Scheduling VBR Traffic for Achievable QoS in a Shared Un-buffered Environment

Jeffrey M. Capone and Ioannis Stavrakakis

Communications and Digital Signal Processing (CDSP) Center
Department of Electrical and Computer Engineering
Northeastern University
Boston, MA 02115

Abstract - **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 service cycle are dropped. The Quality of Service (QoS) vector is equivalently described in terms of (diverse) packet dropping probabilities (p_i), or (diverse) packet dropping rates (d_i). From a precisely defined region of achievable QoS vectors, easy to implement scheduling policies that deliver the achievable QoS vectors are derived. As an example, a policy that delivers a target QoS vector is derived; its performance is verified through simulations.**

I. INTRODUCTION

The problem of sharing transmission resources among disperse users with Variable Bit Rate (VBR) traffic and diverse Quality of Service (QoS) requirements is address in this paper. The transmission resource is assumed to be allocated in a well defined service cycles. Packets (information units) that are not transmitted during the service cycle following their arrival time are considered to have excess delay and are dropped. Applications with such QoS requirements are considered to un-buffered. For example, a wireless networking environment supporting real time applications has characteristics of a shared un-buffered environment in which management of the transmission resources is critical due to its scarcity.

In this work, the shared transmission resources are defined to be the slots (packet transmission times) of a TDMA frame (service cycle). This resource structure has been widely considered in both cellular systems [4] and wireless LANs [5], as well as in recent work toward the development of wireless ATM networks [6], [7]. VBR applications with distinct QoS requirements and traffic characteristics are considered. A method of scheduling the transmissions for the VBR traffic in this environment is examined.

Consider for instance the transmission scheduling problem in a cellular (voice) system. Since the supported (voice) applications are identical - in terms of traffic characteristics and QoS requirements - the transmission scheduling problem 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 $p_S \leq 0.01$. 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 TDMA frame. The uniformity in the QoS requirements and traffic characteristics of the supported voice applications simplifies substantially the transmission scheduling problem. A uniform, non-discriminating scheduling policy would induce the same (system) packet dropping probability to all supported applications.

When sources have diverse QoS requirements and traffic characteristics, a non-discriminating scheduling policy will typically not deliver diverse QoS. In this case it is important to design scheduling policies that deliver the target QoS, provided that the target QoS vector is achievable with the available resources. These issues are addressed in this paper. Details regarding the establishment of the region of achievable QoS vectors for given resources may be found in [1].

In the next section, the system model considered in this work is described. A set of equality and inequality constraints are presented in section III that are employed in the establishment of the region of the achievable QoS. Based on the developments of section III, a class of simple scheduling policies that deliver *any* achievable QoS vector is developed in section IV. A numerical example and simulation results are presented in section V, followed by conclusion of this work in section VI.

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.

II. 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

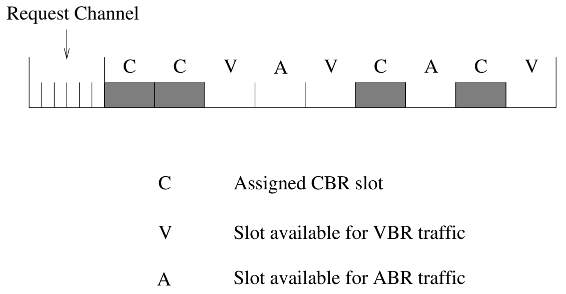

Figure 1. TDMA frame supporting CBR, VBR and ABR traffic classes.

$\lambda_i(n),\ 0 \leq i \leq N$, 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,

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

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.

Suppose that the QoS requirement of application i is defined in terms of a maximum tolerable average 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 $\mathbf{d}$,

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

When the QoS requirement of the 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 III) is whether a given QoS vectors $\mathbf{d}$ is achievable under any policy f. The second question, addressed in section IV, is concerned with the design of scheduling policies that deliver an achievable target QoS vector $\mathbf{d}$.

III. DETERMINATION OF THE REGION OF ACHIEVABLE QOS VECTORS

The establishment of the region of achievable QoS vectors is based on a set of equality/inequality constraints derived by employing work-conserving arguments. Details regarding these derivations may be found in [1]. A scheduling policy f, is work-conserving iff it satisfies the following conditions:

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

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

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 $d_{\{S\}}^f$ denote the average system packet dropping rate under scheduling policy f, denoted by,

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

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

$$d_{\{g\}}^f = E\left[\sum_{i \in g} d_i^f(n)\right] = \sum_{i \in g} d_i^f ,\ g \subseteq S. \quad (6)$$

That is, $d_{\{g\}}^f$ 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. It is given by,

$$b_{\{g\}} = \left\{ E\left[\sum_{i\in g} \lambda_i(n) \;\middle|\; \sum_{i\in g} \lambda_i(n) > T \right] - T \right\} \cdot \\ Pr\left(\sum_{i\in g} \lambda_i(n) > T \right), \; g \subseteq S. \quad (7)$$

It is shown in [1] that necessary and sufficient conditions for a QoS vector $\mathbf{d} = (d_1, d_2, \ldots, d_N)$ to be achieved by a scheduling policy f is that its components satisfy the following constraints,

$$d_{\{g\}} \geq b_{\{g\}} \quad \forall g \subseteq S \quad (8)$$
$$d_{\{S\}} = b_{\{S\}}. \quad (9)$$

Let $\mathcal{D}$ denote the collection of all vectors $\mathbf{d}$ satisfying (8) and (9). Any vector in the set $\mathcal{D}$ can be expressed as a convex combination of extreme points (vertices) of $\mathcal{D}$; that is, $\mathcal{D}$ may be expressed as the convex hull of its extreme points, $\mathcal{D}$=conv[exp($\mathcal{D}$)], see [1]. It is shown in [1] that $\mathbf{d}^*$ is a vertex of the set $\mathcal{D}$ 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.

IV. A CLASS OF POLICIES INDUCING ALL QOS VECTOR IN $\mathcal{D}$

Let C_{O-HoL} denote the class of ordered HoL priority service policies $\boldsymbol{\pi}$ introduced in section III. 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.

For each packet dropping rate vector $\mathbf{d} \in \mathcal{D}$ there exists a policy $f_m \in M_{O-HoL}$ that induces $\mathbf{d}$. The proof follows from the fact that $\mathcal{D}$ can be written as a convex combination of the extreme points (vertices) $\mathbf{d}_{ext-i}$ of $\mathcal{D}$, that is $\mathbf{d} = \sum_{i=1}^{N!} \alpha_i \mathbf{d}_{ext-i}$ 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 each $\mathbf{d}_{ext-i}$ is induced by some policy in C_{O-HoL}, a mixing policy f_m that selects the HoL priority $\boldsymbol{\pi}^i$ (that induces $\mathbf{d}_{ext-i}$) with probability α_i such that $\alpha_i \geq 0$, $1 \leq i \leq N!$, $\sum_{i=1}^{N!} \alpha_i = 1$ will have a packet dropping rate vector $\mathbf{d}^{f_m}$ given by,

$$\mathbf{d}^{f_m} = \sum_{i=1}^{N!} \alpha_i \mathbf{d}_{ext-i} \quad (10)$$

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

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,

$$E_{\boldsymbol{\alpha}}[\mathbf{d}_{ext}] \triangleq \sum_{i=1}^{N!} \alpha_i \mathbf{d}_{ext-i} = \mathbf{d}, \quad (11)$$
$$\boldsymbol{\alpha} \geq \mathbf{0}, \quad (12)$$
$$\mathbf{1} \cdot \boldsymbol{\alpha} = 1, \quad (13)$$

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 $\boldsymbol{\alpha}$.

V. 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 [8]. Source 2 is the superposition of two uncompressed 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 I, II and III; 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 $p_1 = 0.0200$, $p_2 = 0.0400$ and $p_3 = 0.0188$. 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 $d_3 = 0.075$ are obtained. Thus, the QoS vector $\mathbf{d}$ is given by $\mathbf{d} = (d_1, d_2, d_3) = (0.016, 0.064, 0.075)$ and

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

Table I
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 II
Description of Source 2.

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 III
Description of Source 3.

the associated system packet dropping rate is equal to $d_{\{S\}} = d_1 + d_2 + d_3 = 0.155$. By satisfying (8) and (9), it can be established that the minimum amount of resources required in order for $\mathbf{d}$ to be achievable are $T_{min} = 9$ slots per TDMA frame. It is possible for T_{min} to be found when (9) is satisfied with inequality, that is $d_{\{S\}} \geq b_{\{S\}}$. In this case there exits a vector $\mathbf{d}'$ such that $d'_i \leq d_i$, $i = 1, 2, \ldots, N$ and $d'_{\{S\}} = b_{\{S\}}$[1]. $\mathbf{d}'$ is said to have improved performance.

Since $d_{\{S\}} = d_1 + d_2 + d_3 = 0.155 = b_{\{S\}}$ then $\mathbf{d} \in \mathcal{D}$ and thus 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 (11), (12) and (13) may be chosen. The following $\boldsymbol{\alpha}_o$ was found,

$$\boldsymbol{\alpha}_o = \begin{bmatrix} \alpha_1 = 0.1280 \\ \alpha_2 = 0.0000 \\ \alpha_3 = 0.2843 \\ \alpha_4 = 0.1810 \\ \alpha_5 = 0.2678 \\ \alpha_6 = 0.1389 \end{bmatrix} \tag{14}$$

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 it is assumed that $p_1 = 0.2603$ and $p_2 = 0.0619$. By multiplying these probabilities by the corresponding per frame arrival rate, the per frame packet dropping rates $d_1 = 0.781$ and $d_2 = 0.099$ are obtained. Thus, 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 dropping rate 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 target QoS, while

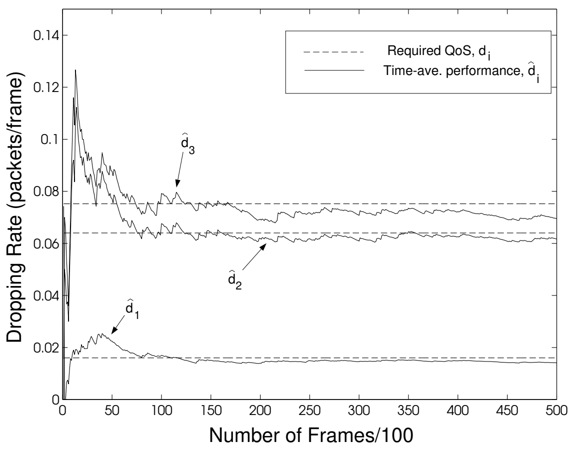

Figure 2. Time-averaged packet dropping rate for each source under the derived policy.

source 1 is experiencing improved performance. Again, requiring (8) and (9) to be satisfied, it can be established that the minimum amount of resources in order for $\mathbf{d}$ to be achievable are $T_{min} = 6$ slots per TDMA frame.

VI. SIMULATION RESULTS

A simulation was performed using the policies, $\boldsymbol{\alpha}$, derived for the example in the previous section. The sources as described in section V were simulated and generated the service demand. Resources (slots) were allocated according to the prescribed policy and the number of dropped packets per frame from each source was recorded. The simulation time was 50000 frames or 2000 seconds (i.e., 40 ms frame). Throughout the simulation in each frame m, for each source i, the time-averaged mean dropping rate was calculated as,

$$\widehat{d}_i(k) = \frac{1}{k} \sum_{m=1}^{k} d_i^f(m), \tag{15}$$

The resulting time-averaged mean packet dropping rate for each source under the policy was found and is displayed in table IV. Throughout the progression of the simulation the time-averaged mean packet dropping rates were recorded and are displayed in fig. 2. Under the policy the objective of mean packet dropping rate d_i for each source i is met, see table IV and Fig. 2.

VII. CONCLUSION

In this work easy to implement transmission scheduling policies were developed for VBR traffic in a shared un-buffered environment. The policies were derived from a precise description of the region of achievable QoS vectors. It was shown that there exists a class of scheduling policies - mixed Ordered HoL policies - that can deliver any achievable QoS vector. A simulation was carried out using a policy derived for a numerical example where the sources were diverse in terms of both the required QoS and the traffic characteristics. The results obtained through the simulation verify the effectiveness of the scheduling policies in delivering the target QoS.

Source, i	Theoretical, d_i	Simulation, $\widehat{d}_i(k)$
1	0.016	0.016
2	0.064	0.064
3	0.075	0.074

Table IV
Comparison of time-averaged and theoretical steady-state mean packet dropping rate for each source under the policy.

REFERENCES

[1] J.M Capone and I. Stavrakakis, "Achievable QoS and Scheduling Policies for Integrated Services Wireless Networks," *CDSP Tech. Report*, TR-CDSP-96-38, 1996; also to appear in, *Performance Evaluation Journal.*

[2] E.G. Coffman and I. Mitrani, "A Characterization of Waiting Time Performance Realizable by Single-Server Queues," *Operations Research*, Vol. 28, No. 3, Part II, pp. 810-821, May-June 1980.

[3] D.J. Goodman, R.A. Valenzeula, K.T. Gayliard, and B. Ramamurthi, "Packet Reservation Multiple Access for Local Wireless Communication," *IEEE Trans. Communication*, Vol. 37, No. 8, pp. 885-890, August 1989.

[4] F. Li and L.F. Merakos, "Voice Data Integration in Digital TDMA Cellular System, " *Proc. International Zurich Seminar on Mobile Communications,* Zurich, Switzerland, March 1994.

[5] D. Bantz, F. Bauchot, "Wireless LAN Design Alternatives," *IEEE Network Magazine*, Vol. 8, No. 2, pp. 43-53, March-April 1994.

[6] D. Raychaudhuri and N. Wilson, "ATM-Based Transport Architecture for Multiservices Wireless Personal Communication Networks," *IEEE J. Selected Areas Communication*, Vol. 12, No. 8, pp. 1401-1414, October 1994.

[7] M. Karol, Z. Liu and K. Eng, "An Efficient Demand-Assignment Multiple Access Protocol for Wireless Packet (ATM) Networks," *Wireless Networks*, Vol. 1, No. 4, pp. 267-279, December 1995.

[8] P.T. Brady, "A model for On-Off Speech Patterns in Two-way Conversation," *Bell Systems Technical Journal,* Vol. 48, pp. 885-890, September 1969.

[9] G. Chen, I. Stavrakakis, "ATM Traffic Management with Diversified Loss and Delay Requirements," *Proc. of IEEE INFOCOM*, March 25-28, 1996, San Francisco, CA.