Comparison of hybrid minimum laxity / first-in -first-out scheduling policies for real-time multiprocessors

Philippe Nain, Don Towsley

INRIA

UNITÉ DE RECHERCHE
INRIA-SOPHIA ANTIPOLIS

Institut National
de Recherche
en Informatique
et en Automatique

Domaine de Voluceau
Rocquencourt
B.P.105
78153 Le Chesnay Cedex
France
Tél (1) 39 63 55 11

Rapports de Recherche

N° 1237

Programme 3
Réseaux et Systèmes Répartis

COMPARISON OF HYBRID MINIMUM LAXITY / FIRST-IN-FIRST-OUT SCHEDULING POLICIES FOR REAL-TIME MULTIPROCESSORS

Philippe NAIN
Don TOWSLEY

Juin 1990

RR-1237

Comparaison de Politiques Hybrides d'Ordonnancement "Minimum Laxity/ Premier-Arrivé-Premier-Servi" pour des Applications Temps-Réel sur Multiprocesseurs

Philippe Nain* and Don Towsley†

Résumé

Cet article étudie le comportement de deux politiques d'ordonnancement dans un système multiserveurs où les clients ont des dates limite de début de traitement. Ces politiques sont telles que leurs performances approchent les performances de la politique optimale *minimum laxity* sans pour autant en supporter le coût. Ceci est réalisé en divisant la file d'attente d'accès aux serveurs en deux parties (tampons): un tampon de taille maximale $n > 0$ (appelé ML) géré suivant la politique *minimum laxity* qui traite en priorité le client le plus proche de sa date limite et un tampon de taille infinie, géré suivant la politique premier-arrivé-premier-servi (PAPS). Sous la politique $F/ML(n)$ le tampon ML est placé devant les serveurs, c'est-à-dire qu'un nouvel arrivant trouvant au moins n clients dans la file d'attente est rangé dans la partie PAPS. Dans ce cas, à chaque départ du système le client en première position dans PAPS va dans ML. Sous la politique $ML(n)/F$ le tampon ML est placé à l'arrière de la file d'attente, c'est-à-dire qu'un nouvel arrivant trouvant au moins n clients rentre dans ML et force le client de ML le plus proche de sa date limite à rejoindre le tampon PAPS. Nous montrons que ces politiques apparemment différentes donnent des performances identiques pour n fixé, qu'il y ait ou non perte des clients ayant dépassé leur date limite. Des propriétés de monotonicité sont ensuite établies. En l'absence de pertes, nous montrons que la variable aléatoire définie comme la différence entre la date d'entrée en service et la date limite d'entrée en service est une fonction décroissante de n pour l'ordre convexe. En cas de pertes de clients, nous montrons que le nombre de clients perdus dans un intervalle donné est une fonction décroissante de n pour l'ordre stochastique fort.

*INRIA, B.P. 109, 06565 Valbonne Cedex, France.

†Department of Computer and Information Science, University of Massachusetts, Amherst, MA 01003, USA.
The work of this author was performed while the author was on sabbatical at Laboratories MASI, University Pierre and Marie Curie, Paris, France. The work of this author was supported in part by NSF under contract ASC 88-8802764 and ONR under contract N00014-87-K-0304.

Comparison of Hybrid Minimum Laxity / First-In-First-Out Scheduling Policies for Real-Time Multiprocessors

Philippe Nain* and Don Towsley†

Abstract

In this paper we study the behavior of two policies for scheduling customers with deadlines until the beginning of service onto multiple servers. Both policies attempt to approximate the performance of the *minimum laxity* scheduling policy without incurring the complete overhead. This is accomplished by dividing the queue into two queues - one, of maximum size $n > 0$, managed using the minimum laxity policy and another of unbounded size managed in a first in first out manner. One policy, $F/ML(n)$ places the ML queue at the front, i.e., customers finding n or more in the system enter the FIFO queue which in turn feeds the ML queue. The other policy, $ML(n)/F$ places the ML queue at the back, i.e, arriving customers enter the ML queue and if the total number in the system exceeds n, forces one customer from the ML queue to the FIFO queue. We show that these seemingly dissimilar policies exhibit exactly the same behavior for a fixed value of n both when customers are allowed to be discarded when they miss their deadlines before entering service and when they are not allowed to be discarded. We also establish monotonicity properties for both policies. In the case that no customer is discarded, we show that the stationary customer tardiness, the difference between the departure time and deadline, is a decreasing function of n, in the sense of convex ordering. In the case that discards are allowed, we show that the number of customers lost within an interval of time is a decreasing function of n in the sense of stochastic ordering.

*INRIA, B.P. 109, 06565 Valbonne Cedex, France.

†Department of Computer and Information Science, University of Massachusetts, Amherst, MA 01003, USA.
The work of this author was performed while the author was on sabbatical at Laboratories MASI, University Pierre and Marie Curie, Paris, France. The work of this author was supported in part by NSF under contract ASC 88-8802764 and ONR under contract N00014-87-K-0304.

1 Introduction

In recent papers Hong, Tan, and Towsley [4] and Zhao and Stankovic [12] proposed and analyzed two variants of the *minimum laxity scheduling policy* (ML) for real-time systems. In the first paper, the authors proposed a policy, $ML(n)$ that divides the queue into two queues. One queue, that can hold a maximum of n customers, is organized according to ML. If the number of customers exceeds n, then arriving customers are stored in a second queue which is handled by a first in first out policy (FIFO). This queue feeds the ML queue. The second paper studies an *improved FCFS* policy (FCFSI) where the deadline of every arriving customer is compared with that of the last customer in the queue. If the deadline of the arriving customer is smaller than the deadline of the customer currently at the end of the queue, then the positions of the two customers are switched; otherwise the new customer is inserted at the end. All customers in front of the last one are served according to a FIFO rule.

In this paper we compare the $ML(n)$ policies first proposed in [4] with a generalization of the FCFSI policy whereby the last n positions in the queue are managed according to ML. We refer to the first class of policies as $F/ML(n)$ and the second class as $ML(n)/F$ to distinguish the relative order of the ML and FIFO portions of the queues. We show the following:

- There is absolutely no difference in the performance of the $ML(n)/F$ and $F/ML(n)$ policies. This is true for systems that require all customers to be served as well as systems that discard customers that miss their deadlines before beginning service;
- In the case of systems that are permitted to discard customers that miss their deadlines, all policies that determine when and which customers to discard exhibit the same behavior when coupled with either the $ML(n)/F$ or $F/ML(n)$ policies;
- We establish monotonicity results for both policies. Specifically we show:
  - The customer tardiness decreases in n in the sense of convex ordering for a system that does not allow customers to be discarded. This is established under the assumptions fthat interarrival times, deadlines, and service times form mutually independent i.i.d. sequences of random variables (r.v.'s). Here the customer tardiness is the difference between the time a customer is scheduled and its deadline;
  - The number of customers lost by time $t > 0$ is a decreasing function of n in the sense of stochastic ordering. This is established under general assumptions for arrival times and deadlines and the requirement that service times form an i.i.d. sequence of exponential r.v.'s. This generalizes an existing result in [4] established under the assumptions of exponential interarrival times and deadlines as well as service times.

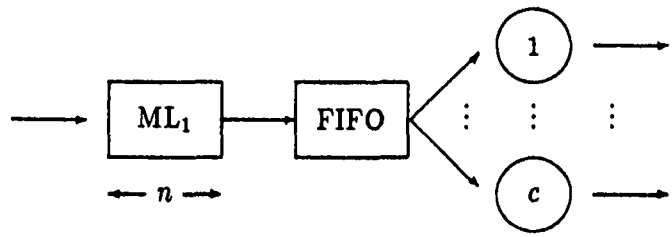
The interest in policies such as $ML(n)/F$ and $F/ML(n)$ lies in the fact that they approximate the performance of ML for small values of n. ML has been shown in a variety of contexts to be the policy, from the class of nonpreemptive policies that use deadline but not service time information, that minimizes the fraction of customers that miss their deadlines (when discards are allowed) [6,7,1,11] or minimizes the customer lag time, in the sense of convex ordering (when discards are not allowed) [10]. The ML policy has a complexity of either $O(m)$ or $O(\log m)$, depending on the implementation, where m is the number of customers waiting in the queue whereas both the $ML(n)/F$ and $F/ML(n)$ policies have a complexity of $O(1)$ for a given value of n. Evidence of how well $F/ML(n)$ and $ML(n)/F$ approximate ML can be found in [4,12].

The remainder of the paper is organized in the following manner. Section 2 contains a formal description of the system and of the $F/ML(n)$ and $ML(n)/F$ policies. Section 3 contains the proofs of the results regarding the equivalence of the $F/ML(n)$ and $ML(n)/F$ policies as well as the insensitivity of the discard policy on the performance of the policies when discards are allowed. Section 4 establishes monotonicity properties for these policies.

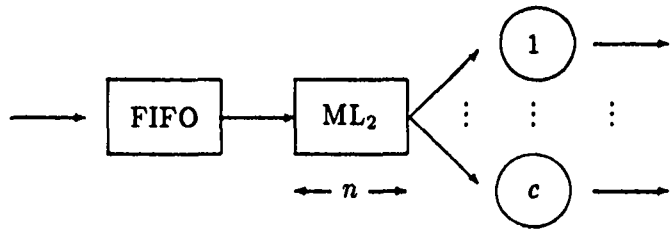
2 Definitions and Notation

We consider c servers serving a single customer arrival stream. Let $0 \leq a_1 < a_2 < \cdots < a_k < \cdots$ denote a sequence of arrival times. Define $\theta_n = a_{n+1} - a_n$ for $n \geq 1$. Let $\sigma_k \geq 0$ denote the service time of the k-th customer to be served. Last, customers have deadlines by which they should begin service. Let $d_k = a_k + \tau_k$ be the deadline by which customer k should begin service, $\tau_k > 0$. We shall refer to τ_k as the *relative deadline* of the k-th customer.

We are interested in the behavior of two scheduling policies, the $ML(n)/F$ and the $F/ML(n)$ policies, $n \geq 1$. Both policies divide the queue into two queues, one managed in a FIFO manner, the other according to the minimum laxity (ML) rule. The ML rule is one that orders the customers according to their deadlines and gives priority to the customer with the minimum laxity (earliest deadline). The ML queue never contains more than n customers whereas the FIFO queue has unlimited space. In both cases, no customer resides in the FIFO queue whenever the number of customers waiting for service is less than or equal to n. The two policies differ according to their behavior when the number of waiting customers exceeds n. At the time of an arrival, $ML(n)/F$ transfers the customer with the smallest deadline from the ML queue to the FIFO queue. The new arrival is placed in the ML queue so that the deadlines remain in increasing order. On the other hand, $F/ML(n)$ places the arrival at the back of the FIFO queue. At the time of a departure, $ML(n)/F$ schedules the customer at the head of the FIFO queue whereas $F/ML(n)$ schedules the customer at the head of the ML queue. In the latter case the customer at the head of the FIFO queue is transferred to the ML queue.



(a)



(b)

Figure 1: (a) The $ML(n)/F$ policy. (b) The $F/ML(n)$ policy.

In the case that the ML queue contains two customers with the same deadline, we assume that the oldest customer is selected first. Last, both policies are assumed to be *nonidling* and *nonpreemptive*.

Two variations of this policy exist according to whether customers are removed or not when they miss their deadlines. We will examine the behavior of each of these variations in later sections.

As it will be cumbersome to distinguish between the $ML(n)/F$ and $F/ML(n)$ policies by their names in subsequent formulae, we will denote them respectively as π_1 and π_2. We are interested in the following performance metrics:

- $L_n^{\pi_i}(t)$, $i = 1, 2$ which denotes the number of customers that miss their deadlines by time t;
- $D_n^{\pi_i}(k)$, the scheduling time of the k-th customer scheduled;
- $I_n^{\pi_i}(k)$, the identity of the k-th customer scheduled;
- $W_n^{\pi_i}(k)$, the wait time of the k-th customer scheduled;
- $R_n^{\pi_i}(k)$, the response time of the k-th customer scheduled, $R_n^{\pi_i}(k) = W_n^{\pi_i}(k) + \sigma_{I_n^{\pi_i}(k)}$;

- $C_n^{\pi_i}(k)$, the tardiness of the k-th customer scheduled, $C_n^{\pi_i}(k) = W_n^{\pi_i}(k) - \tau_{I_n^{\pi_i}(k)}$,

for $i = 1, 2$, $k \geq 1$ where $n \geq 1$ is the policy parameter.

When no confusion is created, we will replace the superscript π_i with i.

In the remainder of the paper we will show that π_1 and π_2 exhibit the same behavior. We will also prove monotonicity results regarding the behavior of the two policies as a function of n. The results of the paper are based on the following assumptions:

- **A1** $\{\theta_k\}_{k=1}^{\infty}$ $\{\tau_n\}_{k=1}^{\infty}$ and $\{\sigma_k\}_{k=1}^{\infty}$ are arbitrary deterministic sequences of nonnegative numbers;
- **A2** $\{\theta_k\}_{k=1}^{\infty}$, $\{\tau_n\}_{k=1}^{\infty}$ and $\{\sigma_k\}_{k=1}^{\infty}$ are arbitrary sequences of i.i.d. r.v.'s. All these random variables are mutually independent;
- **A3** $\{\theta_k\}_{k=1}^{\infty}$ and $\{\tau_n\}_{k=1}^{\infty}$ are arbitrary sequences of nonnegative i.i.d. r.v.'s. $\{\sigma_k\}_{k=1}^{\infty}$ is a sequence of i.i.d. exponential r.v.'s. All these random variables are mutually independent.

Under assumption **A2** and **A3** let θ, σ and τ denote a generic interarrival time, service time and deadline, respectively. We further assume that $\mathrm{E}[\sigma] < \infty$ and $\mathrm{E}[\tau] < \infty$.

3 The Equivalence of $ML(n)/F$ and $F/ML(n)$

3.1 Customers must be served

The main result of this section is that the departure process does not depend on whether $ML(n)/F$ or $F/ML(n)$ is used provided that the number of customers waiting for service at time $t = 0$ is less than or equal to n and the customers in service have the same amount of service time remaining.

3.1.1 Notation and preliminary results

Let $\mathrm{S}^j(t) = (\mathbf{X}^j; \mathbf{Y}^j)$ be the state of the system under policy π_j, $j = 1, 2$ at time t where $\mathbf{X}^j = (x_1, x_2, \ldots, x_m)$ is the state of the ML queue and $\mathbf{Y}^j = (y_1, y_2, \ldots, y_p)$ is the state of the FIFO queue such that $0 \leq m \leq n$, $p = 0$ if $m < n$, and $p \geq 0$ if $m = n$ (with the convention that $\mathbf{X}^j$ and $\mathbf{Y}^j$ are empty if $m = 0$ and $p = 0$, respectively) . We assume that the customers within the ML queue are ordered according to their deadlines, i.e. $d_{x_i} \leq d_{x_{i+1}}$, $i = 1, \ldots, m$ and

the customers in the FIFO queue are ordered according to their time of entry to that queue (here y_1 is the "oldest" customer).

Let t_i^j be the time that the state of the ML queue changes for the i-th time under policy π_j, $i \geq 1$, $j = 1,2$. We let $\mathbf{S}_i^j = \mathbf{S}^j(t_i^{j+})$, $\mathbf{X}_i^j = \mathbf{X}^j(t_i^{j+})$, and $\mathbf{Y}_i^j = \mathbf{Y}^j(t_i^{j+})$. If t is an arrival/departure epoch then when referring to the "state at time t" we mean the state just after t. When speaking of the "number of customers in the system" we mean the number of queued customers, i.e., not including the customers in the servers. Last, ML_j will denote the ML buffer of system j, $j = 1,2$.

Without loss of generality we adopt the convention that, whenever an arrival and a departure occur simultaneously, the departure is first taken into account. We start with two technical lemmas. The first lemma states that the i-th change of ML_1 always occurs before or at the same time as the i-th change of ML_2.

Lemma 1 *Under assumption* **A1** *and given the above initial conditions*

$$t_i^1 \leq t_i^2,$$

for all $i \geq 1$. *Moreover,* $t_i^1 = t_i^2$ *if and only if the* i*-th change in either ML queue occurs when there are at most* $n-1$ *customers in the system or when there are* n *customers and this change is due to a departure.*

Proof. Since σ_k is the service time of the k-th customer that enters a server (whatever the identity of this customer is), assumption **A1** and the fact that both polices are nonidling and nonpreemptive policies readily imply that any time both systems contain the same number of customers.

Let $t > 0$ be the first time when the number of customers exceeds n. Since ML_1 and ML_2 simultaneously change at arrivals and departures when there are less than n customers or when there are n customers and the change is due to a departure, we see that $t_l^1 = t_l^2$ for $l = 1, \ldots, i$ where i is the number of events (arrivals and departures) that occured in $[0, t)$.

Let $t^{(1)}$ be the first time after t when there are fewer than n customers[1] and assume that $t^{(1)} < +\infty$ (the case where $t^{(1)} = +\infty$ can be treated in a similar way). Assume further that k customers entered both systems in $[t, t^{(1)})$ (including the one at time t).

Let $t = a_1 < a_2 < \cdots < a_k < t^{(1)}$ be the successive arrival times in $[t, t^{(1)})$ and let $t < s_1 \leq \cdots \leq s_k \leq s_{k+1} = t^{(1)}$ be the successive departure times in $(t, t^{(1)}]$. Then, by definition of systems

[1] It is to be noted from our convention on simultaneous events that no simultaneous arrival and departure may occur in $(t, t^{(1)})$ when there are n customers since otherwise this time would be $t^{(1)}$ by definition of $t^{(1)}$.

1 and 2, $t^1_{i+l} = a_l$ for $l = 1, \ldots, k$, $t^1_{i+k+1} = t^{(1)}$ and $t^2_{i+l} = s_l$ for $l = 1, \ldots, k+1$. Further a short moment of reflection should convince the reader that $t^1_l < t^2_l$ for $i+1 \leq l \leq i+k$ and $t^1_{i+k+1} = t^2_{i+k+1} = t^{(1)}$.

This last result together with the definition of ML_1 and ML_2 yields $t^1_l = t^2_l$ for $i+k+1 < l \leq i'$, where i' is the largest integer such that $t^j_{i'} < t^{(2)}$ for $j = 1,2$ with $t^{(2)}$ the first time after $t^{(1)}$ when the number of customers exceeds n. Iterating this procedure gives the expected result. ∎

Lemma 2 *Let* $\mathbf{S}^2(t) = (\mathbf{X}^2; \mathbf{Y}^2)$ *be the state under policy* π_2 *at some time* t *with* $\mathbf{X}^2 = (x_1, x_2, \ldots, x_n)$, $\mathbf{Y}^2 = (y_1, y_2, \ldots, y_p)$ *and* $p \geq 1$. *Then* y_1 *is the first customer to arrive after all the* x_i*'s have arrived.*

Proof. First note that y_1 has arrived before $y_2, \ldots, y_p$. Assume there exists a customer z that arrived before y_1 and after all the x_i's. By definition of ML_2 we know that z cannot enter ML_2 before the customer with the smallest deadline among $x_1, \ldots, x_n$ (i.e., x_1 in this case) enters service. Therefore z should still be in $\mathbf{Y}^2$ at time t (and in front of y_1), which implies that z does not exist. ∎

3.1.2 Main result

The following result expresses that the state of both ML buffers is the same just after their i-th change, $i \geq 1$.

Proposition 1 *Under assumption* **A1**, $\mathbf{X}^1_i = \mathbf{X}^2_i$ *for all* $i \geq 1$.

Proof. Assume assumption **A1** holds. Let k be the number of customers at time 0. Assume first that $k < n$. Then, $t^1_1 = t^2_1$ and $\mathbf{X}^1_1 = \mathbf{X}^2_1$.

If $k = n$ and the first event to occur is a departure then $\mathbf{X}^1_1 = \mathbf{X}^2_1 = (x_2, \ldots, x_k)$. Assume now that the first event to occur is an arrival (call it x). Then $t = t^1_1$ and $\mathbf{X}^1_1 = (x_2, \ldots, x_m, x, x_{m+1}, \ldots, x_k)$ if $d_{x_2} \leq \cdots \leq d_{x_m} \leq d_x \leq d_{x_{m+1}} \leq \cdots \leq d_{x_k}$, $m = 1, \ldots, k-1$.

Let t' be the first departure time after t. Then, $t' = t^2_1$ by definition of system 2 and at that time the customer at the head of ML_2 (i.e., x_1) is placed in the server whereas the customer at the head of the FIFO buffer (i.e., x) is placed in ML_2. Then, ML_2 is reordered according to the ML rule and so $\mathbf{X}^2_1 = (x_2, \ldots, x_m, x, x_{m+1}, \ldots, x_k)$, which shows that $\mathbf{X}^1_1 = \mathbf{X}^2_1$.

Assume that $\mathbf{X}_l^1 = \mathbf{X}_l^2$, for $l = 1, 2, \ldots, i-1$. Let us show that $\mathbf{X}_i^1 = \mathbf{X}_i^2$. Two cases have to be distinguished:

(1) There are at most $n-1$ customers when the $(i-1)$-th change in ML_1 occurs or there are n customers and this change is due to departure.

From Lemma 1 we see that $t_{i-1}^1 = t_{i-1}^2$. On the other hand, $\mathbf{X}_{i-1}^1 = \mathbf{X}_{i-1}^2$ from the inductive hypothesis. Consequently this case reduces to the proof of $\mathbf{X}_1^1 = \mathbf{X}_1^2$ (with initial time t_{i-1}^1 and initial state $\mathbf{X}_{i-1}^1$) which was treated above. Hence, $\mathbf{X}_i^1 = \mathbf{X}_i^2$.

(2) There are $n+p$ customers, $p \geq 1$, when the $(i-1)$-th change in ML_1 occurs or there are n customers and this change is due to an arrival.

Let

$$(x_1, \ldots, x_n; y_1, \ldots, y_p, y_{p+1}) \tag{3.1}$$

be the state of queue 1 at time t_{i-1}^{1+}, $p \geq 0$. (Note that $p = 0$ if there were n customers prior to time t_{i-1}^1 and if the $(i-1)$-th change in ML_1 corresponds to an arrival. If there were $n+p$ customers prior to time t_{i-1}^1 with $p \geq 1$, then necessarily the $(i-1)$-th change in ML_1 occurs at an arrival epoch and therefore system 1 contains $n+p+1$ customers at time t_i^{1+}, which agrees with (3.1).)

Let t_a be the arrival time of the first customer that arrives after t_{i-1}^1. Let y be this customer and assume that

$$d_{x_2} \leq \cdots \leq d_{x_{m-1}} \leq d_y \leq d_{x_m} \leq \cdots \leq d_{x_n}, \tag{3.2}$$

$2 \leq m \leq n+1$. Again two cases have to be considered:

(2.1) y_{p+1} completes its service at time $t_{p+1} < t_a$.

Therefore $t_i^1 = t_{p+1}$ and $\mathbf{X}_i^1 = (x_2, \ldots, x_n)$. Since there are exactly n customers in system 1 (and therefore in system 2) at time t_i^{1-} and that t_i^1 is a departure epoch we have from Lemma 1 that $t_i^1 = t_i^2$. By the inductive hypothesis we know that $\mathbf{S}_{i-1}^2 = (x_1, x_2, \ldots, x_n; \ldots)$ and further we know that system 2 contains exactly n customers at time t_i^{2-}. So, necessarily $\mathbf{S}_{i-1}^2 = (x_1, x_2, \ldots, x_n)$ and $\mathbf{X}_i^2 = (x_2, \ldots, x_n)$.

(2.2) y_{p+1} does not complete its service before t_a.

Then $t_i^1 = t_a$ and $\mathbf{X}_i^1 = (x_2, \ldots, x_{m-1}, y, x_m, \ldots, x_n)$ from (3.2).

Since y is the first customer to arrive after $x_1, x_2, \ldots, x_n$ have all arrived (because no customer in $\mathbf{Y}^1$ may have arrived after x_1 by definition of ML_1), Lemma 2 and the inductive hypothesis

$\mathbf{X}^1_{i-1} = \mathbf{X}^2_{i-1}$ imply that the state of system 2 at time t^2_{i-1} is either $(x_1, \ldots, x_n; y, \ldots)$ or $(x_1, \ldots, x_n)$, depending whether or not y has arrived at time t^2_{i-1}. In both cases however t^2_i will be the next service completion time since ML_2 contains n customers.

Assume that y has not yet arrived when x_1 enters the server at time t^2_i. This would imply that $t^2_i < t_a = t^1_i$ which is impossible from Lemma 1. Therefore the state of queue 2 at time t^2_{i-1} is $(x_1, \ldots, x_n; y, \ldots)$ and clearly from (3.2) $\mathbf{X}^2_i = (x_2, \ldots, x_{m-1}, y, x_m, \ldots, x_n)$ which completes the proof. ∎

The following result is a consequence of the above property.

Theorem 1 *Under assumption* **A1** *policies $ML(n)/F$ and $F/ML(n)$ exhibit the same behavior in the sense that*

$$D^{\pi_1}(k) = D^{\pi_2}(k), \tag{3.3}$$

$$I^{\pi_1}_n(k) = I^{\pi_2}_n(k), \tag{3.4}$$

$\forall\, k \geq 1,\ n \geq 1.$

Proof. Relations (3.3) and (3.4) are certainly true in the case of $n = 1$ since both policies schedule customers according to the FIFO rule. In the case of $n > 1$, we can apply Proposition 1. Hence the departure processes are the same due to the fact that the second queue under $ML(n)/F$ preserves the order of customer departures from ML_1. Hence relations (3.3) and (3.4) follow directly. ∎

This leads to the following corollaries.

Corollary 1 *Under assumption* **A1**,

$$\begin{aligned} L^1_n(t) &= L^2_n(t), \\ W^1_n(t) &= W^2_n(t), \\ R^1_n(t) &= R^2_n(t), \\ X^1_n(t) &= X^2_n(t), \end{aligned}$$

for all $t \geq 0$, $n \geq 1$,

Corollary 2 *Under assumption* **A1** *Theorem 1 still holds if the service times are associated with the customers.*

Proof. Let $(x_1, x_2, \ldots, x_p)$ be the state of both systems at time 0, $0 \leq p \leq n$. Let $\hat{\sigma}_k$ be the service time of the customer that arrives at time a_k, $k \geq 1$, and let $\hat{\sigma}_{-j}$ be the service time of x_j, $j = 1, \ldots, p$. Whatever the service time σ_k of the k-th customer scheduled is under policy π_i, it is clear that $I_n^{\pi_i}(k)$ does not depend on σ_k, $k \geq 1$, $i = 1, 2$. Now, since $I_n^{\pi_1}(k) = I_n^{\pi_2}(k)$ for $k \geq 1$ (Proposition 1) the proof follows by letting $\sigma_k = \hat{\sigma}_{I_n^{\pi_1}(k)}$ for $k \geq 1$, with the convention that $\sigma_k = \hat{\sigma}_{-j}$ if $I_n^{\pi_1}(k) = x_j$, $1 \leq j \leq p$. ∎

3.2 Customers are discarded when they miss their deadlines

In this section we establish a similar result for systems where customers can be discarded as soon as they miss their deadlines, if they have not already been served. In such systems, the question arises as to when customers should be discarded. One expects the answer to be - as early as possible. However, we will show that the choice of discard rule does not affect the performance of either policy. Following this, we will apply the results from the previous subsection to show that there is no difference between $ML(n)/F$ and $F/ML(n)$. A customer x is said to be *alive* (resp. *dead*) at time t if $d_x > t$ (resp. $d_x \leq t$). First we introduce some notation. Let

- $\Sigma_1(n)$ be the complete class of nonidling, nonpreemptive policies that use $ML(n)/F$ for scheduling customers. Policies in this class differ from each other as to when waiting customers who miss their deadlines are removed from the queue;
- $\Sigma_2(n)$ be the complete class of nonidling, nonpreemptive policies that use $F/ML(n)$ for scheduling customers;
- $\pi_1 \in \Sigma_1(n)$ be the policy that discards customers at the time that they are scheduled into service;
- $\pi_2 \in \Sigma_2(n)$ be the policy that discards customers at the time that they are scheduled into service.

To make things clear let us specify for π_2 the order of the events at a scheduling epoch s. The following algorithm must be repeated until no dead customer remains in the ML_2 buffer: discard a dead customer from ML_2 and replace it by the customer at the head of the FIFO queue, if any. This procedure is instantaneous and ends at time s^-. Then, at time s a customer is scheduled for service according to the ML rule and replaced (say at time s^+) by the customer at the head of the FIFO buffer (which may be dead or alive), if any. In other words, π_2 behaves exactly like the $F/ML(n)$ policy of Section 3.1 provided that a zero service time is given to a dead customer. Note that all customers in ML_2 are alive at time s^-. Policies in $\pi \in \Sigma_2(n) - \{\pi_2\}$

differ from π_2 in that they may discard customers from both buffers at any time. Again this operation takes no time.

We have the following result concerning $\Sigma_2(n)$.

Lemma 3 *Under assumption* **A1**,

$$D^{\pi_2}(k) = D^{\pi}(k), \tag{3.5}$$

$$I_n^{\pi_2}(k) = I_n^{\pi}(k), \tag{3.6}$$

$\forall \pi \in \Sigma_2(n)$, $k \geq 1$, $n \geq 1$.

Proof. Consider an arbitrary policy $\pi \in \Sigma_2(n)$. The proof is by induction on the scheduling instances $\{s_k\}_{k=1}^{\infty}$ where one or both policies schedule a customer.

Let $Z = \{z_1, \ldots, z_l\}$ be a set of customers. Define $Live(Z,t) = \{z \in Z \mid d_z > t\}$, i.e., the subset of Z containing live customers at time $t > 0$. We establish the following relations: for all $i \geq 1$,

$$Live(\mathbf{Y}^{\pi}(s_i^-), s_i^-) = Live(\mathbf{Y}^{\pi_2}(s_i^-), s_i^-), \tag{3.7}$$

$$\mathbf{X}^{\pi}(s_i^-) = \mathbf{X}^{\pi_2}(s_i^-). \tag{3.8}$$

Note that (3.7)-(3.8) imply relations (3.5) and (3.6).

We recall that $\mathbf{X}^{\pi}(s_i^-)$ and $\mathbf{Y}^{\pi}(s_i^-)$ are understood to be buffer states immediately prior to scheduling a customer into service after the occurence of all the discards.

Basis Step. Trivially true for $i = 1$ since both systems are in the same state time at 0.

Induction Step. Assume that the hypothesis is correct for $i = 1, 2, \ldots, k-1$. There are two cases depending on whether the scheduling at time s_k is triggered by an arrival or a service completion. The former occurs if and only if at least one of the two systems is empty when the customer arrives. But if so the inductive hypothesis tells us that the other system is also empty. Therefore (3.7)-(3.8) still hold for $i = k$.

Let us consider the case that the scheduling is triggered by service completion. It follows from the inductive hypothesis that both policies schedule a customer at s_k. Let z be the customer scheduled into service at s_{k-1}. Again the inductive hypothesis ensures that both policies schedule a customer at time s_{k-1} and that this customer is the same. Let $\mathbf{A}(a,b)$ be the set of customers that arrive in (a,b), $a < b$.

Let $Old(Z,l)$ be the subset of Z containing the l oldest customers if $l < |Z|$ and Z if $l \geq |Z|$. Then, according to the definition of the scheduling policy π,

$$\mathbf{X}^{\pi}(s_k^-) = Live(\mathbf{X}^{\pi}(s_{k-1}^-) - \{z\}, s_k^-) + Old(Live(\mathbf{Y}^{\pi}(s_{k-1}^-) \cup \mathbf{A}(s_{k-1}, s_k), s_k^-), l), \tag{3.9}$$

where $l = n - |Live(\mathbf{X}^{\pi}(s_{k-1}^-) - \{z\}, s_k^-)|$.

Similarly,

$$\mathbf{X}^{\pi_2}(s_k^-) = Live(\mathbf{X}^{\pi_2}(s_{k-1}^-) - \{z\}, s_k^-) + Old(Live(\mathbf{Y}^{\pi_2}(s_{k-1}^-) \cup \mathbf{A}(s_{k-1}, s_k), s_k^-), l). \quad (3.10)$$

But it follows from the inductive hypothesis (3.8) with $i = k - 1$ that

$$Live(\mathbf{X}^{\pi}(s_{k-1}^-) - \{z\}, s_k^-) = Live(\mathbf{X}^{\pi_2}(s_{k-1}^-) - \{z\}, s_k^-), \quad (3.11)$$

and from the inductive hypothesis (3.7) with $i = k - 1$ that

$$Old(Live(\mathbf{Y}^{\pi}(s_{k-1}^-) \cup \mathbf{A}(s_{k-1}, s_k), s_k^-), l) = Old(Live(\mathbf{Y}^{\pi_2}(s_{k-1}^-) \cup \mathbf{A}(s_{k-1}, s_k), s_k^-), l). \quad (3.12)$$

Consequently, $\mathbf{X}^{\pi}(s_k^-) = \mathbf{X}^{\pi_2}(s_k^-)$ by combining relations (3.9)-(3.12). On the other hand,

$$\begin{aligned} Live(\mathbf{Y}^{u}(s_k^-), s_k^-) &= Live(\mathbf{Y}^{u}(s_{k-1}^-, s_k^-) \cup \mathbf{A}(s_{k-1}, s_k), s_k^-) \\ &- Old(Live(\mathbf{Y}^{u}(s_{k-1}^-) \cup \mathbf{A}(s_{k-1}, s_k), s_k^-), l), \end{aligned}$$

for $u \in \{\pi, \pi_2\}$. Consequently, $Live(\mathbf{Y}^{\pi}(s_k^-), s_k^-) = Live(\mathbf{Y}^{\pi_2}(s_k^-), s_k^-)$ from the inductive hypothesis (3.7) and (3.12). This completes the induction step. ∎

Before we prove the analogous result for the class of $ML(n)/F$ policies, we require the following technical lemma.

Lemma 4 *Let the state of the queue under $\pi \in \Sigma_1(n)$ at some point in time $t > 0$ be $\mathbf{S}^{\pi} = (x_1, \ldots, x_n; y_1, \ldots, y_p)$, $p \geq 1$. If $d_{x_2} > t$ then $d_{y_i} \leq d_{x_2}$, $1 \leq i \leq p$, and if $d_{x_2} < t$ then $d_{y_i} < t$.*

Proof. Consider a policy $\pi \in \Sigma_1(n)$ The proof is by induction on the arrival times, service times, and deadlines where there are more than n customers in the queue immediately after the event. Label these times $\{s_k\}_{k=1}^{\infty}$.

Basis step. The system begins with no more than n customers. Hence the first event is an arrival at time s_1 that brings the number of customers to $n + 1$. The resulting state is either $(x, x_2, \ldots, x_n; x_1)$ or $(x_2, \ldots, x_i, x, x_{i+1}, \ldots x_n; x_1)$. The lemma holds in the first case because $d_{x_2} \geq d_{x_1}$ and in the second case because $\min(d_x, d_{x_3}) \geq d_{x_2} \geq d_{x_1}$ as well. Consequently the lemma is true in this case.

Inductive step. There are four types of events. Assume that the theorem is true up to the $(k-1)$-th event and let $\mathbf{S} = (x_1, \ldots, x_n; y_1, \ldots, y_p)$ be the state prior to the k-th event and $\mathbf{S}'$ the state after the k-th event.

Deadline miss. Note that $\mathbf{S}' = \mathbf{S}$. If any customer other than x_2 misses its deadline, then the result follows by induction. If x_2 misses its deadline, then it must have been the case that $d_{y_i} < s_{k-1} \leq s_k$ because $d_{x_2} \geq d_{y_i}$ by induction, and no other deadline misses occur between s_{k-1} and s_k.

Customer discard. If some customer y_i is discarded, then $\mathbf{S}' = (x_1, \ldots, x_n; y_1, \ldots, y_{i-1}, y_{i+1}, \cdots y_p)$ and the result follows from the inductive hypothesis. If customer x_1 is discarded and x_2 is alive at time s_k, then $\mathbf{S}' = (y_p, x_2, \ldots, x_n; y_1, \ldots, y_{p-1})$. This follows from the inductive hypothesis and the result follows directly. If customer x_2 is dead and x_1 is discarded, or x_i is discarded, $i > 2$, then it follows from the inductive hypothesis that $d_{y_i} < s_k$ and the result follows.

Customer arrival. Let customer x arrive at time s_k. If $p = 0$, then the argument used for the basis step is applicable here. Assume that $p > 0$. Then the resulting state will be either $\mathbf{S}' = (x, x_2, \ldots, x_n; y_1, \ldots, y_p, x_1)$ or $\mathbf{S}' = (x_2, \ldots, x_i, x, x_{i+1}, \ldots, x_n; y_1, \ldots, y_p, x_1)$. In both cases the result readily follows from the inductive hypothesis.

Scheduling event. If there are still more than n customers, then the resulting state is $\mathbf{S}' = (x_1, \ldots, x_n; y_j, \ldots, y_p)$ and the inductive hypothesis can be applied to yield the desired result.

This completes the inductive step and proves the lemma. ∎

Lemma 5 *Under assumption* **A1**,

$$D^{\pi_1}(k) = D^{\pi}(k), \tag{3.13}$$

$$I_n^{\pi_1}(k) = I_n^{\pi}(k), \tag{3.14}$$

$\forall \pi \in \Sigma_1(n)$, $k \geq 1$, $n \geq 1$.

Proof. Let $\gamma \in \Sigma_1(n)$ be the policy that discards customers in the queue as soon as they reach their deadlines. We will show that $D^{\gamma}(k) = D^{\pi}(k)$ and $I_n^{\gamma}(k) = I_n^{\pi}(k)$, $\forall \pi \in \Sigma_1(n)$, $k \geq 1$, $n \geq 1$. The proof is simplified by modifying our representation of the system. Let the state of the system at time t be $\mathbf{S}^{\gamma}(t) = (z_p, \ldots, z_1)$ where the contents of the ML portion of the queue is $z_1, \ldots, z_{\min(n,p)}$ with the convention that $d_{z_i} \leq d_{z_{i-1}}$, $2 \leq i \leq \min(n,p)$. We introduce the notation $\mathbf{S}^{\pi}(t) = (z'_{p'}, \ldots, z'_1)$ for an arbitrary policy $\pi \in \Sigma_1(n)$ Define $i^{\pi}(x,t)$ to be the position of customer x in the queue at time $t > 0$ under π. If x is not in the queue, then the position is 0. We establish

$$\mathit{Live}(\mathbf{S}^{\pi}(t), t) = \mathit{Live}(\mathbf{S}^{\gamma}(t), t), \tag{3.15}$$

$$i^{\pi}(x,t) < i^{\pi}(x',t) \text{ iff } i^{\gamma}(x,t) < i^{\gamma}(x',t), \forall x \neq x' \in \mathit{Live}(\mathbf{S}^{\gamma}(t), t), \tag{3.16}$$

for all $t > 0$. The proof of these properties is by induction on the times $\{s_k\}_{k=1}^{\infty}$ that the state changes either under π or γ. These times include arrival, departure and discard times as well as times when a customer dies even if it is not discarded at once. Simultaneous discards of a customer by both policies will be handled by treating the discard under γ first.

Basis step. Follows from the initial conditions.

Inductive step. Assume that the above relations hold for the $(k-1)$-th state change. There are four cases:

Deadline miss. Suppose that customer $z_i \in \mathbf{S}^{\gamma}(s_k^-)$ misses his deadline. By the inductive hypothesis, this customer is also in the system under π and misses his deadline at the same time. Consequently, $Live\,(\mathbf{S}^{\gamma}(s_k), s_k) = Live\,(\mathbf{S}^{\pi}(s_k), s_k)$. The relative order of the live customers in each queue remains unchanged.

Customer discard. Suppose γ discards customer z_i. If $i > n$, then clearly the relative order of the remaining customers is unchanged. If $i = n$, then Lemma 4 ensures that the relative order also remains unchanged. Finally, if $i < n$, there are fewer than n customers and again the relative order is unaffected. Suppose that π discards customer z'_i. Then if $i > n$, the relative order of the remaining customers is unchanged. If $i = n$ and $d_{z'_{n-1}} \geq s_k$, then Lemma 4 ensures that the order is unchanged. If $i = n$ and $d_{z'_{n-1}} < s_k$, then as a consequence of Lemma 4, the live customers have indices less than $n-2$ and their relative positions cannot be affected. Finally, $i < n$ which again implies that all of the live customers have indices less than i as a consequence of Lemma 4 and their relative order is unaffected by the discard.

Customer arrival. There are two cases according to whether $p < n$ or not. It should be clear that $p' \geq p$.
i) $p < n$. In this case it follows from Lemma 4 and the inductive hypothesis that $z'_i = z_i$, $1 \leq i \leq p$ and the remaining customers (if any) under π are dead. The arriving customer (who is alive) will be placed in the same position with respect to these p customers thus establishing the above relationships.
ii) $p \geq n$. It follows from Lemma 4 and the inductive hypothesis that $z'_i = z_i$, $1 \leq i \leq n-1$. Consequently, an arrival will be placed in the same position with respect to these $n-1$ customers thus establishing the above relationships.

Service scheduling. According to the inductive hypothesis, if either policy can schedule a customer at time s_k, the other can also. Also because of the inductive hypothesis, both policies will schedule the same customer. The scheduling cannot affect the relative ordering of the remaining live customers. Hence the relations above hold.

This completes the inductive step. Relations (3.15)-(3.16) readily imply that $D^{\gamma}(k) = D^{\pi}(k)$ and $I_n^{\gamma}(k) = I_n^{\pi}(k)$ for all $k \geq 1$ and for all $\pi \in \Sigma_1(n)$. Letting now $\pi = \pi_1$ in the above

relations yields $D^{\gamma}(k) = D^{\pi_1}(k)$ and $I_n^{\gamma}(k) = I_n^{\pi_1}(k)$ for $k \geq 1$. Therefore, $D^{\pi_1}(k) = D^{\pi}(k)$ and $I_n^{\pi_1}(k) = I_n^{\pi}(k)$ for all $k \geq 1$ and for all $\pi \in \Sigma_1(n)$, which concludes the proof. ∎

Now we state the main result of this section.

Theorem 2 *Under assumption* **A1** *policies $ML(n)/F$ and $F/ML(n)$ exhibit the same behavior in the sense that*

$$L_n^{\pi}(t) = L_n^{\gamma}(t), \tag{3.17}$$

for all $t \geq 0$, $\pi \in \Sigma_1(n)$, $\gamma \in \Sigma_2(n)$, $n \geq 1$.

Proof. As a consequence of Lemmas 3 and 5 we need only consider the behavior of π_1 and π_2. The arguments used to prove theorem 1 are applicable here. The only required modification is to treat the discard of a customer at the time of scheduling as the service of a customer with a service time of 0. Following this, the proof is straightforward. ∎

Remark 1 Different discard implementations can affect the number of customers in the composite queue. Our belief is that it may be easier to design simple and efficientsimple discard policies under $ML(n)/F$ than under $F/ML(n)$.

4 Monotonicity Properties

We first recall the definition of the stochastic ordering and of the convex ordering. Let X and Y be two $\mathbb{R}$-valued random variables. We say that X is *stochastically smaller* than Y, and write $X \leq_{st} Y$, if $\mathrm{E}(f(X)) \leq \mathrm{E}(f(Y))$ for any nondecreasing mapping $f : \mathbb{R} \to \mathbb{R}$ such that these expectations exist. We say that X is smaller than Y for the *convex ordering* (resp. *increasing convex ordering*), and write $X \leq_{cx} Y$ (resp. $X \leq_{icx} Y$), if $\mathrm{E}(f(X)) \leq \mathrm{E}(f(Y))$ for any convex (resp. increasing convex) mapping $f : \mathbb{R} \to \mathbb{R}$ such that these expectations exist.

4.1 System without discards

We have the following monotonicity properties for the customer tardiness.

Lemma 6 *If $f : \mathbb{R} \to \mathbb{R}$ is an increasing convex function, then under assumption* **A1**,

$$\sum_{i=1}^{k} f(C_{n+1}(i)) \leq \sum_{i=1}^{k} f(C_n(i)),\ k \geq 1,\ n \geq 1, \tag{4.1}$$

where $C_n(i)$ denotes the tardiness of the i-th scheduled customer under $F/ML(n)$.

Proof. We define a new policy γ_k, $k \geq 0$ which uses the $F/ML(n+1)$ rule during the first k scheduling instances and the $F/ML(n)$ rule during the remaining scheduling instances. We define a cost function

$$F_j(k) = \sum_{i=1}^{j} f(C^{\gamma_k}(i)),\ j \geq 1,\ k \geq 0, \tag{4.2}$$

where $f : \mathbb{R} \rightarrow \mathbb{R}$ is an increasing convex function. We show now that $F_j(k)$ is a non-increasing function of k for every value of j, i.e., $F_j(k) - F_j(k+1) \leq 0$, $j \geq 1$, $k \geq 0$. The lemma follows once this is established.

We need only consider values of k such that $k < j$. First, we recognize that because of assumption **A1**, the time of the l-th scheduling under γ_k is independent of k. Hence, let D_l denote the l-th scheduling time. Consider policies γ_k and γ_{k+1}. There are two cases according to whether the number of customers in the system prior to the $(k+1)$-th scheduling instant is less than $(n+1)$ or not. In the first case, there is no difference in the behavior of γ_k and γ_{k+1}, i.e., $I^{\gamma_k}(i) = I^{\gamma_{k+1}}(i)$, $i \geq 1$. Consequently $F_j(k) - F_j(k+1) = 0$.

Consider now the case that the number of customers in the system prior to the $(k+1)$-st scheduling exceeds n. Represent the state of the system under γ_k as $\mathbf{S}_k(t) = (z_1, z_2, \ldots, z_p)$ where $a_{z_i} \leq a_{z_{i+1}}$, $1 \leq i \leq p-1$, where a_{z_i} is the arrival time of customer z_i. Policy γ_k behaves in the following way. At the l-th scheduling instant, $l \leq k$, γ_k schedules the customer with the smallest deadline from among the $(n+1)$ oldest customers. If $p < n+1$, then γ_k schedules the customer in the queue with the smallest deadline. If $l > k$, then γ_k applies the same rule but only among the oldest n customers. Clearly $I^{\gamma_k}(i) = I^{\gamma_{k+1}}(i)$, $1 \leq i \leq k$. Let $\mathbf{S}^k(D_{k+1}^-) = \mathbf{S}^{k+1}(D_{k+1}^-) = (z_1, \ldots, z_p)$, $p \geq n+1$. We have two further cases according to whether $d_{z_{n+1}} < \min_{1 \leq i \leq n}\{d_{z_i}\}$ or not.

Case i. $d_{z_{n+1}} < \min_{1 \leq i \leq n}\{d_{z_i}\}$. In this case, γ_k schedules the customer, say z_l, from the oldest n customers with the smallest deadline whereas γ_{k+1} schedules z_{n+1}. The states under the two policies prior to the scheduling decision $k+2$ will be $\mathbf{S}^k(D_{k+2}^-) = (z_1, \ldots, z_{l-1}, z_{l+1}, \ldots, z_n, z_{n+1}, z_{n+2}, \ldots, z_{p'})$ and $\mathbf{S}^{k+1}(D_{k+2}^-) = (z_1, \ldots, z_n, z_{n+2}, \ldots, z_{p'})$, $p' \geq 0$. At this point in time γ_k schedules z_{n+1} and γ_{k+1} schedules z_l. Subsequently, $\mathbf{S}^k(t) = \mathbf{S}^{k+1}(t)$ for $t > D_{k+2}$. From this we conclude that $I^{\gamma_k}(k+1) = I^{\gamma_{k+1}}(k+2)$, $I^{\gamma_k}(k+2) = I^{\gamma_{k+1}}(k+1)$, and $I^{\gamma_k}(i) = I^{\gamma_{k+1}}(i)$, $i \geq k+3$. Hence, for $j \geq k+2$

$$\begin{aligned} F_j(k) - F_j(k+1) &= f(C^{\gamma_k}(k+1)) + f(C^{\gamma_k}(k+2)) - f(C^{\gamma_{k+1}}(k+1)) - f(C^{\gamma_{k+1}}(k+2)), \\ &= f(D_{k+1} - d_{z_l}) + f(D_{k+2} - d_{z_{n+1}}) - f(D_{k+1} - d_{z_{n+1}}) - f(D_{k+2} - d_{z_l}), \\ &\geq 0. \end{aligned}$$

The latter inequality is a consequence of the convexity of f and the fact that $d_{z_l} > d_{z_{n+1}}$. For $j = k+1$, we have

$$\begin{aligned} F_j(k) - F_j(k+1) &= f(C^{\gamma_k}(k+1)) - f(C^{\gamma_{k+1}}(k+1)), \\ &= f(D_{k+1} - d_{z_l}) - f(D_{k+1} - d_{z_{n+1}}), \\ &\geq 0. \end{aligned}$$

where the last inequality is a consequence of the increasingness of f and the fact that $d_{z_l} > d_{z_{n+1}}$.

Case ii. $d_{z_{n+1}} \geq \min_{1 \leq i \leq n}(d_{z_i})$. In this case γ_k and γ_{k+1} schedule the same customer at the $(k+1)$-th scheduling instance, say z_i for some $i \leq n$. Following this, the policies behave in an identical manner and we have $I^{\gamma_k}(i) = I^{\gamma_{k+1}}(i)$, $i \geq k+1$ from which we conclude that $F_j(k) = F_j(k+1)$.

This completes the proof of the lemma. ∎

Assume that a stationary regime exists, i.e. $\mathrm{E}[\sigma] < c\,\mathrm{E}[\theta]$. Let C_n and W_n denote the stationary customer tardiness and the stationary wait time under the $F/ML(n)$ policy, respectively. We introduce the following assumption (see Remark 2):

- **A4** $c = 1$ (single server case) or $c > 1$ (multiserver case) and $\mathrm{P}(\theta > \sigma) > 0$.

Theorem 3 *Let the service times be associated with the customers. Then, under assumptions* **A2** *and* **A4**,

$$E[f(C_{n+1}] \leq E[f(C_n)], \tag{4.3}$$

for all $n \geq 1$ *and for any convex mapping* $f : \mathbb{R} \to \mathbb{R}$ *such that* $E[|f(C_n|] < \infty$ *and* $E[|f(C_{n+1})|] < \infty$.

Proof. The proof is given in appendix.

Remark 2 When $c > 1$ assumption **A4** covers most cases of interest such as interarrival times that are unbounded (e.g. exponential random variables, r.v.'s with rational Laplace-Stieltjes transform, etc.) and service times that can take values arbitrarily close to zero (exponential random variables, uniform distribution in $[0,1]$, etc.).

4.2 System with discards

Our monotonicity result for this class of systems is based on the comparison of sets of deadlines. We will show that the set of deadlines for live customers waiting under $F/ML(n+1)$ *dominates* the set of deadlines under $F/ML(n)$. Consequently, we begin with our definition of dominance and derive properties that it satisfies.

Consider two sets of nonnegative real numbers $R = \{\alpha_1, \ldots, \alpha_l\}$ and $S = \{\beta_1, \ldots, \beta_m\}$ each ordered so that $\alpha_i \leq \alpha_{i+1}$, $i = 1, \ldots, l$ and $\beta_i \leq \beta_{i+1}$, $i = 1, \ldots, m-1$.

Definition 1 *We say that R dominates S ($R \succ S$) if $l \geq m$ and $\alpha_{i+l-m} \geq \beta_i$, $i = 1, 2, \ldots m$.*

We define the following two operations:

- $Large(R,k) = \{\alpha_{l-k+1}, \alpha_{l-k+2}, \ldots, \alpha_l\}$, $0 \leq k \leq l$;
- $Shift(R,\alpha) = \{\alpha_i - \alpha \mid \alpha_i \geq \alpha\}$.

The following lemma gives conditions under which dominance is preserved when set operations, the *Large* operation, and the *Shift* operation are performed on R and S.

Lemma 7 *If $R \succ S$, then:*

1. $R \cup \{\alpha\} \succ S \cup \{\alpha\}$, *for* $\alpha > 0$;
2. $R - \{\alpha_1\} \succ S$, *when* $l > m$;
3. $R \succ S - \{\beta\}$, *where* $\beta \in S$;
4. $R - \{\alpha\} \succ S - \{\beta\}$, *where* $\alpha \in R$, $\beta \in S$, *and* $\alpha \leq \beta$;
5. *Assume that* $R = \{\alpha_1, \ldots, \alpha_l\}$ *where* $\alpha_i \leq \alpha_{i+1}$, $1 \leq i < l$ *and* $S = \{\beta_1, \ldots, \beta_m\}$ *where* $\beta_i \leq \beta_{i+1}$, $1 \leq i < m$. *Then* $R - \{\alpha_k\} \succ S - \{\beta_j\}$ *for* $k \leq j + l - m$;
6. $Shift(R,\alpha) \succ Shift(S,\alpha)$;
7. $Large(R, |S|) \succ S$.

Proof: The proof of 1, 2, 3, and 6 may be found in [7]. Properties 4, 5 and 7 follow from the operations performed on R and S and the definition of "$\succ$" (observe that 4 $\Rightarrow$ 5). ∎

The main result of this section is:

Theorem 4 *Let the service times be associated with the customers. Under assumption* **A3**,

$$L_{n+1}^{\pi_2}(t) \leq_{st} L_n^{\pi_2}(t),$$

for all $t \geq 0$, $n \geq 1$.

Proof. Condition the behavior of the system on a specific sequence of arrival times, service times, and deadlines. Let $C = \{c_1, \ldots, c_m\}$ be a set of m customers. Let $\mathcal{D}(C) = \{d_{c_1}, \ldots, d_{c_m}\}$ be the set of deadlines associated with the customers in C. Let $\{t_i\}_{i=1}^{\infty}$ be the times of events corresponding to arrivals, departures, and deadline misses under either policy. Let $\mathbf{S}_{j,i} = (\mathbf{X}_{j,i}; \mathbf{Y}_{j,i})$ be the state of the system when the parameter values are $j = n, n+1$ at time t_i where $\mathbf{Y}_{j,i} = (y_{j,i,1}, \ldots, y_{j,i,p_{j,i}})$, $i \geq 1$. We will show

$$\mathcal{D}(\mathbf{X}_{n+1,i} \cup \mathbf{Y}_{n+1,i}) \succ \mathcal{D}(\mathbf{X}_{n,i} \cup \mathbf{Y}_{n,i}), \quad i \geq 1. \tag{4.4}$$

This is equivalent to showing

- $\mathcal{D}(\mathbf{X}_{n+1,i} \cup \mathbf{Y}_{n+1,i}) \succ \mathcal{D}(\mathbf{X}_{n,i} \cup \mathbf{Y}_{n,i})$ whenever $p_{n,i} \leq 1$;
- $\mathcal{D}(\mathbf{X}_{n+1,i} \cup \{y_{n+1,i,1}), \ldots, y_{n+1,i,r}\} \succ \mathcal{D}(\mathbf{X}_{n,i} \cup \{y_{n,i,1}\})$ where $r = p_{n+1,i} - p_{n,i} + 1$, whenever $p_{n,i} > 1$, and $p_{n,i} \leq p_{n+1,i} + 1$. Note that in this case $y_{n+1,i,r+l} = y_{n,i,1+l}$ for $l = 1, \ldots p_{n,i} - 1$, i.e., these are the newest customers in the FIFO queue of the queue. By application of property 1 in Lemma 7, we obtain relation (4.4) in this case.

Basis step. Trivially true for t_1.

Inductive step. Assume that the above dominance relations are true at times $t_1, t_2, \ldots, t_{k-1}$. We now establish them for t_k. We consider separately arrivals, departures and deadline misses.

Customer arrival. Property 1 of Lemma 7 can be applied in this instance to show that dominance holds at $t = t_k$.

Customer departure. Because the service times are exponentially distributed, we can couple them so that either there is a departure under both policies or only under $F/ML(n+1)$ (this only occurs whenever $F/ML(n)$ is empty). If $|\mathbf{X}_{n,k-1}| = 0$, then the properties trivially hold at time $t = t_k$. If $|\mathbf{X}_{n,k-1}| > 0$, then the properties can be shown to hold through the use of property 5 of Lemma 7.

Deadline miss. If the deadline miss is in the FIFO queue of both systems, then nothing is affected. If it is in the ML queue under $F/ML(n)$, then property 4 of Lemma 7 is applicable. If the miss is under $F/ML(n)$ only, then property 3 can be applied. Finally, if the deadline

miss is under $F/ML(n+1)$, then clearly $\mathcal{D}(\mathbf{X}_{n+1,k-1} \cup \mathbf{Y}_{n+1,k-1}) \succ \mathcal{D}(\mathbf{X}_{n,k-1} \cup \mathbf{Y}_{n,k-1})$ and property 2 is applicable.

This completes the inductive step.

Clearly the dominance that we have shown between the sets of deadlines under the two policies implies that there are fewer losses under $F/ML(n+1)$ by time t. Removal of the conditioning on the arrival times, service times, and deadlines yields the desired result. ∎

A Appendix

Proof of Theorem 3: As a consequence of Lemma 6

$$\sum_{k=1}^{K} E[f(C_n(k))] \leq \sum_{k=1}^{K} E[f(C_{n+1}(k))], \tag{A.1}$$

when f is an increasing convex function. Note that the independence of the service times are required so as to allow service time interchanges as required in the previous lemma.

For sake of simplicity and without loss of generality we assume that the first busy period starts at time 0 (i.e. $a_1 = 0$ and $W_n^{\pi_2}(1) = 0$, see Section 2). We first notice that under assumption **A4** the system empties with probability one (see Sigman [8], p. 395, for the case where $c > 1$ and $\mathrm{P}(\theta > \sigma) > 0$). We next observe that the busy period durations do not depend on n. For $c = 1$ this directly follows from the fact that the $F/ML(n)$ policies are work conserving and because the service times do no depend on the scheduling policy. For the same reason this is also true when $c > 1$ since the service times are given to the customers at scheduling epochs.

Let $\underline{n}_i$ be the number of customers served in the i-th busy period under policy $F/ML(n)$, $i \geq 1$. Note that $\underline{n}_i$ does not depend on n from what preceeds. Under assumptions **A2**, **A4** it is easily seen that the sequence $\{f(C_n(k)\}_{k=1}^{\infty}$ is regenerative with respect to the renewal sequence $\{\underline{n}_i\}_{i=1}^{\infty}$. Therefore (see Cohen [2]):

$$\mathrm{E}[f(C_n)] = \frac{1}{E[\underline{n}]} \mathrm{E}\left[\sum_{k=1}^{\underline{n}} f(C_n(k))\right], \tag{A.2}$$

where $\underline{n}$ is the number of customers served in a busy period (note that $\mathrm{E}[\underline{n}] < \infty$ under our assumptions).

Define

$$\nu(K) = \min\{i : \underline{n}_1 + \cdots + \underline{n}_i > K\} - 1. \tag{A.3}$$

Then,

$$\begin{aligned}
\mathrm{E}\left[\sum_{k=1}^{K} f(C_n(k))\right] &= \mathrm{E}\left[\sum_{i=1}^{\nu(K)} \sum_{k=1}^{\underline{n}_i} f(C_n(k+\underline{n}_1+\cdots+\underline{n}_{i-1}))\right] \\
&+ \mathrm{E}\left[\sum_{k=\underline{n}_1+\cdots+\underline{n}_{\nu(K)}+1}^{K} f(C_n(k))\right], \\
&= \mathrm{E}\left[\sum_{i=1}^{\nu(K)} \sum_{k=1}^{\underline{n}_i} f(C_n(k))\right] + \mathrm{E}\left[\sum_{k=1}^{K-(\underline{n}_1+\cdots+\underline{n}_{\nu(K)})} f(C_n(k))\right], \quad \text{(A.4)}
\end{aligned}$$

for $n \geq 1$ by using the fact that $\{f(C_n(k))\}_{k=1}^{\infty}$ is regenerative with respect to $\{\underline{n}_i\}_{i=1}^{\infty}$. Let $X_n(i) = \sum_{k=1}^{\underline{n}_i} f(C_n(k))$ for $i \geq 1$, $n \geq 1$.

Clearly, for fixed $n \geq 1$, $\{X_n(i)\}_{i=1}^{\infty}$ is an i.i.d. sequence of r.v.'s. Let $\mathcal{F}_i$ be the σ–field generated by the random variables $\{\underline{n}_1, \ldots, \underline{n}_i\}$, $i \geq 1$. Then, it is easily seen that $\nu(K)$ is an $\mathcal{F}_i$–stopping time (i.e. $\{\nu(K) \leq i\} \subset \mathcal{F}_i$ for all $i \geq 1$). Since $\mathcal{F}_i \cap \sigma(X_n(i+1)) = \emptyset$, Wald's identity applies (Loeve [5], p. 377), and so

$$\begin{aligned}
\mathrm{E}\left[\sum_{i=1}^{\nu(K)} \sum_{k=1}^{\underline{n}_i} f(C_n(k)\right] &= \mathrm{E}\left[\sum_{k=1}^{\underline{n}} f(C_n(k))\right] \mathrm{E}[\nu(K)], \\
&= \mathrm{E}[f(C_n)]\,\mathrm{E}[\underline{n}]\,\mathrm{E}[\nu(K)], \quad \text{(A.5)}
\end{aligned}$$

from (A.2).

Combining (A.1), (A.4) and (A.5), dividing by K and letting K go to infinity yields

$$\begin{aligned}
\mathrm{E}[f(C_n)] - \mathrm{E}[f(C_{n+1})] &\geq \lim_{K\to\infty} \frac{1}{K} \mathrm{E}\left[\sum_{k=1}^{K-(\underline{n}_1+\cdots+\underline{n}_{\nu(K)})} f(C_n(k))\right] \\
&- \lim_{K\to\infty} \frac{1}{K} \mathrm{E}\left[\sum_{k=1}^{K-(\underline{n}_1+\cdots+\underline{n}_{\nu(K)})} f(C_{n+1}(k))\right], \quad \text{(A.6)}
\end{aligned}$$

where we used the renewal theorem (see e.g. [3], p. 87) to show that $\lim_{K\to\infty} \mathrm{E}[\nu(K)]/K = 1/\mathrm{E}[\underline{n}]$.

Let us show that both expectations in the right-hand side of (A.6) are uniformly bounded in K which will complete the proof of (4.3) in the case where f is convex increasing and when the service times are given at scheduling epochs.

Fix $n \geq 1$. We have

$$\begin{aligned}
\left| \mathrm{E}\left[\sum_{k=1}^{K-(\underline{n}_1+\cdots+\underline{n}_{\nu(K)})} f(C_n(k)) \right] \right| &\leq \mathrm{E}\left[\sum_{k=1}^{K-(\underline{n}_1+\cdots+\underline{n}_{\nu(K)})} |f(C_n(k))| \right], \\
&\leq \mathrm{E}\left[\sum_{k=1}^{\underline{n}} |f(C_n(k))| \right], \\
&= \mathrm{E}[|f(C_n)|]\,\mathrm{E}[\underline{n}], \qquad \text{(A.7)} \\
&< \infty,
\end{aligned}$$

from the assumption on the integrability of $|f(C_n)|$ and where to establish (A.7) we used the property that the sequence $\{|f(C_n(k))|\}_{k=1}^{\infty}$ is regenerative with respect to the renewal sequence $\{\underline{n}_i\}_{i=1}^{\infty}$.

It remains to prove that (4.3) actually holds for any convex function. If we can show that $\mathrm{E}[C_n] = \mathrm{E}[C_{n+1}]$ then it follows from Theorem 1.3.1 in [9] that $C_n \geq_{cx} C_{n+1}$. Now, $\mathrm{E}[C_n] = \mathrm{E}[W_n] - \mathrm{E}[\tau]$. We claim that $\mathrm{E}[W_n]$ does not depend on the value of n. Indeed, since the wait time sequence $\{W_n^{\pi_2}(k)\}_{k=1}^{\infty}$ is regenerative with respect to the renewal sequence $\{\underline{n}_i\}_{i=1}^{\infty}$, we have

$$\begin{aligned}
\mathrm{E}[W_n] &= \frac{1}{\mathrm{E}[\underline{n}]}\,\mathrm{E}\left[\sum_{k=1}^{\underline{n}} W_n^{\pi_2}(k) \right], \\
&= \frac{1}{\mathrm{E}[\underline{n}]}\,\mathrm{E}\left[\sum_{k=1}^{\underline{n}} D_n^{\pi_2}(k) - a_{I_n^{\pi_2}(k)} \right], \\
&= \frac{1}{\mathrm{E}[\underline{n}]}\,\mathrm{E}\left[\sum_{k=1}^{\underline{n}} D_n^{\pi_2}(k) \right] - \frac{1}{\mathrm{E}[\underline{n}]}\,\mathrm{E}\left[\sum_{k=1}^{\underline{n}} a_{I_n^{\pi_2}(k)} \right]. \qquad \text{(A.8)}
\end{aligned}$$

Now, since the arrival and service time processes over a busy period do not depend on n it is easily seen that both expectations in the right-hand side of (A.8) do not depend on n as well.

The last step is to prove that (4.3) still holds when the service times are associated with the customers. Fix the input sequences $\{\theta_k, \sigma_k, \tau_k\}_{k=1}^{\infty}$ and let the system run under the $F/ML(n)$ policy when the service times are associated with the customers. Let $I_n(k)$ be the identity of the k-th customer to enter service, $k \geq 1$. Assume now that the input sequences are $\{\theta_k, \sigma_{I_n}(k), \tau_k\}_{k=1}^{\infty}$ and let the system run under the $F/ML(n)$ policy when the service time are given at scheduling epochs (i.e. the service time of the k-customer to enter service is $\sigma_{I_n(k)}$). Clearly, both systems exhibit the same behavior, and moreover, the distributions of both input sequences are the same under assumption **A2**. Therefore, $\mathrm{E}[f(C_n)]$ has the same value under

both service assignment policies, which concludes the proof. ∎

References

[1] P. P. Bhattacharya, A. Ephremides, Optimal scheduling with strict deadlines, *IEEE Trans. Automatic Control* **34**, 7, 721-728 (1989).

[2] J. W. Cohen, *On Regenerative Processes in Queueing Theory*, Lect. Notes Econ. and Math. Syst. **121**, Springer Verlag, New York (1976).

[3] A. Gut, *Stopped Random Walks. Limit Theorems and Applications*, Appl. Prob. Series **5**, Springer Verlag, New York (1988).

[4] J. Hong, X. Tan, D. Towsley, A performance analysis of minimum laxity and earliest deadline scheduling in a real-time system, *IEEE Transactions on Computers* **C-38**, 12, 1736-1744 (1989).

[5] M. Loeve, *Probability Theory I*, 4th ed. Springer Verlag, New York (1977).

[6] S. S. Panwar, Time Constrained and Multi-access Communications, Ph.D. Thesis, Department of Electrical and Computer Engineering, University of Massachusetts, Amherst, Feb. 1986.

[7] S. S. Panwar, D. Towsley, and J. K. Wolf, Optimal scheduling policies for a class of queues with customer deadlines to the beginning of service, *Journal of the ACM* **35**, 4, 832-844 (1988).

[8] K. Sigman, Regeneration in tandem queues with multiserver stations, *J. Appl. Prob.* **25**, 391-403 (1988).

[9] D. Stoyan, *Comparison Methods for Queues and Other Stochastic Processes*, John Wiley & Sons (1983).

[10] D. Towsley, F. Baccelli, Comparisons of service disciplines in a tandem queueing network with real-time constraints, COINS Technical Report 89-63, July 1989. To appear in *Operations Research Letters*.

[11] D. Towsley, S. S. Panwar, On the optimality of minimum laxity and earliest deadline scheduling for real-time multiprocessors, to appear in *Proceedings of the Euromicro Real-Time Workshop*, Denmark, June 1990.

[12] W. Zhao, J. A. Stankovic, Performance evaluation of FCFS and improved FCFS scheduling for dynamic real-time computer systems, *Proc. Real-Time Systems Symposium*, 156-165, Dec. 1989.

Imprimé en France
par
l'Institut National de Recherche en Informatique et en Automatique

ISSN 0249-6399