Cell Multiplexing in ATM Networks

Zvi Rosberg

Abstract—**In this paper, we formulate, analyze, and compare among several connection multiplexing algorithms for a multiplexer residing in the *equivalent terminal* of the asynchronous transfer mode (ATM) layer at the user premise. The primary goal is to find an algorithm that efficiently combats the cell delay variation (CDV) introduced by the multiplexer. Several performance criteria are examined, one of which is unique to ATM networks. This one is the proportion of arriving cells that do not conform to the traffic contract of the connection. The conformance is being tracked by a *generic cell rate algorithm* (GCRA) recommended (but not mandatory) by CCITT. Other criteria are the classical buffer requirements and the cell delay. *Service fairness* among connections is also evaluated with respect to each performance criterion. The impact of the following five multiplexing policies on the performance criteria are evaluated for constant bit rate (CBR) traffic sources. The policies are first in, first out (FIFO), round robin (RR), least time to reach bound (LTRB), most behind expected arrival (MBEA), and golden ratio (GR). Extensive numerical examples reveal that there is no single policy that is best across all criteria. With respect to cell conformance, most behind expected arrival (MBEA) emerges as the preferred one. FIFO is best with respect to cell delay, except for high utilizations where RR dominates. The LTRB is marginally better than all other policies with respect to buffer requirements.**

I. Introduction

ASYNCHRONOUS transfer mode (ATM) is a transport architecture for broadband integrated services digital networks being standardized by CCITT and ATM Forum. A complete description can be found in the standards specifications [3], [11], and in the books [18] and [22]. In ATM, user messages are packetized into fixed-length quanta called *cells* that are transmitted during fixed durations, called *slots*. Each cell consists of 48 octets for user data and five octets for the header. The duration of a time slot depends on the physical link capacity. During each time slot on every ATM link, at most, one cell is transmitted. Cell streams arriving at a switching node from different incoming links are merged by the ATM switch into one or more outgoing links using a statistical multiplexer.

As ATM links have very high capacities (over 155 Mbps), it is common practice to multiplex several user traffic sources over a single ATM link. This is done at the user premise multiplexer (see Fig. 1) from which the merged traffic enters the ATM user network interface (UNI) switch through an access link. Merging various types of traffic sources enables flexibility in allocating traffic resources but may also create a take-over by a heavy traffic source. Arbitration between traffic

Manuscript received April 1994; revised February 10, 1995 and July 18, 1995; approved by IEEE/ACM Transactions on Networking Editor D. Towsley.

Z. Rosberg is with Haifa Research Laboratory of Science and Technology, IBM Israel MATAM, Israel (e-mail: rosberg@haifasc3.vnet.ibm.com).

Publisher Item Identifier S 1063-6692(96)01368-4.

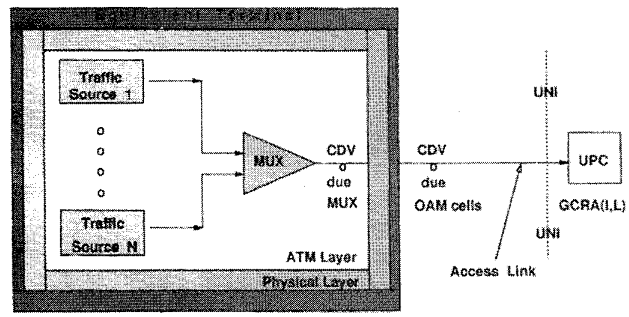


Fig. 1. Equivalent terminal and ATM access link.

sources in ATM is facilitated by a contract on the *quality of service* (QoS) and on the traffic parameters, as well as by a *generic cell rate algorithm* (GCRA). The latter is defined in CCITT recommendation I.371 [10] and in the ATM Forum [Section III, 3]. For every arriving cell, GCRA determines whether or not the cell is conforming with the traffic contract of the connection. Thus, it is used as an operational definition of the conformance of the actual traffic to the negotiated traffic parameters. (The GCRA details are depicted in Fig. 2 and specified in Section II-A.) The network provider is not obligated by the CCITT recommendation to use this algorithm for *usage parameter control* (UPC) purposes at the UNI (see Fig. 1). Rather, it may use any UPC whose performance is consistent with the QoS objectives for the connection.

Upon *connection* setup, the network and the user negotiate a contract for a QoS class and for traffic parameters. A QoS [1] is specified by several performance and integrity measures (e.g., cell transfer delay, cell delay variation, and cell loss ratio). Connection traffic parameters are defined by the *peak cell rate* (PCR) and the *sustainable cell rate* (SCR). The PCR is mandatory and applies to connections supporting *constant bit rate* (CBR) and *variable bit rate* (VBR) services. The SCR is optional and can be used to bound from above the actual average cell rate of a connection supporting VBR services. For VBR services, the burst tolerance is also used to characterize the traffic source. The cell rate control algorithm (GCRA) has the provision to discard cells that do not conform with the agreed upon traffic parameters (see, e.g., [3] and [27]). CBR traffic sources (that generate periodic cell arrivals) are expected to be a major traffic component during the first years of deployment [9]. In this study, we consider only such traffic sources, hence, PCR is the only relevant traffic parameter.

The ATM layer functions at the user premise (within the *equivalent terminal*), Fig. 1, alter the traffic characteristics of each traffic source, by introducing *cell delay variation*

(CDV). That is, when cells from two or more traffic sources (connections) are multiplexed, cells from any given connection may be delayed when cells of another connection are being inserted into the multiplexer outgoing link. The multiplexing scheme has a crucial impact on the CDV of each connection and therefore on how many cells could be discarded by the PCR-based GCRA at the network provider premise. Note that cells could still be discarded, irrespective of the size of the *CDV tolerance value* (defined below) used by the GCRA. This follows from the fact that the CDV increases with the number of multiplexed connections and with the rate variability among the traffic sources. Thus unbounded CDV may result. A proper multiplexing scheme can serve as a traffic shaper to reduce the magnitude of the CDV and attain other desired characteristics for the cell stream.

There is also an optional traffic control function at the UNI, traffic shaping [3, Section III], whose algorithm is not specified. This function may be used, irrespective of the multiplexing scheme (see Remark 1). When used for conformance with negotiated parameters of a traffic descriptor, however, a natural choice is one that mimics the conformance definition. Note that the traffic shaper function and the multiplexer both alter the characteristics of the cell stream. Hence, either one has the potential to attain the cell spacing objective. Indeed, the motivation for the most behind expected arrival (MBEA) multiplexer scheme, defined below, is the same as for using a traffic shaper function that mimics the conformance definition.

The major theme of this paper is to study how various multiplexing schemes impact the CDV of each connection (in the absence of a traffic shaper control function) and to propose a scheme resulting in connection cell streams with desired traffic characteristics. Note that there is another source for CDV beyond the multiplexer control, after the multiplexer and before the entrance at the UNI. This is the cell insertion by the physical and ATM layers operation and maintenance (OAM) within the user *equivalent terminal*, Fig. 1 (F1 to F5 OAM cells). These cells are generated by two protocol layers, and with respect to the data cells generated by potentially many connections, we believe that their impact is negligible compared to the multiplexing impact.

A major traffic characteristic is the magnitude of the CDV of each connection. This is highly correlated to the number of nonconforming cells the user may wish to reduce. There are other performance criteria that may be considered: cell delay and buffer requirements at the multiplexer, and a *fair* multiplexing among the connections. The relative importance of each criterion is application dependent, and therefore, we study and present results for all of them. Note that the buffer requirements (and the cell delay) may either be bounded or unbounded depending on the multiplexing algorithm. This has been shown in [4], [12], and [15]. Another aspect for selecting a multiplexing scheme is its feasibility to support the speed of the UNI access link without wasting outgoing slots. The multiplexing schemes in this paper are all feasible with this respect.

The impact of CDV on cell rate control algorithms (especially the leaky bucket) has been studied for CBR connections in several papers. In [6]–[8], [16], [17], and [23], the multiplexing algorithm has been bypassed, and the incoming perturbed cell stream has been assumed from the outset as part of the model definition. In [7], [16], and [23], the incoming CDV process is governed from the outset by an *a priori* Markov process. In [6] and [17], it is governed by a renewal process. In [8], a discrete-time model is assumed. In [5], the incoming multiplexed connections have been modeled by two traffic streams. One is the primary connection stream, and the other is an aggregated background batch stream. The jitter process (CDV) incurrd by the primary connection resulting from the background stream by a first in, first out (FIFO) multiplexer has been formulated and exactly analyzed. In this paper, as opposed to the aggregated stream in [5], we model each incoming connection separately. We also study several multiplexing algorithms and show that there is no single algorithm that is best with respect to all the performance criteria. With respect to cell discarding, however, the special designed algorithm (MBEA) emerges as the preferred one. The paper is structured as follows: In Section II, we formulate a model, describe the GCRA details, and define the performance criteria and the multiplexing algorithms. In Section III, we present the solution methodology, and in Section IV, we present the comparison data and draw our conclusions.

II. Performance Model Formulation

We start by formulating a queueing model for the user traffic sources and the multiplexer. Then we describe the GCRA details and turn to the multiplexing algorithms. We conclude this section with the performance criteria definitions.

A. Queueing Model

The model objective is to provide an evaluation tool for the number of nonconforming cells, the buffer requirement and the cell delay at the multiplexer, under various multiplexing schemes. Thus, the traffic sources (see Fig. 1) are modeled by multiple independent CBR arrival streams. The multiplexer with the ATM access link is presented by a single server. To see that this indeed serves our needs, note that from our discussion in Section I, we do not lose much by ignoring the CDV due to PL-OAM cell insertion. Thus, cell arrivals at the UNI/UPC are a fixed time shift of their departures from the multiplexer. Therefore, a cell is nonconforming at the multiplexer departure stream if and only if it is nonconforming at the UNI/UPC.

To have a sufficiently rich state information for a multiplexing policy, each arriving cell is identified and recorded by its traffic source and its last bit arrival time. The data structure at the multiplexer will greatly affect the algorithm execution path and will depend on the algorithm and on the multiplexer platform. In practice, the algorithm and its associated data structure can be realized either by micro-code or by a specially designed chip. Based on a preliminary assessment, we may assume that the server capacity is determined by the speed of the access link.

Let $1, 2, \cdots, N$ be the connection labels corresponding to the N traffic sources in Fig. 1, and $S_i > 0$ ($i = 1, \cdots, N$), be

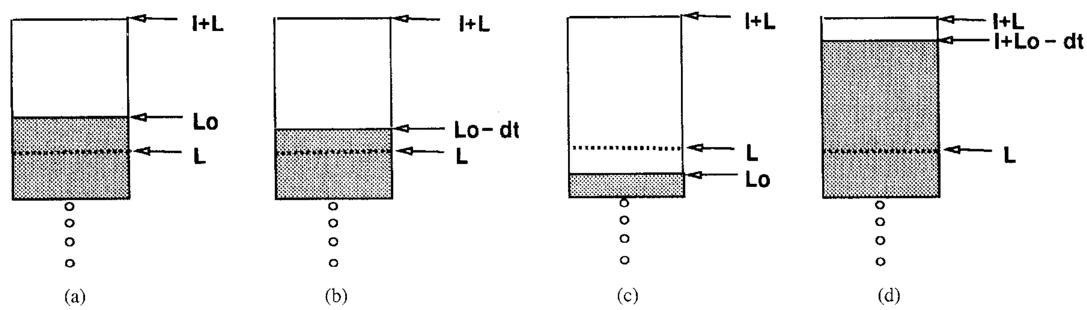


Fig. 2. GCRA(I,L) operation; (a) Bucket level at cell arrival at time t (nonconforming); (b) bucket level at time $(t + dt)$; (c) bucket level at cell arrival at time $t1$ (conforming); (d) bucket level at time $(t1 + dt)$.

the peak rate (in b/s) by which bits enter from connection i. The model of arrivals is the standard gradual input or noninstantaneous input model, often used to study switches and communication networks. It has appeared extensively in the literature in the analysis of these systems [2], [13], [14], [21], and [25]. The noninstantaneous input model describes more accurately than instantaneous input models, real systems for which the inter-arrival times between messages are limited by the speeds of the input channels. Here, ATM cells of fixed size are gradually loaded into the multiplexer buffer as they arrive from each connection, as opposed to arriving instantaneously. Bits arriving from connection i are stored in an infinite buffer. (The maximal buffer requirement, however, is computed below.) The arriving bits form *ATM cells* of a fixed length (53 octets) denoted by M. Each ATM cell is labeled by its connection and by its last bit arrival time. These labels can be used as state information for the multiplexing algorithms defined below.

A single server, representing the multiplexer and the ATM access link whose service rate is S (b/s), serves the cells from the buffer according to a given multiplexing policy defined below. A cell is released from the buffer only after its transmission is complete. We assume that the aggregate rate of the input connections is, at most, the service rate. That is, $\sum_{i=1}^{N} S_i \leq S$. The server is restricted to serve only *complete* cells. Thus, if a buffer contains only part of a cell, that cell cannot be served until the complete cell is present in the buffer. In addition, cells are served without interruption so that cell fragments cannot be served. Finally, the server is *work-conserving*, i.e., it is not idle if there is a complete cell in the multiplexer buffer. Since cells are of a fixed size and only complete cells are served, the epochs at which the server decides what cell to handle next are the beginnings of outgoing cell (*out-cell*) time slots when at least one complete cell resides in the buffer.

Having CBR arrival streams, cells arrive at their buffer periodically. In the case of a single connection, the cell transmission times are also periodic, generating an evenly spaced cell output process without CDV. In case of multiple connections, cell transmission times of each connection are no longer periodic (except for extremely rare cases) and CDV occurs. Clearly, the CDV process greatly depends on the cell multiplexing algorithm. (In this context, the terms multiplexing, scheduling, and spacing algorithms are interchangeable.)

Before turning to the multiplexing algorithms, it will be useful to specify the GCRA details. The GCRA (see [3, Section 3], [10], or [22, Section 7.3]) is specified by two equivalent definitions, namely *virtual scheduling algorithm* and *continuous-state leaky bucket*. For any realization of cell arrivals, both algorithms determine the same cells to be conforming. Here, we give the continuous leaky bucket definition whose operation is illustrated in Fig. 2. The GCRA depends on two parameters: *Increment—I*, and *Limit—L*, i.e., GCRA (I, L). Each connection is associated with a finite capacity bucket whose real-valued content drains out at a continuous rate of one unit of content per time unit. For each conforming cell, its content is incremented by I. If at a cell arrival, the bucket content is less than or equals the limit L, then the cell is conforming [Fig. 2(c) and (d)]. Otherwise, it is not [Fig. 2(a) and (b)]. The bucket capacity is $L + I$ and is initialized at level zero. Different connections may use different GCRA parameters. Every nonconforming cell could be discarded and lost. In practice, nonconforming cells will be marked and discarded only when the system actually needs additional resources. The *CDV tolerance value* mentioned in Section I is defined by the GCRA parameter L. Note that for $L = 0$, every two consecutive conforming cells arrive at least I time units apart. By taking a positive L, inter-arrival times between consecutive conforming cells may fluctuate around I. The larger I is, the larger the fluctuations are. For CBR arrival streams, the increment parameter I in the GCRA associated with connection i is set to $\frac{M}{S_i}$ $(1 \leq i \leq N)$. The tolerance level parameter L is subject to tuning and optimization.

Given the existence of a GCRA-like policer at the UNI/UPC and the absence of a traffic shaper control function, we would like to select a multiplexing algorithm that will combat the CDV magnitude. As this is highly correlated to the number of nonconforming cells, the latter will serve as a major performance criterion. Other important performance criteria are the buffer requirement and the cell delay at the multiplexer. The buffer requirements introduced by various multiplexing algorithms have been studied in [4], [12], and [15]. It has been shown that it may be either bounded or unbounded, depending on the algorithm. From *Little's* Theorem, the same observation holds for the cell delay. Thus, these two criteria cannot be ignored when selecting a multiplexing algorithm. Next, we introduce five multiplexing algorithms to be analyzed and compared.

B. Multiplexing Algorithms

Based on preliminary assessment, all the algorithms below can be realized to operate within the time constraints imposed by an outgoing link with a speed of at least 155 Mb/s. An algorithm can schedule a cell for the next out-cell slot, based on all cell identities (connection labels and last bit arrival times). Two of the algorithms are the simplest to implement and do not exercise any particular measure to reduce cell discarding. These are the FIFO and the round robin (RR) regimes. From a comparison study perspective, they serve as a control group (borrowing from *hypothesis testing* terminology). Among the other three algorithms, one is periodic [a generalized round robin—golden ratio (GR)] and two are adaptive policies. The periodic GR and the RR policies we consider are work-conserving versions in the following sense: If a connection turn with no complete cells is reached, the turn is moved on to the next connection with a complete cell, according to an *a priori* periodic service order.

FIFO Multiplexer: At the beginning of every out-cell, the server considers for transmission the cell whose last bit arrival time is the earliest. If more than one cell qualifies, any arbitration may be used.

The FIFO policy induces the following Markov process: $\mathcal{X}(FIFO) = \{((N_i(t), B_i(t), \mathbf{T}_i(t)), 1 \leq i \leq N) : t \geq 0\}$. Here, $N_i(t)$ is the number of complete cells from connection i at time t. $B_i(t)$ is the number of bits in the incomplete cell from connection i (equals zero, if no incomplete cells). $\mathbf{T}_i(t)$ is a vector of size $(N_i(t)+1)$ whose first $N_i(t)$ elements are the last bit arrival times of the complete cells from connection i at time t and the last one is nil, indicating an incomplete cell.

The following RR policy is a work-conserving with a pre-specified renewal state version of the classical RR.

RR Multiplexer: At the beginning of every out-cell, given that the last connection served is $j(j < N)$, the server chooses to serve the connection from among those with complete cells that is first according to the order $(j+1, j+2, \cdots, N, 1, 2, \cdots, j)$. Given that the last connection served is N, or no connection is served yet, or the system returns to a pre-specified renewal state, the server chooses to serve the connection from among those with complete cells which is first according to the order $(1, 2, \cdots, N)$. If the selected connection contains more than one complete cell, FIFO arbitration is used.

The RR policy induces the following Markov process: $\mathcal{X}(RR) = \{((N_i(t), B_i(t), \mathbf{T}_i(t), J(t)), 1 \leq i \leq N) : t \geq 0\}$, where $N_i(t), B_i(t), \mathbf{T}_i(t)$, are as above, and $J(t)$ is the last connection served before time t.

The next policy is an adaptive one and will be referred to as the *least time to reach bound* (LTRB) algorithm. In [4], it has been shown that the LTRB policy is optimal with respect to buffer size requirement. As buffer requirement and cell delay are tightly coupled, it seems an attractive candidate. Moreover, from [26, ch. 5.4], low average cell delay is indicative of low CDV.

The LTRB policy serves the connection (among those with a complete cell), with the least time to reach a total of $2 \cdot M$ b in the buffer (see Fig. 3). To formally define the LTRB policy,

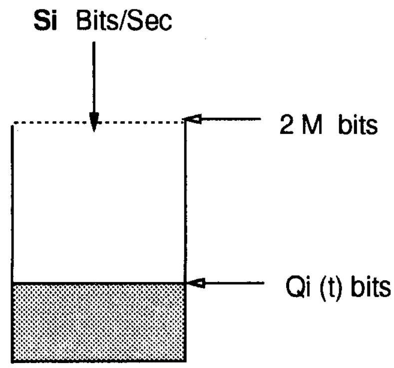


Fig. 3. LTRB algorithm variables.

let $Q_i(t)$ $(i = 1, \cdots, N)$, $t \geq 0$, be the total number of queued bits from connection i at time t (Fig. 3). Define $\theta_i(t)$ as

$$\theta_i(t) = \frac{2 \cdot M - Q_i(t)}{S_i}. \tag{1}$$

Note that if $Q_i(t) \leq 2 \cdot M$, then $\theta_i(t)$ is simply the time it takes for connection i to fill in $2 \cdot M$ b, assuming no bits are removed. Hence, the algorithm is as follows.

LTRB Multiplexer: At the beginning of every out-cell, the server chooses to serve a connection with the minimal $\theta_i(t)$ among those connections with a complete cell. If the selected connection contains more than one complete cell, FIFO arbitration is used.

In [4], it has been shown that under LTRB, the number of bits in the buffer from each connection never exceeds $2 \cdot M$ b (independently of the number of incoming links and their relative speeds). With this respect, LTRB requires the minimum buffer size. (This result has been shown to hold true also for variable bit rate arrival streams and variable packet sizes where M is taken as the maximum packet size.) The LTRB policy induces the following Markov process: $\mathcal{X}(LTRB) = \{((N_i(t), B_i(t), \mathbf{T}_i(t)), 1 \leq i \leq N) : t \geq 0\}$, where $N_i(t), B_i(t), \mathbf{T}_i(t)$, are as above.

Most Behind Expected Arrival (MBEA) Multiplexer: This algorithm is a naturally greedy one to combat GCRA cell discarding. To specify it, imagine two leaky buckets (Fig. 4). One is the GCRA leaky bucket and the other is a virtual leaky bucket whose content levels correspond to the nonpositive interval $(-\infty, 0]$. The virtual leaky bucket starts to drain out at a continuous rate of one unit of content per time unit at the moment when the GCRA leaky bucket becomes empty. For each conforming cell, its content is refilled to the top. The virtual leaky bucket starts full at the initialization. Each connection has its own pair of leaky buckets labeled by $1, 2, \cdots, N$.

For every pair of leaky buckets, i, define the following variables that measure the bucket levels. Let $L_i(t)$ and $R_i(t)$ be the content levels at time t of the GCRA leaky bucket i and of the virtual leaky bucket i, respectively. Note that $R_i(t)$ assumes only nonpositive values and is reset to zero for each conforming cell. Thus, R_i is negative when L_i is zero and zero when L_i is positive. Fig. 4(a) and (b) depicts these variables for the case where the GCRA bucket is not empty,

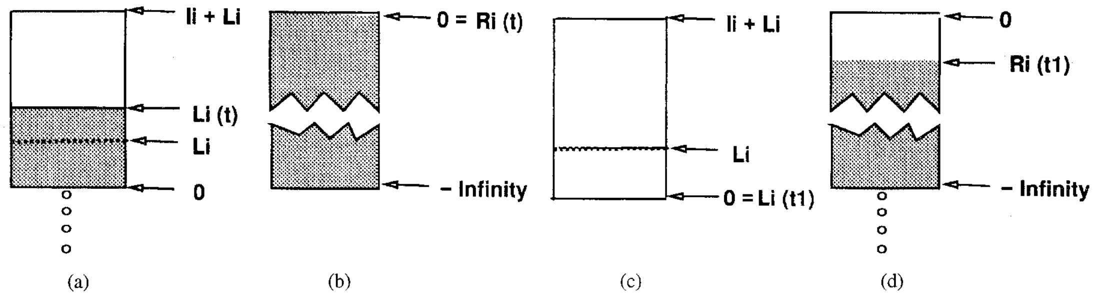


Fig. 4. MBEA algorithm variables; (a) GCRA bucket level at time t; (b) virtual bucket level at time t; (c) GCRA bucket level at time $t1$; (d) virtual bucket level at time $t1$.

and Fig. 4(c) and (d), when it is. To define the MBEA policy, set

$$\eta_i(t) = L_i(t) - R_i(t). \quad (2)$$

The index $\eta_i(t)$ measures the time difference from t to the closest instance when the GCRA bucket will, or has, become empty. Now, the policy is as follows.

MBEA Multiplexer: At the beginning of every out-cell, the server chooses to serve a connection with the minimal $\eta_i(t)$ among those connections with a complete cell. If the selected connection contains more than one complete cell, FIFO arbitration is used.

The MBEA policy induces the following Markov process: $\mathcal{X}(\text{MBEA}) = \{((N_i(t), B_i(t), \mathbf{T}_i(t)), 1 \leq i \leq N) : t \geq 0\}$, where $N_i(t), B_i(t), \mathbf{T}_i(t)$, are as above. Note that given T_i, L_i, and the last arrival time, the values $L_i(t)$, $R_i(t)$ can be computed and, therefore, are not required for the state of the Markov process.

GR Multiplexer: This is a periodic policy that generalizes the RR policy for heterogeneous arrival streams. For such streams, it is more sensible to allocate different amount of slots to different connection types. The GR policy is a general device to achieve that, in a way that any two consecutive out-cell slots allocated to each connection i is most uniformly spaced. That is, it attempts to reduce CDV. This device was first proposed in [20] for a multiple-access channel and is interesting to study in our context.

Like the RR policy, it is also a TDM policy. Unlike RR, it uses a periodic order sequence where the number of appearances of each connection is proportional to its cell arrival rate. In the case where all arrival rates are equal, the GR sequence reduces to an RR sequence. To define the golden ratio periodic order we need the notation $x_i = S_i / \sum_{j=1}^{N} S_j$. The ratio x_i is the relative arrival rate of cells from connection i, which is also the fraction of out-cells allocated to connection i in the GR sequence. First, we select a sequence period K and integers K_i $(i = 1, 2, \cdots, N)$ such that $\frac{K_i}{K}$ is a sufficiently good approximation to x_i $(1 \leq i \leq N)$ and $\sum_i^N K_i = K$. (Note that since the rational numbers are dense in the real numbers, by increasing the period K, we refine the grid of the rational numbers used to approximate x_i. Thus, the above approximation can be improved as we please.) The reader is referred to [20] for the special properties induced on the sequence when K is taken as a Fibonacci number. The golden ratio $\phi^{-1} = (\sqrt{5} - 1)/2 \approx 0.6180339887$ plays a major role in this algorithm.

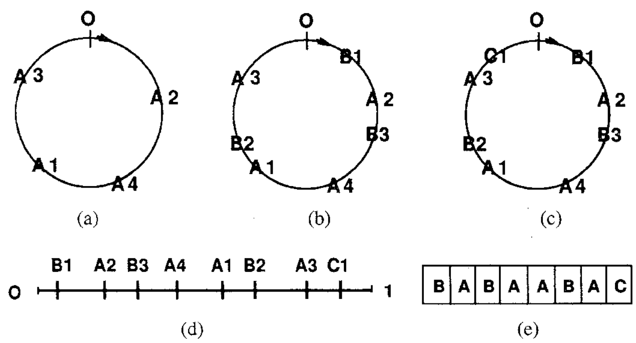


Fig. 5. Golden ratio algorithm: (a) connection 1 placements; (b) connection 1, 2 placements; (c) connection 1, 2, 3 placements; (d) connection 1, 2, 3 placements after opening and straightening the circle; (e) golden ratio periodic loop for connections 1, 2, 3.

Before giving a formal algorithmic definition, we first illustrate it by a pictorial example given in Fig. 5. Take a circle whose contour is one, and mark an origin point by zero. For a given set of connections $\{1, 2, \cdots, N\}$, period K and allocated slots per period $\{K_1, K_2, \cdots, K_N\}$, do as follows. (In Fig. 5, $N = 3, K = 8, K_1 = 4, K_2 = 3$, and $K_3 = 1$.) Starting from the origin zero, and traveling clockwise, place K consecutive points on the circle so that each point is placed ϕ^{-1} units after the previous one. The first K_1 placed points are associated with connection one [see points $A1, A2, A3, A4$ in that order, in Fig. 5(a)]. The next K_2 placed points are associated with connection two [see points $B1, B2, B3$ in that order, in Fig. 5(b)]. Note that point $B1$ is placed ϕ^{-1} units after $A4$. Finally, the K_3 placed points are associated with connection three [see point $C1$ in Fig. 5(c)]. Next, tear the circle at point zero and straighten it to get the interval $[0, 1)$ in Fig. 5(d). The resulting K points induce a periodic order with period K, among the N connections, depicted in Fig. 5(e).

A formal definition of the GR periodic order is as follows: Let $\lfloor x \rfloor$ be the largest integer smaller than or equal to x, $frac(y) = y - \lfloor y \rfloor$, $a_j = frac(j\phi^{-1})$, and $A_K = \{a_j | j = 0, 1, \cdots, K-1\}$ (here the tth smallest point in A_K is associated with the tth out-cell of each period). Now, define the GR sequence $(\sigma_1, \sigma_2, \cdots, \sigma_K)$ by replacing each set of the K_i elements $(i = 1, \cdots, N)$ $\{a_j | \sum_{m=1}^{i-1} K_m \leq j < \sum_{m=1}^{i} K_m\} \subset A_N$, with the connection identity i.

GR Multiplexer: The GR policy is defined exactly as the RR policy, but with the periodic order $(\sigma_1, \sigma_2, \cdots, \sigma_K)$ rather than with $(1, 2, \cdots, N)$.

In [20] and [19], it has been shown that the GR policy results in close to optimal delay and throughput compared to other deterministic periodic policies. Apart from these two properties, its almost uniform inter-allocation times make it an attractive candidate to reduce the number of discarded cells. The GR policy induces the following Markov process: $\mathcal{X}(GR) = \{((N_i(t), B_i(t), \mathbf{T}_i(t), J(t)), 1 \le i \le N) : t \ge 0\}$, where $N_i(t), B_i(t), \mathbf{T}_i(t)$, are as above, and $J(t)$ is the GR sequence element used to determine the last connection served before time t.

Remark 1: Note that none of the algorithms above delay cells that do not conform to the CGRA rule. One may use a special traffic shaping function to delay such cells in order to completely prevent nonconforming cells. Such voluntary delay trades conforming cells with additional cell delay and buffer requirement. We believe that voluntarily delaying cells will not gain popularity in practice for two reasons. One is that cells will probably be marked and be discarded only if additional resources are required. Hence, some probabilistic gain is traded with some definite cost. The second reason is mentioned in Section I, namely, the network provider is not obligated by the CCITT recommendation to use this GCRA algorithm for UPC purposes at the UNI. Rather, it may use any UPC whose performance is consistent with the QoS objectives for the connection. Therefore, the user may be reluctant to inflict voluntary cell delay when its impact on cell discarding is not clear.

C. Performance Criteria

The user is obliged to its traffic contract verified by the GCRA operation and identification of nonconforming cells. A conforming stream of cells is the main measure by which the network can plan its resource allocation to guarantee the committed QoS. Thus, in the absence of a traffic shaping function, increasing the proportions of conforming cells may be regarded as a major objective of a multiplexing algorithm. Denote by $XmtProp_\pi(i)$, the long-run proportion of conforming cells from connection i, transmitted through the outgoing link under policy π during the entire system operation. Let $XmtProp_\pi = (XmtProp_\pi(1), XmtProp_\pi(2), \cdots, XmtProp_\pi(N))$, and $AXmtProp_\pi = \frac{1}{N}\sum_{i=1}^{N} XmtProp_\pi(i)$ be the arithmetic average. A nonconforming cell occupies an out-cell slot, but we regard its slot as a potential waste. Thus, we define the *outgoing link utilization* under policy π, $Util_\pi$, as the long-run proportion of all conforming cells transmitted through the outgoing link during the entire system operation. By definition, $Util_\pi = \frac{1}{S}\sum_{i=1}^{N} S_i \cdot XmtProp_\pi(i)$.

As discussed above, the buffer size requirement and cell delay at the multiplexer are two other important criteria. Denote by $MaxBuf_\pi(i)$, the maximum number of bits belonging to cells from connection i, that has been observed in a given instant at the multiplexer buffer during the entire system operation. Let $MaxBuf_\pi = (MaxBuf_\pi(1), MaxBuf_\pi(2), \cdots, MaxBuf_\pi(N))$, and $AMaxBuf_\pi = \frac{1}{N}\sum_{i=1}^{N} MaxBuf_\pi(i)$ be the arithmetic average.

Denote by $Delay_\pi(i)$, the long-run average delay at the multiplexer, of a cell from connection i under policy π. Here, cell delay is the time from its arrival instant until it is considered for transmission. Let $Delay_\pi = (Delay_\pi(1), Delay_\pi(2), \cdots, Delay_\pi(N))$, and $ADelay_\pi = \frac{1}{N}\sum_{i=1}^{N} Delay_\pi(i)$ be the arithmetic average.

Another important consideration is *service fairness.* In general, fairness is a measure of variability around some average performance measure. In our system, it may measure, e.g., the variability of the $(XmtProp_\pi(1), XmtProp_\pi(2), \cdots, XmtProp_\pi(N))$ around some average value of the transmission proportions. In a system where the agreed QoS determines the performance, it is not desirable that connections with low arrival rates be discriminated against. Thus, a *service fairness* measure should be taken as the variance around the simple arithmetic average rather than around the rate-based weighted average.

We define a *service fairness* measure for each performance criterion by the standard deviation from its corresponding arithmetic average. It is reasonable to expect that a *fair* algorithm will not introduce any performance discrimination among the connections, that is, have low standard deviations. Given a performance measure tuple $(X(1), X(2), \cdots, X(N))$ and its arithmetic average AX, its standard deviation SX is defined by $SX = \{\frac{1}{N}\sum_{i=1}^{N}(X(i) - AX)^2\}^{1/2}$. In our model, we compute the standard deviation of the buffer requirement, cell delay and transmission proportions, denoted by $SMaxBuf_\pi$, $SDelay_\pi$, and $SXmitProp_\pi$, respectively.

III. Solution Methodology

The solution methodology is based on the *mean ergodic theorem* (also called the *renewal theorem* (see, e.g., [24, ch. 7]). Since we consider CBR traffic sources only, the Markov process under any multiplexing policy has a deterministic evolution. Thus, a general method to compute any long-run average statistic of the process is to find a renewal state (if such exists) and to write a simulation program that calculates the corresponding averages in the process realization between two consecutive renewal states. Note, given that one starts with the renewal state, the simulation stopping time is the moment that the process reaches the renewal state again. Given that the time between two consecutive renewals is not too long, this method is most powerful as all renewal functions (e.g., $XmtProp_\pi$, $Util_\pi$, $MaxBuf_\pi$, $Delay_\pi$, $SMaxBuf_\pi$, $SDelay_\pi$, and $SXmitProp_\pi$) can be computed in a single run.

In our model, we assume the case where the slot beginnings of the incoming connections and the outgoing link are synchronized at time zero, and all cells are being served and sent out on the outgoing link (including the nonconforming ones). In the case where a CBR traffic source arrives not at its slot beginning, there is an additional *synchronization cell delay.* Assuming a uniformly distributed arrival instant, this delay equals half a slot on the average.

We will explicitly derive the renewal state and show how to precompute the number of out-cells that need to be simulated in order to precisely compute our performance measures. This procedure has merit since, in general, complex systems require very long simulation time and provide measures only within confidence intervals. It turns out that when using our procedure, the simulation time is independent of the policy π, and it is extremely fast.

Assume that $\frac{S}{S_i}(1 \leq i \leq N)$ are rational numbers. (In practical terms, this is not a real restriction.) Thus, there is a minimum integer k (denoted by K), for which k times the out-cell transmission time is an integral multiple of all cell-in transmission times. That is, $KS_i = K_iS$, for some integers $K_1, K_2, \cdots, K_N$. Observe that the following hold true for the time period consisting of the first K out-cell slots under any work-conserving policy π:

i) As all slot beginnings are synchronized at time zero, they are becoming synchronized again at the end of the K out-cell slot.
ii) During the first K out-cell slots, exactly K_i complete cells from each connection i $(1 \leq i \leq N)$ arrive at the server (multiplexer). Also, no further bits arrive during that period.
iii) The last arriving cell from each connection completes its arrival at the end of out-cell slot K. Thus, at that moment, the server buffer contains the last arriving cell from each connection.
iv) Since all cells are of a fixed size and π is work-conserving, the total number of served cells (including the nonconforming ones) is invariant of π. (This can simply be verified by ignoring the cell identities and observing that any realization of the total number of cells in the system is independent of π.)

Denote by S_0, the state with one complete cell from each connection with zero arrival time. In the following theorem, we show that under any policy π, if we start with state S_0, after K out-cell slots, the Markov process returns to state S_0 (up to a fixed shift in the arrival times). That is, S_0 is a renewal state.

Theorem 1: Under any work-conserving policy π, the state S_0 is a renewal state (up to a fixed time shift) that repeats every K out-cell slots.

Proof: Let π be a work-conserving policy and $R(\pi, S_0)$ be the set of cell identities that are left in the buffer at the end of out-cell slot K. We will show that $R(\pi, S_0) = S_0$, by considering the following two cases.

Case 1: All K out-cell slots are busy. As the out-cell slot duration is $\frac{M}{S}$ units, the total number of arriving bits during the first K out-cell slots is $M \cdot K \cdot \frac{1}{S}\sum_{i=1}^{N} S_i$. The total number of departing bits is $M \cdot K$. Since $\frac{1}{S}\sum_{i=1}^{N} S_i \leq 1$, we have

$$M \cdot K \geq M \cdot K \cdot \frac{1}{S}\sum_{i=1}^{N} S_i.$$

That is, the total number of arriving bits does not exceed the total number of departing bits. Since at time zero, there are $N \cdot M$ bits in the buffer, it follows that the total number of bits in $R(\pi, S_0)$ is, at most, $N \cdot M$. From observation iii), $R(\pi, S_0)$ contains the last arriving cell from each connection (which sums up to $N \cdot M$ b). Thus, $R(\pi, S_0) = S_0$ (up to a fixed shift in the arrival times).

Case 2: There are idle out-cell slots among the first K. Denote by k^* the last idle slot. Since π is work-conserving, there are no complete cells in the buffer at the beginning of out-cell slot k^*. That is, the total number of bits at that instant is *strictly less* than $N \cdot M$. (Otherwise, there would have been at least one complete cell.)

The total number of arriving bits during slots $k^*, \cdots, K$, is $M \cdot (K - k^* + 1) \cdot \frac{1}{S}\sum_{i=1}^{N} S_i$. The total number of departing bits is $M \cdot (K - k^* + 1)$. Similarly to the argument in Case 1 above, the number of added bits does not exceed M. Thus, including the bits at the beginning of slot k^*, the total number of bits in $R(\pi, S_0)$ is *strictly less* than $(N + 1) \cdot M$. Now, observe that only complete cells are served, and from observation ii), there are no incomplete cells at the end of slot K. Therefore, $R(\pi, S_0)$ consists of $N \cdot M$ b at most. As in Case 1, the same result holds. □

Since the process evolution is deterministic, the realizations between any two consecutive renewal states replicate each other (up to a fixed shift in the time arrivals). Thus, our performance measures can be precisely computed based on the following corollary.

Corollary 1: Under any policy π above, the performance measures $MaxBuf_\pi$, $AMaxBuf_\pi$, $SMaxBuf_\pi$, $Delay_\pi$, $ADelay_\pi$, $SDelay_\pi$, $XmtProp_\pi$, $AXmtProp_\pi$, and $SXmtProp_\pi$ can be computed as follows: Set the initial state to S_0 and simulate the system for K outgoing slots during which all the required statistics above are being recorded, then stop.

Note that we evaluate the system only for the case where all slot beginnings are synchronized at time zero. If the incoming slot beginnings are shifted around the out-cell slot beginning by fixed rational numbers, then the same method as above can be used to precisely compute the performance measures. In the case where a CBR traffic source arrives not at its slot beginning, there is an additional *synchronization cell delay*. Assuming a uniformly distributed arrival instant, this delay equals half a slot on the average. Our numerical results relate only to the synchronized case.

IV. Algorithm Comparison

In our numerical study we considered five sets of examples (one set for each of the following number of connections $N = 3, 5, 10, 20, 40$), from which we draw our conclusions. Here, due to space constraints, we present only a subset of them. Each example set covers a range of maximum outgoing link utilizations ρ (when all cells are conforming) and several CDV tolerance parameters L. From these sets, we also learn about the trend of each performance measure as a function of L, N, and ρ. The subset presented here consists only of $N = 5, 20, 40$, $I = 0, 2$, and $\rho = 0.66, 0.9$. The numerical results in the tables below present the arithmetic averages of the connection performance measures and their fairness measures. The individual connection measures are given only for Examples 1 and 2, but the relation among them are representative.

TABLE I
PERFORMANCE MEASURES FOR $N = 5$, $L = 0$, AND $\rho = 0.658$

	LTRB	MBEA	GR	FIFO	RR
Util	.517	.508	.508	.475	.508
AXmtProp	.727	.762	.748	.753	.755
SXmtProp	.166	.051	.071	.126	.077
AMaxBuf	1.283	1.388	1.528	1.428	1.457
SMaxBuf	.052	.108	.179	.322	.299
Adelay	.982	.825	.855	.533	.761
Sdelay	.904	.448	.572	.389	.348

TABLE II
PERFORMANCE MEASURES FOR $N = 5$, $L = 2$, AND $\rho = 0.658$

	LTRB	MBEA	GR	FIFO	RR
Util	.65	.633	.608	.617	.625
AXmtProp	.975	.953	.895	.957	.963
SXmtProp	.05	.044	.091	.065	.045
AMaxBuf	1.283	1.395	1.528	1.428	1.457
SMaxBuf	.052	.148	.179	.322	.299
Adelay	.982	.770	.855	.533	.761
Sdelay	.904	.489	.572	.389	.348

TABLE III
PERFORMANCE MEASURES FOR $N = 20$, $L = 0$, AND $\rho = 0.658$

	LTRB	MBEA	GR	FIFO	RR
Util	.517	.485	.425	.475	.508
AXmtProp	.727	.743	.633	.753	.755
SXmtProp	.166	.041	.057	.126	.077
AMaxBuf	1.251	1.461	1.433	1.385	1.450
SMaxBuf	.100	.135	.121	.312	.259
Adelay	5.427	4.569	4.880	3.633	4.547
Sdelay	3.784	1.744	2.322	1.917	1.787

TABLE IV
PERFORMANCE MEASURES FOR $N = 20$, $L = 2$, AND $\rho = 0.658$

	LTRB	MBEA	GR	FIFO	RR
Util	.517	.533	.467	.475	.508
AXmtProp	.727	.804	.698	.753	.755
SXmtProp	.166	.033	.077	.126	.077
AMaxBuf	1.251	1.454	1.433	1.385	1.450
SMaxBuf	.100	.150	.121	.312	.259
Adelay	5.427	4.699	4.880	3.633	4.547
Sdelay	3.784	2.005	2.322	1.917	1.787

Another important measure in a communication network is the tail distribution of the end-to-end cell delay. It is beyond the scope of this study, however, as we only consider the first hop of the entire route. Its tail distribution bares almost no information on the end-to-end tail distribution. We start by presenting the numerical examples. All time units are given in number of out-cells and buffer units in number cells. Observe that in Examples 1 and 2 there are instances where the cell delay (not including transmission time) is zero. This occurs when all cells belonging to a particular connection arrive to an empty buffer, and at an out-cell beginning. When the system is not saturated, and the link speeds are rational numbers, this is indeed possible.

Example 1: In this example we take $N = 5$, $L = 0$, $S = 1.5$ Mb/s, maximum outgoing link utilization 0.658, and the connections rates are $(S_1, \cdots, S_N) = (.1, .15, .1875, .25, 0.3)$Mb/s. The average values and fairness measures are presented in Table I. The connection individual measures are given below:

	$MaxBuf$	$Delay$	$XmtProp$
LTRB :	(1.333, 1.3, 1.250, 1.333, 1.2)	(2.375, 1.667, .667, .20, 0.000)	(0.500, .667, .667, .80, 1.000)
MBEA :	(1.333, 1.3, 1.375, 1.333, 1.6)	(1.500, 1.083, .867, .30, 0.375)	(0.750, .667, .800, .80, 0.792)
GR :	(1.333, 1.5, 1.375, 1.833, 1.6)	(1.500, 1.583, .600, .30, 0.292)	(0.625, .750, .733, .80, 0.833)
FIFO :	(1.067, 1.2, 1.375, 1.500, 2.0)	(0.000, 0.333, .467, .70, 1.667)	(1.000, .667, .733, .70, 0.667)
RR :	(1.133, 1.4, 1.250, 1.500, 2.0)	(1.000, 1.333, .467, .55, 0.458)	(0.625, .883, .733, .75, 0.833)

Example 2: All parameters are as in Example 1, except for CDV tolerance $L = 2$. The average values and fairness measures are presented in Table II. The connection individual measures are given below:

	$MaxBuf$	$Delay$	$XmtProp$
LTRB :	(1.333, 1.3, 1.250, 1.333, 1.2)	(2.375, 1.667, .667, .200, 0.000)	(.875, 1.000, 1.000, 1.000, 1.000)
MBEA :	(1.333, 1.5, 1.375, 1.667, 1.6)	(1.111, 1.500, .467, .105, 0.667)	(.889, 0.917, 1.000, 1.000, 0.958)
GR :	(1.333, 1.5, 1.375, 1.833, 1.6)	(1.500, 1.583, .600, .300, 0.292)	(.750, 0.833 0.933, 1.000, 0.958)
FIFO :	(1.067, 1.2, 1.375, 1.500, 2.0)	(0.000, 0.333, .467, .700, 1.667)	(1.000, 1.000, 1.000, 0.950, 0.833)
RR :	(1.133, 1.4, 1.250, 1.500, 2.0)	(1.000, 1.333, .467, .550, 0.458)	(1.000, 1.000, 1.000, 0.900, 0.917)

Example 3: In this example, we take $N = 20$, $L = 0$, $S = 1.5$ Mb/s, maximum outgoing link utilization is 0.658, and the connections rates are $(S_1, \cdots, S_N) =$ (.025, .025, .025, .025, .0375, .0375, .0375, .0375, .046875, .046875, .046875, .046875, .0625, .0625, .0625, .0625, .075, .075, .075, .075) Mb/s. The average values and fairness measures are presented in Table III.

Example 4: All parameters are as in Example 3, except for CDV tolerance $L = 2$. The average values and fairness measures are presented in Table IV.

Example 5: In this example we take $N = 40$, $L = 0$, $S = 1.5$ Mb/s, maximum outgoing link utilization is 0.658, and the connections rates are $(S_1, \cdots, S_N) =$ (.0125, .0125, .0125, .0125, .0125, .0125, .0125, .0125, .01875, .01875, .01875, .01875, .01875, .01875, .01875, .01875, .0234375, .0234375, .0234375, .0234375, .0234375, .0234375, .0234375, .0234375, .03125, .03125, .03125, .03125, .03125, .03125, .03125, .03125, .0375, .0375, .0375, .0375, .0375, .0375,

TABLE V
PERFORMANCE MEASURES FOR $N = 40$, $L = 0$, AND $\rho = 0.658$

	LTRB	MBEA	GR	FIFO	RR
Util	.517	.486	.411	.475	.508
AXmtProp	.727	.747	.625	.753	.755
SXmtProp	.166	.045	.046	.126	.077
AMaxBuf	1.256	1.458	1.449	1.389	1.455
SMaxBuf	.098	.135	.130	.313	.258
Adelay	11.38	9.631	10.213	7.767	9.593
Sdelay	7.725	3.614	4.225	3.866	3.609

TABLE VI
PERFORMANCE MEASURES FOR $N = 40$, $L = 2$, AND $\rho = 0.658$

	LTRB	MBEA	GR	FIFO	RR
Util	.517	.515	.441	.475	.508
AXmtProp	.727	.782	.661	.753	.755
SXmtProp	.166	.030	.053	.126	.077
AMaxBuf	1.256	1.468	1.449	1.389	1.455
SMaxBuf	.098	.149	.130	.313	.258
Adelay	11.38	9.688	10.213	7.767	9.593
Sdelay	7.725	3.941	4.225	3.866	3.609

TABLE VII
PERFORMANCE MEASURES FOR $N = 5$, $L = 0$, AND $\rho = 0.9$

	LTRB	MBEA	GR	FIFO	RR
Util	.525	.65	.6	.525	.55
AXmtProp	.687	.842	.710	.665	.686
SXmtProp	.168	.160	.217	.175	.173
AMaxBuf	1.455	1.885	1.69	1.705	1.705
SMaxBuf	.105	.602	.369	.733	.733
Adelay	3.463	1.308	3.23	.824	.766
Sdelay	4.099	.444	4.027	.454	.512

TABLE VIII
PERFORMANCE MEASURES FOR $N = 5$, $L = 2$, AND $\rho = 0.9$

	LTRB	MBEA	GR	FIFO	RR
Util	.875	.825	.875	.85	.825
AXmtProp	.933	.887	.933	.967	.955
SXmtProp	.133	.126	.133	.041	.056
AMaxBuf	1.455	1.683	1.69	1.705	1.705
SMaxBuf	.105	.301	.369	.733	.733
Adelay	3.463	.993	3.23	.824	.766
Sdelay	4.099	.607	4.027	.454	.512

TABLE IX
PERFORMANCE MEASURES FOR $N = 20$, $L = 0$, AND $\rho = 0.9$

	LTRB	MBEA	GR	FIFO	RR
Util	.725	.662	.612	.525	.55
AXmtProp	.787	.794	.729	.666	.686
SXmtProp	.176	.131	.145	.176	.173
AMaxBuf	1.364	1.761	1.592	1.622	1.622
SMaxBuf	.159	.436	.283	.652	.652
Adelay	15.553	6.458	12.051	4.627	4.393
Sdelay	16.659	2.685	13.520	2.088	2.272

TABLE X
PERFORMANCE MEASURES FOR $N = 20$, $L = 2$, AND $\rho = 0.9$

	LTRB	MBEA	GR	FIFO	RR
Util	.75	.781	.675	.625	.625
AXmtProp	.807	.915	.773	.737	.744
SXmtProp	.164	.081	.138	.14	.128
AMaxBuf	1.364	1.769	1.592	1.622	1.622
SMaxBuf	.159	.467	.283	.652	.652
Adelay	15.553	6.578	12.051	4.627	4.393
Sdelay	16.659	4.028	13.520	2.088	2.272

.0375, .0375) Mb/s. The average values and fairness measures are presented in Table V.

Example 6: All parameters are as in Example 5, except for CDV tolerance $L = 2$. The average values and fairness measures are presented in Table VI.

Example 7: In this example we take $N = 5$, $L = 0$, $S = 10$ Mb/s, maximum outgoing link utilization is 0.9, and the connections rates are $(S_1, \cdots, S_N) = (.25, .75, 1.5, 2.5, 4)$ Mb/s. The average values and fairness measures are presented in Table VII.

Example 8: All parameters are as in Example 7, except for CDV tolerance $L = 2$. The average values and fairness measures are presented in Table VIII.

Example 9: In this example we take $N = 20$, $L = 0$, $S = 10$ Mb/s, maximum outgoing link utilization is 0.9, and the connections rates are $(S_1, \cdots, S_N)$ = (.0625, .0625, .0625, .0625, .1875, .1875, .1875, .1875, .375, .375, .375, .375, .625, .625, .625, .625, 1, 1, 1, 1) Mb/s. The average values and fairness measures are presented in Table IX.

Example 10: All parameters are as in Example 9, except for CDV tolerance $L = 2$. The average values and fairness measures are presented in Table X.

Example 11: In this example we take $N = 40$, $L = 0$, $S = 10$ Mb/s, maximum outgoing link utilization is 0.9, and the connections rates are $(S_1, \cdots, S_N)$ = (.03125, .03125, .03125, .03125, .03125, .03125, .03125, .03125, .09375, .09375, .09375, .09375, .09375, .09375, .09375, .09375, .1875, .1875, .1875, .1875, .1875, .1875, .1875, .1875, .3125, .3125, .3125, .3125, .3125, .3125, .3125, .3125, .5, .5, .5, .5, .5, .5, .5, .5) Mb/s. The average values and fairness measures are presented in Table XI.

Example 12: All parameters are as in Example 9 except, for CDV tolerance $L = 2$. The average values and fairness measures are presented in Table XII.

We draw two types of conclusions: which algorithm performs best with respect to each of the performance criteria and how the performance measures vary as a function of N, ρ and L.

A. Algorithm Preferences

The first observation is that there is *no single policy which is best with respect to all criteria.*

1) Buffer Requirements: With respect to buffer requirements, LTRB is best (as expected from [4]). It also achieves

TABLE XI
PERFORMANCE MEASURES FOR $N = 40$, $L = 0$, AND $\rho = 0.9$

	LTRB	MBEA	GR	FIFO	RR
Util	.725	.662	.593	.525	.55
AXmtProp	.787	.826	.727	.666	.686
SXmtProp	.176	.139	.155	.176	.173
AMaxBuf	1.353	1.791	1.626	1.626	1.626
SMaxBuf	.135	.470	.258	.652	.652
Adelay	31.407	13.516	24.693	9.753	9.287
Sdelay	33.003	6.621	25.367	4.206	4.572

TABLE XII
PERFORMANCE MEASURES FOR $N = 40$, $L = 2$, AND $\rho = 0.9$

	LTRB	MBEA	GR	FIFO	RR
Util	.725	.722	.634	.525	.55
AXmtProp	.787	.853	.752	.666	.686
SXmtProp	.176	.111	.148	.176	.173
AMaxBuf	1.353	1.794	1.626	1.626	1.626
SMaxBuf	.135	.502	.258	.652	.652
Adelay	31.407	14.022	24.693	9.753	9.287
Sdelay	33.003	8.154	25.367	4.206	4.572

the best *service fairness* (standard deviation) value for this criterion. Nevertheless, the differences from other policies are small. Therefore, when other criteria are also involved, buffer requirements are given a relatively low weight. The similarity in the buffer requirements follows from the fact that cells are of fixed size, and all policies are work-conserving.

2) Conforming Cells: The conforming cells are measured by their total long-run proportion (given in the tables under the variable $Util$) and by their arithmetic average over the connection long-run proportions (given by the variable $AXmtProp$). With respect to the former criterion, there is a tight competition between LTRB and MBEA, with some advantage to LTRB. With respect to the latter criterion, MBEA is best, except for rare cases where FIFO or LTRB are better. With respect to *service fairness* of conforming cells, MBEA is best most of the time, except for rare cases. To conclude, it appears that MBEA deals best with the *nonconforming cells* phenomenon.

3) Cell Delay: With respect to *long-run average cell delay*, all policies are the same as they are *work-conserving*. However, with respect to the *arithmetic average over the cell delays of the individual connections*, FIFO turned out to be the best, except when ρ is high, where RR becomes best. At least to the author's surprise, both of them appear as significantly better than all other policies. With respect to *service fairness* in cell delay, RR is best most of the time. FIFO is quite close, and in some cases MBEA is best. To conclude, it appears that FIFO and RR deal best with cell delay.

B. Performance Trends

From the graphs we produced (not presented here), the following trends of the conforming cells and arithmetic average cell delay were found.

The number of *conforming cells* exhibits similar trends, either when measured by *outgoing link utilization*, or by *arithmetic average*:

- Both increase with CDV tolerance L with a decreasing rate, and the increasing rate is amplified when reducing the number of connections. Thus, increasing L is beneficial until some threshold point. This threshold increases when the number of connections decreases.
- Both decrease with respect to N with a decreasing rate (except for GR, for that they may increase in the range of small N's). Thus, for a large number of connections, the impact of CDV is much stronger, and a proper policy selection is vital.
- Both increase with respect to ρ with an increasing rate (except for LTRB and MBEA for which they decrease in the range of small ρ's for $\tau = 0$). (It should be noted that with respect to the arithmetic average, in several instances under MBEA, the conforming-cells tend to decrease as a function of ρ or L.)

The *arithmetic average cell delay* exhibits the following trends:

- As a function of the CDV tolerance L, it is constant except under MBEA, where it fluctuates. Thus, when using MBEA, a careful CDV tolerance must be selected in order to trade-off between utilization and cell delay.
- As a function of N, it is almost a linearly increasing function.
- As a function of ρ, except for MBEA, delay increases with an increasing rate for large N's, but with a decreasing rate for small N's. For MBEA, delay increases with ρ, with a decreasing rate for all N's. For FIFO and RR, the change in the rates is almost zero.

V. CONCLUSIONS

From our comparison study we can first conclude that there is no single best multiplexing scheme with respect to all criteria. Nevertheless, if we wish to select one scheme, then the MBEA emerges as the preferred one, if cell delay is not crucial; otherwise, FIFO or RR are preferred.

From the performance trends analysis, we can also conclude that increasing CDV tolerance L is useful in reducing the number of nonconforming cells, only up to some threshold value. That threshold value depends on the number of connections.

REFERENCES

[1] M. E. Anagnostou, M. E. Theologou, K. M. Vlakos, D. Tournis, and E. N. Protonotarios, "Quality of service requirements in ATM-based ISDN's," *Comput. Commun.*, vol. 14, pp. 197–204, May 1991.

[2] D. Anick, D. Mitra, and M. M. Sondhi, "Stochastic theory of a data-handling system with multiple sources," *Bell Syst Tech. J.*, vol. 61, no. 8, pp. 1871–1894, Oct. 1982.

[3] ATM Forum Technical Committee, "Draft recommendation section 3: ATM layer specification," 1993.

[4] A. Birman, H. R. Gail, S. L. Hantler, Z. Rosberg, and M. Sidi, "An optimal service policy for buffer systems," *J. ACM*, vol. 42, no. 3, pp. 641–657, 1995.

[5] C. C. Bisdikian, W. Matragi, and K. Sohraby, "A study of the jitter in atm multiplexers," in *Proc. 5th Int. Conf. Data Commun. Syst. Perform.*, Raleigh, NC, Oct. 1993.

[6] S. Blaabjerg, *Estimating the Effect of Cell Delay Variation by an Application of the Heavy Limit Theorem*, COST 242, TD(92) 017, 1992.

[7] ——, "Cell delay variation in a fifo queue: a diffusion approach," in *Proc. 5th Int. Conf. Data Commun. Syst. Perform.*, Raleigh, NC, Oct. 1993.

[8] P. Castelli, A. Tonietti, and A. Forcina, "Dimensioning criteria for policing functions in atm networks," in *Proc. INFOCOM*, 1992.

[9] CCITT, Study XVIII, CCITT Recommendation I.211, BISDN Service Aspects, (rep. R34), Geneva, Switzerland, June 1990.

[10] CCITT, Draft Recommendation I.371, "Traffic control and congestion control in B-ISDN," Geneva, June 1992.

[11] CCITT, Study XVIII, CCITT SG XVIII Draft Recommendation I.35B, 1992.

[12] I. Cidon, I. Gopal, G. Grover, and M. Sidi, "Real time packet switching: a performance analysis," *IEEE J. Select. Areas Commun.*, vol. SAC-6, pp. 1576–1586, Dec. 1988.

[13] J. W. Cohen, "Superimposed renewal processes and storage with gradual input," *Stochastic Processes Applications*, vol. 2, pp. 31–58, 1974.

[14] R. L. Cruz, "A calculus for network delay, Part I: Network elements in isolation," *IEEE Trans. Inform. Theory*, vol. 37, no. 1, pp. 114–131, Jan. 1991.

[15] H. R. Gail, G. Grover, R. Guérin, S. L. Hantler, Z. Rosberg, and M. Sidi, "Buffer size requirements under longest queue first," *Perform. Eval.*, Sept. 1993.

[16] F. Guillemin, and J. W. Roberts, "Jitter and bandwidth enforcement," in *Proc. IEEE GLOBCOM*, Dec. 1991.

[17] F. Guillemin, P. Boyer, A. Dupuis, and L. Romoeuf, "Peak rate enforcement in ATM networks," in *Proc. INFOCOM*, 1992.

[18] R. Handel, and M. N. Huber, *Integrated Broadband Networks: An Introduction to ATM-Based Networks*. Wokingham, England: Addison-Wesley, 1991.

[19] M. Hofri and Z. Rosberg, "Packet delay under the golden ratio weighted TDM policy in a multiple-access channel," *IEEE Trans. Inform. Theory*, vol. IT-33, pp. 341–349, 1987.

[20] A. Itai, and Z. Rosberg, "A golden ratio control policy for a multiple-access channel," *IEEE Autom. Contr.*, vol. AC-29, pp. 712–718, 1984.

[21] H. Kaspi and M. Rubinovitch, "The stochastic behavior of a buffer with non-identical input lines," *Stochastic Processes Applications*, vol. 3. pp. 73–88, 1975.

[22] M. de Prycker, *Asynchronous Transfer Mode: Solution for Broadband ISDN*. New York: Ellis Horwood, 1993.

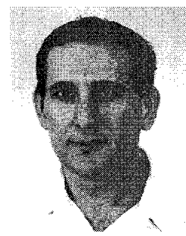
[23] J. W. Roberts, and F. Guillemin, "Jitter in ATM networks and its impact on peak rate enforcement," *Perform. Eval.*, vol. 16, pp. 35–48, Nov. 1992.

[24] S. M. Ross, *Introduction to Probability Models*. New York: Academic, 1980.

[25] M. Rubinovitch, "The output of a buffered data communication system," *Stochastic Processes Applications*, vol. 1, pp. 375–382, 1973.

[26] D. Stoyan, *Comparison Methods for Queues and Other Stochastic Models*. New York: Wiley, 1983.

[27] E. D. Sykas, K. M. Vlakos, and M. J. Hillyard, "Overview of ATM Networks: Functions and Procedures," *Comput. Commun.*, vol. 14, pp. 615–625, Dec. 1991.



Zvi Rosberg received the B.Sc., M.A., and Ph.D. degrees from the Hebrew University of Jerusalem.

He was a Senior System Analyst in the Central Computing Bureau of the Israeli government from 1972 to 1978, where he was one of the chief designers of a new on-line Israeli population registration system. He held a research fellowship at the Center of Operation Research and Econometric (CORE), Catholic University of Louvain, Belgium, from 1978 to 1979. He was a Visiting Professor in the Department of Business Administration at the University of Illinois, Urbana, from 1979 to 1980 and was with the Computer Science Department, Technion, Israel, from 1980 to 1990. Since then, he has been with the Haifa Research Laboratory, Science and Technology, IBM, Israel, holding a position of Program Manager of Communication Networks. He was a Visiting Scientist at the IBM Thomas J. Watson Research Center, Yorktown Heights, NY, the Department of Electrical and Computer Engineering, University of Massachusetts, Amherst, the Department of Electrical Engineering and Computer Science, University of California, Berkeley, and the Radio Communication Systems, Royal Institute of Technology. He has published numerous papers in the areas of communication networks, queueing theory, control of queueing systems, applied probability, analysis of algorithms in communication and computing systems, power control and resource allocation in cellular radio systems, and performance evaluation. He led a project that developed an IBM licensed product for network topology and routing design. His current research interests include resource allocation and power control in cellular radio systems, mobile computing, and high-speed networking.