Overload Performance of Several Processor Queueing Disciplines for the *M/M/1* Queue

BHARAT T. DOSHI, MEMBER, IEEE, AND HARRY HEFFES, SENIOR MEMBER, IEEE

Abstract—**In a variety of overloaded queueing systems (e.g., an overloaded call processing system), long delays can result either in poor service given to the customer or in customers, unknown to the system, turning "bad." For example, in switching systems, long dial tone delays can result in customers initiating dialing before receiving dial tone. In this case the system will not receive all the digits and an unsuccessful call results. This can lead to the system expending real time on unsuccessful services and, therefore, reduces the effective throughput. Thus, there is a need for control schemes which reduce the load offered to the processor by selectively refusing service to some customers in such a way as to keep delays, for those customers which are selected for service, small. This fact has been recognized and has led to improved strategies for local switches. In this paper we analyze and compare the performance of various queueing and service disciplines for an *M/M/1* queue. We consider LIFO and FIFO schemes with customer rejection mechanisms corresponding to pushing out or timing out older customers in queue. Delay distributions for served customers are obtained and comparisons based upon throughput-delay tradeoffs are presented. For the situation where customers can turn "bad" at a random time after their arrival, we compare the throughput of good customers. The results presented are a mixture of classical results, which are briefly stated, and new results which are developed in more detail. The numerical results show a dramatic effect of the queueing and service disciplines on the overload performance and a strong dependence of the throughput of successful services on the mechanism for customers turning "bad." Although results are obtained for a single server queue, they can be used to approximately analyze overload control schemes which control access to distributed systems.**

I. INTRODUCTION

THE main function of a processor overload control is to protect the processor by reducing the load offered to it, if necessary, in a way which is consistent with the desire to maintain a high throughput in the system and small processing delays. Maintaining a high throughput requires that calls are blocked only when necessary. Because of statistical load fluctuations some unnecessary blocking is unavoidable, but this should be kept to a minimum. The delays, on the other hand, are important from several points of view. First, long delays are clearly undesirable from the customer's point of view. Second, in many systems, customers may become impatient and either abandon after some wait or take action which makes subsequent service unsuccessful (i.e., customers turn "bad" [1]). For example, in switching systems customers may abandon while waiting for dial tone, or may start dialing before receiving dial tone, due to long dial tone delays. In the latter case the system will not receive all the digits and an unsuccessful call results. Thus, long delays can lead to the system expending real time on unsuccessful calls and can therefore reduce the effective throughput. This fact has been recognized for some time and has resulted in improved strategies for local switches [2].

In general, there is a need for control mechanisms which limit the load offered to the processor by selectively refusing service to some customers, if necessary, to a level at which the delays for those customers who are served are small. The selective refusal of service to those customers which would have seen long delays should assure that customers who do get served will be served quickly and successfully. Of course, such a scheme should not refuse service when the arrival rate is small. Thus, we have a tradeoff between the throughput and delay of customers who get served.

In this paper we analyze and compare the performance of several control schemes for the *M/M/1* queue. We consider LIFO and FIFO schemes with customer rejection mechanisms corresponding to pushing out or timing out older customers in queue. Delay distributions for served customers are obtained and comparisions based on throughput–delay tradeoff characteristics are presented. For the situation where customers can turn "bad" at random time after their arrival, we present results for the throughput of good customers ("goodput"). The results show a significant effect of the control mechanism on performance as well as a strong dependence of the throughput of good customers on the mechanism for customers turning bad. Although results are obtained for a single server queue, they can be used to approximately analyze overload control schemes which control access to distributed systems [3].

The five control schemes considered differ in the service discipline and the customer rejection mechanism. Two service disciplines are considered, the first-in first-out (FIFO) and the last-in first-out (LIFO), and three different situations are considered as to which customers are refused service. The first two are window-based with window size N which corresponds to a finite buffer of size $N - 1$. In the pure blocking case a customer arriving to see a full buffer is refused service (blocked), whereas in the pushout case, a customer arriving to see a full buffer joins the buffer while pushing out the oldest customer waiting in the buffer. The third scheme is based on waiting time where, theoretically, an infinite buffer size is available but a customer is timed out (refused service) after it has spent T time units in the buffer.

Specifically, the five disciplines we consider are as follows.

i) FIFO-Blocking (FIFO-BL): First-in first-out (FIFO) service; a finite buffer of size $N - 1$; a customer arriving to see a full buffer leaves immediately. This is the classical $M/M/1/N$ queue.

ii) FIFO-Pushout (FIFO-PO): FIFO service; a finite buffer of size $N - 1$; a customer arriving to see a full buffer pushes out the oldest customer in the buffer and joins the queue.

iii) LIFO-Pushout (LIFO-PO): Last-in first-out (LIFO)

Paper approved by the Editor for Computer Communications Theory of the IEEE Communications Society. Manuscript received March 27, 1984; revised December 1, 1985. This paper was presented at the ORSA/TIMS Special Interest Meeting on Applied Probability in Biology and Engineering, Lexington, KY, July 1983.

The authors are with AT&T Bell Laboratories, Holmdel, NJ 07733.

IEEE Log Number 8608507.

0090-6778/86/0600-0538$01.00

service; a finite buffer of size $N-1$; a customer arriving to see a full buffer pushes out the oldest customer in the buffer.

iv) FIFO-Timeout (FIFO-TO): FIFO discipline; infinite buffer; every arriving customer joins the buffer but will leave at time T after arrival if it is still in the buffer at that time.

v) LIFO-Timeout (LIFO-TO): Same as (iv) but the service discipline is LIFO.

In order to compare the control schemes from both the customer's perspective and the system's perspective, we need to evaluate the conditional waiting time distributions for the various schemes involving pushouts and timeouts. Some results are available in the literature, and they will be briefly stated, while other results are new and will be derived in greater detail.

Section II-A briefly states the formulas for the FIFO-BL scheme which are available from the classical $M/M/1/N$ solution [4]. Section II-B addresses the FIFO-PO scheme. Here, the conditional waiting time distribution and the recursions for its moments are completely new (only mean values were considered by the authors in [5]). Two approaches are presented for this control scheme, one a transform approach and the other a time domain approach. The solution technique exploits the structure of the derived three-dimensional recursions. Although the LIFO-PO discipline has been treated earlier [6], the analysis presented in Section II-C avoids the consideration of eigenvalues of an $(N-1)\times(N-1)$ matrix used in the earlier treatment. Also included in this section is a discussion relating to analysis of the LIFO-PO scheme for a nonexponential service distribution. While the FIFO-TO problem has a well-known solution [7], which is briefly stated in Section II-D, we present a nonclassical derivation of the results based on level crossing ideas [8] in the Appendix. We further note that the level crossing approach is extendable to general service time distributions. The last scheme, LIFO-TO, has been treated by one of the authors in [1] and the results briefly stated in Section II-E. This section also includes a discussion of how to generalize the analysis for the $M/G/1$ case. In Section III we present new results for determining the throughput of successful services for the situation where customers can turn bad at random time after their arrival. Finally, numerical results are presented in Section IV and a discussion in Section V. We thus see that the analytic results presented are a mixture of classical and new results, presented in a unified way for our primary objective of making the desired performance comparisons.

II. Analysis of the Control Schemes

In this section we obtain expressions for the performance measures for the five schemes described in Section I. In what follows we let λ correspond to the arrival rate of the Poisson process and μ the service rate corresponding to the exponential service time distribution. Our interest is in the overload situation where λ is close to or exceeds μ.

A. FIFO-Blocking (FIFO-BL)

Let $\rho=\lambda/\mu$ and let p_i be the probability of having i customers in the system (buffer plus processor), $i=0,1,\cdots,N$. Then, from classical results [4] we get

$$p_i=\frac{\rho^i}{\sum_{j=0}^{N}\rho^j}=\frac{\rho^i(1-\rho)}{1-\rho^{N+1}}$$

$$P_s=P\{\text{an arrival gets served}\}=\frac{\sum_{j=0}^{N-1}\rho^j}{\sum_{j=0}^{N}\rho^j}=\frac{1-\rho^N}{1-\rho^{N+1}}$$

$\lambda_T=\text{throughput}=\lambda P_S$

$M=$ mean waiting time for the customers who get served

$$=\frac{\sum_{j=1}^{N-1}jp_j}{\mu P_S}$$

$$=\begin{cases}\dfrac{1}{\mu P_S}\left[\dfrac{1}{1-\rho}-\dfrac{(N+1)\rho^N}{1-\rho^{N+1}}\right]-\dfrac{1}{\mu}, & \rho\neq 1\\[2ex] \dfrac{N-1}{2\mu}, & \rho=1\end{cases}$$

$f_s(t)=$ density function for the waiting time of the customers who get served

$$f_s(t)=\frac{\sum_{j=1}^{N-1}p_j\dfrac{e^{-\mu t}\mu^j t^{j-1}}{(j-1)!}}{P_S}\qquad(0<t<\infty)\tag{2.1.1}$$

with an atom at 0 of

$$F_S(0)=\frac{p_0}{P_S}.\tag{2.1.2}$$

The distribution function for the waiting time of the customers who get served is then given by

$$F_s(t)=F_s(0)+\int_{0+}^{t}f_s(\tau)\,d\tau.$$

B. FIFO-PO

Here, p_i, $i=0,1,\cdots,N$, P_S, and λ_T for this scheme are the same as for the FIFO-blocking scheme. We next evaluate the moments and the distribution of the waiting time of the customers who get served.

Suppose a customer arrives to join the system in position i, $i=1,2,\cdots,N$ (position 1 is the processor). Let us follow the movement of this tagged customer. If $i=1$, then the customer is already in service and its waiting time is zero. If $i>1$, then this customer is in the buffer. When a service completion occurs and this tagged customer is in position j, it moves to position $j-1$. When an arrival occurs and the buffer is not full, it joins the buffer behind the waiting customers. If an arrival occurs and the buffer is full, it joins the buffer in position N and the customer in position 2 gets pushed out. If the tagged customer is in position 2, it gets pushed out; otherwise it moves from position j to $j-1$. Thus, the movement of the tagged customer depends on both its own position and the number of customers behind it (or, equivalently, the total number in the system). For the tagged customer consider the state (j,k) where j is its own position and k is the number behind it. Initially, the state is $(i,0)$ with probability p_{i-1} for $i\le N-1$ and with probability $p_{N-1}+p_N$ if $i=N$. Let

$f(j,k)$

$=P\{\text{a tagged customer in state } (j,k) \text{ will get served eventually}\}$

$G(j,k,t)$

$=P\{\text{a tagged customer in state } (j,k) \text{ will get served and its remaining waiting time will not exceed } t\}$

$$g^*(j,\ k,\ \theta) = \int_{0-}^{\infty} e^{-\theta t}\, dG(j,\ k,\ t).$$

Then, we have the following recursions for f:

$$f(1,\ k) = 1 \qquad \text{for all } k$$

$$f(j,\ k) = \frac{\lambda}{\lambda+\mu} f(j,\ k+1) + \frac{\mu}{\lambda+\mu} f(j-1,\ k) \qquad (j \geq 2,\ j+k<N)$$

$$f(j,\ N-j) = \frac{\lambda}{\lambda+\mu} f(j-1,\ N-j+1) + \frac{\mu}{\lambda+\mu} f(j-1,\ N-j) \qquad (j>2)$$

$$f(2,\ N-2) = \frac{\mu}{\lambda+\mu}.$$

We can solve for $f(j, k)$ by using the above recursions in the order $(2, N-2)$, $(2, N-3)$, $\cdots$, $(2, 0)$, $(3, N-3)$, $(3, N-4)$, $\cdots$, $(N, 0)$.

Similarly, for g^* we get the following:

$$g^*(1,\ k,\ \theta) = 1 \qquad \text{for all } k,\ \theta$$

$$g^*(j,\ k,\ \theta) = \frac{\lambda}{\lambda+\mu+\theta} g^*(j,\ k+1,\ \theta) + \frac{\mu}{\lambda+\mu+\theta} g^*(j-1,\ k,\ \theta) \qquad (j \geq 2,\ j+k<N) \quad (2.2.1)$$

$$g^*(j,\ N-j,\ \theta) = \frac{\lambda}{\lambda+\mu+\theta} g^*(j-1,\ N-j+1,\ \theta) + \frac{\mu}{\lambda+\mu+\theta} g^*(j-1,\ N-j,\ \theta) \qquad (j>2) \quad (2.2.2)$$

$$g^*(2,\ N-2,\ \theta) = \frac{\mu}{\lambda+\mu+\theta}. \qquad (2.2.3)$$

These recursions can be used in a variety of ways. First, we obtain the recursions for the mean and higher moments. Let

$$M_n(j,\ k) = \int_{0-}^{\infty} t^n\, dG(j,\ k,\ t) = (-1)^n \lim_{\theta \to 0} g^{*(n)}(j,\ k,\ \theta).$$

$M_0(j, k) = f(j, k)$ are easily obtained from the earlier recursions for f. Next, from (2.2.1)–(2.2.3) we get

$$M_n(1,\ k) = 0 \qquad \text{for all } k$$

$$M_n(j,\ k) = \frac{n}{\lambda+\mu} M_{n-1}(j,\ k) + \frac{\lambda}{\lambda+\mu} M_n(j,\ k+1) + \frac{\mu}{\lambda+\mu} M_n(j-1,\ k) \qquad (j>2,\ j+k<N)$$

$$M_n(j,\ N-j) = \frac{n}{\lambda+\mu} M_{n-1}(j,\ N-j) + \frac{\lambda}{\lambda+\mu} \cdot M_n(j-1,\ N-j+1) + \frac{\mu}{\lambda+\mu} M_n(j-1,\ N-j) \qquad (j>2)$$

$$M_n(2,\ N-2) = \frac{n}{\lambda+\mu} M_{n-1}(2,\ N-2).$$

Starting with $M_0 = f$ we can use the above recursions to get $M_n(j, k)$ for all $n \geq 1$. The conditional nth moment of the waiting time for the customers who get served can now be evaluated by using

$$M_n = \frac{\sum_{j=1}^{N-1} p_j M_n(j+1,\ 0) + p_N M_n(N,\ 0)}{P_S}.$$

In particular,

$$M = M_1 = \frac{\sum_{j=1}^{N-1} p_j M_1(j+1,\ 0) + p_N M_1(N,\ 0)}{P_S}.$$

We can also use the recursions in (2.2.1)–(2.2.3) to obtain $g^*(j, k, \theta)$ for any j, k, and θ. These can then be used together with a numerical inversion of the transform (e.g., [9]) to obtain $g(j, k, t) = G'(j, k, t)$. Then the conditional waiting time distribution is given by

$$F_S(0) = \frac{P_0}{P_S} \qquad (2.2.4)$$

$$f_S(t) = \frac{\sum_{j=1}^{N-1} p_j g(j+1,\ 0,\ t) + p_N g(N,\ 0,\ t)}{P_S}, \qquad t>0. \qquad (2.2.5)$$

We can also obtain $g(j, k, t)$ in a different way. Suppose we consider an event as either an arrival or a service completion. Then these events occur according to a Poisson process at rate $\lambda + \mu$ as long as the system is not empty. If a customer arrives when the system is not empty, then it has to wait and during its waiting time the system will never be empty. Also, it will either go into service or will get pushed out after a random but finite number of events. Thus, the waiting time of a customer in the buffer is a mixture of gamma distributed random variables. In particular,

$$g(j,\ k,\ t) = \sum_{m=1}^{\infty} h(j,\ k,\ m) \cdot \frac{e^{-(\lambda+\mu)t}(\lambda+\mu)^m t^{m-1}}{(m-1)!} \qquad (j \geq 2),$$

where

$$h(j,\ k,\ m) = P\left\{\begin{array}{l}\text{a customer in state } (j,\ k) \text{ will get} \\ \text{served eventually and its service will start} \\ \text{after } m \text{ events}\end{array}\right\}.$$

Thus, it suffices to obtain $h(j, k, m)$ for all j, k, and m. We obtain these by the following recursions:

$$h(1,\ k,\ 0) = 1 \qquad \text{for all } k$$

$$h(1,\ k,\ m) = 0 \qquad \text{for all } k \text{ and } m \geq 1$$

$$h(j,\ k,\ m) = 0 \qquad \text{for all } m > N+j-k-3$$

$$h(j,\ k,\ m) = 0 \qquad \text{for all } j \geq m+2$$

$$h(j,\ k,\ m)=\frac{\lambda}{\lambda+\mu}\,h(j,\ k+1,\ m-1)+\frac{\mu}{\lambda+\mu}\cdot h(j-1,\ k,\ m-1)\qquad (j\geq 2,\ j+k<N)$$

$$h(j,\ N-j,\ m)=\frac{\lambda}{\lambda+\mu}\,h(j-1,\ N-j+1,\ m-1)+\frac{\mu}{\lambda+\mu}\,h(j-1,\ N-j,\ m-1)\qquad (j>2)$$

$$h(2,\ N-2,\ m)=\frac{\mu}{\lambda+\mu}\,h(1,\ N-2,\ m-1).$$

The relevant density, $g(j+1, 0, t)$, is then given by the finite sum

$$g(j+1,\ 0,\ t)=\sum_{m=j}^{j+N-2} h(j+1,\ 0,\ m)e^{-(\lambda+\mu)t}\,\frac{(\lambda+\mu)^m t^{m-1}}{(m-1)!}.$$

A different way of getting the delay distribution, based on viewing delay as the time until absorption in a Markov process, is reported in [10].

C. LIFO-PO

Once again p_i, P_S, and λ_T are the same as for FIFO-blocking and FIFO-PO. We now obtain the moments and the distribution of the waiting time of the customers who get served. Suppose the positions in the buffer are numbered $2, 3, \cdots, N$ with the processor numbered 1. An arriving customer goes into service (position 1) if the system is empty; otherwise it goes into position 2. If an arrival occurs while this customer is waiting in position 2, it moves to position 3 and the new arrival moves to position 2. Of course, a service completion brings the positions of all the waiting customers down by 1. If a customer is in position N and an arrival occurs, then it gets pushed out and the new arrival joins position 2. Let

$$f(j)=P\{\text{a customer in position } j \text{ will get served eventually}\}$$

$$G(j,\ t)=P\left\{\begin{array}{l}\text{a customer in position } j \text{ will get served}\\ \text{eventually and its remaining waiting}\\ \text{time will be no greater than } t\end{array}\right\}$$

$$g(j,\ t)=G'(j,\ t)\qquad (t>0)$$

$$M_n(j)=\int_0^\infty t^n g(j,\ t)\,dt.$$

Then,

$$F_S(0)=\frac{p_0}{P_S}$$

and

$$f_S(t)=\frac{(1-p_0)g(2,\ t)}{P_S}=\frac{\lambda}{\mu}\,g(2,\ t),\qquad (t>0).\quad (2.3.1)$$

We obtain the following recursions for $f(j)$:

$$f(j)=\frac{\mu}{\mu+\lambda}f(j-1)+\frac{\lambda}{\lambda+\mu}f(j+1)$$

with boundary conditions

$$f(1)=1,\quad f(N+1)=0.$$

This has the solution

$$f(j)=\frac{\sum_{i=0}^{N-j}\rho^i}{\sum_{i=0}^{N-1}\rho^i};\qquad 1\leq j\leq N.$$

Applying Little's law to those customers in the queue which get served, we obtain [5]

$$E[\text{delay}|\text{served}]=\begin{cases}\dfrac{\rho[1-\rho^{2N-1}-\rho^{N-1}(1-\rho)(2N-1)]}{\mu(1-\rho)(1-\rho^N)^2}; & \rho\neq 1\\[2ex] \dfrac{(N-1)(2N-1)}{6\mu N}; & \rho=1.\end{cases}\qquad (2.3.2)$$

For $g^*(j,\theta)=\int_{0-}^{\infty} e^{-\theta t}\,dG(j,t)$ we obtain the following:

$$g^*(1,\ \theta)=1,\quad g^*(N+1,\ \theta)=0\qquad \text{for all } \theta$$

$$g^*(j,\ \theta)=\frac{\mu}{\lambda+\mu+\theta}\,g^*(j-1,\ \theta)+\frac{\lambda}{\lambda+\mu+\theta}\,g^*(j+1,\ \theta)\qquad (1<j<N)$$

$$g^*(N,\ \theta)=\frac{\mu}{\lambda+\mu+\theta}\,g^*(N-1,\ \theta).$$

These can be solved to yield

$$g^*(j,\ \theta)=\frac{r_1(\theta)^{j-1}r_2(\theta)^N-r_1(\theta)^N r_2(\theta)^{j-1}}{r_2(\theta)^N-r_1(\theta)^N}\qquad (2.3.3)$$

with

$$r_{1,2}(\theta)=\frac{(\lambda+\mu+\theta)\pm\sqrt{(\lambda+\mu+\theta)^2-4\lambda\mu}}{2\lambda}.$$

Transform inversion for $j=2$ and using (2.3.1) gives the desired delay distribution.

We remark that in [6] an approach is given to obtain an alternate form for the transform of the delay distribution of served calls (as a finite sum) in terms of eigenvalues of an $(N-1)\times(N-1)$ matrix, as well as for obtaining recursions for the moments of the delay distribution. Here, we obtain expressions for higher moments of the delay distribution for served calls from

$$M_n(2)=(-1)^n g^{*(n)}(2,\ \theta)|_{\theta=0}$$

and

$$M_n=E[(\text{delay})^n|\text{served}]=\frac{1-p_0}{P_S}\,M_n(2).$$

We note that the approach in this paper can be modified to handle other service time distributions [11]. For example, the $M/D/1$ LIFO-PO discipline can be analyzed by solving the partial differential-difference equations for the probability that a tagged customer in a given queue position gets served with remaining waiting time not exceeding a given value, conditioned on the elapsed service time of the customer in service. Note that this is a conditioned version of the quantity $G(j, t)$ of this section.

D. FIFO-TO

This discipline corresponds to the classical system with waiting time limited by a constant [7]. A nonclassical approach to obtaining these results, which appears in the Appendix, uses level crossing ideas [8] and is extendable to the nonexponential service time case [11]. We briefly state the results for exponential service times below.

$$P_s=\begin{cases}\dfrac{1-\rho e^{-(\mu-\lambda)T}}{1-\rho^2 e^{-(\mu-\lambda)T}}, & \rho\neq 1\\[2ex] \dfrac{1+\mu T}{2+\mu T}, & \rho=1.\end{cases}$$

The mean delay of customers that receive service is given by

$$M=M_1=\begin{cases}\dfrac{\rho[1-e^{-(\mu-\lambda)T}-\mu T(1-\rho)e^{-(\mu-\lambda)T}]}{\mu(1-\rho)(1-\rho e^{-(\mu-\lambda)T})}, & \rho\neq 1\\[2ex] \dfrac{\mu T}{1+\mu T}\dfrac{T}{2}, & \rho=1\end{cases}$$

with the delay distribution of served customers given by

$$F_s(t)=\begin{cases}\dfrac{1-\rho e^{-(\mu-\lambda)t}}{1-\rho e^{-(\mu-\lambda)T}}, & \rho\neq 1,\quad 0\le t\le T\\[2ex] \dfrac{1+\mu t}{1+\mu T}, & \rho=1,\quad 0\le t\le T\\[2ex] 1, & t>T.\end{cases}$$

Finally, the throughput is clearly given by $\lambda_T = \lambda P_S$.

E. LIFO-TO

Since this discipline has been studied in [1], we briefly summarize the results. Let $B(\cdot)$ be the distribution function of the busy period started by one customer in the usual $M/M/1$ queue. Then,

$$P_S=\frac{1}{1+\rho(1-B(T))},$$

the throughput λ_T is given by

$$\lambda_T=\lambda P_S=\frac{\lambda}{1+\rho(1-B(T))}$$

$$M_n=\rho\int_0^T t^n\,dB(t)$$

$$p_0=\frac{1-\rho B(T)}{1+\rho(1-B(T))}$$

and for $t \le T$

$$F_S(t)=1-\rho[B(T)-B(t)]. \tag{2.5.1}$$

We note that M_n can be obtained from a single transform inversion by defining

$$M_n(\tau)=\rho\int_0^\tau t^n\,dB(t),$$

recognizing

$$\tilde{M}_n(\theta)=L[M_n(\tau)]=(-1)^n\frac{\rho}{\theta}\frac{d^n}{d\theta^n}\tilde{b}(\theta)$$

where [12]

$$\tilde{b}(\theta)=\frac{\lambda+\mu+\theta-[(\lambda+\mu+\theta)^2-4\mu\lambda]^{1/2}}{2\lambda} \tag{2.5.2}$$

and inversion of $\tilde{M}_n(\theta)$ at the point $\tau = T$.

The above analysis need only be slightly modified to handle general service time distribution, with $B(t)$ replaced by $B_r(t)$, the $M/G/1$ busy period distribution initiated by the forward recurrence time of the service time distribution [13].

III. Throughput of Good Customers (Goodput)

The next set of results corresponds to the situation where a customer in queue, unknown to the system, turns "bad" at a random time after its arrival [1]. This arises in a variety of call processing systems. Thus, serving a customer with delay in excess of this random time results in a "bad" (unsuccessful) service. Clearly, the delay distribution of served customers and the distribution of the time at which a customer turns bad determine the rate at which the system serves good customers (goodput). Defining $P(t)$ = Pr [customer in queue for t seconds is good], we consider two cases.

$$\text{Case I:}\quad P(t)=e^{-\alpha t}$$

and

$$\text{Case II:}\quad P(t)=\begin{cases}1 & t\le\tau\\ 0 & t>\tau.\end{cases}$$

Since the goodput V is given by

$$V=\lambda P_S\int_{0^-}^{\infty}P(t)\,dF_S(t)$$

we have

$$V_I=\lambda P_S L[f_s(t)]_{\theta=\alpha}$$

for Case I. To obtain the Case I goodputs we either use the transform results, or transform the result of Section II and obtain the following.

FIFO-BL:

$$V_I=\lambda p_0+\lambda\sum_{j=1}^{N-1}p_j\left[\frac{\mu}{(\alpha+\mu)}\right]^j. \tag{3.1}$$

FIFO-PO:

$$V_I=\lambda p_0+\lambda p_N g^*(N,\ 0,\ \alpha)+\lambda\sum_{j=1}^{N-1}p_j g^*(j+1,\ 0,\ \alpha) \tag{3.2}$$

where $g^*(j, k, \alpha)$ can be obtained from the recursions (2.2.1)–

(2.2.3) or directly from

$$g^*(j+1,\ 0,\ \alpha)=\sum_{m=j}^{j+N-2} h(j+1,\ 0,\ m)\left[\frac{\lambda+\mu}{\alpha+\lambda+\mu}\right]^m$$

LIFO-PO:

$$V_I=\lambda p_0+\lambda(1-p_0)g^*(2,\ \alpha) \tag{3.3}$$

with $g^*(2, \alpha)$ obtained from (2.3.3).

FIFO-TO:

$$V_I=A+\frac{A\lambda}{\mu+\alpha-\lambda}\left[1-e^{-(\mu+\alpha+\lambda)T}\right] \tag{3.4}$$

with

$$A=\begin{cases}\dfrac{\lambda(1-\rho)}{1-\rho^2 e^{-(\mu-\lambda)T}}, & \rho\neq 1\\[2ex] \dfrac{\mu}{2+\mu T}, & \rho=1.\end{cases}$$

LIFO-TO:

$$V_I=\lambda p_0+\lambda(1-p_0)\int_0^T e^{-\alpha t}b(t)\ dt.$$

To evaluate the integral we define

$$I(u)=\int_0^u e^{-\alpha t}b(t)\ dt$$

with

$$\tilde{I}(\theta)=L[I(u)]=\frac{1}{\theta}L[e^{-\alpha t}b(t)]=\frac{1}{\theta}\tilde{b}(\theta+\alpha),$$

where $\tilde{b}(\theta)$ is given by (2.5.2), and numerically invert $\tilde{I}(\theta)$ at $u = T$. Thus,

$$V_I=\lambda p_0+\lambda(1-p_0)L^{-1}\left[\frac{1}{\theta}\tilde{b}(\theta+\alpha)\right]_{u=T}. \tag{3.5}$$

For Case II we clearly have

$$V_{II}=\lambda P_S F_S(\tau).$$

IV. Numerical Results

In this section we present system performance measures for each of the queueing disciplines studied in overload. Specifically we present numerical results for tails of delay distributions, throughput–delay tradeoffs, and the effect of customers turning "bad."

In Fig. 1 we show the delay distribution tail result for the overload case $\rho = 1.5$ where the unit of time is the mean service time μ^{-1}. For the purpose of these comparisons, the timeout parameters have been chosen to match the throughput with that of the finite buffer schemes. We observe that the LIFO-TO and LIFO-PO schemes are comparable over the entire range and give the best performance up to $t = 7$ (the particular t of interest may correspond to a delay criterion), with FIFO-PO and FIFO-TO giving the next best performances, respectively, in this range. For larger t, as expected, the long tails of the LIFO schemes dominate. The closeness of the LIFO results for the timeout and pushout schemes indicates no strong need for timing customers in queue if pushout implementation is simpler.

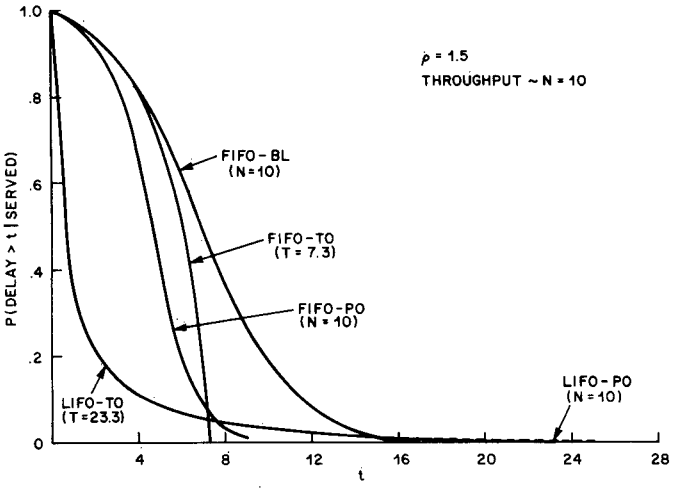

Fig. 1. Delay distribution comparisons.

The mean delays of served customers,

LIFO-TO	LIFO-PO	FIFO-PO	FIFO-TO	FIFO-BL
1.69	1.73	4.6	5.44	7.18

have the same ordering as the delay tail results for $t < 7$, and present clear choices if a mean delay criterion exists.

This type of comparison can be extended by considering the throughput–mean delay tradeoff comparisons generated by varying the control parameters (T, N) to further limit the traffic and resulting mean delays. This is shown in Fig. 2, where we see the previous ordering of mean delays preserved. In cases where the tail of the delay distribution is important, the corresponding throughput–delay tail tradeoffs would provide the needed comparisons.

Fig. 3 shows distribution results for $\rho = 0.9$ with throughput corresponding to $N = 10$ and timeouts adjusted to match this. With less control here, the FIFO results are clustered together, with the LIFO results exhibiting their characteristically longer tail behavior. The corresponding mean delays for this example are given by

LIFO-TO	LIFO-PO	FIFO-PO	FIFO-TO	FIFO-BL
2.59	2.73	3.28	3.38	3.65

which show a much tighter range than the $\rho = 1.5$ overload case. The larger mean delays, relative to $\rho = 1.5$, for the LIFO schemes are explainable by the fact that as ρ gets large, customers who do not get served quickly are more likely to get pushed or timed out. The FIFO-PO scheme also exhibits a peaking in the *mean delay* of served customers as a function of ρ [5], since the rate at which a waiting customer changes position in the queue increases with λ due to arriving customers to a full buffer, pushing down customers in queue.

Comparisons for a larger buffer, $N = 20$, are shown in Fig. 4, where we see more dramatic differences with the LIFO schemes (which are indistinguishable) outperforming the FIFO schemes over a much wider range of t. This is explainable by the fact that under heavy loads, served LIFO customers get service almost immediately, whereas served FIFO customers must still step down the entire queue before entering service with larger N, resulting in more time in queue. The mean delays of served customers also show a dramatic effect,

LIFO-TO	LIFO-PO	FIFO-PO	FIFO-TO	FIFO-BL
1.99	1.99	11.26	13.41	17.01

In the previous examples we compared the delay perform-

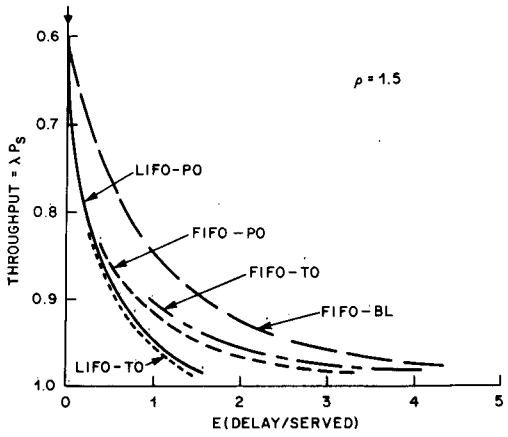

Fig. 2. Throughput-mean delay tradeoffs.

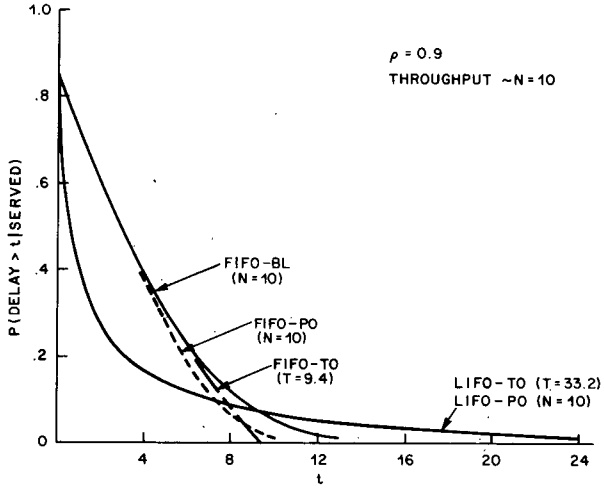

Fig. 3. Delay distribution comparisons.

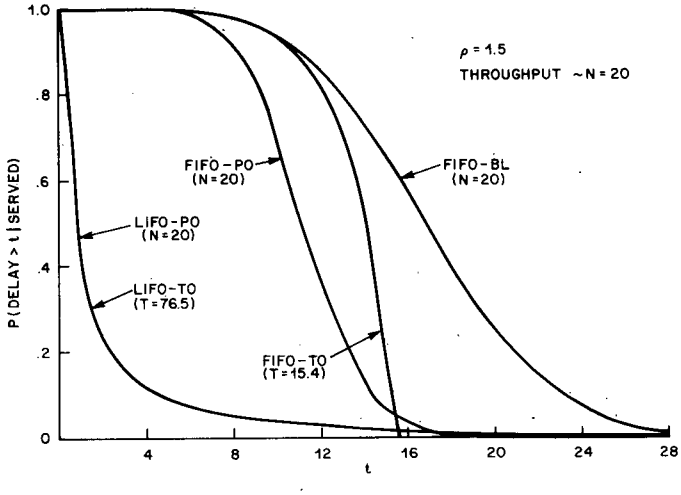

Fig. 4. Delay distribution comparisons.

ance of the various queueing disciplines when the throughputs were matched. Here we look at the goodput performance for each scheme, where we choose the control parameters to maximize the goodput, as a function of the severity of the overload. Fig. 5 shows the results for Case I where the average time a customer remains good is $\alpha^{-1} = 15$ service times. We see the superiority of the LIFO schemes here with LIFO-TO providing a 13 percent larger goodput than FIFO-

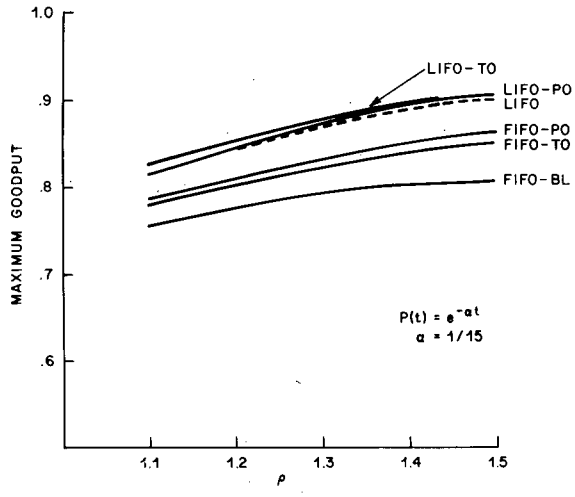

Fig. 5. Maximum goodput comparisons—case I.

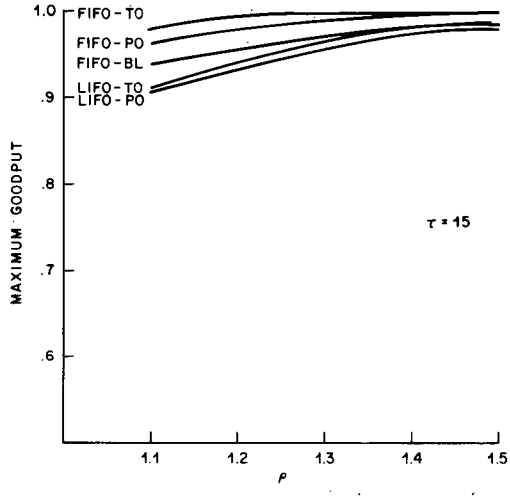

Fig. 6. Maximum goodput comparisons—case II.

BL at $\rho = 1.5$. Also shown in the figure is the result for a pure LIFO scheme which is known to be the optimum [1] *work conserving* discipline, and note the slight goodput improvement of the LIFO-TO scheme due to rejecting customers.

In Fig. 6 we show the maximum goodput results for Case II, which corresponds to the case where customers turn bad exactly τ time units after arrival. We choose $\tau = 15$ which gives the same average time for a customer in queue remaining good as Case I. The FIFO-TO discipline, with $T = \tau$, is optimum [14] in terms of maximizing the goodput; however, it should be noted that the mean time spent by a customer in queue who enters service (good or bad) is large (13 service times), whereas the FIFO-PO scheme can achieve close to maximum goodput (for $\rho \geq 1.3$) with approximately half the time in queue. The LIFO-schemes, which have excellent delay performance, have more limited goodput for this case. In heavy overload ($\rho = 1.5$) the 1.5–2 percent goodput reduction can be traded off by a 4:1 reduction in the average time a customer (good or bad) spends in queue for the LIFO-TO case.

These results clearly indicate the strong dependence of the goodput on the mechanism for customers turning bad. If the mechanism corresponds to sending machine timeouts, then the FIFO-TO and PO schemes are desirable from the goodput point of view, assuming the larger mean delays are acceptable. If mean delay is more important, then the LIFO schemes are

attractive. If the mechanism for customers turning bad is customer abandonments, then the results depend on the nature of $P(t)$, with LIFO schemes attractive from both the goodput and delay points of view for the convex $P(t)$ of Case I. A way of minimizing the effect of customers turning bad is to have the system check for the goodness of the customer before performing its work.

V. Discussion

We have put together a mixture of classical results and new results (particularly the FIFO-PO and goodput results) for the purpose of making comparisons of the overload performance of several queueing, service, and buffer management schemes. We have observed a dramatic effect of the control scheme on performance resulting in, for example, up to almost an order of magnitude difference in mean delays experienced by served customers. In an environment where customers remain "good," the LIFO schemes perform well and we note that there is no significant effect of timing customers in the queue relative to the LIFO pushout buffer management scheme. On the other hand, under FIFO the pushout buffer management actually performs better (under a mean delay criterion) than timing customers. If, however, the environment is such that customers can turn "bad," then the comparisons depend strongly on the distribution of time that a customer remains "good." If this distribution is exponential, which could represent customer behavior, then the LIFO schemes continue to perform well. As a matter of fact, the LIFO-TO scheme performs better than the optimal *work-conserving* queueing discipline (in the sense of maximizing the throughput of successful services). If, on the other hand, the time at which customers turn bad is deterministic, which could result from the service request coming from an upstream switch with a timeout mechanism, the FIFO-TO scheme performs best in the sense of maximizing the throughput of successful services. These results clearly indicate the importance of knowledge of the environment when selecting an overload control strategy.

Appendix

Level Crossing Analysis [8] of the FIFO-TO Control Scheme

As far as the probability of getting served, the throughput and the waiting time for the customers who get served are concerned, the FIFO-TO scheme is equivalent to the following scheme.

Let x_t denote the work in the system (processor + buffer) at time t. If an arrival occurs at time t and $x_t < T$, then it joins the buffer. It will be served and its waiting time will be x_t. If, on the other hand, an arrival at time t sees $x_t > T$, then it will leave.

Let J denote the distribution of the work in the equivalent system. Then

$$P_S = P\{x < T\} = J(T)$$

$$F_S(0) = \frac{J(0)}{P_S}$$

$$f_S(t) = \frac{J'(t)}{P_S} = \frac{j(t)}{P_S}, \qquad 0 < t < T$$

and for $n \geq 1$

$$M_n = \frac{\int_0^\infty t^n j(t)\ dt}{P_S}.$$

Thus, it suffices to obtain $J(0)$ and $j(t)$, $0 < t < \infty$. By the level crossing arguments [8]

$$j(t) = \lambda \int_0^t j(y) e^{-\mu(t-y)}\ dy + \lambda J(0) e^{-\mu t}, \qquad (0 < t < T) \tag{A.1}$$

and

$$j(t) = \lambda \int_0^T j(y) e^{-\mu(t-y)}\ dy + \lambda J(0) e^{-\mu t}, \qquad (t \geq T). \tag{A.2}$$

Here, (A.1) and (A.2) result from balancing the rate of down-crossings of the work level t with the rate of up-crossings of the level t. We also have

$$J(0) + \int_{0^+}^\infty j(t)\ dt = 1. \tag{A.3}$$

Equations (A.1)–(A.3) can be solved easily to obtain

$$J(0) = \frac{A}{\lambda}$$

$$j(t) = Ae^{-(\mu-\lambda)t} \qquad 0 < t < T$$

$$j(t) = Ae^{\lambda T} e^{-\mu t} \qquad T \leq t < \infty$$

where, for $\rho \neq 1$,

$$A = \frac{\lambda(1-\rho)}{1-\rho^2 e^{-(\mu-\lambda)T}}.$$

Thus,

$$P_S = J(0) + \int_{0^+}^T j(t)\ dt$$

$$= \frac{1-\rho e^{-(\mu-\lambda)T}}{1-\rho^2 e^{-(\mu-\lambda)T}}, \qquad \rho \neq 1.$$

Also,

$$M = M_1 = \frac{\int_0^T tj(t)\ dt}{P_S}$$

$$= \frac{1}{P_S}\left[\frac{\rho(1-e^{-(\mu-\lambda)T} - \mu T(1-\rho)e^{-(\mu-\lambda)T})}{\mu(1-\rho)(1-\rho^2 e^{-(\mu-\lambda)T})}\right]$$

$$= \frac{\rho[1-e^{-(\mu-\lambda)T} - \mu T(1-\rho)e^{-(\mu-\lambda)T}]}{\mu(1-\rho)(1-\rho e^{-(\mu-\lambda)T})}, \qquad \rho \neq 1.$$

Finally, for $\rho \neq 1$,

$$F_S(0) = \frac{J(0)}{P_S} = \frac{1-\rho}{1-\rho e^{-(\mu-\lambda)T}}$$

$$f_S(t) = \frac{j(t)}{P_S} = \frac{\lambda(1-\rho)e^{-(\mu-\lambda)t}}{1-\rho e^{-(\mu-\lambda)T}} \qquad 0 < t < T \tag{A.4}$$

$$F_s(t) = \begin{cases} \dfrac{1-\rho e^{-(\mu-\lambda)t}}{1-\rho e^{-(\mu-\lambda)T}} & 0 \leq t \leq T \\ 1 & t > T. \end{cases}$$

We note that these delay results are obtained in [7] using less direct methods and that the level crossing approach is extendable to general service time distributions [11].

For $\rho = 1$, we obtain

$$A = \frac{\mu}{2+\mu T}$$

$$P_S = \frac{1+\mu T}{2+\mu T}$$

$$M = M_1 = \frac{\mu}{1+\mu T} \cdot \frac{T^2}{2}$$

$$F_s(t) = \begin{cases} \dfrac{1+\mu t}{1+\mu T} & 0 \le t \le T \\ 1 & t \ge T. \end{cases}$$

References

[1] B. T. Doshi and E. H. Lipper, "Comparison of service disciplines in queueing systems with delay dependent behavior," in *Applied Probability-Computer Science: The Interface*, Vol II, R. L. Disney and T. J. Ott, Eds. Cambridge, MA: Birkhauser, 1982, pp. 269-301.

[2] J. Borchering, L. J. Forys, A. A. Fredericks, and G. Hejny, "Coping with overload," *Bell Lab. Rec.*, July-Aug. 1981.

[3] B. T. Doshi and H. Heffes, "Analysis of overload control schemes for a class of distributed switching machines," in *Proc. 10th Int. Teletraffic Congr.*, Montreal, P.Q., Canada, 1983.

[4] D. Gross and C. M. Harris, *Fundamentals of Queueing Theory*. New York: Wiley, 1974.

[5] B. T. Doshi and H. Heffes, "Comparison of control schemes for a class of distributed systems," in *Proc. 21st IEEE Conf. Decision Contr.*, Orlando, FL, Dec. 8-10, 1982, pp. 846-853.

[6] P. Kuehn, "On a combined delay and loss system with different queue disciplines," in *Trans. 6th Prague Conf. Inform. Theory, Statist. Decision Functions, Random Processes*, Czechoslovak Acad. Sci., Prague, 1973, pp. 501-528.

[7] B. V. Gnedenko and I. N. Kovalenko, *Introduction to Queueing Theory*, Israel Program for Sci. Transl., Jerusalem, 1968.

[8] P. H. Brill and M. J. M. Posner, "Level crossings in point processes applied to queues: Single server case," *Oper. Res.*, vol. 25, pp. 662-674, July-Aug. 1977.

[9] D. L. Jagerman, "An inversion technique for the Laplace transform with application to approximation," *Bell Syst. Tech. J.*, vol. 57, pp. 669-710, Mar. 1978.

[10] D. M. Lucantoni and Y. C. Jenq, unpublished work.

[11] H. Heffes, "Analysis of overload performance for a class of *M/D/1* processor queueing disciplines," in *Proc. 11th Int. Teletraffic Congr.*, Kyoto, Japan, Sept. 1985, paper 2.1B-1.

[12] L. Kleinrock, *Queueing Systems, Vol. 1: Theory*. New York: Wiley, 1975.

[13] L. Kleinrock, *Queueing Systems, Vol. 2: Computer Applications*. New York: Wiley, 1976.

[14] B. Doshi, unpublished work.

★

Bharat T. Doshi (M'83) received the B. Tech. degree in mechanical engineering from the Indian Institute of Technology, Bombay, in 1970, and the M.S. and Ph.D. degrees in operations research from Cornell University, Ithaca, NY, in 1973 and 1974, respectively.

He was an Assistant Professor at Rutgers University, New Brunswick, NJ, from 1974 to 1979. He joined AT&T Bell Laboratories, Holmdel, NJ, in 1979, and was promoted to Supervisor in the Performance Analysis Department in March 1982. His recent technical work includes analysis of processor schedules, protocols, flow and congestion controls for a variety of AT&T products. His research interests are queueing theory and scheduling theory applied to performance analysis of computer, communications, and production systems.

Dr. Doshi is a member of the Operations Research Society of America, and Associate Editor of the journals *OR Letters* and *Queueing Systems*.

★

Harry Heffes (M'66-SM'82) received the B.E.E. degree from the City College of New York, New York, NY, in 1962, and the M.E.E. and Ph.D. degrees in electrical engineering from New York University, New York, in 1964 and 1968, respectively.

He joined AT&T Bell Laboratories, Whippany, NJ, in 1962 in the Military Systems Area, where he applied modern control and estimation theory results to problems relating to guidance, navigation, tracking, and trajectory optimization. Since 1973 his primary concern has been with the modeling and analysis of teletraffic processes and systems. Most recently he has been concerned with the performance analysis of computer based systems and services, including digital switching systems. He is the author of over 20 papers in such areas as Kalman filtering, control system theory, approximation theory, communication theory, air traffic control, teletraffic theory, queueing theory, simulation, switching systems, data traffic, overload control, communication network survivability, and integrated voice/data systems. He is also an Adjunct Professor of Computer Science at Stevens Institute of Technology, Hoboken, NJ.

Dr. Heffes received the AT&T Bell Laboratories Distinguished Technical Staff Award in 1983. He is a member of Tau Beta Pi, Eta Kappa Nu, American Men and Women of Science, the Operations Research Society of America, and the Association for Computing Machinery.