

## On the optimality of exhaustive service policies in multiclass queueing systems with modulated arrivals and switchovers

Y NARAHARI\* and N HEMACHANDRA

Department of Computer Science and Automation, Indian Institute of Science,  
Bangalore 560 012, India  
e-mail: [hari,hema]@csa.iisc.ernet.in

**Abstract.** Consider a single-server multiclass queueing system with  $K$  classes where the individual queues are fed by  $K$ -correlated interrupted Poisson streams generated in the states of a  $K$ -state stationary modulating Markov chain. The service times for all the classes are drawn independently from the same distribution. There is a setup time (and/or a setup cost) incurred whenever the server switches from one queue to another. It is required to minimize the sum of discounted inventory and setup costs over an infinite horizon. We provide sufficient conditions under which exhaustive service policies are optimal. We then present some simulation results for a two-class queueing system to show that exhaustive, threshold policies outperform non-exhaustive policies.

**Keywords.** Multiclass-queue; MMPP arrivals; setup times; exhaustive policies.

### 1. Introduction

Hofri & Ross (1987) investigated the following scheduling problem in multiclass queues with setup times (see figure 1): There is a single server attending to two classes of customers from two queues fed by independent homogeneous Poisson processes. The service times for the two classes are drawn independently from the same (general) distribution. A switchover time with or without a constant monetary cost per switch made is involved whenever the server switches from one queue to the other. An inventory holding cost, linear in queue length and having the same rate at the two queues is also included. Two separate cost structures are considered:

- (1) Sum of discounted switchover cost and inventory holding cost over an infinite horizon
- (2) The long-run average switchover cost plus the inventory costs

\* Author for correspondence

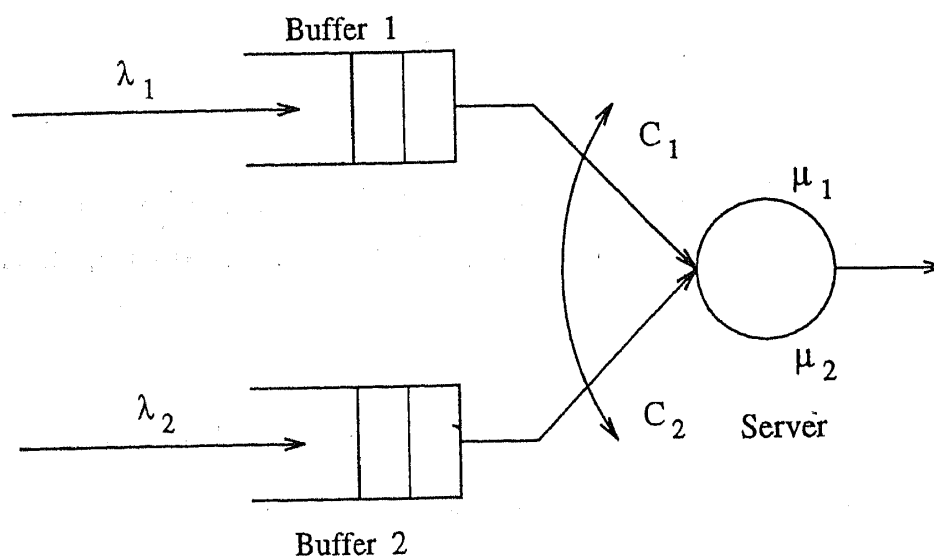


Figure 1. A two-class queue with independent Poisson arrivals.

In both cases above, Hofri & Ross (1987) showed that the policies to minimize the cost:

- (1) are necessarily *exhaustive*, i.e., server may switch to the other queue only when the current one is empty, and
- (2) are likely to be *threshold* policies, i.e., the server switches (from an empty queue) only when the other reaches a critical size.

The first result, in fact, holds for three or more queues. Also, for the two-queue case, a detailed queueing-based procedure was given for computing the optimal thresholds.

In this paper, we look at the following variant of the problem: The input streams to the two queues are no longer independent but are correlated in a Markov-modulated Poisson process (Fischer & Meier-Hellstern 1993) sense. (A Markov-modulated Poisson process can be informally described as an arrival process in which the Poisson arrivals have their rate modulated by a finite state, irreducible continuous-time Markov chain. In a  $K$ -state MMPP, the arrivals are Poisson at a certain rate (say,  $\lambda_i$ ) so long as the CTMC is in state  $i$ ,  $\forall i \in \{1, \dots, K\}$ . This is a popular model for nonrenewal input to queues.) More specifically, we look into the system (depicted in figure 2) where the input streams are the two interrupted Poisson processes generated in the states of a two-state homogeneous, stationary modulating Markov chain. Consequently, the inputs to the queues are correlated. The motivation for considering input processes of this type comes from several situations in manufacturing and communication networks (Fischer & Meier-Hellstern 1993; Frost & Melamed 1994). For example, under Markovian switching, the output from a multiclass flexible machine constitutes an MMPP (Hemachandra & Narahari 1995). Similarly, the output of a failure-prone Markovian queue has the so-called on-off arrival feature – a Poisson output for an exponential amount of time when the machine is working and zero output for another exponential amount of time when it is not-working. We remark that the input to each class is an interrupted Poisson process which is stochastically equivalent to a hyperexponential renewal process (Fischer & Meier-Hellstern 1993).

The problem and the investigations in this work are directly inspired by the work of Hofri & Ross (1987). Indeed, we follow the same line of argument and technical conditions

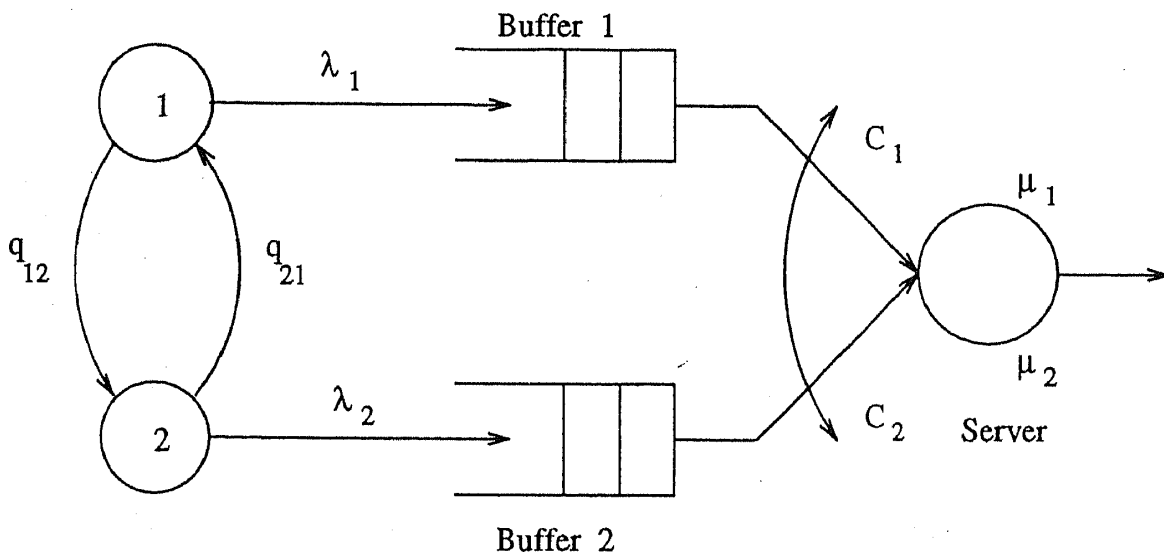


Figure 2. A two-class queue with modulated arrivals.

to corroborate our results. In this paper, we focus on the discounted cost criterion and investigate whether the optimal scheduling policies have to be necessarily exhaustive, like in the case of the Hofri–Ross formulation. Investigation of the correctness of this conjecture leads us to a set of sufficient conditions under which the conjecture is indeed true.

Apart from the seminal work of Hofri & Ross (1987), the literature that is relevant to this work is concerned with optimal scheduling in the presence of setups. The classic economic lot scheduling problem (Elmaghraby 1978) is the deterministic version of the multiclass scheduling problem with setups and has very elegant closed-form solutions. In the stochastic version which is more relevant here, two separate formulations exist. In the setup cost problem, a setup cost is incurred on each setup and in the setup time problem, a random setup time is incurred when the server switches class. The setup time problem is more realistic than the setup cost problem (Reiman & Wein 1994) but is also more difficult to solve.

Several researchers have studied the setup problem with more than two classes of customers, using a variety of techniques such as dynamic programming and heavy traffic queueing theory. A comprehensive survey appears in Reiman & Wein (1994) and Ravikumar (1996). All the authors however assume independent arrival processes into the individual queues. In § 2, we present the notation for the multiclass queueing model considered in this paper and describe the objective function to be optimized. In § 3, we investigate the conjecture that exhaustive policies are optimal and this leads to a set of sufficient conditions under which this conjecture is indeed true. In order to explore the validity of the conjecture, we carry out simulations on a two-class queueing system in § 4, which show that exhaustive, threshold policies outperform non-exhaustive policies.

## 2. The model

We consider a 2-class queue, representative of multiclass queues as a whole. Let the rates of the interrupted arrival processes to the two queues be  $\lambda_1$  and  $\lambda_2$  respectively, so that

the overall arrival process is a two-state MMPP. Let  $\alpha_1$  ( $\alpha_2$ ) be the rate from State 1 to State 2 (from State 2 to State 1) of the two-state Markov chain modulating the arrivals. Let  $\pi_1$  and  $\pi_2$  be the stationary probabilities of the modulating Markov chain so that  $\pi_1 = \alpha_2/(\alpha_1 + \alpha_2)$  and  $\pi_2 = \alpha_1/(\alpha_1 + \alpha_2)$ . Service time distribution for a customer from either of the queues is the same and let  $S_1, S_2, S_3, \dots$  be the sequence of service times (*i.i.d.*) for customers of Queue 1 and  $T_1, T_2, T_3, \dots$  the sequence of service times (*i.i.d.*) for customers of Queue 2. Note that  $\{S_n\}$  and  $\{T_n\}$  are from the same service time distribution, with finite mean. Let  $\{C_{1,n}\}$  and  $\{C_{2,n}\}$  be two (possibly different) sequences of *i.i.d.* positive random variables with some general distributions, corresponding to the sequence of setup times at Queue 1 and Queue 2 respectively. The state of the system is then a four-tuple  $(u, x_1, x_2, v)$ , where,

- $u$  gives the status of the server taking values in the set  $\{I_1, I_2, 1, 2, 12, 21\}$  where  $I_i$  means server is idle but setup for class  $i$  ( $i = 1, 2$ ); 1 denotes that the server is currently serving Class 1 customers; 12 means that the server is switching from Class 1 to 2, and the other symbols have similar meaning.
- $x_i$  is the number in the queue (including the one in service, if any)  $i$  ( $i = 1, 2$ ), taking integer values.
- $v$  takes values in the set  $\{1, 2\}$  giving the state of the modulating Markov chain.

Let  $S$  be this state space. As in Hofri & Ross (1987), the decision epochs will be at service completions, switch completions and arrival instants. The actions at each such decision epoch are from the set  $\mathcal{A}$  with  $\mathcal{A} = \{S, C, I\}$  where these symbols successively mean that the server chooses to serve, change and idle. So the problem is a semi-Markov decision problem with state space  $S$  and action space  $\mathcal{A}$  given above.

Let  $X_i^{\pi,z}(t)$  be the number of customers in queue  $i$  ( $i = 1, 2$ ) at time  $t$ , given that the initial state of the system is  $z$ , and a policy  $\pi$  is used. Let

$$X^{\pi,z}(t) = X_1^{\pi,z}(t) + X_2^{\pi,z}(t)$$

be the number in the system at time  $t$ . Let  $R^{\pi,z}(t)$  be the number of switches made by the server up to time  $t$  under policy  $\pi$ . Then, the expected discounted cost, with discount factor  $\beta > 0$ ,  $V_\pi(z)$ , incurred when a policy  $\pi$  is followed for a system starting in state  $z$ , is given by

$$V_\pi(z) = \mathbb{E} \left[ \int_0^\infty e^{-\beta t} X^{\pi,z}(t) dt + a \int_0^\infty e^{-\beta t} dR^{\pi,z}(t) \right] \quad (1)$$

assuming that the holding costs per customer per unit time in both the queues are the same and are normalized to 1 and that  $a$  is the one-time switch charge levied at the start of each switch. A policy  $\pi'$  is said to be optimal if it attains

$$V(z) := V_{\pi'}(z) = \inf_{\pi} V_\pi(z) \quad \forall z \in S.$$

To suit the analysis ahead, we cast the objective function as in Harrison (1975): Let  $A_i^{\pi,z}(t)$  be the number of arrivals up to time  $t$  and  $D_i^{\pi,z}(t)$  the number of departures up to time  $t$  for the queue  $i$  ( $i = 1, 2$ ) when a policy  $\pi$  is followed in a system starting at state  $z$ . Let  $x_1^0$  and  $x_2^0$  be the number in Queue 1 and Queue 2 respectively at time  $t = 0$ . Then, we have,

$$X_i^{\pi,z}(t) = x_i^0 + A_i^{\pi,z}(t) - D_i^{\pi,z}(t) \quad i = 1, 2, \quad (2)$$

so that from (1) and (2) we have,

$$V_\pi(z) = \frac{x_1^0 + x_2^0}{\beta} + \frac{\pi_1 \lambda_1 + \pi_2 \lambda_2}{\beta^2} - \mathbb{E} \left[ \int_0^\infty e^{-\beta t} (D_1^{\pi,z}(t) + D_2^{\pi,z}(t)) dt - a \tilde{R}^{\pi,z}(\beta) \right], \quad (3)$$

where  $\tilde{R}^{\pi,z}(\beta)$  is the Laplace–Stieltjes transform (LST) of the distribution  $R^{\pi,z}(t)$ . Since,

$$\begin{aligned} & \mathbb{E} \left[ \int_0^\infty e^{-\beta t} (D_1^{\pi,z}(t) + D_2^{\pi,z}(t)) dt - a \tilde{R}^{\pi,z}(\beta) \right] \\ &= \frac{1}{\beta} \mathbb{E} \left[ \sum_{k \geq 1} e^{-\beta T_k^{\pi,z}} - a \beta \tilde{R}^{\pi,z}(\beta) \right], \end{aligned}$$

where,  $T_k^{\pi,z}$  is the  $k$ th departure epoch from the system under policy  $\pi$ , we have,

$$V_\pi(z) = \frac{x_1^0 + x_2^0}{\beta} + \frac{\pi_1 \lambda_1 + \pi_2 \lambda_2}{\beta^2} - \frac{1}{\beta} J_\pi(z),$$

with  $J_\pi(z)$  defined as,

$$J_\pi(z) := \mathbb{E} \left[ \sum_{k \geq 1} e^{-\beta T_k^{\pi,z}} - a \beta \tilde{R}^{\pi,z}(\beta) \right].$$

The optimizing problem is now equivalent to maximizing  $J_\pi(z)$ . If  $J(z)$  is the supremum of  $J_\pi(z)$  over all policies, then by Lippman (1973), there is an optimal pure policy which achieves  $J(z)$ .

### 3. Optimality of exhaustive policies

Let  $g$  be any non-exhaustive policy; we shall argue below that the performance of any such policy could be improved upon. This means that non-exhaustive policies are suboptimal. Thus, optimal policies have to be necessarily exhaustive.

First, we synthesize from the existing arrival process, a new arrival process that depends on the scheduling policy followed. Let  $f$  be any such policy.

Let  $N_1(t)$  and  $N_2(t)$  be the arrival processes to Queues 1 and 2. Assume that  $\bar{N} := (\bar{N}_1(t), \bar{N}_2(t))$  is stochastically equivalent but independent of  $N(t) := (N_1(t), N_2(t))$ . Let  $(\Omega, \mathcal{F}, P)$  support the independent processes  $N(t)$ ,  $\bar{N}(t)$  and the sequences  $\{S_n\}$ ,  $\{T_n\}$ ,  $\{C_{1,n}\}$ ,  $\{C_{2,n}\}$ . As in Hofri & Ross (1987), form a new arrival process from these  $N(t)$  and  $\bar{N}(t)$  by collecting customers from  $N(t)$  or  $\bar{N}(t)$  as described below. Let  $\tau^f$  be the first instant that the server begins to serve a customer from Queue 1 under policy  $f$ . Let  $S_1$  be the service time of this customer from Queue 1. Define a new arrival process  $A^f(t)$  as:

$$\begin{aligned} A^f(t) &= N(t), \quad 0 \leq t \leq \tau^f, \\ &= N(\tau^f) + \bar{N}(t - \tau^f), \quad \tau^f \leq t \leq \tau^f + S_1, \\ &= \bar{N}(S_1) + N(t - S_1), \quad t \geq \tau^f + S_1. \end{aligned} \quad (4)$$

That is, collect customers from  $N(t)$  until  $\tau^f$ , then on from  $\bar{N}(t)$  until  $\tau^f + S_1$ ; and revert back to  $N(t)$  indefinitely after  $\tau^f + S_1$ . Note that  $A^f(t)$  is defined pathwise and its construction depends on policy  $f$ .

Since  $g$  is non-exhaustive, there exists some state,  $z$ , such that either  $g$  causes the server to idle at a nonempty queue or abandons a nonempty queue. Without loss of generality, assume that this occurs at Queue 1. Then, if  $g(u, x_1, x_2, v)$  denotes the scheduling decision in the state  $(u, x_1, x_2, v)$ ,

$$\exists x_1 > 0 \quad \ni g(1, x_1, x_2, v) \neq S,$$

i.e.,

$$g(1, x_1, x_2, v) \in \{C, I\} \quad \text{with} \quad x_1 > 0.$$

Let this  $z_0 := (1, x_1, x_2, v)$  be the initial state in the sequel. Let  $Z^{f, z_0}(t)$  denote the state of the system at time  $t$  when a policy  $f$  is followed by the system starting in a state  $z_0$ . Write  $Z^\pi(t)$  for  $Z^{\pi, z_0}(t)$ .

We now exhibit a policy  $\pi$  that outperforms the policy  $g$  in minimizing the total cost. Policy  $\pi$  says that the system beginning in state  $z_0$  should first *immediately* serve a job from Queue 1 (let  $S_1$  be the service time of this customer). Then,  $\pi$  follows policy  $g$  for a random amount of time  $\tau^g$  from  $S_1$  to  $\tau^g + S_1$  with a lag of  $S_1$ . From  $\tau^g + S_1$  onwards it simply follows policy  $g$  with no delay. So,  $\tau^\pi = 0$ . Writing out  $A^\pi(t)$  and  $A^g(t)$ , the arrival processes corresponding to policies  $\pi$  and  $g$  using (4), we have that they are related as:

$$\begin{aligned} A^\pi(t) &= A^g(t + \tau^g) - A^g(\tau^g), & 0 \leq t \leq S_1, \\ &= A^g(t - S_1) + A^\pi(S_1), & S_1 \leq t \leq \tau^g + S_1, \\ &= A^g(t), & t \geq \tau^g + S_1. \end{aligned}$$

Note that the departure processes,  $D^\pi(t)$  and  $D^g(t)$ , are related as

$$\begin{aligned} D^\pi(t) &= 0, & 0 \leq t \leq S_1, \\ &= 1 + D^g(t - S_1), & S_1 \leq t \leq \tau^g + S_1, \\ &= D^g(t), & t \geq \tau^g + S_1. \end{aligned} \tag{5}$$

Essentially, we have states coupled (stochastically) by the policies  $g$  and  $\pi$  in such a way that

$$Z^\pi(t) = Z^g(t) \quad \text{for } t > \tau^g + S_1.$$

Here is the crucial assumption.

*Assumption A.* Let the arrival processes  $A^\pi(t)$  and  $A^g(t)$  formed above be stochastically equivalent, so that the systems when driven by any of these processes have the same performance measures when the same policy is used for both the systems.

*Remark.* This assumption is satisfied by independent Poisson processes as noted by Hofri and Ross (1987). The validity of this assumption in the case of general arrival processes is an interesting issue to be looked into.

**PROPOSITION 1**

Let (i) at least one of the values  $a$ ,  $\mathbb{E}[C_1]$ ,  $\mathbb{E}[C_2]$  be strictly greater than zero and (ii) the processes  $A^\pi(t)$  and  $A^g(t)$  be equal in law. Then, any non-exhaustive policy  $g$  is non-optimal.

*Proof.* Using (5) above, we show that the difference,

$$\begin{aligned} J_\pi(z_0) - J_g(z_0) &= \beta \mathbb{E} \left[ \int_{t \geq 0} e^{-\beta t} [D^\pi(t) - D^g(t)] dt \right] - a\beta \mathbb{E} \left[ \int_{t \geq 0} e^{-\beta t} [dR^\pi(t) - dR^g(t)] \right], \end{aligned} \quad (6)$$

is positive. This difference has two terms on the RHS, the first one capturing the holding cost and the other the switching cost.

Let  $\sigma^g$  be the first time that  $g$  switches from Queue 1 when the initial state is  $z_0$ . On each of the sets

$$\Lambda_1 := \{\tau^g < \sigma^g\}, \quad \Lambda_2 := \{\tau^g = \sigma^g = \infty\}, \quad \Lambda_3 := \{\tau^g \geq \sigma^g < \infty\},$$

we compute this difference assuming that  $\mathbb{E}[C_2] > 0$  or  $a > 0$ .

*Case 1* ( $\tau^g < \sigma^g$ ). Here we have  $R^\pi(t) = R^g(t)$ ,  $t \geq 0$  and hence the difference in switching costs as captured by the second term above is zero. Since

$$\begin{aligned} D^\pi(t) &= D^g(t) + 1, \quad t \in [S_1, \tau^g + S_1], \\ &= D^g(t), \quad t \notin [S_1, \tau^g + S_1], \end{aligned}$$

we have,

$$\begin{aligned} \beta \mathbb{E} \left[ \int_{t \geq 0} e^{-\beta t} [D^\pi(t) - D^g(t)] dt \mid \Lambda_1 \right] &= \mathbb{E}[e^{-\beta S_1} (1 - e^{-\beta \tau^g}) \mid \Lambda_1] \\ &= \mathbb{E}[\mathbb{E}[e^{-\beta S_1} (1 - e^{-\beta \tau^g}) \mid \Lambda_1, \tau^g] \mid \Lambda_1] \\ &= \mathbb{E}[(1 - e^{-\beta \tau^g}) \mathbb{E}[e^{-\beta S_1} \mid \Lambda_1, \tau^g] \mid \Lambda_1] \\ &= \mathbb{E}[e^{-\beta S_1}] \mathbb{E}[(1 - e^{-\beta \tau^g}) \mid \Lambda_1] \end{aligned}$$

Now,  $\mathbb{E}[e^{-\beta S_1}] =: \tilde{S}(\beta)$  and since  $\tau^g$  is strictly positive a.s. we have this term greater than zero and thus (6) is positive in Case 1.

*Case 2* ( $\tau^g = \sigma^g = \infty$ ). Policy  $g$  simply idles forever and hence the difference in switching costs is zero and the difference in holding costs is given by

$$\beta \mathbb{E} \left[ \int_0^\infty e^{-\beta t} [D^\pi(t) - D^g(t)] dt \mid \Lambda_2 \right] = \mathbb{E}[e^{-\beta S_1} \mid \Lambda_2] = \tilde{S}(\beta)$$

so that, here also (6) is positive.

*Case 3* ( $\tau^g \geq \sigma^g < \infty$ ). Many scenarios are possible here with policy  $g$  deciding to opt for a switching before taking up a job from Queue 1 (in finite time a.s.). In all of them, however,  $R^\pi(t)$  lags behind  $R^g(t)$  by  $S_1$  till  $\tau^g + S_1$  and from then on they are the same. Here, we show that the difference (6) is strictly positive by showing that the first term (holding cost difference) is strictly positive if  $\mathbb{E}[C_2] > 0$ , and the negative of the second term (switching cost difference) is also strictly positive, if  $a > 0$ .

First, it follows as in Hofri & Ross (1987) again, that the negative of the switching cost difference in (6) above can be lower bounded,

$$-a\beta\mathbb{E}\left[\int_0^\infty e^{-\beta t} [dR^\pi(t) - dR^g(t)]|\Lambda_3\right] \geq a\beta\mathbb{E}[e^{-\beta\sigma^g}(1 - e^{-\beta S_1})|\Lambda_3]$$

and hence, is strictly positive if  $a > 0$ .

If  $\mathbb{E}[C_2] > 0$ , it follows that the holding cost difference in (6) is strictly positive as shown below. Write

$$\begin{aligned} & \beta\mathbb{E}\left[\int_0^\infty e^{-\beta t} [D^\pi(t) - D^g(t)]dt|\Lambda_3\right] \\ &= \beta\int_0^\infty e^{-\beta t} \left(\sum_{k \geq 1} [\mathbb{P}(D^\pi(t) \geq k, \Psi|\Lambda_3) - \mathbb{P}(D^g(t) \geq k, \Psi|\Lambda_3)]\right) dt \end{aligned}$$

where  $\Psi := \{t \leq S_1 + \tau^g\}$ . Note that departure processes are stochastically equal for  $t > S_1 + \tau^g$ . With  $T_i^g$  denoting the  $i$ th departure epoch from the system under policy  $g$ , ( $T_0^g := 0$ ), set  $U_i := T_i^g - T_{i-1}^g$ . Then, with  $U_k \geq_{st} S_1$ , we have,

$$\begin{aligned} \mathbb{P}(U_1 + U_2 + \dots + U_{k-1} + S_1 \leq t) &\geq \mathbb{P}(U_1 + U_2 + \dots + U_k \leq t) \\ &\quad \forall t \geq 0, k \geq 0 \\ \Rightarrow \mathbb{P}(D^\pi(t) \geq k, \Psi|\Lambda_3) &\geq \mathbb{P}(D^g(t) \geq k, \Psi|\Lambda_3) \quad \forall t \geq 0, k \geq 0 \end{aligned}$$

so that, the above is indeed positive. If,  $\mathbb{E}[C_2] > 0$ , then from the fact that  $U_1$  includes the service time of a job as well as switchover or idling time, we have,

$$\mathbb{P}(S_1 \leq t, \Psi|\Lambda_3) \geq \mathbb{P}(U_1 \leq t, \Psi|\Lambda_3) + \epsilon,$$

for some  $\epsilon > 0$  and  $t$  in an open interval. This means that

$$\mathbb{P}(D^\pi(t) \geq 1, \Psi|\Lambda_3) \geq \mathbb{P}(D^g(t) \geq 1, \Psi|\Lambda_3) + \epsilon,$$

for some  $\epsilon > 0$  and  $t$  in an open interval. So, finally, we have (6) strictly positive under the hypothesis.

*Remark 1.* It is easy to see that the above arguments go through for queues with more than two classes; we require that at least one of  $a, \{\mathbb{E}[C_{i,j}]\}$  be strictly positive.

*Remark 2.* The proof above uses the stochastic coupling idea and the requirement on the part of the arrival processes, as set out in assumption A, facilitates coupling of the evolution of the systems following the policies *pathwise*, after the coupling epoch,  $\tau^g + S_1$ . Since the technical nature of MMPP is not used in the above proof, we conjecture that the above result is valid for a much more general arrival process that satisfies assumption A.



*Remark 3.* Policy  $\pi$  is more exhaustive than policy  $g$  in the sense that it serves one job from the same queue in the very beginning. As the above argument is sample pathwise, by induction, this means that exhaustive policies are better than non-exhaustive policies.

*Remark 4.* We left open here the decisions the server has to take when the queue to which it is attached becomes empty. One class of policies to use in this context is that of the *threshold policies* in which Hofri & Ross (1987) show that such policies are optimal when arrivals are Poisson.

*Remark 5.* Verifying the sufficiency condition involving the arrival process (Assumption A) is difficult; indeed it may be necessary to pursue another track of investigation to get around this problem (M Hofri 1996, personal communication).

#### 4. Simulation results

Since it is difficult to verify the validity of the sufficient conditions concerning the equivalence of the arrival processes (assumption A), we simulated quite a few systems operating under different policies and looked at the cost incurred by them. This way, we investigated

**Table 1.** Setup costs and inventory holding costs for a bursty scenario.

$n_1 n_2$	$N_1 N_2$	$SW_1$	$SW_2$	Switching cost	Holding cost	Total cost
<b>1 1</b>	<b>2 3</b>	<b>1337.6</b>	<b>1252.4</b>	<b>2590.0</b>	<b>798.0</b>	<b>3388.0</b>
1 2	2 3	1342.5	1267.4	2609.9	1526.3	4136.2
1 3	2 3	1456.1	1404.4	2860.5	2228.2	5088.7
2 1	2 3	1375.8	1506.3	2882.1	1656.1	4538.2
3 1	2 3	1817.8	1909.5	3727.3	2142.1	5869.4
2 2	2 3	1387.9	1532.7	2920.6	2329.9	5250.5
2 3	2 3	1504.2	1675.4	3179.6	3008.8	6188.2
<b>1 1</b>	<b>2 2</b>	<b>1342.5</b>	<b>1275.4</b>	<b>2617.9</b>	<b>591.1</b>	<b>3209.0</b>
1 2	2 2	1456.1	1412.3	2868.4	1325.2	4193.6
1 3	2 2	1785.4	1755.5	3540.9	2190.7	5731.7
2 1	2 2	1387.4	1540.7	2928.6	1394.6	4323.2
3 1	2 2	2001.6	2132.1	4133.7	1780.2	5913.9
<b>1 1</b>	<b>3 2</b>	<b>1160.9</b>	<b>1101.5</b>	<b>2262.4</b>	<b>732.5</b>	<b>2994.9</b>
1 2	3 2	1259.0	1224.1	2483.1	1507.8	3991.0
1 3	3 2	1401.4	1377.6	2779.0	2441.1	5220.1
1 5	3 2	1401.9	1379.2	2781.1	4154.4	6935.5
2 1	3 2	1112.3	1275.4	2387.7	1511.3	3899.0
2 2	3 2	1225.9	1412.3	2638.3	2245.4	4883.7
2 3	3 2	1555.2	1755.5	3310.7	3111.0	6421.7
2 4	3 2	1633.7	1835.9	3469.6	3972.3	7441.9
3 1	3 2	1202.3	1357.6	2559.9	2116.1	4676.0
3 2	3 2	1318.6	1499.7	2818.3	2831.4	5649.8
3 3	3 2	1869.5	2065.4	3934.9	3628.2	7563.1
4 1	3 2	1816.0	1949.0	3765.0	2501.7	6266.7
4 2	3 2	2187.1	2348.6	4535.7	3073.3	7608.9
5 1	3 2	2237.9	2343.8	4580.8	2907.0	7487.8
5 2	3 2	3018.9	3158.6	6177.6	3428.1	9605.6

**Table 2.** Setup and inventory holding costs for exhaustive policies.

$n_1 n_2$	$N_1 N_2$	$SW_1$	$SW_2$	Switching cost	Holding cost	Total cost
1 1	1 1	1734.3	1683.7	3418.0	294.1	3712.1
1 1	1 2	1618.1	1540.7	3158.8	474.4	3633.2
1 1	1 3	1606.0	1506.3	3112.3	735.9	3848.2
1 1	1 4	1257.2	1149.2	2406.4	964.8	3371.1
1 1	1 5	1217.1	1096.6	2313.6	1256.0	3569.6
1 1	2 1	1456.1	1412.7	2868.8	433.9	3302.7
1 1	2 2	1342.5	1275.4	2617.8	591.1	3209.0
1 1	2 3	1337.6	1252.4	2589.9	797.4	3387.3
1 1	2 4	1232.9	1137.4	2370.4	936.8	3307.1
1 1	2 5	1184.9	1075.7	2260.6	1233.0	3493.7
1 1	3 1	1259.0	1224.5	2483.5	616.5	3100.0
<b>1 1</b>	<b>3 2</b>	<b>1160.9</b>	<b>1101.5</b>	<b>2262.4</b>	<b>732.5</b>	<b>2994.9</b>
1 1	3 3	1160.1	1082.4	2242.5	930.6	3173.1
1 1	3 4	1056.1	970.1	2026.2	1008.6	3034.8
1 1	3 5	1008.5	910.2	1918.7	1268.7	3187.4
1 1	4 1	1122.2	1091.3	2213.4	940.6	3154.1
1 1	4 2	1024.7	971.6	1996.3	1027.7	3024.0
1 1	4 3	1019.7	950.9	1970.5	1209.7	3180.1
1 1	4 4	1019.2	939.8	1959.0	1150.3	3109.3
1 1	4 5	1001.0	905.5	1906.5	1324.0	3230.5
1 1	5 1	1096.3	1067.1	2163.8	1035.9	3199.3
1 1	5 2	1016.4	965.3	1981.7	1094.2	3075.9
1 1	5 3	1015.8	949.0	1964.7	1263.5	3228.2
1 1	5 4	1005.1	927.3	1932.4	1201.0	3133.4
1 1	5 5	998.8	905.2	1904.0	1371.3	3275.3

the performance of the exhaustive policies. Also, we confine our experimentation to the class of exhaustive, threshold policies motivated by the results of Hofri & Ross (1987) concerning the optimality of threshold policies in the case of independent Poisson arrivals. We remark here that exhaustive, threshold policies may not be the optimal policies.

A typical policy we considered is captured by the four tuple  $(n_1, n_2, N_1, N_2)$ . Here,  $n_1$  and  $n_2$  are *idle* thresholds while  $N_1$  and  $N_2$  are *switching* thresholds.  $n_1$  ( $n_2$ ) is the minimum number of jobs required in Queue 1 (Queue 2) for the server to process a job from that queue when attached to the queue. In other words, even though the server is attached to Queue 1 (Queue 2) and the number of customers in that queue is less than  $n_1$  ( $n_2$ ), it will not process any job from that queue. Note that a service policy is exhaustive if  $n_1 = n_2 = 1$ .  $N_1$  ( $N_2$ ) is the minimum number of jobs required in Queue 1 (Queue 2) for the server, currently attached to Queue 2 (Queue 1), to switch to Queue 1 (Queue 2).  $N_1$  and  $N_2$  are called the switching thresholds.

Note that, in view of the exhaustive nature of the policies,  $n$  takes precedence over  $N$ . So, in an empty system, the server when attached to a queue, say Queue 1, potentially idles there till the number in this queue is greater than or equal to  $n_1$ . If, during this idling period, the number in Queue 2 becomes at least  $N_2$ , it switches. Similarly, the server continues to serve the jobs of Queue 1, till there are less than  $n_1$  jobs in this queue, upon which, it switches to Queue 2 if there are more than  $N_2$  jobs in Queue 2, otherwise the server idles till work builds up in one of the queues. We present two such examples below.

**Table 3.** Setup and inventory holding costs for a non-bursty scenario.

$n_1 n_2$	$N_1 N_2$	$SW_1$	$SW_2$	Switching cost	Holding cost	Total cost
<b>1 1</b>	<b>4 5</b>	<b>692.3</b>	<b>550.8</b>	<b>1243.1</b>	<b>1189.5</b>	<b>2432.6</b>
1 2	4 5	694.2	556.6	1250.9	1766.7	3017.6
1 3	4 5	735.0	599.6	1334.5	2329.9	3664.4
1 4	4 5	863.1	732.0	1595.2	2814.6	4409.4
2 1	4 5	764.0	618.6	1382.6	2098.3	3480.9
2 2	4 5	775.9	634.8	1410.6	2653.2	4063.8
2 3	4 5	800.2	663.4	1463.7	3238.8	4702.4
2 4	4 5	929.0	796.1	1725.1	3702.2	5427.4
3 1	4 5	886.5	737.9	1624.4	3088.9	4713.3
3 2	4 5	898.4	754.8	1653.2	3624.8	5278.1
3 3	4 5	999.8	857.6	1857.5	4175.7	6033.1
3 4	4 5	1129.3	992.0	2121.3	4606.1	6727.4
4 1	4 5	888.3	737.9	1626.2	4027.6	5653.8
4 2	4 5	900.2	754.9	1655.1	4563.0	6218.1
4 3	4 5	1001.7	857.7	1859.4	5113.2	6972.5
4 4	4 5	1131.7	992.0	2123.7	5537.7	7661.3
5 1	4 5	1222.0	1049.4	2271.4	5140.6	7412.0
5 2	4 5	1336.9	1167.2	2504.1	5705.8	8209.9
5 3	4 5	1501.0	1342.8	2843.8	6118.0	8961.8
6 1	4 5	1323.9	1146.8	2470.7	6124.7	8595.4
6 2	4 5	1442.9	1270.0	2712.9	6666.4	9379.3
6 3	4 5	1711.6	1542.7	3254.3	6968.2	10222.6
<b>1 1</b>	<b>4 4</b>	<b>835.9</b>	<b>699.1</b>	<b>1535.1</b>	<b>949.2</b>	<b>2484.3</b>
1 2	4 4	876.7	742.2	1618.9	1559.7	3178.5
1 3	4 4	1004.8	875.6	1880.4	2074.3	3954.7
1 4	4 4	1012.0	890.8	1902.9	2746.0	4648.9
2 1	4 4	917.7	777.3	1695.0	1834.3	3529.3
<b>1 1</b>	<b>5 4</b>	<b>831.2</b>	<b>698.8</b>	<b>1530.0</b>	<b>1062.2</b>	<b>2592.2</b>
1 2	5 4	871.5	741.8	1613.4	1685.7	3299.1
1 3	5 4	999.2	875.3	1874.5	2217.1	4091.6
1 4	5 4	1006.3	890.5	1896.8	2890.0	4786.8
2 1	5 4	835.0	699.2	1534.1	1936.9	3470.9
<b>1 1</b>	<b>5 5</b>	<b>688.8</b>	<b>550.4</b>	<b>1239.2</b>	<b>1268.1</b>	<b>2507.3</b>
1 2	5 5	690.7	556.3	1245.0	1847.2	3094.2
1 3	5 5	731.0	599.2	1330.2	2423.3	3753.5
2 1	5 5	691.3	550.8	1242.1	2177.2	3419.4

#### 4.1 Examples

Here, we look at the performance costs incurred by two systems as they follow policies which are more and more exhaustive, i.e., with  $n_1$ ,  $n_2$  or both, becoming less. For the systems we considered, the simulations show that the exhaustive policies incur the least cost.

*Example 1.* (Bursty arrivals). Recall that a central motivation for the use of MMPP as an arrival process is to capture bursty arrivals to a queue. We wish to look at such a scenario now in the context of service and scheduling policies. Let  $\alpha_1$ ,  $\alpha_2$ ,  $\lambda_1$ , and  $\lambda_2$  have the usual significance. The numerical data are:  $\alpha_1 = 1/10$ ,  $\alpha_2 = 1/60$ ,  $\lambda_1 = 0.2$ ,  $\lambda_2 = 0.04$ . Let the setup time be deterministic at a value 10 for both the queues and the mean service time deterministic at 0.50. The discount factor  $\beta = 0.001$ , while the setup cost  $a = 250$ .

**Table 4.** Setup and inventory holding costs for exhaustive policies.

$n_1$	$n_2$	$N_1$	$N_2$	$SW_1$	$SW_2$	Switching cost	Holding cost	Total cost
11	11	11	11	1486.9	1360.7	2847.5	183.0	3030.5
11	11	11	12	1426.8	1286.4	2713.2	334.7	3047.5
11	11	11	13	1149.1	1001.1	2150.9	644.3	2794.5
11	11	11	14	1047.6	897.4	1945.0	782.3	2727.3
11	11	11	15	892.4	737.9	1630.4	1085.1	2715.5
11	11	11	16	813.7	653.6	1467.3	1346.2	2813.5
11	11	11	21	1482.0	1360.5	2842.5	247.0	3089.5
11	11	11	22	1422.2	1286.3	2708.6	395.4	3103.9
11	11	11	23	1145.3	1001.1	2146.4	694.0	2840.4
11	11	11	24	1043.9	897.3	1941.2	831.3	2772.6
11	11	11	25	890.2	737.9	1628.0	1114.6	2742.7
11	11	11	26	811.9	653.5	1465.4	1367.2	2832.7
11	11	11	31	1236.1	1119.6	2355.7	394.3	2750.0
11	11	11	32	1218.9	1090.5	2309.4	483.1	2792.5
11	11	11	33	943.0	806.8	1749.8	744.0	2493.9
11	11	11	34	918.6	777.3	1695.9	846.6	2542.5
11	11	11	35	765.0	618.6	1383.6	1110.7	2494.3
11	11	11	36	681.7	533.0	1214.8	1338.9	2553.6
11	11	11	37	644.5	494.2	1138.7	1478.8	2617.6
11	11	11	41	1013.5	900.2	1913.7	543.8	2457.5
11	11	11	42	1006.3	882.2	1888.5	611.0	2499.5
11	11	11	43	876.7	743.0	1619.7	824.1	2443.8
11	11	11	44	835.9	699.1	1535.1	949.1	2484.2
<b>11</b>	<b>11</b>	<b>11</b>	<b>45</b>	<b>692.3</b>	<b>550.8</b>	<b>1243.2</b>	<b>1189.6</b>	<b>2432.8</b>
11	11	11	46	615.7	469.7	1085.5	1392.7	2478.2
11	11	11	47	579.1	430.9	1010.0	1522.9	2532.9
11	11	11	48	576.2	424.6	1000.7	1646.8	2647.6
11	11	11	49	499.4	348.2	847.6	1968.9	2816.6
11	11	11	51	1007.5	899.8	1907.3	698.9	2606.1
11	11	11	52	1000.4	881.8	1882.2	764.7	2646.9
11	11	11	53	871.5	742.6	1614.1	950.0	2564.1
11	11	11	54	831.21	698.8	1530.0	1062.1	2592.1
11	11	11	55	688.76	550.4	1239.2	1268.0	2507.2
11	11	11	56	612.5	469.7	1082.1	1473.2	2555.4
11	11	11	61	986.0	881.7	1867.7	902.5	2770.2
11	11	11	62	979.0	865.0	1844.0	957.3	2801.3
11	11	11	63	850.3	726.3	1576.5	1131.5	2708.0
11	11	11	64	827.3	698.0	1525.4	1150.9	2676.3
11	11	11	65	685.31	550.0	1235.3	1352.4	2587.8

This system was simulated over a sufficiently long simulation run length of 10,000 time units and the total cost (as given by (1)) was computed for different sets of values of  $n_1, n_2, N_1, N_2$ . It was found that the run length of 10,000 was an adequate indicator of the infinite horizon discounted cost since the accumulated cost became virtually saturated for this run length. The simulation clock was made to start ticking only after ensuring that the modulating Markov chain was well into the steady state.

Table 1 shows for different values of  $n_1, n_2, N_1, N_2$  the values of the accumulated discounted switching costs from Queue 1 and Queue 2, respectively, as  $SW_1, SW_2$ , the total switching cost ( $SW_1 + SW_2$ ), the accumulated (discounted) inventory holding cost, and the overall discounted cost. Observe that three sets of values have been explored for

$N_1$  and  $N_2$ , namely,  $N_1 = 2, N_2 = 3$ ;  $N_1 = 2, N_2 = 2$ ;  $N_1 = 3, N_2 = 2$ . From table 1 we see that in all these three sets, exhaustive policies as given by  $n_1 = n_2 = 1$  have the best performance.

Next, having seen that exhaustive policies perform better, we investigate the effectiveness of threshold switching by looking at the cost incurred for different values of thresholds  $N_1$  and  $N_2$  (see table 2). For this system, the threshold  $N_1 = 3, N_2 = 2$  offers the least overall discounted cost among threshold policies considered.

*Example 2.* (Non-bursty arrivals). This scenario comes up when the streams of jobs have comparable intensities with a correlation as captured by the 2-state Markov chain. We have here,  $\alpha_1 = \alpha_2 = 1/60, \lambda_1 = 0.20, \lambda_2 = 0.10$ . The service is deterministic at value 0.50. The setup cost is 200 while the setup times are deterministic at values 5 and 5. The discount factor  $\beta = 0.001$ . The notation in tables 3 and 4 is as in the previous example. Table 3 demonstrates that exhaustive policies are better when thresholds are taken as (4,5); (4,4); (5,4); and (5,5). Then, fixing the policy as exhaustive, i.e., with  $n_1 = n_2 = 1$ , we search for the thresholds which offer the least overall discounted cost; Table 4 shows that the thresholds  $N_1 = 4, N_2 = 5$  offer the best performance among the policies considered.

## 5. Conclusions

For a single-server multiclass queue fed by the streams of an MMPP, with non-preemptive switchovers involving a switchover cost and/or a switchover time, we argued that exhaustive policies are optimal as long as the arrival process satisfies a technical condition regarding its law and the service times of the various classes are the same. The verification of this technical requirement on the part of the arrival process appears to be difficult. Hofri & Ross (1987) showed that, when the queue is fed by independent Poisson streams, these policies are optimal; here, this technical condition is true. To test our claim, we carried out quite an extensive simulation of two examples in the case of a two-class queue. Exhaustive policies were found to outperform non-exhaustive policies in all cases. Further, these simulations showed that threshold policies are the best performing among all exhaustive, threshold policies. Apart from having a direct argument about the optimality of exhaustive policies, the question of effectiveness of the threshold policies needs to be looked into in future.

This research was supported in part by the Office of Naval Research and the Department of Science and Technology grant N00014-93-1017. We would also like to acknowledge the excellent facilities at the Intelligent Systems Laboratory, Department of Computer Science and Automation, Indian Institute of Science.

## References

- Elmaghraby S E 1978 The economic lot scheduling problem (ELSP): Review and extensions. *Manage. Sci.* 28: 587-598

- Fischer W, Meier-Hellstern K 1993 The Markov-modulated Poisson process (MMPP) cookbook. *Performance Evaluation* 18: 149-171
- Frost V S, Melamed B 1994 Traffic modeling for telecommunication networks. *IEEE Commun. Mag.* 10: 70-81
- Harrison J M 1975 A priority queue with discounted linear costs. *Oper. Res.* 23: 260-269
- Hemachandra N, Narahari Y 1995 A linear programming approach to Markovian switching among Poisson streams to a queueing system. Technical report, Department of Computer Science and Automation, Indian Institute of Science, Bangalore
- Hofri M, Ross K W 1987 On the optimal control of two queues with server setup times and its analysis. *SIAM J. Comput.* 16: 399-420
- Lippman S 1973 Semi-Markov decision processes with unbounded rewards. *Manage. Sci.* 19: 717-731
- Ravikumar K 1996 *Dynamic and stochastic scheduling of multi-product queues with setups: A diffusion approach*. Ph D dissertation, Department of Computer Science and Automation, Indian Institute of Science, Bangalore
- Reiman M I, Wein L M 1994 Dynamic scheduling of a two-class queue with setups. Technical report, Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA