$, $\boldsymbol{\alpha} \geq \mathbf{0}$, $\mathbf{1} \cdot \boldsymbol{\alpha} = 1$. *Let* M_{O-HoL} *denote the class of such policies.* $\Box$

The following theorem establishes the main results of this section.

Theorem 5.1 *For each packet dropping rate vector* $\mathbf{d} \in \mathcal{D}$ *there exist a policy* $f_m \in M_{O-HoL}$ *that induces* $\mathbf{d}$*.* $\square$

Proof. Let $\mathbf{d} \in \mathcal{D}$. Then $\mathbf{d} = \sum_{i=1}^{N!} \alpha_i \mathbf{d}_{ext-i}$ for some $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_{N!})$ where $\alpha_i \geq 0,\ 1 \leq i \leq N!,\ \sum_{i=1}^{N!} \alpha_i = 1$, since any point in $\mathcal{D}$ can be written as a convex combination of the extreme points (vertices) $\mathbf{d}_{ext-i}$ of $\mathcal{D}(=\mathcal{D}^u)$; each $\mathbf{d}_{ext-i}$ is induced by some policy in C_{O-HoL} (Theorem 3.4).

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

$$\mathbf{d}^{f_m} = \sum_{i=1}^{N!} \alpha_i \mathbf{d}_{ext-i} \tag{35}$$

and thus f_m induces $\mathbf{d}$. $\square$

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

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

$$E_{\boldsymbol{\alpha}}[\mathbf{d}_{ext}] \stackrel{\triangle}{=} \sum_{i=1}^{N!} \alpha_i \mathbf{d}_{ext-i} = \mathbf{d}, \tag{36}$$

$$\boldsymbol{\alpha} \geq \mathbf{0}, \tag{37}$$

$$\mathbf{1} \cdot \boldsymbol{\alpha} = 1, \tag{38}$$

where $E_{\boldsymbol{\alpha}}[.]$ *is weighted average of the set of extreme points,* $\mathbf{d}_{ext}$*, of* $\mathcal{D}$ *with respect to the Probability Mass Function* $\boldsymbol{\alpha}$*.* $\square$

6 A Numerical Example

In this section a numerical example is carried out for a system with three sources competing for T slots in a TDMA frame. The source packet arrival processes are assumed to be mutually independent. Each arrival process is described in terms of an underlying finite-state Markov chain embedded at the frame boundaries. The number of packets generated (and requesting service) by a source in the current frame boundary is (probabilistically) determined by the present state of the associated underlying Markov chain.

Consider in this example the typical TDMA frame in which each slot represents a 32 Kbps channel. Let $E^k = \{0, 1, \ldots, M^k - 1\}$ denote the state space of the Markov chain associated with source k, $k = 1, 2, 3$. In this example $M^1 = 2$, $M^2 = 3$ and $M^3 = 4$. When in state i, a source generates $2i$ packets in the current frame. Source 1, has a maximum transmission rate of 64 Kbps, it can model an uncompressed voice source [10]. Source 2 is the superposition of two uncompressed

Source 1 State, i	α_i^1	β_i^1	State probability
0	0.667	-	0.6
1	-	1.0	0.4

Table 1: Description of source 1.

Source 2 State, i	α_i^2	β_i^2	State probability
0	1.330	-	0.36
1	0.667	1	0.48
2	-	2	0.16

Table 2: Description of source 2.

voice sources. Source 3 could be used to model a low rate video transmission with maximum rate 6 slots/frame = 192 Kbps and step size of 64 Kbps. Further description of all three sources, including transition rates, can found in Tables 1, 2 and 3; α_i^k (β_i^k) denote the transition rates from state i to state $i+1$ (state i to $i-1$), of source k; $i,\ i-1,\ i+1 \in E^k$, $k = 1, 2, 3$. The per frame arrival rates are easily computed and are equal to $\lambda_1 = 0.8$, $\lambda_2 = 1.6$ and $\lambda_3 = 4.0$ packets per frame.

Let p_k denote the target packet dropping probability, for source k, $k = 1, 2, 3$. In this example it is assumed that $p_1 = 0.02$, $p_2 = 0.04$ and $p_3 = 0.02$. By multiplying these probabilities by the corresponding per frame arrival rate, the per frame packet dropping rates $d_1 = 0.016$, $d_2 = 0.064$ and $d_3 = 0.08$ are obtained. Thus, the worst-tolerable QoS vector $\mathbf{d}$ is given by $\mathbf{d} = (d_1, d_2, d_3) = (0.016, 0.064, 0.08)$ and the associated system packet dropping rate is equal to $d_{\{S\}} = d_1 + d_2 + d_3 = 0.160$. By applying Theorem 4.2 it is established that the minimum amount of resources required

Source 3 State, i	α_i^3	β_i^3	State probability
0	2.7000	-	0.05
1	4.4000	0.9	0.15
2	0.6818	1.2	0.55
3	-	1.5	0.25

Table 3: Description of source 3.

in order for $\mathbf{d}$ to be in the call-acceptance region are $T_{min} = 9$ slots per TDMA frame.

In this case $d_{\{S\}} \geq b_{\{S\}} = 0.1552$ and any vector $\mathbf{d}'$ satisfying conditions in Theorem 4.1 can be selected. The (non-unique) vector $\mathbf{d}' = (0.016, 0.064, 0.0752)$ was found by requiring (last constraint in Theorem 4.1) that $d'_{\{S\}} = b_{\{S\}} = 0.1552$. Since $\mathbf{d}' \in \mathcal{D}$, there exists a mixing ordered HoL priority policy $f_m \equiv \boldsymbol{\alpha}$ achieving exactly the QoS vector $\mathbf{d}'$. Any $\boldsymbol{\alpha}$ satisfying the conditions of Corollary 5.1, may be chosen. For this example and employing linear programming techniques, the following $\boldsymbol{\alpha}_o$ was found,

$$\boldsymbol{\alpha}_o = \begin{bmatrix} \alpha_1 = 0.1280 \\ \alpha_2 = 0.0000 \\ \alpha_3 = 0.2843 \\ \alpha_4 = 0.1810 \\ \alpha_5 = 0.2678 \\ \alpha_6 = 0.1389 \end{bmatrix} \tag{39}$$

As it 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 achievability of the resulting policies. For instance, among all mixing policies inducing $\mathbf{d}'$, the one which minimizes the variance of the service provided to certain sources may be identified. Such additional objectives will be pursued in the future.

To further illustrate the design concept, consider the following two user system. In this example $M^1 = 3$, $M^2 = 4$. As in the first example, when in state i a source generates $2i$ packets in the current frame. Source 1 and source 2 have probability state distributions, $(0.1, 0.3, 0.6)$ and $(0.5, 0.3, 0.1, 0.1)$, respectively. The per frame arrival rates are easily computed and are equal to $\lambda_1 = 3.0$ and $\lambda_2 = 1.6$ packets per frame.

In this example the worst-tolerable QoS vector $\mathbf{d}$ is given by $\mathbf{d} = (d_1, d_2) = (0.781, 0.099)$ and the associated system packet dropping rate is equal to $d_{\{S\}} = d_1 + d_2 = 0.790$. In this example, the system packet dropping rate can be satisfied with $T = 5$ slots, that is $d_{\{S\}} = b_{\{S\}} = 0.790$. Although with $T = 5$, $d_2 < b_2 = 0.10$, and thus the worst-tolerable QoS vector is not achievable. In this case the best possible performance that source 2 can receive is if it is given priority, resulting in dropping rates of 0.780 and 0.10 for source 1 and 2 respectively. The overall system performance is satisfied, but source 2 is experiencing poorer service than its worst-tolerable, while source 1 is experiencing improved performance. Applying Theorem 4.2 it is established that the minimum amount of resources required in order for $\mathbf{d}$ to be in the call-acceptance region are $T_{min} = 6$ slots per TDMA frame.

7 Conclusion

In this work the region of achievable QoS vectors for an integrated services wireless network was precisely determined. The QoS in a shared un-buffered environment (wireless network) is measured by the packet dropping probability, p_i. The determination of the region of achievability for this environment was first made possible by noticing that a conservation law applies to the packet dropping rate, d_i, from which a packet dropping probability can be inferred as $p_i = \frac{d_i}{\lambda_i}$. Secondly, the super-modularity of the set function $b_{\{g\}}$, describing the lower bound on the cumulative packet dropping rate, made it possible to classify the extreme points (vertices) of the region of achievability.

Knowledge of the region of achievable QoS vectors led to the development of an admission control rule and resource allocation techniques for the applications in the integrated services wireless network. Admission control for a special case of a homogeneous system was considered. The anticipated result, $d_v \geq \frac{b_{\{S\}}}{N}$, or equivalently, $p_i \geq \frac{b_{\{S\}}}{N\ \lambda_v}$, which had been seen in [3] as a sufficient condition for admission to a cellular voice network, was derived.

Once it was established that the target QoS vector was in the achievable region, easy to implement scheduling policies were derived that achieve the desired result. It was shown that there exists a class of scheduling policies - mixed Ordered HoL policies - that can achieve any achievable QoS vector.

8 Appendix

Lemma 8.1 *b_A is a super-modular set function, that is,*

$$b_A + b_B \leq b_{\{A \cup B\}} + b_{\{A \cap B\}} \tag{40}$$

and equality hold only if $A \subset B$ or $B \subset A$. $\square$

Proof. Let λ_A (λ_B) be the generic random variables representing the number of packets generated by sources in set A (B) over a frame, $A, B, \subseteq S$. Clearly,

$$\lambda_A + \lambda_B = \lambda_{\{A \cup B\}} + \lambda_{\{A \cap B\}} \quad \forall A, B \subseteq S. \tag{41}$$

Case (I): Assume A and B are disjoint ($A \cap B = \emptyset$). In this case $\lambda_{\{A \cup B\}} = \lambda_A + \lambda_B$ and,

$$b_{\{A \cup B\}} \quad = \quad \left\{E\left[\lambda_A + \lambda_B \;\middle|\; \lambda_A + \lambda_B > T\right] - T\right\} Pr\left(\lambda_A + \lambda_B > T\right). \tag{42}$$

$$\tag{43}$$

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, then,

$$\begin{aligned} Pr\left(\lambda_A + \lambda_B > T\right) = Pr\left(\lambda_A > T\right) P\left(\lambda_B \leq T\right) + Pr\left(\lambda_A \leq T\right) P\left(\lambda_B > T\right) \\ + Pr\left(\lambda_A \leq T, \lambda_B \leq T, \lambda_A + \lambda_B > T\right) \\ + Pr\left(\lambda_A > T\right) P\left(\lambda_B > T\right), \end{aligned} \tag{44}$$

therefore (42) may be written as,

$$\begin{aligned} b_{\{A \cup B\}} = \left\{E\left[\lambda_A + \lambda_B \;\middle|\; \lambda_A > T, \lambda_B \leq T\right] - T\right\} Pr\left(\lambda_A > T\right) Pr\left(\lambda_B \leq T\right) \\ + \left\{E\left[\lambda_A + \lambda_B \;\middle|\; \lambda_A \leq T, \lambda_B > T\right] - T\right\} Pr\left(\lambda_A \leq T\right) Pr\left(\lambda_B > T\right) \\ + \left\{E\left[\lambda_A + \lambda_B \;\middle|\; \lambda_A > T, \lambda_B > T\right] - T\right\} Pr\left(\lambda_A > T\right) Pr\left(\lambda_B > T\right) \\ + \left\{E\left[\lambda_A + \lambda_B \;\middle|\; \lambda_A \leq T, \lambda_B \leq T, \lambda_A + \lambda_B > T\right] - T\right\} \\ Pr\left(\lambda_A \leq T, \lambda_B \leq T, \lambda_A + \lambda_B > T\right). \end{aligned} \tag{45}$$

Expanding the expected value operator over λ_A and λ_B and rearranging terms, (45) can be written as,

$$b_{\{A \cup B\}} = \left\{E\left[\lambda_A \;\middle|\; \lambda_A > T\right] - T\right\} Pr\left(\lambda_A > T\right) \tag{46}$$

$$
\begin{aligned}
&+\left\{E\left[\lambda_B \,\middle|\, \lambda_B > T\right] - T\right\} Pr\left(\lambda_B > T\right) \\
&\qquad +E\left[\lambda_A \,\middle|\, \lambda_A \le T\right] Pr\left(\lambda_A \le T\right) Pr\left(\lambda_B > T\right) + E\left[\lambda_B \,\middle|\, \lambda_B \le T\right] \\
&\qquad Pr\left(\lambda_A > T\right) P\left(\lambda_B \le T\right) + T Pr\left(\lambda_A > T\right) P\left(\lambda_B > T\right) \\
&\qquad +\left\{E\left[\lambda_A + \lambda_B \,\middle|\, \lambda_A \le T, \lambda_B \le T, \lambda_A + \lambda_B > T\right] - T\right\} \\
&\qquad Pr\left(\lambda_A \le T, \lambda_B \le T, \lambda_A + \lambda_B > T\right) \\
&\qquad = b_{\{A\}} + b_{\{B\}} + C^{\lambda_A, \lambda_B}
\end{aligned}
$$

where,

$$
\begin{aligned}
C^{\lambda_A,\lambda_B} = {} & E\left[\lambda_A \,\middle|\, \lambda_A \le T\right] Pr\left(\lambda_A \le T\right) Pr\left(\lambda_B > T\right) + E\left[\lambda_B \,\middle|\, \lambda_B \le T\right] \\
& Pr\left(\lambda_A > T\right) P\left(\lambda_B \le T\right) + T Pr\left(\lambda_A > T\right) P\left(\lambda_B > T\right) \\
& +\left\{E\left[\lambda_A + \lambda_B \,\middle|\, \lambda_A \le T, \lambda_B \le T, \lambda_A + \lambda_B > T\right] - T\right\} \\
& Pr\left(\lambda_A \le T, \lambda_B \le T, \lambda_A + \lambda_B > T\right).
\end{aligned} \tag{47}
$$

Notice that $C^{\lambda_A,\lambda_B} > 0$ (for the cases of interest corresponding to $T > 0$). Thus, from (46), and since $b_{\{A\cap B\}} = 0$ for $\{A \cap B\} = \emptyset$,

$$
b_A + b_B = b_{\{A\cup B\}} - C^{\lambda_A,\lambda_B} = b_{\{A\cup B\}} + b_{\{A\cap B\}} - C^{\lambda_A,\lambda_B}, \tag{48}
$$

or,

$$
b_A + b_B < b_{\{A\cup B\}} + b_{\{A\cap B\}}. \tag{49}
$$

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 $\{B - \{A \cap B\}\}$ and using the results from Case I (46), the following can be written.

$$
b_{\{A\cup B\}} = b_{\{A\cup\{B-\{A\cap B\}\}} = b_A + b_{\{B-\{A\cap B\}\}} + C^{\lambda_A,\lambda_{\{B-\{A\cap B\}\}}} \tag{50}
$$

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

$$
b_B = b_{\{A\cap B\}} + b_{\{B-\{A\cap B\}\}} + C^{\lambda_{\{A\cap B\}},\lambda_{\{B-\{A\cap B\}\}}} \tag{51}
$$

or,

$$
b_{\{B-\{A\cap B\}\}} = b_B - b_{\{A\cap B\}} - C^{\lambda_{\{A\cap B\}},\lambda_{\{B-\{A\cap B\}\}}}. \tag{52}
$$

Finally, by combining (50) and (52), the following is obtained,

$$
b_{\{A\cup B\}} + b_{\{A\cap B\}} = b_A + b_B + C^{\lambda_A,\lambda_{\{B-\{A\cap B\}\}}} - C^{\lambda_{\{A\cap B\}},\lambda_{\{B-\{A\cap B\}\}}} \tag{53}
$$

If $A \subset B$ then,

$$
C^{\lambda_A,\lambda_{\{B-\{A\cap B\}\}}} - C^{\lambda_{\{A\cap B\}},\lambda_{\{B-\{A\cap B\}\}}} = C^{\lambda_A,\lambda_{\{B-A\}}} - C^{\lambda_A,\lambda_{\{B-A\}}} = 0 \tag{54}
$$

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

$$C^{\lambda_A, \lambda_{\{B-\{A\cap B\}\}}} - C^{\lambda_{\{A\cap B\}}, \lambda_{\{B-\{A\cap B\}\}}} > 0, \tag{55}$$

completing the proof of the lemma. □

Lemma 8.2 *The sets g_j, $j = 1, 2, \ldots, N$, that satisfy equation (20), defining the vertices of the polytope $\mathcal{D}^u$, must be strictly included in each other; that is*

$$\textit{either } h_j = g_j - (g_j \cap g_k) = \emptyset \textit{ or } h_k = g_k - (g_k \cap g_j) = \emptyset, \tag{56}$$

for all g_j and g_k in (20). □

Proof. Let g_1 and g_2 be two sets satisfying (20). By adding the corresponding equations the following is obtained.

$$b_{\{g_1\}} + b_{\{g_2\}} = \sum_{i\in g_1} d_i^* + \sum_{i\in g_2} d_i^* \quad = \sum_{i\in\{g_1\cup g_2\}} d_i^* + \sum_{i\in\{g_1\cap g_2\}} d_i^* \tag{57}$$

or,

$$\sum_{i\in\{g_1\cup g_2\}} d_i^* = b_{\{g_1\}} + b_{\{g_2\}} - \sum_{i\in\{g_1\cap g_2\}} d_i^*. \tag{58}$$

Since for any subset of S (17) must be satisfied, then for $\{g_1 \cap g_2\}$, $\sum_{i\in\{g_1\cap g_2\}} d_i^* \geq b_{\{g_1\cap g_2\}}$, the following is obtained from (58),

$$\sum_{i\in\{g_1\cup g_2\}} d_i^* \leq b_{\{g_1\}} + b_{\{g_2\}} - b_{\{g_1\cap g_2\}}. \tag{59}$$

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 non-empty. Then, Lemma 8.1 implies that,

$$b_{\{g_1\}} + b_{\{g_2\}} < b_{\{g_1\cup g_2\}} + b_{\{g_1\cap g_2\}} \tag{60}$$

or,

$$b_{\{g_1\}} + b_{\{g_2\}} - b_{\{g_1\cap g_2\}} < b_{\{g_1\cup g_2\}}. \tag{61}$$

Therefore,

$$\sum_{i\in\{g_1\cup g_2\}} d_i^* \leq b_{\{g_1\}} + b_{\{g_2\}} - b_{\{g_1\cap g_2\}} < b_{\{g_1\cup g_2\}} \tag{62}$$

or,

$$\sum_{i\in\{g_1\cup g_2\}} d_i^* < b_{\{g_1\cup g_2\}}, \tag{63}$$

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

References

[1] G. Chen and I. Stavrakakis. ATM Traffic Management with Diversified Loss and Delay Requirements. In *Proceedings of IEEE INFOCOM*, San Francisco, CA, March 1996.

[2] D.J. Goodman, R.A. Valenzeula, et al. Packet Reservation Multiple Access for Local Wireless Communication. *IEEE Transactions on Communications*, 37(8):885–890, August 1989.

[3] F. Li and L.F. Merakos. Voice Data Integration in Digital TDMA Cellular System. In *Proceedings of International Zurich Seminar on Mobile Communications*, Zurich, Switzerland, March 1994.

[4] D. Bantz and F. Baucho. Wireless LAN Design Alternatives. *IEEE Network Magazine*, 8(2):43–45, March-April 1994.

[5] D. Raychaudhuri and N. Wilson. ATM-Based Transport Architecture for Multiservices Wireless Personal Communication Networks. *IEEE Journal of Selected Areas Communications*, 12(8):1401–1414, October 1994.

[6] M. Karol, Z. Liu, and K. Eng. An Efficient Demand-Assignment Multiple Access Protocol for Wireless Packet (ATM) Networks. *Wireless Networks*, 1(4):267–279, December 1995.

[7] E.G. Coffman and I. Mitrani. A Characterization of Waiting Time Performance Realizable by Single-Server Queues. *Operations Research*, 28(3):810–82, May-June 1980.

[8] Arne Brøndsted. *An Introduction to Convex Polytopes.* Springer-Verlag, 1983.

[9] W.W Sharkey. Cooperative Games with Large Cores. *International Journal of Game Theory*, 11:175–182, 1982.

[10] P.T. Brady. A Model for On-Off Speech Patterns in Two-way Conversation. *Bell Systems Technical Journal*, 48:885–890, September 1969.

[11] S. Nanda, D.J. Goodman, and U. Timor. Performance of PRMA: A Packet Voice Protocol for Cellular Systems. *IEEE Transactions on Vehicular Technology*, 40(3):584–589, August 1991.

[12] Keith W. Ross and David D. Yao. Optimal Dynamic Scheduling in Jackson Networks. *IEEE Transactions on Automatic Control*, 34(1):47–53, January 1989.

[13] D.J.A. Welsh. *Matroid Theory.* Academic Press, 1976.