THE BINARY SIMULATION OF THE MINIMUM LAXITY AND EARLIEST DEADLINE SCHEDULING POLICIES FOR REAL-TIME SYSTEMS

J. Hong, X. Tan, and D. Towsley
Department of Computer and Information Science
University of Massachusetts
Amherst, MA 01003

COINS Technical Report 89-70

July 1989

The Binary Simulation of the Minimum Laxity and Earliest Deadline Scheduling Policies for Real-Time Systems

J. Hong,*
X. Tan, D. Towsley†
Department of Computer & Information Science
University of Massachusetts
Amherst, MA 01003

July 1989

Abstract

The behavior of the minimum laxity scheduling policy on a multiprocessor and of the earliest deadline scheduling policy on a single processor are examined in this paper. The state of the system under either policy can be represented by binary strings and the state evolution is described by a simple transition function. Formal arguments are given to show that these binary string representations and the transition functions may be used to simulate these policies correctly. This model can be used to generate probabilistic models for estimating the performance of both policies.

*The work of this author was partially supported by the National Science Foundation under grant DCI-87-96236.

†The work of the last two authors was partially supported by the Office of Naval Research under contract N00014-87-k-0796 and the National Science Foundation under grant DCR-85-00332.

1 Introduction

The design and analysis of policies for scheduling customers with real-time constraints is receiving increasing attention these days. When customers have deadlines until the beginning of service, it has been shown that the minimum laxity policy maximizes the fraction of customers that begin service by their deadlines [2] for a large class of systems. When the deadlines are until the end of service, similar results have been established for the earliest deadline policy [2]. The problem of analyzing the *performance* of these policies has not been solved. A primary reason for the lack of a successful analysis is the lack of a formal mathematical model to describe the behavior of these policies.

The purpose of this paper is to remove this deficiency. A formal model is developed which represents the state of the system at some point in time t under either the minimum laxity or earliest deadline policies by a *diagram* consisting of groups of customers and binary strings. There is a binary string associated with each group of customers that encodes a partial history of the system up until that time t. Simple rules are given that describe how the state of the system changes over time in response to different events such as arrivals, departures, and deadline misses. Formal arguments are given to show that the diagrams and transition rules can be used to correctly simulate both the minimum laxity policy on multiple servers and the earliest deadline policy on a single server. Last, these results are applied in several cases to obtain Markov chains representations of these two policies.

The paper is structured as follows. In sections 2 and 3 we focus on the minimum laxity scheduling policy. Section 2 introduces the concept of a diagram and a transition function that can be used to describe the dynamics of the system. Section 3 contains formal arguments that the diagram and transition function can be used to correctly simulate minimum laxity. Section 4 outlines similar results for the earliest deadline scheduling policy, and section 5 summarizes the paper.

2 Events, History and the Minimum Laxity Policy

In this section, we develop a descriptive model of the minimum laxity scheduling policy.

Definition 1 *The minimum laxity policy (ML) is the non-idling policy that schedules the customer closest to its deadline. Customers that miss their deadlines while in the queue are removed from the queue.*

We consider its behavior in a system containing c identical servers. Given a set of customers $\{c_i\}_{i=1}^{\infty}$ with arrival times $a = \{a_i\}_{i=1}^{\infty}$, such that $0 \leq a_1 < a_2 < \cdots$, service times $\sigma = \{\sigma_i\}_{i=1}^{\infty}$, and deadlines $d = \{d_i\}_{i=1}^{\infty}$, then the behavior of ML can be represented by a sequence of events coupled with the times at which they occur. There are three events corresponding to arrivals, departures and deadline misses. An arrival of customer c_i will be denoted by i. A departure of a customer will be denoted by $\underline{i}$ where c_i is the customer that is scheduled to the server following the departure. The event that no customer enters service is denoted by $\underline{0}$. Last, $\bar{i}$ corresponds to the event that the deadline of customer c_i expires. We use the notation E to denote an event. and $s = (a, \sigma, d)$ to denote the *input sample.*

Definition 2 *A history $H(s)$ is a sequence $(C_1, t_1), (C_2, t_2), \cdots$ where C_i denotes the i-th event and t_i the time at which it occurs. The history depends on the input sample s.*

Example. Consider a single server system. Let $a = \{1, 5, 7, 10, 11, 15\}$, $\sigma = \{8, 4, 69, 12, 16, 8\}$ and $d = \{32, 22, 27, 25, 90, 33\}$. Th history of this system is

$$(1,1),(2,5),(3,7),(\underline{2},8),(4,10),(5,11),(\underline{4},12),(6,15),(\bar{2},22),$$
$$(\underline{3},24),(\bar{4},25),(\bar{3},27),(\bar{1},32),(\bar{6},33),(\bar{5},90),(\underline{0},93)$$

Definition 3 *An event history $H_e(s)$ is the sequence of events $E_1, E_2, \cdots$ corresponding to the history $H(s)$.*

The event history for the above example is

$$1,2,3,\underline{2},4,5,\underline{4},6,\bar{2},\underline{3},\bar{4},\bar{3},\bar{1},\bar{6},\bar{5},\underline{0}$$

When there is no source of confusion, we will omit the argument s from $H(\cdot)$ and $H_e(\cdot)$.

We now introduce the concept of a *diagram* which can be thought of as a partial state description of the system. Its usefulness in describing the behavior of ML will become apparent in section 3.

Definition 4 *A diagram is either an integer from the set $\{0, 1, \cdots, c\}$ or a sequence of $g > 0$ groups. The ith group consists of a set S_i containing n_i distinct customer identities and a word $w_i \in 1^+0^+$ of length n_i, The word in the last group need not contain 0's, i.e., $w_g \in 1^+0^*$.*

The system can be represented by a diagram after each event. We describe now how the diagram changes with the occurrence of each event. Let D_i denote the diagram for the system after the i-th event. We define a function ψ that takes two arguments, an event and a diagram, and produces a new diagram.

- *The arrival event j:*

  If the diagram consists of an integer less than c, then the next diagram corresponds to the next integer,

  $$\psi(j,n) = n+1, \ \ 0 \leq n < c.$$

  If the diagram consists of the integer c, then the next diagram consists of a single group containing the customer that arrived and $w = 1$,

  $$\psi(j,c) = \{j\}1.$$

  If the diagram consists of a sequence of groups and the word associated with the last group consists of all 1's, then add the customer to the last group and add an additional 1 to the corresponding binary string,

  $$\psi(j, DS1^{|S|}) = D(S \cup \{j\})1^{|S|+1}; \ D \in \mathcal{D}^+; \ S \subset \boldsymbol{N}.$$

  where $\boldsymbol{N}$ is the set of natural numbers, $\mathcal{D}^+$ is the set of diagrams of which the associated word of the last group has at least one 0. The set of all possible diagrams is denoted by $\mathcal{D}$. The formal descriptions of $\mathcal{D}^+$ and $\mathcal{D}$ are given later in this section.

  If the word associated with the last group ends in a 0, then add a new group consisting of the arriving customer and associate a 1 with it,

  $$\psi(j, DSw0) = DSw0\{j\}1; \ D \in \mathcal{D}^+; \ S \subset \boldsymbol{N}; \ w \in 1^+0^*; \ |w| = |S|.$$

- *The deadline miss event $\bar{j}$:*

  If the customer that misses its deadline is not contained in any group, then the diagram does not change,

  $$\psi(\bar{j}, n) = n, \ \ 0 \leq n < c,$$

  $$\psi(\bar{j}, D) = D, \quad D \in \mathcal{D}; j \notin S(D); \ |S(D)| \geq 1.$$

Here $S(D)$ is the union of all of the groups of customers, $S(D) = \cup_{i=1}^{g} S_i$. If the customer belongs to a group whose associated word has at least two zeroes, then remove the customer from the group and remove one zero from the word,

$$\begin{aligned}\psi(\bar{j}, D_1 S w 00 D_2) &= D_1(S - \{j\}) w 0 D_2; D_1 \in \mathcal{D}^+; D_2 \in \mathcal{D}; \\ & j \in S; S \subset \boldsymbol{N}; w \in 1^+0^*.\end{aligned}$$

If the customer belongs to a group for which the associated word contains a single 0, then merge the group with the next group after removing the customer and the 0,

$$\begin{aligned}\psi(\bar{j}, D_1 S_1 1^{|S_1|-1} 0 S_2 w D_2) &= D_1(S_1 \cup S_2 - \{j\}) 1^{|S_1|-1} w D_2, D_1 \in \mathcal{D}^+; D_2 \in \mathcal{D}; \\ & j \in S_1; S_1, S_2 \subset \boldsymbol{N}.\end{aligned}$$

If the customer is contained in the last group and the associated word consists of no 0, then remove that customer from the last group and delete one 1 from the associated word.

$$\psi(\bar{j}, D S 1^{|S|}) = D(S - \{j\}) 1^{|S|-1}), \quad D \in \mathcal{D}.$$

If the last group consists solely of that customer, then remove that group from the diagram,

$$\begin{aligned}\psi(\bar{j}, D\{j\}1) &= D, \quad D \in \mathcal{D}, \\ \psi(\bar{j}, \{j\}1) &= c.\end{aligned}$$

- *The departure event $\underline{j}$:* If the present diagram is a positive integer, then the next diagram is that integer less one,

$$\psi(\underline{j}, n) = n - 1, \quad 1 \le n \le c.$$

If the word associated with the last group contains two or more 1's, then the next diagram is obtained by changing the rightmost 1 to 0,

$$\psi(\underline{j}, D S 1^n 0^{|S|-n}) = D S 1^{n-1} 0^{|S|-n+1}, \quad D \in \mathcal{D}; S \subset \boldsymbol{N}, |S| \ge 2; \ 1 < n \le |S|.$$

If the word associated with the last group contains only one 1, then the last group is merged with the next to last group,

$$\psi(\underline{j}, D S_1 w S_2 1 0^{|S_2|-1}) = D(S_1 \cup S_2) w 0^{|S_2|}, D \in \mathcal{D}; \ S_1, S_2 \subset \boldsymbol{N}; \ w \in 1^+0^+; \ |w| = |S_1|.$$

If the present diagram consists of a single group whose associated word contains one 1, then the next diagram is the integer c

$$\psi(D, S10^{|S|-1}) = c, \quad S \subset \boldsymbol{N}.$$

Here $\mathcal{D}^+$ and $\mathcal{D}$ are defined as

$$\begin{aligned} \mathcal{D}^+ &= \{S_1 w_1 S_2 w_2 \cdots S_g w_g : S_i \subset \boldsymbol{N}, w_i \in 1^+0^+, 1 \le i \le g, \\ &\qquad S_i \cap S_j = \emptyset, i \ne j, 1 \le i, j \le g, g = 0, 1, \cdots\} \end{aligned}$$

and

$$\begin{aligned} \mathcal{D} &= \{S_1 w_1 S_2 w_2 \cdots S_g w_g : S_i \subset \boldsymbol{N}, w_i \in 1^+0^+, 1 \le i \le g-1, w_g \in 1^+0^*, \\ &\qquad S_i \cap S_j = \emptyset, i \ne j, 1 \le i, j \le g, g = 0, 1, \cdots\}. \end{aligned}$$

Beginning with the diagram 0, the sequence of diagrams corresponding to the example in the previous section is given below,

$$\begin{aligned} 0 \xrightarrow{1} 1 \xrightarrow{2} \{2\}1 \xrightarrow{3} \{2,3\}11 \xrightarrow{\underline{2}} \{2,3\}10 \xrightarrow{4} \{2,3\}10\{4\}1 \xrightarrow{5} \\ \{2,3\}10\{4,5\}11 \xrightarrow{\underline{4}} \{2,3\}10\{4,5\}10 \xrightarrow{6} \\ \{2,3\}10\{4,5\}10\{6\}1 \xrightarrow{\bar{2}} \{3,4,5\}110\{6\}1 \xrightarrow{\underline{3}} \\ \{3,4,5\}100\{6\}1 \xrightarrow{\bar{4}} \{3,5\}10\{6\}1 \xrightarrow{\bar{3}} \\ \{5,6\}11 \xrightarrow{\bar{1}} \\ \{5,6\}11 \xrightarrow{\bar{6}} \{5\}1 \xrightarrow{\bar{5}} 1 \xrightarrow{\underline{0}} \ 0 \end{aligned}$$

We let $\phi(i, s)$ denote the diagram that occurs from applying the first i events of $H_e(s)$ to the initial diagram 0. Again, when there is no source of confusion, we will omit the parameter s.

3 The meaning of the diagrams

Consider a set of arrival times, service times and deadlines. Let $D = \phi(j) = \phi(j, s)$. Let $S_i(D)$ denote the i-th group, $w_i(D)$ the binary word associated with $S_i(D)$, $m_i(D)$ the total number of 0's in $w_i(D)$, and $n_i(D) = |S_i(D)|$ the total number of customers in set

$S_i(D)$. When there is no ambiguity, we will omit the argument D. Consider a finite set of positive real numbers, T. We define $Small(n,T)$ to be the subset of T that contains the n smallest elements in T. Let S be a set of customers. We define $Min(m,S)$ to be the subset of S, which consists of the m customers with the smallest deadlines. In other words, if $T = \{d_k : k \in S\}$, then $k \in Min(m,S)$ iff $d_k \in Small(m,T)$.

Lemma 1 *If S, T are mutually exclusive sets of customers such that $|S| = m$, then*

$$((T - Min\{n,T\}) \cup S) - Min\{m,(T - Min\{n,T\}) \cup S\} = (T \cup S) - Min\{n+m, T \cup S\}$$

whenever $n \leq |T|$.

Proof. Let $A = (T \cup S) - Min\{n+m, T \cup S\}$ and $B = ((T - Min\{n,T\}) \cup S) - Min\{m,(T - Min\{n,T\}) \cup S\}$. Assume $x \in A$ and $x \notin B$. This implies that there are n elements in $Min\{n,T\}$ that are less than x and m additional elements in $(T - Min\{n,T\})$ less than x. Hence we conclude that $x \notin A$ which contradicts our assumption. Hence we have $A \subseteq B$. The Lemma follows from this and the fact that $|A| = |B|$. □

This lemma states that if we remove the n smallest elements from a set T, add m new elements and remove the smallest m elements from the resulting set, we obtain a set identical to the one obtained by adding the m elements first to T and then removing the $n + m$ smallest elements from this set.

Consider a history under ML up to and including the t-th event. Let $D = \phi(t)$ contain g groups $S_1, S_2, \cdots, S_g$. The event history can be divided into $g + 1$ periods labeled $i = 0, 1, \cdots, g$. Period 0 contains all events prior to the arrival of any customer contained within S_1. Period i begins with the event of the first arrival of a customer in S_i and contains all events prior to the arrival of any customer in S_{i+1}, $i = 1, 2, \cdots$. Last, period g begins with the arrival of the first customer within S_g and includes all events up through E_t. An illustration of this decomposition is given for diagrams $\{2,3\}10\{4,5\}10\{6\}1$ and $\{3,5\}10\{6\}1$ from our earlier example. For the first of these diagrams, there are four periods, period 0 - 1, period 1 - 2,3,4, period 2 - 5,6,7, period 3 - 8. The second diagram has the following three periods, period 0 - 1,2, period 1 - 3,4,5,6,7, period 2 - 8,9,10,11.

Clearly, all customers contained in the set $S_1 \cup S_2 \cup \cdots \cup S_g$ have arrived before time t and are alive, since we have added their names to this set when they arrived, and have deleted their names if they die before time t. More specifically, S_i is the set of customers that are still alive and that arrived during the i-th period, $i = 1, \cdots, g$. We can prove the following lemma.

Lemma 2 *Consider the system at time t. Under ML, the number of customers alive at time t that departed during the i-th period is equal to* m_i, $i = 1, \cdots, g$.

Proof. The proof is by induction on t.
Basis Step. It is obviously true when $t = 0$, since the diagram is the integer 0 and there are no periods.
Inductive Step. Now assume that the lemma is true for t. We show that it is also true for $t+1$. There are three cases corresponding to the different events that can occur at time t. The parameters g, S_i, w_i, and m_i, $i = 1, \cdots, g$ correspond to t and g', S'_i, w'_i, and m'_i correspond to $t+1$.

Case one. If the $(t+1)$-th event is an arrival of a new customer, then there are two subcases.
Subcase 1. The associated word w_g of the last group ends with a 1. In this case, the new customer is placed into the last group and a 1 is added to the word w_g. This produces $w'_i = w_i$, $S_i = S'_i$, and $m'_i = m_i$ for $i = 1, 2, \cdots, g-1$. The lemma is true in this case for $t+1$ since no new departure occurred during the last period.
Subcase 2. The string w_g ends with a 0. In this case we add a new group, the $g' = (g+1)$-th one. The lemma is clearly true in this case.

Case two. The $(t+1)$-th event is a deadline miss, $\overline{k}$. Let $c_k \in S_i$. There are three subcases.
Subcase 1. The string w_i contains more than one 0. In this case, one 0 is deleted and c_k is removed from S_i. We have $m'_j = m_j$, $S'_j = S_j$ for $j \neq i$ and $m'_i = m_i - 1$, $S'_i = S_i - \{k\}$. The lemma holds in this case provided that c_k departed during the i-th period. We show this by contradiction. Assume that c_k did not depart the queue during the i-th period. Since the policy is ML, c_k could not have been in the queue when the customers corresponding to the m'_i 0's in w'_i departed. Hence the arrival of c_k could only have occurred during period $j > i$. This contradicts the fact that $k \in S_i$.
Subcase 2. There is only one 0 in w_i. In this case, we delete this 0, remove the customer's name from the i-th group and merge group i with group $i+1$, if $i < g$. We have $S'_i = S_i \cup S_{i+1} - \{k\}$. An application of lemma 1 along with an argument similar to that given for the previous subcase yields the desired result.

Subcase 3. There is no 0 in w_i. In this case the ith group must be the last group; we delete the name of the customer from the last group and delete a 1 from the word w_i. The lemma is obviously true.

Case three. If the $(t+1)$-th event is a departure, we convert the rightmost 1 to 0. There are two subcases.
Subcase 1. If there is more than one 1 in w_g, then the rightmost 1 is converted to 0. Then we have $m'_g = m_g + 1$, which is exactly the total number of customers that are still alive at time $t+1$ and that departed during the last period.
Subcase 2. If there is only one 1 in w_g, then this 1 converts to a 0, the last two words are concatenated, and the last two groups are merged. That is, $g' = g-1$, $S'_{g'} = S_{g-1} \cup S_g$, $m'_{g'} = m_{g-1} + m_g + 1$. The diagram at time $t+1$ has one less period associated with it. The number of customers that are still alive and that have departed during period $i = 1, \cdots, g'-1$ has not changed, but since the new last period, g', consists of the last two periods with respect to $\phi(t)$ in addition to event $t+1$, the number of departures during this last period is the sum of the departures during the old two last periods and the departure at $t+1$. This corresponds to $m_{g'}$ and the lemma holds for this case.

This completes the proof of the lemma. □

A customer c_k is lost under ML if it misses its deadline before it begins service.

Theorem 1 *When the event* $\overline{k}$ *appears,* c_k *is removed from some set* S_i. *The customer* c_k *is lost under ML iff and a 1 is removed from the word* w_i *in the diagram immediately preceding that event.*

Proof. If c_k is removed from some S_i and a 1 is removed from w_i, then the customer must appear in the last group, $i = g$, and there is no 0 in the word w_g. By the preceding lemma, this means that c_k arrived during the last period, and during this period no live customer has departed the queue. Therefore, customer c_k did not depart the queue before missing his deadline.

On the other hand, if c_k is taken away from S_i and w_i contains some 0's, then by the preceding lemma, during the i-th period some customers departed the queue. Since all customers in $S_1 \cup S_2 \cup \cdots \cup S_i$ are still alive at time t, c_k is among them. Because c_k is about to die (miss its deadline), c_k's deadline is smaller than that of any customer in the queue. We know that during period i, $m_i > 0$ customers departed and according to ML, c_k must be among them. Hence, c_k began service before its death. □

Lemma 3 *The queue length after the t-th event is the total number of 1's in all words w_i, $i = 1, 2, ..., g$, in the diagram $\phi(t)$. If $g = 0$, then the diagram is an integer p which corresponds to the number of busy processors.*

Proof. The total number of customers (up to event t), which are still alive at time t and which arrived during period i, is n_i. The total number of customers that are still alive at time t and that departed the queue during period i is m_i. Therefore, the total number of customers in the queue after the t-th event is

$$\sum_{i=1}^{g}(n_i - m_i)$$

which is exactly the total number of 1's in all w_i, $i = 1, 2, \cdots, g$.

Therefore, the queue length is 0 if the total number of 1's in all words w_i, $i = 1, 2, \cdots, g$, is 0. This means that $g = 0$ and the diagram is an integer corresponding to the number of busy processors. □

4 The Policy ED

In this section we treat the earliest deadline policy (ED) operating on a single server in a similar manner as we treated ML in the preceding two sections. Because of the similarity, we omit many of the details and just describe the results.

Definition 5 *The earliest deadline policy (ED) is the non-idling policy that schedules the customer closest to its deadline. Customers that miss their deadlines while in the system are immediately removed.*

There are two variants, depending on whether preemptions are allowed. We consider solely the preemptive policy. Henceforth ED refers to this policy.

Given a set of customers $\{c_i\}_{i=1}^{\infty}$ with arrival times $\{t_i\}_{i=1}^{\infty}$, such that $0 \leq t_1 < t_2 < \cdots$, service times $\{\sigma_i\}_{i=1}^{\infty}$, and deadlines $\{d_i\}_{i=1}^{\infty}$, the behavior of ED can be represented by a history and event history in the exactly the same way as ML.

Example. Consider a single server system. Let $a = \{1, 5, 7, 10, 11, 15\}$, $\sigma = \{8, 4, 69, 12, 16, 8\}$ and $d = \{32, 22, 27, 25, 90, 33\}$. The event history is

$$1, 2, 3, \underline{2}, 4, 5, 6, \overline{2}, \underline{4}, \overline{2}, \overline{4}, \overline{3}, \underline{1}, \overline{1}, \overline{6}, \underline{5}, \overline{5}.$$

Diagrams under ED are similar to those under ML.

Definition 6 *A diagram under ED is either the integer 0 or a sequence of $g > 0$ groups where the ith group consists of a set S_i containing n_i distinct customer identities and a word $w_i \in 1^+0^+$ of length n_i, The word in the last group need not contain 0's, i.e., $w_g \in 1^+0^*$.*

The system can be represented by a diagram after each event. We describe how the diagram changes with the occurrence of each event. We define a diagram transition function ψ that takes two arguments, an event and a diagram, and produces a new diagram.

- *The arrival event j:*
  If the diagram consists of the integer 0, then the next diagram consists of a single group containing the customer that arrived and $w = 1$,

  $$\psi(j, 0) = \{j\}1.$$

  In all other cases, ψ behaves in the same way as for ML.

- *The deadline miss event $\overline{j}$:* The transition function ψ has the same behavior under ED-P as ML except if the diagram consists of a single group with one customer and the associated word contains no 0's. In that case,

  $$\psi(\overline{j}, \{j\}1) = 0.$$

- *The departure event $\underline{j}$:* The transition is identical to the transition under ML with the exception of when the diagram consists of a single group with a single 1 in the binary word. In this case, it is

  $$\psi(\underline{j}, S10^n) = 0, \;\; j \in S, \;\; S \subset N, \;\; n = 0, 1, \cdots.$$

The sequence of diagrams corresponding to the example given earlier in this section is given below.

$$
\begin{aligned}
&0 \xrightarrow{1} \{1\}1 \xrightarrow{2} \{1,2\}11 \xrightarrow{3} \{1,2,3\}111 \xrightarrow{\underline{2}} \{1,2,3\}110 \xrightarrow{4} \{1,2,3\}110\{4\}1 \xrightarrow{5} \\
&\{1,2,3\}110\{4,5\}11 \xrightarrow{6} \{1,2,3\}110\{4,5,6\}111 \xrightarrow{\underline{4}} \{1,2,3\}110\{4,5,6\}110 \xrightarrow{\overline{2}} \\
&\{1,3,4,5,6\}11110 \xrightarrow{\overline{4}} \{1,3,5,6\}1111 \xrightarrow{\overline{3}} \{1,5,6\}111 \xrightarrow{\underline{1}} \{1,5,6\}110 \xrightarrow{\overline{1}} \\
&\{5,6\}11 \xrightarrow{\overline{6}} \{5\}1 \xrightarrow{\underline{5}} 0
\end{aligned}
$$

Arguments similar to those given in the previous sections can be applied to establish the following properties of the diagram and transition function for ED.

Theorem 2 *The customer c_k is lost under ED whenever the event $\overline{k}$ appears, c_k is removed from some set S_i and a 1 is removed from the word w_i in the diagram immediately preceding that event.*

Theorem 3 *The number of customers in the system after the n-th event is the total number of 1's in all words w_i, $i = 1, 2, ..., g$, in the diagram $\phi(n)$. If $g = 0$, then there are no customers in the system.*

5 Applications

The above results can be used to develop probabilistic models of systems that use either ML or ED. For example, if the arrival process is Poisson with rate λ, the service times are exponential r.v.'s with parameter μ and the time between a customer's deadline and arrival time is an exponential r.v. with parameter α, then either ML or ED can be modeled as a Markov chain with states corresponding to the diagrams without the customer identities, i.e., the S_i is unnecessary.[1] A description of the equilibrium equations that the stationary probabilities must satisfy for these systems is found in [1].

The assumption that the arrival process be Poisson can be relaxed to be any renewal process by embedding a Markov chain immediately prior to arrival instants. In this

[1] Removing the groups creates a state subject to two interpretations, namely state 1. This ambiguous interpretation is removed by defining the state to be a pair (n, D) where n represents the number of busy servers and D is a concatenation of the binary strings associated with the groups.

case, the state description is again the same as the diagrams with the customer identities removed.

In the case of a single server, it is possible to allow service times to come from any distribution. In this case, a Markov chain can be embedded at instants of time immediately following departures. The state is again defined to be the diagram with the customer identities removed.

References

[1] J. Hong, X. Tan, D. Towsley, "A Performance Analysis of Minimum Laxity and Earliest Deadline Scheduling in a Real-Time System," to appear in *IEEE Transactions on Computers*, December 1989.

[2] S.S. Panwar, D. Towsley, "On the Optimality of the STE Rule for Multiple Server Queues that Serve Customers with Deadlines," COINS Technical Report 88-81, Dept. of Comp. & Info. Science, Univ. of Massachusetts, July 1988.