Optimality of the Stochastic Earliest Deadline Policy for the G/M/c Queue Serving Customers with Deadlines

Don Towsley *
Dept. of Computer & Information Science
University of Massachusetts
Amherst, MA 01003, USA

S.S. Panwar †
Dept. of Electrical Engineering
Polytechnic University
Brooklyn, NY 11201, USA

September 30, 1991

Abstract

We consider the problem of scheduling customers with deadlines in the G/M/c queue. We assume that customer deadlines are not known but that the scheduling policy has available to it partial information regarding stochastic relationships between the deadlines of eligible customers. We prove three main results, 1) in the case that deadlines are until the beginning of service, the nonpreemptive, non-idling policy that stochastically minimizes the number of customers lost during an interval of time belongs to the class of non-idling *stochastic earliest deadline* (SED) policies, 2) in the case that deadlines are until the end of service, the optimum policy belongs to the class of SED policies, and 3) in the case of deadlines until the end of service, the optimum non-preemptive policy belongs to the class of non-preemptive, non-idling SED policies. The last result assumes that a customer in service that misses its deadline is always removed and thrown away. Here a policy belongs to the class of SED policies if it never schedules a customer whose deadline is known to be stochastically larger than that of some other customer in the queue. We describe several applications for which these classes contain exactly one policy. These include queues where *i)* deadlines are known exactly, *ii)* deadlines are characterized by distributions with increasing failure rate, *iii)* deadlines are characterized by distributions with decreasing failure rate, *iv)* customers fall into several classes, each with its own exponential deadline distribution, and *v)* certain combinations of the previous three applications. The optimal policies for the first four applications are the earliest deadline first first come first serve, last come first serve, and head of the line priority scheduling. The scheduling policy for the last application combines elements of head of the line priority, first come first serve, and last come first serve. The paper concludes with some generalizations to discrete time systems, finite buffers and vacation models.

Keywords: Earliest deadline scheduling, FCFS, G/M/c, LCFS, Optimal scheduling, Priority scheduling, Real-time constraints

*The work of this author was supported by the Office of Naval Research under grant N00014-87-K-0796.

†The work of this author was supported by the Center for Advanced Technology in Telecommunications, Polytechnic University, and by the NSF under grant NCR-8909719.

1 Introduction and Summary

In this paper we consider the problem of scheduling customers with deadlines on multiple server queues when there is only partial information available to the scheduling policy regarding customer deadlines. We assume that scheduling policies do not know either future customer arrival times or customer service times, but have available to them information regarding deadlines in the form of an acyclic directed graph representing a partial ordering of the customers. The presence of an edge in the graph from one customer to another indicates that the deadline of the first customer is stochastically larger than that of the second customer. We prove that the policy that stochastically minimizes the number of customers missing their deadlines by time t *never schedules a customer which is known to have a deadline that is stochastically larger than that of another customer also in the queue.* We refer to this as the class of *stochastically earliest deadline* (SED) policies.

Our results hold for systems in which deadlines are until the beginning of service as well as for systems in which deadlines are until the end of service. In the first variation, we consider non-preemptive policies whereas in the second variation we have results for the class of policies that allow preemption as well. Most of our results require that customer service times form an i.i.d. sequence of r.v.'s independent of arrival times and deadlines.

If the information graph induces a total ordering among the jobs, then the optimal policy is well defined. A total ordering occurs in five cases,

- all deadlines are known to the policy,
- the distribution of *relative deadlines* (times from arrivals to deadlines) has increasing failure rate (IFR),
- the distribution of relative deadlines has decreasing failure rate (DFR),
- K classes of jobs, each with a relative deadline that is an exponential r.v. with a mean that depends on the class,
- a combination of the last three.

The optimal policy in the first four cases are *i)* the earliest deadline scheduling policy, *ii)* first come first serve (FCFS), *iii)* last come first serve (LCFS), and *iv)* fixed priority head of the line (HOL) policy. The optimal policy for the last example combines features from FCFS, LCFS, and a fixed priority HOL policy as will be explained later.

Situations in which policies do not have detailed deadline information arise in reliability problems. Here the deadline may correspond to the time that a component will fail and the servers are responsible for performing preventive maintenance. The problem is to determine the order in which to schedule maintenance of arriving components so as to minimize the probability of component failure. Similar situations occur in medicine.

Several papers have considered the problem of scheduling customers with known deadlines Panwar [5], Bhattacharya and Ephremides [2], Panwar, et al. [6] and Saito [8] in which the earliest deadline policies have been shown to be optimal. These results are applicable to soft-real-time computer systems and to packet switched networks carrying voice or video packets with known time constraints. Also contained in [2] is the proof that the FCFS policy stochastically minimizes the number of customers that miss their deadlines when deadlines are not known but the but the distribution of the relative deadlines has IFR. This was done for the case of one server and when deadlines are until the beginning of service. All of the above results follow as applications of the results contained in this paper.

A number of papers have also treated the problem of scheduling customers with deadlines in systems where they must all be completed. Here typical results are that earliest deadline policies minimize the job tardiness in the sense of convex ordering for single servers, multiple servers, tandem queueing systems, and a variety of parallel processing systems Kallmes, et al. [3], Towsley and Baccelli [10], and Baccelli, etal. [1].

The paper is organized as follows. The next section introduces our system model and our model for partial deadline information. Section 3 includes our results for systems in which deadlines are until the beginning of service. Section 4 deals with systems where the deadlines are until the end of service. Last, Section 5 concludes with generalizations to discrete time systems, systems with finite buffers, and systems in which servers take vacations.

2 Model Description

We consider a G/M/c queue in which customers have deadlines. We treat two variations, queues in which deadlines are until the beginning of service and queues in which deadlines are until the end of service. Let $A_1 < A_2 < \cdots$ where A_i denotes the arrival time of the i-th customer, c_i, $i = 1, \cdots$ and $\tau_i = A_i - A_{i-1}$ denotes the i-th interarrival time, $i = 1, 2, \cdots$. Let $\{S_i\}_{1 \le i}$ be a sequence of random variables where S_i denotes the service time of the i-th customer Let $\{R(\theta), \theta \in \mathcal{S}\}$ be a family of random variables parameterized by an index $\theta \in \mathcal{S}$ where $\mathcal{S}$ is *an index set.* Let $\{\Theta_i\}_{1 \le i}$ be a sequence of r.v.'s where the deadline of the i-th customer is given by $D_i = A_i + R(\Theta_i)$, $i = 1, \cdots$. Here $R(\Theta_i)$ is referred to as the *relative deadline for customer i.* When there is no ambiguity, R_i will represent $R(\Theta_i)$.

We shall use the notation $\boldsymbol{A} = \{A_i\}_{1 \le i}^{\infty}$, $\boldsymbol{\Theta} = \{\Theta_i\}_{1 \le i}^{\infty}$, $\boldsymbol{S} = \{B_i\}_{1 \le i}^{\infty}$, and $\boldsymbol{B} = (\boldsymbol{A}, \boldsymbol{\Theta}, \boldsymbol{S})$, In addition, whenever we focus on a specific sample realization of the above r.v.'s, we shall use lower case notation (i.e., a_i for A_i, etc ...). Furthermore, we shall let $\boldsymbol{a} = \{a_i\}_{1 \le i}$, $\boldsymbol{s} = \{b_i\}_{1 \le i}$, $\boldsymbol{\theta} = \{\theta_i\}_{1 \le i}$. Last, let $\boldsymbol{b} = (\boldsymbol{a}, \boldsymbol{s}, \boldsymbol{\theta})$. This quantity will be referred to as the input sample.

Assumption A_0: *$\{S_i\}_1^{\infty}$ is an independent and identically distributed (i.i.d.) sequence of r.v.'s independent of $\{A_i\}$ and $\{\Theta_i\}$ and $\{R_i\}_1^{\infty}$ is an independent sequence of r.v.'s given that $\boldsymbol{\Theta} = \boldsymbol{\theta}$.*

Assumption A_1: *$\{S_i\}_1^{\infty}$ is an independent and identically distributed (i.i.d.) sequence of exponential r.v.'s independent of $\{A_i\}$ and $\{\Theta_i\}$ and $\{R_i\}_1^{\infty}$ is an independent sequence of r.v.'s*

given that $\Theta = \theta$.

Assumption A_2**:** $\{S_i\}_1^\infty$ *is an independent and identically distributed (i.i.d.) sequence of r.v.'s,* $\{\tau_i\}$ *is an i.i.d. sequence of exponential r.v.'s with mean* $E[\tau]$*, and* $\{\Theta_i\}$ *is an i.i.d. sequence of r.v.'s, and* $\{R_i\}_1^\infty$ *is an independent sequence of r.v.'s given that* $\Theta = \theta$*. Last, the sequences are mutually independent.*

Let us consider the deadline model more closely. Let $R(\theta, t)$ denote the value of $R(\theta)$ given that it exceeds $t > 0$. Let $Y(\theta, t) = R(\theta, t) - t$. We will use the notation $Y_i(t) = Y(\theta_i, t)$ and it should be understood that $R_i = Y_i(0) = R(\theta_i, 0)$.

Definition 1 *Let* X *and* Y *be real valued r.v.'s.* X *is stochastically larger than* Y *(written* $Y \leq_{st} X$*) iff*

$$\Pr[X < x] \leq \Pr[Y < x], \quad -\infty < x < \infty.$$

Information regarding customer deadlines is available to a policy in the form of a *partial order* between customers represented by a directed graph. Let $\{H_{i,j}\}_{i,j=1}^\infty$ be a sequence of r.v.'s where $H_{i,j}$ is the time at which information regarding the relationship (if any) between the i-th and j-th customers becomes available to a scheduling policy. Specifically, whenever $t \geq H_{i,j}$ and either $Y_j(t - A_j) \leq_{st} Y_i(t - A_i)$ or $Y_j(t - A_j) \geq_{st} Y_i(t - A_i)$ then the presence and type of this relationship is available to the scheduling policy. The r.v. $H_{i,j}$ exhibits several properties,

- $\min(\max(A_i, A_j), D_i, D_j) \leq H_{i,j} \leq \min(D_i, D_j)$,
- $H_{i,j} = H_{j,i}$

In addition, we assume without loss of generality that, if $H_{i,j} = t_0$, $H_{j,k} \leq t_0$, $Y_j(t_0 - A_j) \leq_{st} Y_i(t_0 - A_i)$, and $Y_k(t_0 - A_k) \leq_{st} Y_j(t_0 - A_j)$, then $H_{i,k} = t_0$. Similarly, if $H_{i,j} = t_0$, $H_{i,k} \leq t_0$, $Y_j(t_0 - A_j) \leq_{st} Y_i(t_0 - A_i)$, and $Y_i(t_0 - A_i) \leq_{st} Y_k(t_0 - A_k)$, then $H_{k,j} = t_0$.

Let π be a policy that determines what customer in the queue is to be executed (if any) whenever a server is free. This policy makes its decision based on the customers that are eligible for service as well as on the past history of the system. Policy π has no service time information available but has information on the deadlines available in the form of a graph $G_\pi(t) = (V_\pi(t), E_\pi(t))$ where $V_\pi(t)$ is the set of *eligible customers*, i.e., the customers that have neither made nor missed their deadlines at time t. The set of edges is defined to be

$$E_\pi(t) = \{(i, j) : i, j \in V_\pi(t); (H_{i,j} \leq t) \wedge (Y_j(t - A_j) \leq_{st} Y_i(t - A_i))\}.$$

This graph is updated whenever a customer arrives, completes service, misses its deadline, or makes its deadline, or whenever the information that $Y_j \leq_{st} Y_i$ is made available to the policy, i.e., at time $H_{i,j}$, $\forall i, j$, or whenever the information is changed thereafter. This graph satisfies several properties.

- If $(i, j), (j, k) \in E_\pi(t)$ for some $i, j, k > 1$, then $(i, k) \in E_\pi(t)$. This is because $\leq_{st}$ is a transitive relation.

- If $V_{\pi_1}(t) \cap V_{\pi_2}(t) \neq \emptyset$, then $i, j \in V_{\pi_1}(t) \cap V_{\pi_2}(t)$ implies $(i,j) \in E_{\pi_1}(t)$ iff $(j,i) \in E_{\pi_2}(t)$, i.e., all policies have the same information regarding deadlines.

We are interested in the following performance metric: $L_\pi(t)$, the number of customers that miss their deadlines by time $t > 0$ under π.

We are interested in the following classes of policies.

- Σ_0 - the class of non-preemptive policies that operate on a system with deadlines to the beginning of service. Policy $\pi \in \Sigma_0$ cannot use service time information but can use deadline information found in $G_\pi(t)$.
- Σ_1 - the class of preemptive policies that operate on a system with deadlines to the end of service. Policy $\pi \in \Sigma_1$ has no service time information available to it but has deadline information available to it in the form of $G_\pi(t)$. Policies can abort customers that miss their deadlines while in service.
- Σ_2 - the class of non-preemptive policies that operate on a system with deadlines until the end of sevice. Policy $\pi \in \Sigma_2$ has no service time information available but has deadline information available to it in the form of $G_\pi(t)$. Policies can abort customers that miss their deadlines while in service.

The classes of policies Σ_0^{ni}, Σ_1^{ni}, Σ_2^{ni} are the subsets of Σ_0, Σ_1, Σ_2, respectively, that never idle a server while there are customers waiting for service. We are also interested in the following classes of policies

- $SED_k \subset \Sigma_k$, $k = 0, 1, 2$. Policy $\pi \in SED_k$ *never schedules* a customer c_i at time t if there exists a customer c_j such that $(i,j) \in E_\pi(t)$. It may choose to idle a server when there is work in the queue.
- $SED_k^{ni} \subset SED_k$, $k = 0, 1, 2$. Policy $\pi \in SED_k^{ni}$ is a policy in SED_k that never idles a server while there is work in the queue.

We conclude this section with a property of "$\leq_{st}$" that will be very useful in proving our results.

Lemma 1 (Strassen [9]) *Let $\{X_i\}_{i=1}^n$ and $\{Y_i\}_{i=1}^n$ be two sequences of independent r.v.'s. If $X_i \leq_{st} Y_i$, $i = 1, \cdots, n$, then there exist two sequences of r.v.'s $\{\tilde{X}_i\}_{i=1}^n$ and $\{\tilde{Y}_i\}_{i=1}^n$ such that*

1. *both are sequences of independent r.v.'s and*
2. $\Pr[\tilde{X}_1 \leq \tilde{Y}_1, \cdots, \tilde{X}_n \leq \tilde{Y}_n] = 1$, *almost surely.*

3 Deadlines to the Beginning of Service

In this section we prove that for any policy $\pi \in \Sigma_0^{ni}$ there exists a policy $\pi^* \in SED_0^{ni}$ that performs at least as well when deadlines are until the beginning of service. Here $V_\pi(t)$ contains customers that are waiting in the queue but not in service. For any policy π, we define $N_\pi(t)$ to be the number of occupied servers under policy π at time $t > 0$. Without loss of generality, we assume that the first $N_\pi(t)$ servers are always occupied.

Lemma 2 *Let $\pi \in \Sigma_0^{ni}$ deviate from a SED_0^{ni} policy for the first time at time t_0 for a given input sample $\boldsymbol{b}$. There exists another policy $\pi^* \in \Sigma_0^{ni}$ that deviates from a SED_0^{ni} policy for the first time at $t_1 > t_0$ and for which*

$$L_{\pi^*}(t) \leq_{st} L_\pi(t), \quad \forall t > 0$$

under assumption $\boldsymbol{A}_1$.

Proof. Assume that π deviates from a SED_0^{ni} policy for the first time at $t = t_0$ for some sequence of deadlines $D_1, \cdots$ by scheduling c_i when $(i,j) \in E_\pi(t_0)$, $(j,k) \notin E_\pi(t)$, $\forall k \in V_\pi(t_0)$. We define policy π^* as follows.

- at time $t = t_0$, π^* schedules c_j,
- π^* schedules c_i whenever π schedules c_j,
- π^* does nothing when π schedules a customer, say c_l and at that time all servers are busy under π^*,
- if $V_\pi(t) = \emptyset$, $V_{\pi^*}(t) \neq \emptyset$ and a service completion occurs under both policies, then π^* schedules an arbitrary customer from $V_{\pi^*}(t)$,
- otherwise π^* emulates π.

Policy π^* can be defined in this way provided that we can couple both systems so that one of the following four conditions is satisfied for all $t > t_0$. These are

1. $V_\pi(t) = V_{\pi^*}(t)$, $N_\pi(t) = N_{\pi^*}(t)$,
2. $V_\pi(t) - \{c_j\} = V_{\pi^*}(t) - \{c_i\}$, $N_\pi(t) = N_{\pi^*}(t) = c$,
3. $V_{\pi^*}(t) = V_\pi(t) + \{c_l\}$ for some customer c_l, $N_\pi(t) = N_{\pi^*}(t) = c$, or
4. $V_{\pi^*}(t) = V_\pi(t) = \emptyset$, $N_\pi(t) = N_{\pi^*}(t) - 1$.

For the coupling that we will describe below, we will prove that the system always satisfies one of these conditions by induction on the times that important events occur. These events are

- $\mathcal{E}_0$ - arrival to both systems,

- $\mathcal{E}_1$ - completion of a customer in either or both systems,
- $\mathcal{E}_2$ - customer missing deadline under one or both policies,

In order to guarantee that departures occur simultaneously on both systems, we find it useful to assign fictitious customers to servers that are idle and to couple the service completions on each server under both policies. Furthermore, whenever a customer receives service simultanously in both systems, we assign it to the same server so that it will depart at the same time. When an arriving customer is assigned immediately to an idle server, it is given the remaining service time associated with that server. The assumption that times betwen service completions are exponentially distributed r.v.'s guarantees that the service time received by this customer is exponentially distributed. In addition, following the scheduling of π and π^* at time $t = t_0$, we invoke lemma 1 to couple the deadlines of the customers under both policies as follows: customer c_l, $l \neq i, j$ has the same deadline, $D_l^* = D_l$ under both policies; customer c_i under π^* has deadline $D_i^* \leq D_j$. Here D_l denotes a deadline for the system under π and D_l^* under π^* The assumption that $\{R_i\}_1^\infty$ is an independent sequence when conditioned on the values of Θ_i implies that this coupling can be done without affecting either the joint statistics of R_i and the deadlines for all customers other than c_j or the joint statistics of R_j and the deadlines of all other customers except c_i.

Let $(t_0, \sigma_0), (t_1, \sigma_1), \cdots$ be the sequence of times and events that occur at those times, i.e., event σ_i occurs at time t_i where $\sigma_i \in \{\mathcal{E}_0, \mathcal{E}_1, \mathcal{E}_2\}$. We proceed with our inductive argument.

Basis Step: The hypothesis is trivially true for $t = t_0$, as condition 1 holds.

Inductive Step: Assume that one of the four conditions holds for $t_l, \cdots, t_n$. We now show that either it or one of the other conditions will hold at $t = t_{n+1}$. There are three cases according to the type of event, σ_{n+1}.

Case 1 ($\sigma_{n+1} = \mathcal{E}_0$): In this case, the conditions follow trivially from the inductive hypothesis. For example, if condition 1 holds at $t = t_n$, then it also holds at $t = t_{n+1}$. Similarly for conditions 2 and 3. In the case of condition 4, if $N_{\pi^*}(t_n) = c$ then condition 3 holds at $t = t_{n+1}$; otherwise condition 4 continues to hold.

Case 2 ($\sigma_{n+1} = \mathcal{E}_1$): Again, keeping in mind how the departures are coupled in both systems, the conditions follow easily from the inductive hypothesis.

Case 3 ($\sigma_{n+1} = \mathcal{E}_2$): If the same customer misses a deadline under both policies, then the conditions are easy to establish. If condition 2 holds at $t = t_n$ and customer c_j dies under π, then according to the way that the deadline of c_i and c_j are coupled, the system will satisfy condition 3 at $t = t_{n+1}$. If condition 3 holds and c_l dies under π^*, then the system will satisfy condition 1 at $t = t_{n+1}$.

This completes the induction portion of the proof. Clearly the number of completions by time t and the number of customers in the system at time t are each larger under π^* than under π. Hence, since the number of customers that arrive to the system under both policies are the same, we conclude that $L_\pi(t) \geq L_{\pi^*}(t)$. This argument can be conducted on all deadline sequences for

which π deviates from a SED_0^{ni} policy yielding the desired result. ∎

This lemma can be used to prove the following result.

Theorem 1 *For any policy $\pi \in \Sigma_0^{ni}$, there exists a policy $\pi^* \in SED_0^{ni}$ such that*

$$L_{\pi^*}(t) \leq_{st} L_\pi(t), \quad \forall t > 0$$

under assumption $\boldsymbol{A}_1$.

Proof. The proof consists of considering a specific sample path and constructing a sequence of policies $\pi = \pi_0, \pi_1, \cdots, \pi_i, \cdots$ such that $\pi^* = \lim_{i\to\infty} \pi_i$ is a SED_0^{ni} policy satisfying $L_{\pi^*}(t) \leq_{st} L_\pi(t)$, $\forall t > 0$. Specifically, lemma 2 can be used to construct this sequence that exhibits the property that the first time at which each deviates from a SED_0^{ni} policy is increasing in i and that exhibit the property that $L_{\pi_i}(t)$ is a stochastically nondecreasing function of i $\forall t > 0$. Removal of the conditioning on the sample path yields the desired result. ∎

Last, let $\mathcal{T}_\pi(t)$ denote the number of customers that complete under $\pi \in \Sigma_0^{ni}$ by time $t > 0$ and let $\mathcal{T}_\pi = \limsup_{t\to\infty} E[\mathcal{T}_\pi(t)/t]$ denote the *throughput* under policy π. We have the following result regarding the optimality of SED_0^{ni} policies with respect to the throughput for general service times.

Theorem 2 *If SED_0^{ni} contains a single policy, γ, then*

$$\mathcal{T}_\gamma \geq \mathcal{T}_\pi,$$

for all stationary policies $\pi \in \Sigma_0^{ni}$, for a single server queue provided that the arrival times, service times, and deadlines satisfy assumption $\boldsymbol{A}_2$.

Proof. As there is only one SED_0^{ni} policy, we shall refer to it as γ. If π is a stationary policy, then the system is regenerative with regeneration points corresponding to the times that the system empties. Let B_π^n and M_π^n denote the length of the n-th busy period and the number of customers served during the n-th busy period under stationary policy $\pi \in \Sigma_0^{ni}$. As $\{(B_\pi^n, M_\pi^n)\}_1^\infty$ is an i.i.d. sequence of r.v.'s, let (B_π, M_π) be the generic r.v. having the same distribution as (B_π^n, M_π^n). Clearly $E[B_\pi] < \infty$ and $E[M_\pi] < \infty$ since the number of customers in the system can be easily shown to be stochastically smaller than that of a M/G/∞ system with the same arrival rate and with service times corresponding to the relative deadlines. Hence, it follows from Ross [7, Theorem 3.6.1] that

$$\mathcal{T}_\pi = E[M_\pi]/E[B_\pi].$$

Similarly, we have $E[B_\pi] = E[M_\pi]E[S_\pi] + 1/\lambda$ which, when substituted into the previous expression yields

$$\mathcal{T}_\pi = 1/E[S_\pi] - (\lambda E[S_\pi]E[B\pi])^{-1}.$$

Hence, $\mathcal{T}_\pi$ is maximized by the policy that maximizes $E[B_\pi]$.

The proof of the theorem is completed by examining a single busy period under π and showing that it is stochastically smaller than a busy period under γ, i.e., $B_\pi^n \leq_{st} B_\gamma^n$, $\forall n > 0$. This can be done either by using the interchange arguments used in the previous results or by using a forward induction argument similar to the one used to prove theorem 2 in Panwar, etal. [6]. ∎

Remark. We conjecture that the above result holds for the more general class of nonstationary policies.

Applications: In general, SED_0^{ni} will not define a unique policy. However, there are several interesting systems for which it does define a single unique policy. These occur when $H_{i,j} = \min(\max(A_i, A_j), D_i, D_j)$ (i.e., the stochastic relationship between the relative deadlines of two customers in a system is available to the scheduler as soon as the last of the two customers arrives) and either

- $\mathcal{S} = I\!R^+$, $R(\theta) = \theta$, i.e., known deadlines,
- $\mathcal{S} = I\!R^+$, $R(0) = R(0,0)$ is a random variable whose distribution has increasing failure rate (IFR) and $R(\theta) = R(0,\theta)$, $\forall \theta \geq 0$, i.e., the customer have IFR deadlines and they can have an age θ at arrival,
- $\mathcal{S} = I\!R^+$, $R(0) = R(0,0)$ is a random variable whose distribution has decreasing failure rate (DFR) and $R(\theta) = R(0,\theta)$, $\forall \theta \geq 0$, i.e., the customer have DFR deadlines and they can have an age θ at arrival,
- $\mathcal{S} = \{1, 2, \cdots, K\}$, $R(k)$ is exponentially distributed with parameter $\mu_1 > \mu_2 > \cdots > \mu_K > 0$.

In the first case, SED_0^{ni} consists of the well known EDF policy. In the second case, SED_0^{ni} consists of the policy which, at time t, schedules the customer with the largest value of $\theta_i + t - a_i$ from among all of the eligible customers. Note that if Θ_i only takes on value 0, then this corresponds to the FCFS policy. In the third case SED_0^{ni} consists of the policy which, at time t, schedules the customer with the smallest value of $\theta_i + t - a_i$ from among all of the eligible customers. This corresponds to the LCFS policy when $\Theta_i = 0$ $\forall i$. In the last case, SED_0^{ni} consists of a head of the line (HOL) fixed priority policy with K priority classes where the priorities are assigned in decreasing order of the μ_k's.

A more complicated example can be constructed by combining the last three examples.

- $\mathcal{S} = \{1, 2, \cdots, K\} \times I\!R^+$,
- For $k = 1, \cdots, K$, $\theta \in \mathcal{S}$, $R((k,0)) = R((k,0),0)$ is a r.v. with a distribution having either IFR or DFR, and $R((k,y),0) = R((k,0),y)$, $y \in I\!R^+$, and
- $\inf_t \mu_k(t) \geq \sup_t \mu_{k+1}$, $k = 1, \cdots, K-1$, where $\mu_k(t)$ is the failure rate of $R((k,0))$ at time $t > 0$.

The policy in this case is a fixed priority HOL policy with K priority classes. Priority is given in decreasing value of the first coordinate of θ. Within each priority class, the customer with the largest value of $y_i + t - a_i$, where $\theta_i = (k, y_i)$, is used if the deadline distribution for that class has IFR and the customer with the smallest value of $y_i + t - a_i$, where $\theta_i = (k, y_i)$, is used if the deadline distribution has DFR.

We conclude this section with an additional result for the case where $\mathcal{S} = \{0\}$ and $R(0)$ is a r.v. with IFR. In this case, FCFS is the optimum policy from Σ_0.

Theorem 3 *If $\mathcal{S} = \{0\}$ and $R(0)$ has increasing failure rate, then*

$$L_{FCFS}(t) \leq_{st} L_\pi(t), \quad \forall \pi \in \Sigma_0; t > 0$$

under assumption $\boldsymbol{A}_0$ provided that the states are the same under FCFS and π.

Proof. The argument is similar to those given for lemma 2 and theorem 1. It consists of conditioning on the arrival times and service times and showing that idle periods can be removed with no decrease in the number of losses over time. The IFR property is required because it ensures that the longer a customer is left in the queue while a server is idle, the more likely it is to miss its deadline. ∎

This generalizes a result in Bhattacharya and Ephremides [2] established for constant deadlines.

Remark. No such result exists if deadlines have a distribution that is DFR. This should be clear as the situation can arise where there is one customer in the queue who has been in the system for a long time. In this case, there may be a benefit in waiting for a customer to arrive whose deadline may be stochastically smaller.

4 Deadlines Until the End of Service

In this section we consider systems where the deadlines are until the end of service. We state two results for two classes of policies, those allowing preemptions and those that do not.

Preemptive policies: In this environment, $V_\pi(t)$ contains all customers present in the system, both waiting in the queue as well as in service. As it is useful to distinguish between these two types of customers, let $C_\pi(t)$ be the set of all customers in service at time $t > 0$.

First we establish that the best policies out of the class of idling policies are non-idling.

Lemma 3 *For every policy $\pi \in \Sigma_1$, there exists a policy $\pi^* \in \Sigma_1^{ni}$ such that*

$$L_{\pi^*}(t) \leq_{st} L_\pi(t), \quad \forall t > 0$$

under $\boldsymbol{A}_0$.

Proof. Condition on a single sample path. Assume that π idles a server at time $t = t_0$ for the first time while there are customers in the queue and schedules a customer subsequently at $t = t_0' > t_0$.

We construct a new policy π' which schedules any customer, say c_j at $t = t_0$ and from $t = t_0'$, simulates π in the following way. If c_j neither completes nor misses its deadline by $t = t_0'$, it is preempted after which π' copies π exactly. If c_j misses its deadline by $t = t_0'$, then the server remains idle until $t = t_0'$ and π' copies π subsequently. Last, if c_j completes by $t = t_0'$ then π' copies π exactly except when π schedules c_j. In this case, π' does nothing.

It is clear that $L_\pi(t) \geq L_{\pi'}(t), \forall t \geq t_0$. Furthermore, this procedure can be repeated to construct a sequence of policies $\pi = \pi_0, \pi_1, \cdots$ such that $\pi_i \in \Sigma_1$ and π_i idles a server while there is work in the queue for the first time at $t = t_i$ where t_i is an increasing function of i. The sequence is either finite and $\pi^* = \pi_n$ where π_n is non-idling, or the sequence is infinite and $\pi^* = \lim_{i\to\infty} \pi_i$ which is also non-idling.

Removal of the conditioning on the sample path yields the desired result. ∎

Lemma 4 *Let $\pi \in \Sigma_1^{ni}$ deviate from a SED_1^{ni} policy for the first time at time t_0 for input sample* ***b****. There exists another policy $\pi^* \in \Sigma_1^{ni}$ that deviates from a SED_1^{ni} policy for the first time at $t_1 > t_0$ and for which*

$$L_{\pi^*}(t) \leq_{st} L_\pi(t), \quad \forall t > 0$$

under assumption $\boldsymbol{A}_1$*.*

Proof. Assume that π deviates from a SED_1^{ni} policy for the first time at time $t = t_0$ for a sequence of deadlines $D_1, \cdots$ by scheduling c_i where $(i,j) \in E_\pi(t_0)$, $(j,k) \notin E_\pi(t)$, $\forall k \in V_\pi(t_0)$. We define policy π^* as follows.

- at time $t = t_0$, π^* schedules c_j,
- the first time $t > t_0$ (if any) that (i,j) is removed from $E_{\pi^*}(t)$ and c_i and c_j remain in the system (this reflects changes in the stochastic relation between the deadlines of c_i and c_j), π^* reschedules c_i and c_j, if necessary so that their status is the same under π^* as under π.
- π^* schedules or preempts c_i whenever π schedules or preempts c_j,
- π^* does nothing when π schedules a customer say c_l, and at that time all servers are busy under π^*,
- if $V_\pi(t) - C_\pi(t) = \emptyset$, $V_{\pi^*}(t) - C_{\pi^*}(t) \neq \emptyset$ and a service completion occurs under the two policies, then π^* schedules an arbitrary customer from $V_{\pi^*}(t)$,
- otherwise π^* emulates π.

Policy π^* can be defined in this way provided we can couple the systems under the two policies in such a way that one of the following three conditions is satisfied for all $t > t_0$. These are

1. $V_\pi(t) = V_{\pi^*}(t)$, $C_\pi(t) = C_{\pi^*}(t)$,

2. $V_\pi(t) - \{c_j\} = V_{\pi^*}(t) - \{c_i\}$, and either $C_\pi(t) = C_{\pi^*}(t)$, or $C_\pi(t) - \{c_j\} = C_{\pi^*}(t) - \{c_i\}$, and

3. $V_{\pi^*}(t) = V_\pi(t) + \{c_l\}$ for some customer c_l, and $C_\pi(t) = C_{\pi^*}(t)$ whenever $|V_\pi(t)| \geq c$. Note that if $|V_\pi(t)| < c$, then $C_\pi(t) = V_\pi(t)$ and $C_{\pi^*}(t) = V_{\pi^*}(t)$

We will prove that, under π^*, the system always satisfies one of these conditions by induction on the times that important events occur. These events are

- $\mathcal{E}_0$ - arrival to both systems,
- $\mathcal{E}_1$ - completion of a customer in either or both systems,
- $\mathcal{E}_2$ - customer missing deadline under one or both policies,

We couple service completions in the same way that we did for lemma 2. The exponential assumption guarantees that the service time received by the customers form an i.i.d. sequence of exponential r.v.'s. We couple the customer deadlines in the following way. The deadlines of all customers are the same in both systems while c_i and c_j are present in both systems, i.e., $D_l^* = D_l$. However, if c_i completes under π while c_j is in the queue (and thus c_j under π^*), then lemma 1 allows us to couple the deadlines of c_i under π^* and c_j under π so that $D_i^* \geq D_j$. Here D_l denotes a deadline under π and D_l^* a deadline under π^*. The assumption that $\{R_i\}_1^\infty$ is an independent sequence allows us to do so without affecting the joint statistics of R_i and all of the relative deadlines for customers other than c_j and the joint statistics of R_j and the deadlines of all other customers except c_i.

Let $(t_0, \sigma_0), (t_1, \sigma_1), \cdots$ be the sequence of times and events that occur at those times, i.e., event σ_i occurs at time t_i where $\sigma_i \in \{\mathcal{E}_0, \mathcal{E}_1, \mathcal{E}_2\}$. The inductive argument is similar to that found in lemma 2 and is omitted here.

Clearly the number of completions by time t and the number of customers in the system at time t are each larger under π^* than under π. Hence, since the number of customers that arrive to the system under the two policies are the same, we conclude that $L_\pi(t) \geq L_{\pi^*}(t)$. Again, the argument can be repeated for all deadline sequences for which π deviates from a SED_1^{ni} policy to yield the desired result. ∎

This lemma can be used to prove the following result.

Theorem 4 *For any policy $\pi \in \Sigma_1$, there exists a policy $\pi^* \in SED_1^{ni}$ such that*

$$L_{\pi^*}(t) \leq_{st} L_\pi(t), \quad \forall t > 0$$

under assumption $\boldsymbol{A}_1$.

Proof. According to lemma 3, we need only consider $\pi \in \Sigma_1^{ni}$. The proof consists of considering sample path $\boldsymbol{b}$ and constructing a sequence of policies $\pi = \pi_0, \pi_1, \cdots, \pi_i, \cdots$ such that

$\pi^* = \lim_{i\to\infty} \pi_i$ is a SED_1^{ni} policy satisfying $L_\pi(t) \geq L_{\pi^*}(t)$, $\forall t > 0$. Specifically, lemma 4 can be used to construct this sequence that exhibits the property that the first time at which each deviates from a SED_1^{ni} policy is increasing in i and that exhibit the property that $L_{\pi_i}(t)$ is a stochastically nondecreasing function of i $\forall t > 0$. Removal of the conditioning on the sample path yields the desired result. ∎

Nonpreemptive policies: The following result is established in exactly the same manner as theorems 1 and 4. However it should be noted that the lemma analogous to lemmas 2 and 4 require greater care when coupling the behavior of π and π^*. In addition, it requires the following assumption regarding the statistics of $Y(\theta, t)$, namely

Assumption *B*: *If* $Y(\theta, t_1) \leq_{st} Y(\alpha, t_2)$, *for* $\theta, \alpha \in \mathcal{S}$, $t_1, t_2 \geq 0$, *then*

$$Y(\theta, t_2 + \Delta) \leq_{st} Y(\alpha, t_1 + \Delta), \Delta > 0.$$

All of the applications described in the previous section satisfy this assumption.

Theorem 5 *For any policy* $\pi \in \Sigma_2^{ni}$, *there exists a policy* $\pi^* \in SED_2^{ni}$ *such that*

$$L_{\pi^*}(t) \leq_{st} L_\pi(t), \quad \forall t > 0$$

under assumptions $\boldsymbol{A}_1$ *and* $\boldsymbol{B}$.

Assumption $\boldsymbol{B}$ is required because, unlike a preemptive policy, a non-preemptive policy is unable to change its decision if the relationship between the deadlines of two customers change. Hence, the optimality of SED_2^{ni} policies requires that this relationship not change.

In general, neither SED_1^{ni} nor SED_2^{ni} will define unique policies. However they do each define a unique policy for the applications given in the previous section.

5 Generalizations of the Results

In this section we describe a number of ways that the results can be generalized to classes of policies that may idle while there is work in the queue, discrete time systems, systems in which servers take vacations, and systems with finite buffers.

Idling Policies: Almost all of our results regarding the optimality of nonpreemptive policies are restricted to non-idling policies. It is possible to show

1. $\forall \pi \in \Sigma_0$, there exists $\pi^* \in SED_0$ such that $L_{\pi^*}(t) \leq_{st} L_\pi(t)$, $t \geq 0$ under assumption $\boldsymbol{A}_0$.
2. $\forall \pi \in \Sigma_2$, there exists $\pi^* \in SED_2$ such that $L_{\pi^*}(t) \leq_{st} L_\pi(t)$, $t \geq 0$ under assumption $\boldsymbol{A}_1$.

Discrete time systems: Consider a discrete time multiple server queue where customers arrive in batches during each time unit and the service times consist of an integer multiple of time

units that are given by geometric r.v.'s with identical means. Similar results to those given in the previous sections can be proven for this system, i.e., that the best policies are contained in SED_0^{ni}, SED_1^{ni}, SED_2^{ni}. This model is of particular use in data communications in the case that the service time is always a single time unit. It forms the basis of many models of statistical multiplexers Oie et al. [4]. In the case that customers require a single time unit of service, there is no distinction between preemptive and non-preemptive systems. Furthermore, there is no distinction between systems in which customers must meet their deadlines either by the time service begins or by the time service completes.

Vacation Models: Theorem 5 can be generalized to include systems in which servers take vacations. This is of interest for at least two reasons. First, vacations can be used to represent server failures. Second, systems in which servers take vacations can be used to model real-time systems with two or more classes of customers. For example, one class of customers may be unable to tolerate missed deadlines. The second class of customers may be able to tolerate some missed deadlines. If the customers in the first class are well understood (i.e., known service times, arrival times), they can be given higher priority than the second class of customers and scheduled independently of the second class. Customers from the second class are like the customers that we have considered in our model.

Let $\{U_{i,j}, W_{i,j}\}_{i=1,\ldots,}$ $j = 1, 2, \cdots, c$ be families of r.v.'s such that $U_{i,j}$ is the length of the i-th time interval during which the j-th server is available for service and $W_{i,j}$ is the length of the i-th time interval during which the j-th server is on vacation (unavailable for service). We state the following result.

Theorem 6 *For any $\pi \in \Sigma_1$, there exists a policy $\pi^* \in SED_1^{ni}$ such that*

$$L_\pi(t) \geq_{st} L_{\pi^*}(t), \quad \forall t > 0$$

under assumption $\boldsymbol{A}_1$ and the assumption that $\{U_{i,j}, W_{i,j}\}_{i=1,\ldots,}$ $j = 1, 2, \cdots, c$ is independent of $\{A_i\}$, $\{S_i\}$, $\{\Theta_i\}$.

Finite buffer systems: Consider our system now with a maximum queue size B. When a customer arrives to a full queue, a decision must be made as to whether to reject that customer, or to admit him and remove some other customer already in the queue. This decision is made by a *buffer overflow policy.* Most work has centered around the buffer overflow policy that always rejects the arriving customer. However, if we define SEDB buffer overflow policies similar to the scheduling policies, i.e., the customer stochastically closest to its deadline is removed from the set of customers in that system that have not missed their deadlines (incl. the arrival), then we have results analogous to theorems 1, 2, 4, 5 where for any combination of scheduling/buffer overflow policy there is a combined policy that belongs to the family of $SED_0^{ni}/SEDB$, $SED_1^{ni}/SEDB$, or $SED_2^{ni}/SEDB$ policies which performs at least as well as the arbitrary policy in a stochastic sense.

Acknowledgments: The authors would like to acknowledge Mr. Zhang-Xue Zhao for his helpful suggestions which improved this paper and for some initial contributions to this work.

References

[1] F. Baccelli, Z. Liu, D. Towsley, "Extremal Scheduling of Parallel Processing with and without Real-Time Constraints", to appear in *J. ACM.*

[2] P. Bhattacharya and A. Ephremides, "Optimal Scheduling with Strict Deadlines," *IEEE Trans. on Automatic Control,* Vol. 34, No. 7 (July 1989), pp. 721-728.

[3] M. Kallmes, D. Towsley, C.G. Cassandras, "Optimality of the last-in-first (LIFO) service discipline in queueing systems with real-time constraints", *Proceedings of the 28-th IEEE Conference on Decision and Control*, pp. 1073-1074, December 1989.

[4] Y. Oie, T. Suda, M. Murata, D. Kolson, M. Miyahara "Survey of Switching Techniques in High-speed Networks and Their Performance," *Proc. IEEE Infocom'90,* pp. 1242-1252.

[5] S.S. Panwar, "Time Constrained and Multi-access Communications," Ph.D. Thesis", Dept. of Electrical & Computer Engineering, Univ. Massachusetts, Feb. 1986.

[6] S.S. Panwar, D. Towsley and J. Wolf, "Optimal Scheduling Policies for a Class of Queues with Customer Deadlines to the Beginning of Service," *J. of the ACM,* Vol. 35, No. 4 (Oct. 1988), pp. 832-844.

[7] S.M. Ross, *Stochastic Processes*, John Wiley & Sons, New York, (1983).

[8] H. Saito, "Optimal Queueing Discipline for Real-Time Traffic at ATM Switching Nodes," *Proc. of IEICI of Japan*, pp. 49-54, Sept. 1988.

[9] V. Strassen, "The Existence of Probability Measures with Given Marginals," *Ann. Math Statist.*, **36** (1965) 423-429.

[10] D. Towsley, F. Baccelli, "Comparisons of service disciplines in a tandem queueing network with real-time constraints", *Operations Research Letters*, **9**, 3, pp. 368-377, April 1991.