An Optimal Service Policy for Buffer Systems

ALEXANDER BIRMAN, H. RICHARD GAIL, AND
SIDNEY L. HANTLER

IBM T. J. Watson Research Center, Yorktown Heights, New York

ZVI ROSBERG

IBM Israel, MATAM, Haifa, Israel

AND

MOSHE SIDI

Technion–Israel Institute of Technology, Technion City, Haifa, Israel

Abstract. Consider a switching component in a packet-switching network, where messages from several incoming channels arrive and are routed to appropriate outgoing ports according to a service policy. One requirement in the design of such a system is to determine the buffer storage necessary at the input of each channel and the policy for serving these buffers that will prevent buffer overflow and the corresponding loss of messages. In this paper, a class of buffer service policies, called Least Time to Reach Bound (LTRB), is introduced that guarantees no overflow, and for which the buffer size required at each input channel is independent of the number of channels and their relative speeds. Further, the storage requirement is only twice the maximal length of a message in all cases, and as a consequence the class is shown to be *optimal* in the sense that any nonoverflowing policy requires at least as much storage as LTRB.

Categories and Subject Descriptors: C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*packet networks*; C.4 [**Performance of Systems**]: *performance attributes*; D.4.4 [**Operating Systems**]: Communications Management–*buffering*; D.4.8 [**Operating Systems**]: Performance–*queueing theory*

General Terms: Algorithms, Design, Performance

Additional Key Words and Phrases: Buffer overflow, gradual input, multiplexor, parallel queues, service discipline, switch

---

This work was done while Z. Rosberg and M. Sidi were at the IBM T. J. Watson Research Center.

Authors' present addresses: A. Birman, H. R. Gail, and S. L. Hantler, IBM T. J. Watson Research Center, Yorktown Heights, NY 10598; Z. Rosberg, Haifa Research Laboratory, Science and Technology, IBM Israel, MATAM, Haifa 31905, Israel; M. Sidi, Department of Electrical Engineering, Technion—Israel Institute of Technology, Technion City, Haifa 32000, Israel.

1. *Introduction*

We consider a system consisting of several input channels and a single output channel. Variable length messages, with a maximum length of L bits, arrive over each input channel and are stored in the buffer associated with that channel. The buffers are emptied into the output channel by a single server whose speed is at least as great as the aggregate speed of the input channels. The system is to be work-conserving subject to the constraint that service can only be provided to complete messages. Accordingly, the server:

(a) may not begin serving a buffer until it contains at least one complete message;
(b) cannot be idle if at least one buffer contains a complete message;
(c) serves complete messages without interruption.

Service is provided by the server according to a rule which, given the contents of each buffer at the beginning of a service period, determines the buffer to be served next. Simple examples of such a service policy include Exhaustive Round Robin and First Come First Served.

The system described above can be found in various devices, for example, packet switches in communication networks. Here the server is the switch itself, and the service policy provides the rule that determines which buffer the switch will serve. In such systems, it is important, for economic reasons, to minimize the amount of storage required in each buffer [Krishna 1990]. Yet the buffers must be large enough to limit overflow (the loss of messages that arrive to a full buffer). Ideally, the buffers would be designed to eliminate overflow. Our interest is in a nonoverflowing policy that minimizes the size of the largest buffer required under any message arrival pattern. In many applications, it is also desirable that the buffer storage required for each channel be independent of the number of channels and their relative speeds. This will enable the reuse of the input channels when the system is reconfigured to allow higher speed channels or a larger number of channels.

This design problem was studied in Cidon et al. [1988], where the Exhaustive Round Robin (ERR) service policy was analyzed. Under this policy, the buffers are served in cyclic order, and once the service of a buffer starts it continues until all complete messages in that buffer are exhausted. It was shown there that when the speeds of the input channels are equal, a buffer at every input channel of capacity $3.35L$ is sufficient to prevent overflow. However, when the speeds are not equal, the required buffer sizes depend on the relative speeds and grow linearly with the number of input channels. Recently [Sasaki 1989], the upper bound for equal speed channels and ERR was improved to $3.307L$, and a lower bound of $3.051L$ was also provided. In the same paper, Gated Round Robin (GRR) was investigated for equal speeds, and an upper bound of $3L$ was found. Other policies that have been studied include First Come First Served (FCFS) and Longest Queue First (LQF). In Birman et al. [1989], the FCFS policy was shown to require buffers of capacity $2L$ to prevent overflow in the case of equal speed channels, but in the unequal speed case the buffer sizes exhibit the same linear behavior in the number of channels as ERR. A similar phenomenon has been established for LQF in Gail et al. [1993]. The required buffer storage to guarantee no overflow is again $2L$ for the equal speed case, but it depends on the number and relative speeds of the input channels in the unequal speed case and increases logarithmically in the number of channels. Finally, we mention a recent paper [Greenberg and Madras 1992] in which a study of a discipline called fair queueing was presented. Translating that work into the context of the above system model, it was shown that an upper bound of $2L$ guarantees no overflow in the equal speed case under the fair queueing service policy.

In this paper, we introduce and analyze a class of buffer service policies, called Least Time to Reach Bound (LTRB), that satisfies (a)–(c), guarantees no overflow, and for which the buffer storage required at each input channel is $2L$ in all cases. The storage required under LTRB is not only independent of the number of channels and their relative speeds, but it is also optimal in the sense that any such nonoverflowing policy requires at least as much storage. The LTRB policy operates as follows: When the service of a message is completed, the next message to be served is chosen from a buffer that would overflow first if all input channels were to remain continuously busy and none were served.

The model of arrivals is the standard gradual input or noninstantaneous input model often used to study switches and communication networks. This model has appeared extensively in the literature in the analysis of these systems (see, for example, Anick et al. [1982], Cohen [1974], Cruz [1991], Kaspi and Rubinovitch [1975], and Rubinovitch [1973]) and has also been used in the analysis of dams [Cohen 1969; Gaver and Miller 1962]. The noninstantaneous input model describes more accurately than instantaneous input models real systems for which the interarrival times between messages are limited by the speeds of the input channels. An input channel may be either "on" (bits are arriving) or "off" (no bits are arriving). Messages are loaded gradually into the buffer as they arrive, as opposed to arriving instantaneously. We emphasize that the results proved here hold for every instance of systems satisfying the assumptions, regardless of the distribution of arrivals, service times and on/off statistics of the input channels.

The remainder of the paper is organized as follows. In Section 2, the noninstantaneous input model is described and the LTRB class of service policies is introduced. In Section 3, a proposition involving the properties of a certain set of difference equations is proved. In Section 4, this proposition is used to establish that LTRB does not overflow with a buffer storage of only twice the maximum message length. In Section 5, a brief discussion of the results is presented.

2. *LTRB Policies*

Consider a system with N input channels, each with an input buffer. Let $S_i > 0$, $i = 1, \dots, N$, be the rate (in bits/s) of channel i. As discussed in the introduction, we assume the gradual input or noninstantaneous input model of arrivals. Bits arriving through channel i are stored in its corresponding input buffer, and if that buffer is full the bits are lost. An input channel may be either in an on state, during which S_i bits/s are arriving, or in an off state, during which no bits are arriving at that channel. The arriving bits form *messages*. Each message consists of not more than L bits. We do not impose any statistical assumptions about the on and off periods of the input channels or on the message lengths.

A single server, whose service rate is S (bits/s), serves the messages residing at the input buffers. Without loss of generality we normalize the server speed to $S = 1$. The rate S_i then represents the speed of channel i relative to the service rate. The aggregate rate of the input channels is at most the service rate, that is, $\Sigma_{i=1}^{N} S_i \leq 1$. The server is restricted to serve only *complete* messages. Thus, if a buffer contains only part of a message, that message cannot be served until the complete message is present in the buffer. In addition, messages are served without interruption, so that message fragments cannot be served. Finally, the server is *work-conserving*, that is, it is not idle if there is a complete message at some buffer.

Since only complete messages are served, the epochs at which the server decides upon the next message to handle are times when at least one buffer contains a complete message. Since messages are served without interruption and the server is work-conserving, these decision epochs are either instants of service completion or, if the server is idle, instants when at least one complete message is formed at some buffer.

We assume that when the system starts to operate no buffer contains more than L bits, so that the total number of bits in the system initially is not greater than NL bits. Since the service rate is at least as large as the aggregate arrival rate when the server is busy, and since no buffer can contain L or more bits when the server is idle, the total number of bits in the system cannot exceed NL bits at any time.

To define the class of LTRB policies we need the following notation. Let $Q_i(t)$, $i = 1, \ldots, N$, $t \geq 0$ be the number of buffered bits at channel i at time t. As noted above, the $Q_i(t)$ satisfy $\Sigma_{i=1}^{N} Q_i(t) \leq NL$ for all t. Define the quantity $\theta_i(t)$ as

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

We note that if $Q_i(t) \leq 2L$, then $\theta_i(t)$ is simply the time it will take for the queue size at channel i to reach a buffer size of $2L$, assuming a continuous stream of bits arrive over the channel. Policies from the LTRB class operate by attempting to serve any buffer with the minimal "time to reach bound." However, since only complete messages can be served, a slight variation to this scheme is necessary.

The Least Time to Reach Bound Class of Policies

At a decision epoch τ, the server chooses to serve a buffer i, from among those buffers with a complete message, for which $\theta_i(\tau) \leq \theta_j(\tau)$ for all buffers j with at least L bits.

One member (policy A) from this class of policies is the following:

At a decision epoch τ, the server chooses to serve a buffer with the minimal $\theta_i(\tau)$ among those buffers with a complete message.

Another variant (policy B) from the LTRB class is the following:

At a decision epoch τ, the server chooses to serve a buffer with the minimal $\theta_i(\tau)$ among those buffers with at least L bits. If no buffer has L or more bits, then any buffer with a complete message is chosen.

We now proceed to give a mathematical description of the LTRB class of policies. At each decision epoch τ, the channel chosen to be served is required to have a complete message to transmit. Note that $Q_i(\tau) \geq L$ is equivalent to $\theta_i(\tau) \leq L/S_i$. Therefore, any channel satisfying $\theta_i(\tau) \leq L/S_i$ must have a complete message, and it is thus eligible to be chosen for service. Define

$$\mathscr{L}(\tau) = \{i\text{: channel } i \text{ has at least } L \text{ bits at } \tau\} \tag{2}$$

and

$$\mathscr{C}(\tau) = \{i\text{: channel } i \text{ has a complete message at } \tau\}. \tag{3}$$

Since L is the maximum message length, we have $\mathscr{L}(\tau) \subseteq \mathscr{C}(\tau)$. Note that $\mathscr{C}(\tau) \neq \varnothing$, although we may have $\mathscr{L}(\tau) = \varnothing$. If channel $F(\tau)$ is chosen to be served, then $F(\tau) \in \mathscr{C}(\tau)$. An LTRB policy is one for which $\theta_{F(\tau)}(\tau) \leq \theta_i(\tau)$ for $i \in \mathscr{L}(\tau)$.

Let $\mathscr{F}(\tau)$ be the set of channels from which the service policy chooses according to the minimal $\theta_i(\tau)$, that is,

$$\mathscr{F}(\tau) = \left\{i\text{: } \theta_i(\tau) \geq \theta_{F(\tau)}(\tau)\right\}. \tag{4}$$

It should now be clear that the relation

$$\mathscr{L}(\tau) \subseteq \mathscr{F}(\tau) \subseteq \mathscr{C}(\tau), \tag{5}$$

for all decision epochs τ, yields the class of LTRB policies. Setting $\mathscr{F}(\tau) = \mathscr{C}(\tau)$ we obtain the first example (policy A) that was discussed above. The choice of channel to serve is the one with minimal time to reach bound over all channels with a complete message. Setting $\mathscr{F}(\tau) = \mathscr{L}(\tau)$ for those decision epochs τ for which $\mathscr{L}(\tau) \neq \varnothing$ gives the other extreme (policy B). The choice of channel to serve is the one among those with at least L bits with minimal time to reach bound. If all channels have less than L bits, then any channel with a complete message may be chosen.

Our main goal is to prove that any member of the LTRB class of policies requires buffer storage of only $2L$ bits at each input channel to prevent overflow, for any number of channels N and any set of channel speeds $\{S_i\}$. That is, we will prove the following:

THEOREM 2.1. *Consider a system of N channels with channel speeds S_i, $i = 1, \ldots, N$, and server speed $S \geq \sum_{i=1}^{N} S_i$ operating under any member of the class of LTRB policies. Then*

$$Q_i(t) \leq 2L \qquad \textit{for} \quad i = 1, \ldots, N, \qquad t \geq 0. \tag{6}$$

In Cidon et al. [1988], it was shown that a buffer size of $2L$ is a lower bound for any policy satisfying properties (a)–(c) of Section 1. Thus, we obtain the following result immediately from Theorem 2.1.

THEOREM 2.2. *The LTRB class is optimal in terms of buffer storage for policies that are work-conserving and for which only complete messages are served.*

3. *Analysis*

3.1. PRELIMINARIES. In order to determine whether or not overflow occurs under a given policy, only the buffer sizes at decision epochs need to be examined. The reason is that overflow occurs if and only if it occurs at a decision epoch (necessarily during a busy period). This can be seen as follows. Consider an arbitrary busy period, and let $\tau_n, n = 0, 1, \ldots,$ represent the beginning of the nth service. Note that these instants of time constitute the decision epochs within that busy period. We claim that it is enough to show that overflow does not occur at these decision epochs. For suppose that overflow occurs at some time t that is not a decision epoch, and let channel i be the one that overflows. We have $\tau_n < t < \tau_{n+1}$ for some n, since channel i will still have at least one complete message at the end of the present service. If channel i has already overflowed at the decision epoch τ_n, then the claim holds. If channel i has not overflowed at τ_n, then clearly it is not being served at t, for otherwise its queue size would not increase between τ_n and t. Therefore, at the next decision epoch τ_{n+1} (the end of the current service), overflow will still occur at channel i. In fact, $Q_i(t) \le Q_i(\tau_n)$ if channel i is being served at time t, while $Q_i(t) \le Q_i(\tau_{n+1})$ if channel i is not being served at time t. Thus, overflow occurs during a busy period if and only if it occurs at a decision epoch within that busy period, and we need only concentrate on these points in time.

Note that the queue size at each channel at the start of the busy period is at most L. That is, the initial value θ_i satisfies $\theta_i \ge L/S_i$, $i = 1, \ldots, N$. Let $\sigma(n)$ be the length of the nth message to be served during the busy period. Then, it also represents the nth service time, since the server speed is normalized to 1. Let $\delta_i(n), i = 1, \ldots, N$, be the time during this service when channel i actually receives input (thus, $0 \le \delta_i(n) \le \sigma(n) \le L$). Let $F(n)$ be the channel that is chosen to be served at the nth decision epoch. Then, when service is completed the value of θ_i for any channel $i \ne F(n)$ that has not been served will decrease by $\delta_i(n)$, while that for channel $F(n)$ will increase by $(\sigma(n)/S_{F(n)}) - \delta_{F(n)}(n)$. That is, at the completion of service we have $\theta_i(n + 1) = \theta_i(n) - \delta_i(n)$ for $i \ne F(n)$, and $\theta_i(n + 1) = \theta_i(n) - \delta_i(n) + \sigma(n)/S_i$ for $i = F(n)$. Note that these equations hold for any service policy F under our modeling assumptions. We will study difference equations of this type under a general framework that includes equations obtained when F belongs to the class of LTRB policies.

3.2. GENERALIZED FRAMEWORK. The above discussion suggests the study of the following mathematical model in a general setting. We are given $L \ge 0$ and $S = (S_1, \ldots, S_N)$ satisfying $0 < S_i < 1$ and $\sum_{i=1}^{N} S_i \le 1$. A state is an N-vector $\theta = (\theta_1, \ldots, \theta_N)$. We are also given an initial state θ satisfying $\theta_i \ge L/S_i$, $i = 1, \ldots, N$. Our interest is in certain sequences $T = (T(n))$, $n = 0, 1, \ldots,$ which are called *triples* and are defined as follows: $T(n) = (\sigma(n), \delta(n), F(n))$ where $\delta(n) = (\delta_1(n), \ldots, \delta_N(n))$, $0 \le \delta_i(n) \le \sigma(n) \le L$ and $F(n) \in \{1, \ldots, N\} = \mathcal{N}$. A triple T determines a state trajectory $\Theta = (\theta(0), \theta(1), \ldots)$ as follows: $\theta(0) = \theta$ and $\theta(n + 1)$ satisfies

$$\theta_i(n + 1) = \theta_i(n) - \delta_i(n), \qquad \text{for } i \ne F(n),$$

$$\theta_{F(n)}(n + 1) = \theta_{F(n)}(n) - \delta_{F(n)}(n) + \frac{\sigma(n)}{S_{F(n)}}. \tag{7}$$

As mentioned above, the evolution of the states of the channels within a busy period as defined by (1) follows (7) for an arbitrary service policy F. However, there are additional constraints on the equations generated from the behavior of the queueing system that models the switch. As an example, messages of length L cannot be consecutively chosen for service from the same channel at the beginning of a busy period, since sufficient time would not have elapsed for the second message to arrive. As another example, the amount of bits in any buffer cannot become negative (the value of θ_i cannot become larger than $2L$). We will prove a proposition on difference equations of type (7) without taking into account such constraints. All equations that model the behavior of the queueing system described above will be included, but they will constitute a small set of the cases covered by our result. Although the proposition is quite general from a mathematical point of view, the significance of the framework developed above is that it allows certain inductive arguments to be carried through which yield the queue size upper bound of Theorem 2.1. Otherwise, such additional constraints would always have to be taken into account, complicating the arguments.

We next describe the triples that we wish to consider. Let T be a triple with corresponding state trajectory Θ, and define

$$\mathscr{L}_T(n) = \left\{ i\colon \theta_i(n) \le \frac{L}{S_i} \right\}. \tag{8}$$

A triple T is admissible at step n if $\theta_{F(n)}(n) \le \theta_i(n)$ for all $i \in \mathscr{L}_T(n)$. T is *admissible* if it is admissible at $n = 0, 1, \dots$. In terms of the queueing model, note that policies from the LTRB class yield triples that are admissible.

One example of an admissible triple is obtained if the index $F(n)$ corresponds to the minimal value of $\theta_i(n)$ at each step n. A triple T is strict at step n if $\theta_{F(n)}(n) \le \theta_i(n)$ for $i = 1, \dots, N$. T is *strict* if it is strict at $n = 0, 1, \dots$. In the queueing model, if at each decision epoch the channel with the minimal θ_i as defined in (1) has a complete message, choosing these channels to serve yields a strict triple. Furthermore, the choices $F(n)$ constitute a policy that is necessarily a member of the LTRB class.

To prove Theorem 2.1, we will show that the trajectory Θ is always positive, namely, $\theta_i(n) > 0$ for $i = 1, \dots, N, n = 0, 1, \dots$. We carry out the proof in three steps. First, we show that if T is a strict triple, then $\theta_i(n) \ge L$ for $i = 1, \dots, N, n = 0, 1, \dots$. We then handle admissible triples T that have nonstrict steps and show that Θ is positive if $\delta_i(n) = \delta_j(n)$ for $n = 0, 1, \dots,$ $i, j \in \mathscr{N}$. This case corresponds in the queuing model to all channels remaining on for the same percentage of time during each service. Finally, we prove that Θ is always positive for any admissible T. Note that this last result yields Theorem 2.1, since for θ_i as defined in (1), θ_i positive at the decision epochs of a busy period implies that no overflow occurs at channel i given a buffer of size $2L$.

Before continuing, we introduce some notation. For a set $\mathscr{J} \subseteq \{1, \dots, N\} = \mathscr{N}$ and two N-vectors $a = (a_1, \dots, a_N)$, $b = (b_1, \dots, b_N)$, define $(a \cdot b)_{\mathscr{J}} = \sum_{i \in \mathscr{J}} a_i b_i$. We also use the notation $a_{\mathscr{J}} = (a \cdot 1)_{\mathscr{J}}$. When $\mathscr{J} = \{1, \dots, N\}$, we simply write $a \cdot b$.

3.3. Strict Triples. We now prove the following proposition about strict triples.

Proposition 3.3.1. *Let T be a triple with corresponding state trajectory Θ. If T is strict, then*

$$\theta_i(n) \ge L \qquad \textit{for} \quad i = 1, \dots, N, n = 0, 1, \dots . \tag{9}$$

PROOF. We will first prove by induction on n that $(S \cdot \theta(n))_{\mathscr{J}} \geq |\mathscr{J}| S_{\mathscr{J}} L$ for $\mathscr{J} \subseteq \mathscr{N}$, $n = 0, 1, \ldots$.

By hypothesis, $(S \cdot \theta(0))_{\mathscr{J}} \geq |\mathscr{J}| L \geq |\mathscr{J}| S_{\mathscr{J}} L$ for $\mathscr{J} \subseteq \mathscr{N}$, so that the claim holds for $n = 0$. Now assume it holds for $n = m$. Pick $\mathscr{J} \subseteq \mathscr{N}$. If $F(m) \in \mathscr{J}$, then we have

$$(S \cdot \theta(m+1))_{\mathscr{J}} = (S \cdot \theta(m))_{\mathscr{J}} - (S \cdot \delta(m))_{\mathscr{J}} + \sigma(m) \\ \geq (S \cdot \theta(m))_{\mathscr{J}} \geq |\mathscr{J}| S_{\mathscr{J}} L$$

by the induction hypothesis and the fact that $(S \cdot \delta(m))_{\mathscr{J}} \leq \sigma(m)$. If $F(m) \notin \mathscr{J}$, then

$$(S \cdot \theta(m+1))_{\mathscr{J}} = (S \cdot \theta(m))_{\mathscr{J}} - (S \cdot \delta(m))_{\mathscr{J}} \geq (S \cdot \theta(m))_{\mathscr{J}} - S_{\mathscr{J}} L.$$

We claim that $(S \cdot \theta(m))_{\mathscr{J}} \geq (|\mathscr{J}| + 1) S_{\mathscr{J}} L$, which will complete the proof. Suppose not, so that

$$(S \cdot \theta(m))_{\mathscr{J}} < (|\mathscr{J}| + 1) S_{\mathscr{J}} L. \tag{10}$$

Since T is strict (at step m in particular), then $\theta_{F(m)}(m) \leq \theta_i(m)$ for $i = 1, \ldots, N$. Multiplying the ith equation by S_i and summing over $i \in \mathscr{J}$, we obtain the inequality $S_{\mathscr{J}} \theta_{F(m)}(m) \leq (S \cdot \theta(m))_{\mathscr{J}}$. Thus, from (10), it follows that

$$\theta_{F(m)}(m) < (|\mathscr{J}| + 1) L. \tag{11}$$

Defining $\mathscr{K} = \mathscr{J} \cup \{F(m)\}$, we have from (10) and (11)

$$(S \cdot \theta(m))_{\mathscr{K}} < |\mathscr{K}| S_{\mathscr{K}} L,$$

which contradicts the induction hypothesis. This completes the proof of the claim.

Specializing to the case $\mathscr{J} = \{i\}$ shows that $\theta_i(n) \geq L$ for $i = 1, \ldots, N$, $n = 0, 1, \ldots$, which proves the proposition. □

It is interesting to note that, for certain queueing models, the channel with the minimal θ_i as defined in (1) is guaranteed to have a complete message, and thus a strict policy can always be implemented. For example, consider the case when $S_i = 1/N$, $i = 1, \ldots, N$, all channels remain continuously on, and all buffers initially contain L bits. For equal speed channels, the channel with the smallest value of θ_i has the largest queue size, as is apparent from (1). Further, since all channels initially start with L bits, $\sum_{i=1}^{N} S_i = 1$, and there are no off periods, the total number of bits in the system always remains at NL. Thus, the maximal queue size is at least L bits at any time, and so the channel with the smallest θ_i has a complete message.

3.4. UNIFORM TRIPLES. We can now prove the main result for *uniform* triples, that is, admissible triples satisfying the additional assumption that $\delta_i(n) = \delta_j(n)$ for $n = 0, 1, \ldots, i, j \in \mathscr{N}$. For the queueing model, this case corresponds to all channels remaining on for the same percentage of time while messages are being served. One example of this behavior occurs when $\delta_i(n) = \sigma(n)$, $i = 1, \ldots, N$, which is the case of continuous input at all channels during the service time of the nth message. The other extreme, when $\delta_i(n) = 0$ for $i = 1, \ldots, N$, corresponds to the case of no input at any of the channels.

When $\delta_i(n) = \delta_j(n)$, $i, j \in \mathscr{N}$, eq. (7) shows that the positions of $\theta_i(n+1)$ compared to those of $\theta_i(n)$ do not change, except for index $F(n)$. That is, we have $\theta_i(n+1) - \theta_j(n+1) = \theta_i(n) - \theta_j(n)$ for $i \neq F(n)$, $j \neq F(n)$. This property is used extensively in the proof of the following proposition:

PROPOSITION 3.4.1. *Let T be a triple with corresponding state trajectory Θ. Assume that $\delta_i(n) = \delta_j(n)$ for $n = 0, 1, \dots, i, j \in \mathscr{N}$. If T is admissible, then*

$$\theta_i(n) > 0 \qquad \text{for} \quad i = 1, \dots, N, n = 0, 1, \dots . \tag{12}$$

PROOF. Consider any step $n > 0$. If T is not strict at n, then the theorem clearly holds, since the minimal θ_i must satisfy $\theta_i(n) > L/S_i$ by (8). Thus, we may restrict attention to uniform triples that are strict at n. Such triples may be classified by the number $K = K(T)$ of nonstrict steps in $\{0, \dots, n-1\}$. We will prove the theorem by induction on K. The case $K = 0$, which corresponds to triples that are strict at the first n steps, holds by Proposition 3.3.1. Assume the theorem is true for all uniform triples with $K = \kappa$ nonstrict steps, and we will show it is true for $K = \kappa + 1$. Suppose T is a uniform triple with $\kappa + 1$ nonstrict steps in $\{0, \dots, n-1\}$. Let m be the last nonstrict step before n, so that T is strict at steps $m+1, \dots, n$. Define

$$\mathscr{R} = \left\{i \colon \theta_i(m) < \theta_{F(m)}(m)\right\}. \tag{13}$$

Note that $\mathscr{R} \neq \varnothing$ by assumption. Since T is an admissible triple, we have $\mathscr{R} \subseteq \mathscr{N} \setminus \mathscr{L}_T(m)$, that is,

$$\theta_i(m) > \frac{L}{S_i} \qquad \text{for} \quad i \in \mathscr{R}. \tag{14}$$

The index $F(m+1)$ must correspond to the minimal θ_i at step m, since T is not strict at m but is strict at $m+1$, and $\delta_i(m) = \delta_j(m)$ for $i, j \in \mathscr{N}$. Therefore, $F(m+1) \in \mathscr{R}$, and we have

$$\theta_{F(m+1)}(m+1) = \theta_{F(m+1)}(m) - \delta_{F(m+1)}(m) > \frac{L}{S_{F(m+1)}} - L > 0.$$

Thus, the theorem holds if $m + 1 = n$, and so we may assume that $m + 1 \leq n - 1$.

The proof of the induction step splits into two cases, depending on whether or not only members of $\mathscr{R}$ are chosen between steps $m+1$ and n.

Case 1. First, suppose that

$$\{F(m+1), \dots, F(n)\} \subseteq \mathscr{R}. \tag{15}$$

We claim that (15) implies (12). It is sufficient to prove that $\theta_{F(n)}(n) > 0$, because T is strict at n. Set $\mathscr{K} \overset{\text{def}}{=} \{F(m+1), \dots, F(n)\}$. Since $\mathscr{K}$ includes all of the indices chosen during steps $m+1, \dots, n-1$ and $F(m) \notin \mathscr{K}$, we may write

$$\begin{aligned}(S \cdot \theta(n))_{\mathscr{K}} &= (S \cdot \theta(m))_{\mathscr{K}} - (S \cdot \delta(m))_{\mathscr{K}} \\ &\quad - \sum_{p=m+1}^{n-1} (S \cdot \delta(p))_{\mathscr{K}} + \sum_{p=m+1}^{n-1} \sigma(p) \\ &\geq (S \cdot \theta(m))_{\mathscr{K}} - (S \cdot \delta(m))_{\mathscr{K}}.\end{aligned}$$

Since $\mathscr{K} \subseteq \mathscr{R}$ and $(S \cdot \delta(m))_{\mathscr{K}} \leq L$, we have using (14)

$$(S \cdot \theta(n))_{\mathscr{K}} > (|\mathscr{K}| - 1)L. \tag{16}$$

Note that this proves the claim if $F(m+1) = \cdots = F(n)$, that is, if $|\mathscr{K}| = 1$.

If $|\mathscr{K}| \geq 2$, there is $i \in \mathscr{K}$, $i \neq F(n)$. Let $l \in \{m+1, \dots, n-1\}$ be the last step prior to n such that $i = F(l)$. This implies

$$\theta_{F(l)}(n) = \theta_{F(l)}(l) - \sum_{p=l}^{n-1} \delta_{F(l)}(p) + \frac{\sigma(l)}{S_{F(l)}}.$$

Since T is strict at l, $\theta_{F(l)}(l) \leq \theta_{F(n)}(l)$, so that

$$\theta_{F(l)}(n) \leq \theta_{F(n)}(l) - \sum_{p=l}^{n-1} \delta_{F(n)}(p) + \frac{\sigma(l)}{S_{F(l)}} \leq \theta_{F(n)}(n) + \frac{\sigma(l)}{S_{F(l)}}$$

where we have used the fact that $\delta_{F(l)}(p) = \delta_{F(n)}(p)$ for $p = l, \dots, n-1$. Therefore, since $\sigma(l) \leq L$, we have shown that

$$S_i \theta_i(n) \leq S_i \theta_{F(n)}(n) + L \tag{17}$$

for $i \in \mathscr{K}$, $i \neq F(n)$. Summing eq. (17) over such i, we obtain

$$(S \cdot \theta(n))_{\mathscr{K}} \leq S_{\mathscr{K}} \theta_{F(n)}(n) + (|\mathscr{K}| - 1)L. \tag{18}$$

From eqs. (16), (18), and $S_{\mathscr{K}} > 0$, we find that $\theta_{F(n)}(n) > 0$. This completes the proof of the induction step when (15) holds.

Case 2. We now suppose that

$$\{F(m+1), \dots, F(n)\} \not\subseteq \mathscr{R}. \tag{19}$$

Let $k^* + 1$ be the first step in $\{m+1, \dots, n\}$ such that $F(k^*+1) \notin \mathscr{R}$. We claim that

$$\theta_{F(m)}(k^* + 1) - \frac{\sigma(m)}{S_{F(m)}} \leq \theta_{F(k^*+1)}(k^* + 1). \tag{20}$$

This obviously holds if $F(k^* + 1) = F(m)$, so we may assume that $F(k^* + 1) \neq F(m)$. Since $F(k^* + 1)$ was not chosen between steps m and k^*, we have

$$\theta_{F(k^*+1)}(k^* + 1) = \theta_{F(k^*+1)}(m) - \sum_{p=m}^{k^*} \delta_{F(k^*+1)}(p).$$

Similarly, $F(m)$ was only chosen at step m, and so

$$\theta_{F(m)}(k^* + 1) = \theta_{F(m)}(m) - \sum_{p=m}^{k^*} \delta_{F(m)}(p) + \frac{\sigma(m)}{S_{F(m)}}.$$

Using $\delta_{F(m)}(p) = \delta_{F(k^*+1)}(p)$ for $p = m, \dots, k^*$, these inequalities yield

$$\theta_{F(m)}(k^* + 1) - \frac{\sigma(m)}{S_{F(m)}} - \theta_{F(k^*+1)}(k^* + 1) = \theta_{F(m)}(m) - \theta_{F(k^*+1)}(m).$$

However, $\theta_{F(m)}(m) \leq \theta_{F(k^*+1)}(m)$ since $F(k^* + 1) \notin \mathscr{R}$, and so (20) holds.

Let $k + 1$ be the first step in $\{m + 1, \dots, n\}$ satisfying (20). Note that for $i \in \mathscr{R} \neq \emptyset$,

$$\begin{aligned}\theta_{F(m)}(m+1) - \frac{\sigma(m)}{S_{F(m)}} &= \theta_{F(m)}(m) - \delta_{F(m)}(m)\\ &> \theta_i(m) - \delta_i(m)\\ &= \theta_i(m+1),\end{aligned}$$

since $\theta_{F(m)}(m) > \theta_i(m)$ and $\delta_{F(m)}(m) = \delta_i(m)$. As T is strict at $m + 1$, this implies

$$\theta_{F(m)}(m+1) - \frac{\sigma(m)}{S_{F(m)}} > \theta_{F(m+1)}(m+1),$$

and so $k + 1 \in \{m + 2, \dots, n\}$. We wish to "delay" choosing $F(m)$ until step k to obtain a triple with only κ nonstrict steps.

Consider the triple $U = (\nu(l), \gamma(l), G(l))$ defined by

$$\begin{aligned}U(l) &= T(l) && \text{for} \quad l = 0, \dots, m-1\\ U(l) &= T(l+1) && \text{for} \quad l = m, \dots, k-1\\ U(k) &= T(m) && \\ U(l) &= T(l) && \text{for} \quad l = k+1, k+2, \dots\end{aligned} \tag{21}$$

and let $\Phi = (\phi(0), \phi(1), \dots)$, where $\phi(0) = \theta$, be the corresponding state trajectory. Clearly, $\phi(l) = \theta(l)$, for $l = 0, \dots, m$. For $m \leq l \leq k$, we have

$$\begin{aligned}\phi_i(l) &= \theta_i(l+1) + \delta_i(m), \qquad \text{for } i \neq F(m)\\ \phi_{F(m)}(l) &= \theta_{F(m)}(l+1) + \delta_{F(m)}(m) - \frac{\sigma(m)}{S_{F(m)}},\end{aligned} \tag{22}$$

which follows from (21).

We want to show that U is strict at steps $m, \dots, n$ and that $\phi(n) = \theta(n)$. Then the induction hypothesis will give the proposition. For $l + 1 \in \{m + 1, \dots, k\}$, we have $\theta_{F(m)}(l + 1) - \sigma(m)/S_{F(m)} > \theta_{F(l+1)}(l + 1)$, so that $G(l) = F(l + 1) \neq F(m)$. Also recall that $\delta_i(m) = \delta_j(m), i, j \in \mathscr{N}$. Therefore, for $i \neq F(m)$,

$$\phi_{G(l)}(l) = \theta_{F(l+1)}(l+1) + \delta_{F(l+1)}(m) \leq \theta_i(l+1) + \delta_i(m) = \phi_i(l)$$

since T is strict at $l + 1$. Also, for $F(m)$,

$$\begin{aligned}\phi_{G(l)}(l) &= \theta_{F(l+1)}(l+1) + \delta_{F(l+1)}(m)\\ &< \theta_{F(m)}(l+1) + \delta_{F(m)}(m) - \frac{\sigma(m)}{S_{F(m)}}\\ &= \phi_{F(m)}(l).\end{aligned}$$

Thus, U is strict for $l \in \{m, \ldots, k-1\}$. At step k, we have

$$\phi_{G(k)}(k) = \phi_{F(m)}(k) = \theta_{F(m)}(k+1) + \delta_{F(m)}(m) - \frac{\sigma(m)}{S_{F(m)}}$$
$$\leq \theta_{F(k+1)}(k+1) + \delta_{F(k+1)}(m),$$

from (20). Thus, for $i \neq F(m)$,

$$\phi_{G(k)}(k) \leq \theta_i(k+1) + \delta_i(m) = \phi_i(k)$$

since T is strict at $k+1$. This shows that U is also strict at k.

We now claim that $\phi(k+1) = \theta(k+1)$. To see this, note that, for $i \neq F(m)$, we have

$$\phi_i(k+1) = \phi_i(k) - \gamma_i(k) = \phi_i(k) - \delta_i(m) = \theta_i(k+1),$$

and for $F(m)$ we have

$$\phi_{F(m)}(k+1) = \phi_{F(m)}(k) - \gamma_{F(m)}(k) + \frac{\nu(k)}{S_{F(m)}}$$
$$= \phi_{F(m)}(k) - \delta_{F(m)}(m) + \frac{\sigma(m)}{S_{F(m)}} = \theta_{F(m)}(k+1).$$

Thus, $\phi(l) = \theta(l)$ for $l = k+1, k+2, \ldots$. As a consequence of this result, the triple U is strict for $l = k+1, \ldots, n$ since U and T (strict) agree there. Thus U has κ nonstrict steps in $\{0, \ldots, n-1\}$, and so $\theta_i(n) = \phi_i(n) > 0$, $i = 1, \ldots, N$, by the induction hypothesis. This completes the proof of Proposition 3.4.1. □

Relating the above proof to the queueing model, in case 1, only channels with less than L bits at the mth decision epoch are chosen for service between $m+1$ and n. These channels, which correspond to the minimal θ_i for epochs $m+1, \ldots, n$, have a small queue size at m, and a direct calculation shows that no overflow occurs by only serving them. In case 2, a policy with less nonstrict choices is constructed by delaying the service of the channel chosen originally at epoch m until later in the busy period.

3.5. ADMISSIBLE TRIPLES. We now extend the above result to admissible triples with arbitrary $\delta(n)$. For a triple $T = (\sigma(n), \delta(n), F(n))$, let $\delta_{\max}(n) = \max_{i=1,\ldots,N} \delta_i(n)$, $n = 0, 1, \ldots$, and define the set

$$\mathscr{I}(T) = \{(i,n): \delta_i(n) \neq \delta_{\max}(n)\} = \{(i,n): \delta_i(n) < \delta_{\max}(n)\}.$$

Further, define $I = I(T)$ to be the cardinality of $\mathscr{I}(T)$.

We can now prove the main proposition on admissible triples.

PROPOSITION 3.5.1. *Let T be a triple with corresponding state trajectory Θ. If T is admissible, then*

$$\theta_i(n) > 0 \qquad \textit{for} \quad i = 1, \ldots, N,\ n = 0, 1, \ldots. \tag{23}$$

PROOF. It is clear that, for each n, there is an admissible triple U_n that agrees with T for $l = 0, \ldots, n-1$ and satisfies $I(U_n) < \infty$. Thus, we need only prove the proposition for admissible triples T with $I(T) < \infty$. The proof is by induction on $I(T)$.

When $I(T) = 0$, we have $\delta_i(n) = \delta_{\max}(n) = \delta_j(n)$ for $n = 0, 1, \ldots, i, j, \in \mathcal{N}$. Therefore, the result holds in this case by Proposition 3.4.1. Now assume the proposition holds for admissible triples with $I = \kappa$, and we show it holds when $I = \kappa + 1$. Let T be admissible with $I(T) = \kappa + 1$, and choose step n and $j \in \mathcal{N}$ such that $\delta_j(n) < \delta_{\max}(n)$.

The proof of the proposition splits into two cases, depending on whether or not the index j is ever chosen after step n. In each case, we will identify an admissible triple U related to T with $I(U) = \kappa$, and then use the induction hypothesis.

Case 1. First, suppose that

$$\theta_j(l) - \delta_{\max}(n) + \delta_j(n) \geq \theta_{F(l)}(l) \qquad \text{for} \quad l = n+1, n+2, \ldots. \tag{24}$$

In this case $\theta_j(l) > \theta_{F(l)}(l)$, so that $j \neq F(l)$ for $l = n+1, n+2, \ldots$. That is, the index j is not chosen after step n. Define the triple U as follows:

$$\begin{aligned}
U(l) &= T(l) && \text{for} \quad l = 0, \ldots, n-1 \\
\nu(n) &= \sigma(n) && \\
\gamma_i(n) &= \delta_i(n) && \text{for} \quad i \neq j \\
\gamma_j(n) &= \delta_{\max}(n) && \\
G(n) &= F(n) && \\
U(l) &= T(l) && \text{for} \quad l = n+1, n+2, \ldots
\end{aligned} \tag{25}$$

Note that $I(U) = \kappa$. Let $\Phi = (\phi(0), \phi(1), \ldots)$ be the state trajectory corresponding to U. Clearly, $\phi(l) = \theta(l)$ for $l = 0, \ldots, n$. For $l = n+1, n+2, \ldots$, we have

$$\begin{aligned}
\phi_i(l) &= \theta_i(l), \qquad \text{for} \quad i \neq j \\
\phi_j(l) &= \theta_j(l) - \delta_{\max}(n) + \delta_j(n).
\end{aligned} \tag{26}$$

Since $j \neq F(l)$ for such l, we have using (24)

$$\phi_{G(l)}(l) = \phi_{F(l)}(l) = \theta_{F(l)}(l) \leq \theta_j(l) - \delta_{\max}(n) + \delta_j(n) = \phi_j(l). \tag{27}$$

It now follows that U is admissible, because T is. Also, $\theta_i(l) \geq \phi_i(l)$ for $l = 0, 1, \ldots,\ i = 1, \ldots, N$, and so $\theta_i(l) > 0$ by the induction hypothesis.

Case 2. Next suppose that

$$\theta_j(l) - \delta_{\max}(n) + \delta_j(n) < \theta_{F(l)}(l) \tag{28}$$

for some $l \in \{n+1, n+2, \ldots\}$. Let k be the first such step ($k > n$). Define the triple $U = (\nu(l), \gamma(l), G(l))$ as follows:

$$\begin{aligned}
U(l) &= T(l) && \text{for} \quad l = 0, \ldots, n-1 \\
\nu(n) &= \sigma(n) \\
\gamma_i(n) &= \delta_i(n) && \text{for} \quad i \neq j \\
\gamma_j(n) &= \delta_{\max}(n) \\
G(n) &= F(n) \\
U(l) &= T(l) && \text{for} \quad l = n+1, \ldots, k-1 \\
\nu(k) &= S_j\big(\delta_{\max}(n) - \delta_j(n)\big) \\
\gamma(k) &= (0, \ldots, 0) \\
G(k) &= j \\
U(l) &= T(l-1) && \text{for} \quad l = k+1, k+2, \ldots.
\end{aligned} \tag{29}$$

It is easy to verify that U is a triple. For example,

$$0 \leq S_j\big(\delta_{\max}(n) - \delta_j(n)\big) \leq \delta_{\max}(n) \leq \sigma(n) \leq L$$

shows that $0 \leq \nu(k) \leq L$. Further, we have $I(U) = \kappa$. Let $\Phi = (\phi(0), \phi(1), \ldots)$ be the state trajectory corresponding to U.

Clearly, $\phi(l) = \theta(l)$ for $l = 0, \ldots, n$. For $n+1 \leq l \leq k$, the state $\phi(l)$ is given by (26). For $l \in \{n+1, \ldots, k-1\}$, we have $\phi_j(l) \geq \phi_{G(l)}(l)$ (see the derivation of eq. (27)), so that U is admissible at l. Next, by definition of k,

$$\phi_{G(k)}(k) = \phi_j(k) = \theta_j(k) - \delta_{\max}(n) + \delta_j(n) < \theta_{F(k)}(k).$$

Thus, U is admissible at k since T is.

We now consider step $k+1$. For $i \neq j$, we have

$$\phi_i(k+1) = \phi_i(k) = \theta_i(k),$$

while for j we have

$$\phi_j(k+1) = \phi_j(k) + \frac{S_j\big(\delta_{\max}(n) - \delta_j(n)\big)}{S_j} = \theta_j(k).$$

This implies that $\phi(l) = \theta(l-1)$ for $l = k+1, k+2, \ldots,$ and so U is admissible for such l. Therefore, U is an admissible triple such that $I(U) = \kappa$. For $i = 1, \ldots, N$, we have $\theta_i(l) \geq \phi_i(l), l = 0, \ldots, k-1$, and $\theta_i(l) = \phi_i(l+1), l = k, k+1, \ldots$. Thus, $\theta_i(l) > 0$ by the induction hypothesis. This completes the proof of Proposition 3.5.1. □

To describe the above proof in terms of the queueing model, recall that the case when all channels remain on for the same percentage of time while messages are being served is covered by Proposition 3.4.1. To handle the general case, channel j is turned on for an additional length of time $\delta_{\max}(n) - \delta_j(n)$ during the nth service, so that the on periods of the channels are more uniform. That is, $S_j(\delta_{\max}(n) - \delta_j(n)) \leq L$ "dummy bits" are added to the buffer at channel j. Case 1 corresponds to channel j never being chosen for service (for example, it may never contain a complete message). Case 2 corresponds to serving the dummy bits while keeping all channels turned off during their service.

4. *Optimality of LTRB*

We now use the results of Section 3 to prove the optimality of the least time to reach bound (LTRB) class of service policies introduced in Section 2. Our interest is in a service policy for which the buffer size required at each channel to prevent overflow is independent of the number of channels and their speeds. We will show that not only do the LTRB policies possess such a property, but, in addition, the members of this class are optimal in terms of buffer size. Our approach is to examine any instance of the evolution of the behavior of the system and apply Proposition 3.5.1 of the previous section to conclude that no overflow will occur. The proof will yield an upper bound of $2L$ for the buffer size at each channel to prevent overflow. However, as mentioned in Section 2, the value $2L$ is a lower bound for any policy that is work-conserving and for which only complete messages are served. Thus, the proof of the upper bound shows that LTRB is optimal among all such policies.

Recall that L is the maximum message length, and S_i, $i = 1, \ldots, N$, are the relative speeds of the N channels. The number of bits in storage at channel i at time t is denoted $Q_i(t)$, and, for $Q_i(t) \le 2L$, $\theta_i(t) = (2L - Q_i(t))/S_i$ is the time for the queue size at channel i to reach $2L$, assuming a continuous flow of bits at rate S_i and no service given to channel i. Since overflow occurs if and only if it occurs at a decision epoch during a busy period, we may concentrate on these instants of time. Now recall that the states θ_i at decision epochs during a busy period satisfy difference equations of the form (7). Here $\sigma(n)$ is the length of the message chosen for service at the nth epoch, $F(n)$ is the channel containing that message, and $\delta_i(n), i = 1, \ldots, N$, is the length of the corresponding on period for channel i. Further, the queue size at channel i at the beginning of a busy period is at most L, so that the initial state θ satisfies $\theta_i \ge L/S_i$, $i = 1, \ldots, N$. With these definitions, the sequence $(\sigma(n), \delta(n), F(n))$, $n = 0, 1, \ldots,$ represents a triple as defined in Section 3. It is also clear that members of the LTRB class yield triples that are admissible, and so the results of the previous section may be applied to these policies.

As mentioned before, certain restrictions on the triples arise naturally in the queueing model. For example, the buffer size $Q_i(n)$ cannot be negative (the value of $\theta_i(n)$ cannot go above $2L$), while this is allowed in the triples studied in Section 3. Also, in the queueing model a message cannot be transmitted until sufficient time has elapsed for it to arrive (e.g., it may not be possible for a channel to transmit L bits in two successive steps). Thus, the set of triples generated by the queueing model is a small subset of the set of all possible triples considered in Section 3. However, we may apply the general result obtained in that section to show that no overflow can occur for a system operating under an LTRB policy which has buffers of size $2L$.

THEOREM 4.1. *Consider a system of N channels with channel speeds S_i, $i = 1, \ldots, N$, and server speed $S \ge \sum_{i=1}^{N} S_i$ operating under any member of the class of LTRB policies. Then*

$$Q_i(t) \le 2L \qquad \text{for} \quad i = 1, \ldots, N, t \ge 0. \tag{30}$$

PROOF. As discussed above, only decision epochs $\tau_n, n = 0, 1, \ldots,$ within a particular busy period need to be considered. When all channels remain on for the same percentage of time during the service of each message, then $\delta_i(n) =$

$\delta_j(n)$ for $i, j \in \mathcal{N}$; $n = 0, 1, \ldots,$ which is the situation covered by Proposition 3.4.1. The more general case when input flow to the channels can be turned on and off arbitrarily is handled in Proposition 3.5.1. □

Theorem 4.1 shows that the bound of $2L$ is valid for any number of channels and any set of channel speeds. However, in certain cases, an even smaller bound may hold. For example, suppose all channel speeds are identical ($S_i/\Sigma_j S_j = 1/N$). In this case, $\theta_i(t) = C(2L - Q_i(t))$, where C is a constant independent of the channel. Thus, choosing the minimal θ_i is equivalent to choosing the buffer with the maximal queue size among all channels with a complete message, which shows that the Longest Queue First policy (LQF) is in the LTRB class. Using results from Gail et al. [1993] for LQF, the upper bound on buffer storage for this case is, in fact, $(2 - 1/N)L < 2L$.

We have shown that the policies from the LTRB class require buffer storage of $2L$, which is independent of the number of channels and their relative speeds. These policies are also optimal in terms of storage. To see this, first note that, in Cidon et al. [1988], an example was given to show that a buffer size of $2L$ is a lower bound for any policy that satisfies the properties (a)–(c) discussed in Section 1. Let us briefly review the lower bound example. Consider an N buffer system with channel speeds that are identical and such that their aggregate speed is equal to the server speed ($S_i = 1/N$ for all i). Suppose that initially all N buffers contain a maximal length message of L bits. After serving all but one of the buffers at least once in an order dictated by the policy (this takes time at least $(N - 1)L$ since $N - 1$ of the maximal length messages must have been served and the server speed is unity), the final buffer that has not yet been served will contain at least $L + (1/N)(N - 1)L = L(2 - 1/N)$ bits. As $N \to \infty$, we obtain the lower bound of $2L$ for any such policy.

Thus, the following is an immediate consequence of Theorem 4.1:

THEOREM 4.2. *The LTRB class is optimal in terms of buffer storage for policies that are work-conserving and for which only complete messages are served.*

5. *Discussion*

In this paper, we introduced a new class of service policies, called Least Time to Reach Bound (LTRB), for servicing messages that reside in the input buffers of a switch. According to this policy, once a message has completed service, the next message to be chosen is taken from a buffer that would overflow first assuming a continuous flow of bits to all input channels and no further service. We proved that operating under this class of policies guarantees no overflow (and thus no message loss) when the buffer storage at each input channel is only *twice* the maximal length of a message for any number of channels and any set of speeds. This class is optimal in the sense that any nonoverflowing policy (satisfying conditions (a)–(c)) requires at least as much storage as LTRB.

There are obvious advantages of storage requirements that do not increase with the number of incoming channels when compared to the logarithmic growth under the Longest Queue First (LQF) service policy or the linear growth under the Exhaustive Round Robin (ERR) and First Come First Served (FCFS) service policies. The buffer sizes are much smaller and need not be changed every time an input channel is added to the switch, and thus the system is more robust. Yet, to gain these advantages, the switch must be able to determine the number of bits in each of the incoming buffers at the end of service of each message and must also know the speeds of the incoming channels. With ERR the switch is simpler, since the queue lengths of other buffers need not be observed while some buffer is served. The switch with LQF is similar to that with LTRB, except that there is no need to know the speeds of the incoming channels. In conclusion, we observe a trade-off between the amount of storage required and the complexity of the switch.

REFERENCES

ANICK, D., MITRA, D., AND SONDHI, M. M. 1982. Stochastic theory of a data-handling system with multiple sources. *Bell Syst. Tech. J. 61*, 8 (Oct.), 1871–1894.

BIRMAN, A., CHANG, P. C., CHEN, J. S. C., AND GUÉRIN, R. 1989. Buffer sizing in an ISDN frame relay switch. IBM Res. Rep. RC14836. IBM Thomas J. Watson Research Center, Yorktown Heights, N.Y.

CIDON, I., GOPAL, I., GROVER, G., AND SIDI, M. 1988. Real time packet switching: A performance analysis. *IEEE J. Sel. Areas Commun. SAC-6*, (Dec.), 1576–1586.

COHEN, J. W. 1969. *The Single Server Queue*. North-Holland, Amsterdam, The Netherlands.

COHEN, J. W. 1974. Superimposed renewal processes and storage with gradual input. *Stochast. Proc. Their Appl. 2*, 31–58.

CRUZ, R. L. 1991. A calculus for network delay, Part I: network elements in isolation. *IEEE Trans. Inf. Theory 37*, 1 (Jan.) 114–131.

GAIL, H. R., GROVER, G., GUÉRIN, R., HANTLER, S. L., ROSBERG, Z., AND SIDI, M. 1993. Buffer size requirements under longest queue first. *Perf. Eval. 18*, 2 (Sept.) 133–140.

GAVER, D. P., AND MILLER, R. G. 1962. Limiting distributions for some storage problems. In *Studies in Applied Probability and Management Science*. Arrow, K., Karlin S., and Scarf, H., eds., Stanford Univ. Press, Stanford, Calif.

GREENBERG, A. G., AND MADRAS, N. 1992. How fair is fair queuing? *J. ACM, 39*, 3 (July) 568–598.

KASPI, H., AND RUBINOVITCH, M. 1975. The stochastic behavior of a buffer with non-identical input lines. *Stochast. Proc. Their Appl. 3*, 73–88.

KRISHNA, A. 1990. Communication with few buffers: Analysis and design. Univ. Illinois Rep. UILU-ENG-90-2259. Univ. Illinois, Urbana-Champaign, Ill.

RUBINOVITCH, M. 1973. The output of a buffered data communication system. *Stochast. Proc. Their Appl. 1*, 375–382.

SASAKI, G. 1989. Input buffer requirements for round robin polling systems. In *Proceedings of 27th Annual Allerton Conference on Communication, Control, and Computing*. Univ. Illinois, Urbana-Champaign, Ill., pp. 397–406.

RECEIVED MARCH 1991; REVISED NOVEMBER 1994; ACCEPTED JANUARY 1995