Centrum voor Wiskunde en Informatica

REPORT*RAPPORT*

Assigning a Single Server to Inhomogeneous Queues with Switching Costs

G. Koole

Department of Operations Reasearch, Statistics, and System Theory

BS-R9405 1994

Assigning a Single Server to Inhomogeneous Queues with Switching Costs

Ger Koole

CWI

P.O. Box 94079, 1090 GB Amsterdam, The Netherlands

Abstract

In this paper we study the assignment of a single server to two queues. Customers arrive at both queues according to Poisson processes, and all service times are exponential, but with rates depending on the queues. The costs to be minimized consist of both holding costs and switching costs. The limiting behavior of the switching curve is studied, resulting in a good threshold policy. Numerical results are included to illustrate the complexity of the optimal policy and to compare the optimal policy with the threshold policy.

AMS Subject Classification (1991): 90B22, 90C40
Keywords & Phrases: Control of queues, threshold policies, value iteration, polling models
Note: Supported by the European Grant BRA-QMIPS of CEC DG XIII
Running Head: Server assignment to queues with switching costs

1. Introduction

Our model consists of 2 queues with Poisson arrivals (with rate λ_i at queue i) and exponential service times (with rate μ_i at queue i). There are holding costs (c_i at queue i) for each time unit a customer spends in a queue. There is a single server, which has to divide its time between the queues. When the server moves from one queue to the other, switching costs are incurred (equal to s_{ij} if the server moves from queue i to queue j). The objective of this paper is to study the optimal preemptive dynamic assignment of the server to the queues, with respect to the long run discounted or average costs.

A special case of this model, with $\mu_1 = \mu_2$ and $c_1 = c_2$ (and with switching times instead of switching costs), has been studied in Hofri & Ross [2] and in Liu et al. [4]. In both papers it is shown that the optimal policy serves each queue exhaustively. (In [4] a more general model is considered, allowing for more than 2 queues and different information structures.) In [2] it is conjectured that it is optimal for the server to switch from an empty queue to the other if the number of customers in the other queue exceeds a certain level. Such a policy is called a *threshold policy*. Our numerical results are in compliance with this conjecture.

Another special case, the one with $s_{12} = s_{21} = 0$, has been studied extensively. For this model the μc-rule is known to be optimal (e.g., Buyukkoc et al. [1]). The μc-rule serves, amongst the non-empty queues, a customer in the queue with highest $\mu_i c_i$.

In this paper we study the general case with arbitrary parameters.

Before going into the technical details of the paper, let us first do a numerical experiment, to obtain some insight in the model. This we do using dynamic programming (dp). Using standard arguments (on which we elaborate in section 2), we can reformulate our continuous time problem into a discrete time problem, and derive its dp equation.

Using a computer program, we computed the actions minimizing the α-discounted costs, (for a state space truncated at a sufficiently high level) for $\lambda_1 = \lambda_2 = 1$, $\mu_1 = \mu_2 = 6$, $c_1 = 2$, $c_2 = 1$, $s_{12} = s_{21} = 20$, and $\alpha = 0.95$. The results can be found in table 1. We denote the state of the system with (x, y), with $x = (x_1, x_2)$ the numbers of customers in the queues, and $y \in \{1, 2\}$ the position of the server. A "−" at position $x = (x_1, x_2)$ denotes that if the server is in state $(x, 1)$, then it is optimal to switch from queue 1 to queue 2. A "+" denotes that it is optimal to switch to queue 1 in state $(x, 2)$. A "·" denotes that the server stays at the present queue. In the table the state space is truncated at x_1, $x_2 \leq 15$, but the computations were done for higher truncation levels.

$x_2 = 15$	−	·	·	·	+	+	+	+	+	+	+	+	+	+	+	+
14	−	·	·	·	+	+	+	+	+	+	+	+	+	+	+	+
13	−	·	·	·	+	+	+	+	+	+	+	+	+	+	+	+
12	−	·	·	·	+	+	+	+	+	+	+	+	+	+	+	+
11	−	·	·	·	+	+	+	+	+	+	+	+	+	+	+	+
10	−	·	·	·	+	+	+	+	+	+	+	+	+	+	+	+
9	−	·	·	·	+	+	+	+	+	+	+	+	+	+	+	+
8	−	·	·	·	+	+	+	+	+	+	+	+	+	+	+	+
7	−	·	·	·	+	+	+	+	+	+	+	+	+	+	+	+
6	−	·	·	·	+	+	+	+	+	+	+	+	+	+	+	+
5	−	·	·	·	·	+	+	+	+	+	+	+	+	+	+	+
4	−	·	·	·	·	+	+	+	+	+	+	+	+	+	+	+
3	−	·	·	·	·	·	+	+	+	+	+	+	+	+	+	+
2	·	·	·	·	·	·	+	+	+	+	+	+	+	+	+	+
1	·	·	·	·	·	·	·	+	+	+	+	+	+	+	+	+
0	·	·	+	+	+	+	+	+	+	+	+	+	+	+	+	+
	$x_1 = 0$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Table 1. The optimal switching policy for $\lambda_1 = \lambda_2 = 1$, $\mu_1 = \mu_2 = 6$, $c_1 = 2$, $c_2 = 1$, $s_{12} = s_{21} = 20$, $\alpha = 0.95$

Several interesting conclusions can be drawn from this table. In the first place, as the policy in the table does not have a simple form, it seems unlikely that the optimal policy can be described easily.

Switching from queue 1 to queue 2 occurs only if $x_1 = 0$, i.e., queue 1 is served exhaustively. This we can prove (see section 2), for all cases with $\mu_1 c_1 \geq \mu_2 c_2$. Thus the queue that would get higher priority under the μc-rule in the case without switching costs, is served exhaustively. We also show in section 2 that if $x_1 = 0$, then queue 2 is served exhaustively. This is also in compliance with the μc-rule.

From table 1 it is clear that it is not optimal always to serve queue 2 exhaustively; if there are sufficient customers in queue 1, it pays to switch to queue 1. Note that serving queue 1 reduces the holding costs at a faster rate than by serving queue 2. (This is the intuitive explanation of the optimality of the μc-rule.) However, to reduce costs at a faster rate we have to invest in the form of switching costs. This investment only pays if there are enough customers in queue 1 to serve. This suggests a threshold level for x_1, at which to switch to queue 1. In our example, this threshold clearly depends on x_2, but becomes constant for

$x_2 > 5$ (and remains the same for values of x_2 well beyond 15). The intuition behind this is that if there are only a few customers in queue 2, it is better to serve these first before switching to queue 1, thereby avoiding having to switch back to queue 2 after serving queue 1 exhaustively. This complex behavior lends itself hardly for analysis, but for the discounted costs criterion, and for x_2 big enough, we can prove that the optimal policy does not depend on x_2. This is done in section 3, and we show how the optimal policy for x_2 large can be computed. In section 4 we compare numerically the optimal policy and several threshold policies, one of which is based on the limiting policy obtained in section 3.

2. Exhaustive policies

In this section we formally derive the discrete time dp equation, and prove some properties of this model, which partially describes the optimal policy.

In Serfozo [6] it is shown that each continuous time Markov decision process with uniformly bounded transition rates is equivalent to a discrete time Markov decision chain, for the discounted cost criterion. In our model, the sum of the transition rates is bounded by $\gamma = \lambda_1 + \lambda_2 + \mu$ in each state (with $\mu = \max_i \mu_i$). By adding fictitious transitions from a state to itself, we can assume that the sum of the rates in each state is equal to γ. Let the costs at t in the continuous time model be discounted with a factor β^t. Then, according to [6], the optimal policy in the continuous time model is the same as the optimal policy in the discrete time model with the transition rates divided by γ as transition probabilities, and with discount factor $\alpha = \gamma/(\log(\beta^{-1}) + \gamma)$. In each state, the transition probabilities sum to one due to the fictitious transitions. The minimal discounted costs in both models are equal up to a multiplicative factor γ. A similar result holds for the average cost case.

In our model we do not allow for idleness of the server at the current queue, if there are customers available at that queue. If $c_i > 0$, it can indeed be shown that idleness is suboptimal, in the same way as Liu et al. [4] show it for their model. Note however, that the optimal policy need not be work conserving: in the example of the previous section the server remains in state $((0,1),1)$ at the empty queue 1, while there is a customer waiting in queue 2.

Assume, without restricting generality, that $\lambda_1 + \lambda_2 + \mu = 1$. Recall that $x = (x_1, x_2)$ denote the queue lengths, and that y is the queue presently being served. The dp equation of the discrete time model is then as follows (with $e_1 = (1,0)$ and $e_2 = (0,1)$):

$$V^n(x,y) = \min\left\{\hat{V}^n(x,y), s_{yz} + \hat{V}^n(x,z)\right\},\ z = y \bmod 2 + 1, \tag{2.1}$$

$$\hat{V}^{n+1}(x,y) = x_1c_1 + x_2c_2 + \alpha\lambda_1 V^n(x+e_1,y) + \alpha\lambda_2 V^n(x+e_2,y) + \tag{2.2}$$

$$\alpha\mu_y V^n((x-e_y)^+,y) + \alpha(\mu-\mu_y)V^n(x,y),$$

with $V^0(x,y) = 0$. Here $\hat{V}^n$ serves as an intermediate variable, making the notation easier. It is well known that (under some technical conditions) $V^n(x,y)$ converges to the minimal discounted costs $V^\alpha(x,y)$ for all x and y, and that the actions minimizing $V^n(x,y)$ converge to the minimizing actions in (x, y). If $\alpha = 1$, then $V^{n+1}(x, y) - V^n(x, y)$ converges to the minimal average costs.

In the remainder of this paper we study the discrete time model, whose dp equation is given by (2.1) and (2.2). Assume that $\mu_1 c_1 \geq \mu_2 c_2 \geq 0$, and that $s_{12}, s_{21} \geq 0$. To show that queue 1 should always be served exhaustively, we need a technical lemma. Define $\bar{\mu}_i = \mu - \mu_i$, and note that $\alpha < 1$ is equivalent to taking $\lambda_1 + \lambda_2 + \mu < 1$ and $\alpha = 1$. Therefore we will suppress α in the notation, and drop the condition that $\lambda_1 + \lambda_2 + \mu = 1$. (The case $\lambda_1 + \lambda_2 + \mu = 1$ represents the average cost case.)

Lemma 2.1 *For $n = 0, 1, \ldots$, we have*

$$\mu_1 V^n(x - e_1, 1) + \bar{\mu}_1 V^n(x, 1) \leq \mu s_{12} + \mu_2 V^n(x - e_2, 2) + \bar{\mu}_2 V^n(x, 2), \;\; x > 0, \tag{2.3}$$

and

$$V^n(x, y) \leq V^n(x + e_i, y). \tag{2.4}$$

Proof. It is easily seen that

$$V^n(x, y) \leq s_{yz} + V^n(x, z), \;\; z \neq y. \tag{2.5}$$

We will show (2.3) and (2.4) inductively, starting with (2.3). For $n = 0$ the inequality holds. Assume it holds up to n. Instead of proving (2.3) for $n + 1$, we prove

$$\mu_1 \hat{V}^{n+1}(x - e_1, 2) + \bar{\mu}_1 \hat{V}^{n+1}(x, 2) \leq \mu_2 V^{n+1}(x - e_2, 2) + \bar{\mu}_2 V^{n+1}(x, 2).$$

This is clearly sufficient, as $V^{n+1}(x, 1) \leq s_{12} + \hat{V}^{n+1}(x, 2)$ for all x. We have to distinguish between all combinations of actions in $(x - e_2, 2)$ and $(x, 2)$. The optimal actions in these states are denoted with a_1 and a_2, respectively. If $a_1 = a_2 = 1$, we have to show

$$\mu_1 \hat{V}^n(x - e_1, 2) + \bar{\mu}_1 \hat{V}^n(x, 2) \leq \mu s_{21} + \mu_2 \hat{V}^n(x - e_2, 1) + \bar{\mu}_2 \hat{V}^n(x, 1).$$

If we insert (2.1) and use (2.5) (and the fact that $s_{21} \geq 0$), this inequality follows without difficulty.

If $a_1 = a_2 = 2$, we also write the inequality in terms of V^n, and then we use, by induction, (2.3) (or (2.4), if $x_2 = 1$).

For $a_1 = 1$ and $a_2 = 2$, we have, for the terms concerning arrivals,

$$\lambda_i \mu_1 V^n(x - e_1 + e_i, 2) + \lambda_i \bar{\mu}_1 V^n(x + e_i, 2) \leq$$

$$\lambda_i \mu_2 s_{21} + \lambda_i \mu_2 \hat{V}^n(x - e_2 + e_i, 1) + \lambda_i \bar{\mu}_2 \hat{V}^n(x + e_i, 2),$$

by induction and (2.5).

For the terms concerning departures in $\mu_1 \hat{V}^{n+1}(x - e_1, 2) + \bar{\mu}_1 \hat{V}^{n+1}(x, 2)$ we have

$$\mu_1\mu_2 V^n(x-e_1-e_2,2)+\mu_1\bar{\mu}_2 V^n(x-e_1,2)+\bar{\mu}_1\mu_2 V^n(x-e_2,2)+\bar{\mu}_1\bar{\mu}_2 V^n(x,2)\le$$

$$\mu\mu_2 s_{21}+\mu_2\mu_1 V^n(x-e_1-e_2,1)+\mu_2\bar{\mu}_1 V^n(x-e_2,1)+$$

$$\bar{\mu}_2\mu_2 V^n(x-e_2,2)+\bar{\mu}_2\bar{\mu}_2 V^n(x,2),$$

which holds again by induction and (2.5). The r.h.s. consists of the departure terms in $\mu_2 V^{n+1}(x-e_2,1)+\bar{\mu}_2 V^{n+1}(x,2)$ if $a_1=1$ and $a_2=2$. Summing these terms, together with $0\le(1-\lambda_1-\lambda_2-\mu)s_{21}$, gives the inequality.

The case with $a_1=a_2=2$ follows in a similar way.

Equation (2.4) follows easily, by taking the action in (x,y) equal to the optimal action in $(x+e_i,y)$. □

Now we can show that queue 1 should always be served exhaustively.

Theorem 2.2 *For $n=0,1,\ldots,$ we have*

$$\hat{V}^n(x,1)\le s_{12}+\hat{V}^n(x,2)\ \textit{if}\ x_1>0, \tag{2.6}$$

showing that queue 1 should be served exhaustively.

Proof. Again by induction. Using (2.5), it is easily seen that

$$\lambda_i V^n(x+e_i,1)\le\lambda_i s_{12}+\lambda_i V^n(x+e_i,2).$$

By (2.3) (or (2.4), if $x_2=0$) we have

$$\mu_1 V^n(x-e_1,1)+\bar{\mu}_1 V^n(x,1)\le\mu s_{12}+\mu_2 V^n((x-e_2)^+,2)+\bar{\mu}_2 V^n(x,2).$$

Summing the inequalities (and using that $s_{12}\ge 0$) gives (2.6) for $n+1$. As $\lim_{n\to\infty}\hat{V}^n(x,y)=\hat{V}^\alpha(x,y)$, (2.6) holds also for $\hat{V}^\alpha$. Thus, if $x_1>0$, the action minimizing $V^n(x,1)$, is staying at queue 1. This shows that the optimal discounted policy serves queue 1 exhaustively. □

Theorem 2.3 *For $n=0,1,\ldots,$ we have*

$$\hat{V}^n(x,2)\le s_{21}+\hat{V}^n(x,1)\ \textit{if}\ x_1=0, \tag{2.7}$$

showing that queue 2 should be served as long as queue 1 is empty.

Proof. The proof is similar to that of theorem 2.2. It is easily seen that

$$\lambda_i V^n(x+e_i,2)\le\lambda_i s_{21}+\lambda_i V^n(x+e_i,1),$$

and

$$\mu_2 V^n((x-e_2)^+,2) + \bar{\mu}_2 V^n(x,2) \le \mu s_{21} + \mu V^n(x,1),$$

holds by (2.4) and (2.5). □

If $s_{12} = s_{21} = 0$ it follows from the theorems 2.2 and 2.3 that the μc-rule is optimal. Indeed, because in this case $V^n(x,y) = \min\{\hat{V}^n(x,1), \hat{V}^n(x,2)\}$, it follows from theorem 2.2 that queue 1 should always be served if $x_1 > 0$; by theorem 2.3 we know that if $x_1 = 0$ and $x_2 > 0$ queue 2 should be served. A simpler iterative proof of the optimality of the μc-rule can be found in Hordijk & Koole [3].

I tried to generalize the results of this section to more than 2 queues using similar arguments, but failed.

Remark 1. We assumed the arrivals to be Poisson, but without losing the results of this section, we can allow them to be more general, for example a Markov arrival process (MAP). Note however that in this case the optimal policy will also depend on the state of the arrival process. In [3] the MAP is used for a related control model.

Remark 2. In the literature on polling models it is customary to study the expected (weighted) waiting time of an arbitrary customer. The problem of finding the optimal policy for this criterion is equivalent to finding the policy that minimizes the expected (weighted) sojourn time, as each customer's expected service time is fixed. By Little's theorem, minimizing $\sum_i \lambda_i \hat{c}_i \mathbb{E} W_i$ is equivalent to minimizing $\sum_i \hat{c}_i \mathbb{E} L_i$, where $\hat{c}_i$ is a weighting factor, and L_i is the stationary queue length at queue i. But this is equivalent to the average cost case studied in this section, with $c_i = \hat{c}_i$. Thus the results proved in this section hold also if the objective is to minimize the expected waiting times.

3. Asymptotic analysis

In this section we will study the actions minimizing $V^\alpha(x,y)$ for x_2 large. To do this, we consider the optimal actions in $V^n(x,y)$ for $n \le x_2$, and n large (and thus also x_2 large). These results are used to derive ϵ-optimal policies for the discounted cost criterion.

Lemma 3.1 *If $x_2 \ge n$, then $V^n(x+e_2,y) = V^n(x,y) + \frac{1-\alpha^n}{1-\alpha} c_2$.*
Furthermore, the optimal actions in $V^n(x,y)$ and $V^n(x+e_2,y)$ are equal.

Proof. We use induction to n. Assume that $V^n(x+e_2,y) - V^n(x,y) = \frac{1-\alpha^n}{1-\alpha} c_2$ for all x with $x_2 \ge n$. Then the actions minimizing $V^n(x,y)$ and $V^n(x+e_2,y)$ are equal. (Note that adding a constant to $V^n(x,y)$ does not change the optimal action, giving the second part of the theorem.) Now look at $V^{n+1}(x'+e_2,y) - V^{n+1}(x',y)$, with $x'_2 \ge n+1$. For all states x that can be reached from x' in one step, we have $x_2 \ge n$. Therefore

$$V^{n+1}(x'+e_2,y) - V^{n+1}(x',y) = c_2 + \alpha \frac{1-\alpha^n}{1-\alpha} c_2 = \frac{1-\alpha^{n+1}}{1-\alpha} c_2.$$

□

Thus, if $x_2 \geq n$, the optimal policy does not depend on x_2. To study the limiting behavior as both n and x_2 go to ∞, we consider a model with states (x_1, y), where x_1 is the number of customers in the single queue of the system, and $y \in \{1,2\}$ the position of the server: the server is at the queue only if $y = 1$. The dp equation is:

$$W^n(x_1,y) = \min\left\{\hat{W}^n(x_1,y), s_{yz} + \hat{W}^n(x_1,z)\right\},\ z = y \bmod 2 + 1,$$

$$\hat{W}^{n+1}(x_1,1) = x_1c_1 + \alpha\lambda_1 W^n(x_1+1,1) + \alpha\lambda_2\Big(\frac{1-\alpha^n}{1-\alpha}c_2 + W^n(x_1,1)\Big)$$

$$+\alpha\mu_1 W^n((x_1-1)^+,1) + \alpha(\mu-\mu_y)W^n(x_1,1),$$

and

$$\hat{W}^{n+1}(x_1,2) = x_1c_1 + \alpha\lambda_1 W^n(x_1+1,1) + \alpha\lambda_2\Big(\frac{1-\alpha^n}{1-\alpha}c_2 + W^n(x_1,2)\Big)$$

$$-\alpha\mu_2\frac{1-\alpha^n}{1-\alpha}c_2 + \alpha\mu W^n(x_1,2).$$

This dp equation can be interpreted as originating from the original model but with an infinite number of class 2 customers. Indeed, if $y = 2$ the costs are reduced with a factor $\alpha\mu_2\frac{1-\alpha^n}{1-\alpha}c_2$. This is equal to the probability of a class 2 departure, times the expected costs incurred for a class two customer who stays in the system for the remaining n periods.

Note that the dp equation has a somewhat unusual form, as the costs depend on n. However, the discounted costs are equal to those for the model with α^n replaced by 0 (which can by proved by considering the optimality equation). To distinguish between both models, we will refer to the original model as the V-model, and to the model with value function W^n as the W-model.

Lemma 3.2 *If $x_2 \geq n$, then $V^n(x,y) = W^n(x_1,y) + x_2\frac{1-\alpha^n}{1-\alpha}c_2$.*

Proof. The proof is similar to that of lemma 3.1. Assume that $V^n(x,y) - W^n(x_1,y) = x_2\frac{1-\alpha^n}{1-\alpha}c_2$ for all x with $x_2 \geq n$. For x' with $x_2' \geq n+1$, it is straightforward to show, using lemma 3.1, that

$$V^{n+1}(x',y) - W^{n+1}(x_1',y) = x_2c_2 + \alpha x_2\frac{1-\alpha^n}{1-\alpha}c_2 = x_2\frac{1-\alpha^{n+1}}{1-\alpha}c_2.$$

□

Now we can compute W^α and the optimal actions in each state. Based on this optimal policy for the W-model we define a policy R_T for the original V-model, which takes as action in (x,y) the optimal action for the W-model, in state (x_1,y). Note that under R_T, the original model and the optimal policy for the W-model treat the class 1 customers exactly the same. Obvious improvements can be made to R_T, like not switching from queue 1 to queue 2 if $x = (0,0)$.

Theorem 3.3 *For all $\epsilon > 0$, there is a N such that $|V^\alpha_{R_T}(x,y) - V^\alpha(x,y)| < \epsilon$ if $x_2 \geq N$.*

Proof. Using the triangle inequality, we have

$$|V^\alpha_{R_T}(x,y) - V^\alpha(x,y)| \leq |V^\alpha(x,y) - V^{x_2}(x,y)| + |W^\alpha(x_1,y) - W^{x_2}(x_1,y)| +$$

$$|V^{x_2}(x,y) - W^{x_2}(x_1,y) - x_2c_2\tfrac{1-\alpha^{x_2}}{1-\alpha}| + |x_2c_2\tfrac{1-\alpha^{x_2}}{1-\alpha} - x_2c_2\tfrac{1}{1-\alpha}| +$$

$$|V^\alpha_{R_T}(x,y) - W^\alpha(x_1,y) - x_2c_2\tfrac{1}{1-\alpha}|.$$

For each of the terms on the r.h.s. we show that it goes to 0, as x_2 tends to ∞.

Let us start with $V^\alpha(x,y) - V^{x_2}(x,y)$. Discounting can be interpreted as taking the total costs over a geometrically distributed horizon. Thus $V^\alpha(x,y)$ can be seen as the minimal costs for a control problem with horizon X, where X is geometrically distributed with parameter α. Similarly, $V^n(x,y)$ can be seen as a problem with horizon $\max\{X,n\}$. Note that the policies used to calculate V^n differ from the optimal discounted policy; a different horizon gives different optimal policies. As the direct costs for V^n are positive, it is easily seen that $V^n(x,y) \leq V^\alpha(x,y)$. Now we bound the costs for the case that $X > n$, which occurs with probability α^n. At time n there are $x_1 + x_2 + n$ customers in the system or less (the number $x_1 + x_2 + n$ corresponds to all events being arrivals). If these customers are not served, their costs after n are bounded by $\alpha^n(x_1 + x_2 + n)\max\{c_1,c_2\}/(1-\alpha)$. The costs of a customer arriving at time k can be bounded by $\alpha^k \max\{c_1,c_2\}/(1-\alpha)$. Summing this for $k = n$ to ∞ gives a bound for customers arriving after n, which is equal to $\alpha^n \max\{c_1,c_2\}/(1-\alpha)^2$. The switching costs can be bounded by $\max\{s_{12},s_{21}\}/(1-\alpha)$. Together, this gives

$$V^n(x,y) \leq V^\alpha(x,y) \leq V^n(x,y) + \alpha^n \tfrac{(x_1+x_2+n+(1-\alpha)^{-1})\max\{c_1,c_2\}+\max\{s_{12},s_{21}\}}{1-\alpha},$$

from which we conclude that $V^\alpha(x,y) - V^{x_2}(x,y) \to 0$, as $x_2 \to \infty$.

In a similar way, we can find bounds for W^α. The negative costs after n are bounded by $\alpha^n\mu_2c_2/(1-\alpha)^2$. Therefore

$$W^n(x_1,y) \leq W^\alpha(x_1,y) + \alpha^n\mu_2c_2/(1-\alpha)^2.$$

On the other hand

$$W^\alpha(x_1,y) \leq W^n(x_1,y) + \alpha^n \tfrac{(x_1+n+(1-\alpha)^{-1})\max\{c_1,c_2\}+\max\{s_{12},s_{21}\}}{1-\alpha},$$

giving that $W^\alpha(x_1,y) - W^{x_2}(x_1,y) \to 0$, as $x_2 \to \infty$.

By lemma 3.2 we know that $V^{x_2}(x,y) = W^{x_2}(x_1,y) + x_2c_2\frac{1-\alpha^{x_2}}{1-\alpha}$, making the third term 0.

Obviously, $x_2c_2\frac{1-\alpha^{x_2}}{1-\alpha} - x_2c_2\frac{1}{1-\alpha} \to 0$, as $x_2 \to \infty$.

To prove that the last term tends to 0, we do not consider V^n or W^n, with a possibly different policy for each n, but we restrict ourself to the policy R_T. If we add a term x_2c_2

to the direct costs in W^n, resulting in $\bar{W}^n$, the term becomes $|V^{\alpha}_{R_T}(x, y) - \bar{W}^{\alpha}(x_1, y)|$. Now, if R_T is employed at all times, then the direct costs in both systems are equal, until $x_2 = 0$. Note that the switching costs and the holding costs in queue 1 are always equal. Therefore, using similar arguments as above, we have

$$|V^{\alpha}_{R_T}(x, y) - \bar{W}^{\alpha}(x_1, y)| \leq \alpha^{x_2} c_2/(1-\alpha)^2.$$

This gives the bound on the fifth term. □

The next step in finding good and simple policies is characterizing the policy R_T. Although we conjecture that R_T is a threshold policy, we were not able to prove it. Instead, we computed R_T for various choices of parameters. In all instances a threshold policy was found. Note that the computation of R_T takes little time compared to the overall optimal policy, due to the reduction in size of the state space. In the next section we report on the numerical results.

Remark. In the literature, I found one paper on the control of a queue with switching costs between the actions, Lu & Serfozo [5]. However, a detailed study of it learned that there is an unrepairable error in the proof of the main result. Indeed, in the proof of Lemma 1 on p. 1128 of [5] it is stated that $R_n = 0$ for case C2 (which contains the holding cost case). This is not correct, and examples contradicting (25) can be constructed. This disproves Lemma 1 for case C2, on which the main result, Theorem 1, is based.

4. Numerical results

First we computed the optimal W-policy for several instances. A table showing the optimal policy, for the same parameters as used in table 1, can be found in table 2.

−	·	·	·	+	+	+	+	+	+	+	+	+	+	+	+
$x_1 = 0$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Table 2. The optimal W-policy for $\lambda_1 = \lambda_2 = 1$, $\mu_1 = \mu_2 = 6$, $c_1 = 2$, $c_2 = 1$, $s_{12} = s_{21} = 20$, $\alpha = 0.95$

We see that the server switches from serving queue 2 to queue 1 as soon as x_1 reaches the threshold level (in this case 4), which is what we conjectured to be the limiting behavior for the V-model. For all considered instances the optimal W-policy has the same structure.

From this we construct the policy R_T for the V-model, by taking in $((x_1, x_2), y)$ the action which is optimal in (x_1, y) for the W-model. We make an exception for the states with $x_2 = 0$; there we assume the policy to be work conserving, that is, it switches to queue 1 only if there are customers at queue 1, and in state $((0, 0), 1)$ the server does not switch to queue 2 (which is obviously suboptimal). Thus R_T becomes as depicted in table 3.

	$x_1=0$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$x_2=8$	−	·	·	·	+	+	+	+	+	+	+	+	+	+	+	+
7	−	·	·	·	+	+	+	+	+	+	+	+	+	+	+	+
6	−	·	·	·	+	+	+	+	+	+	+	+	+	+	+	+
5	−	·	·	·	+	+	+	+	+	+	+	+	+	+	+	+
4	−	·	·	·	+	+	+	+	+	+	+	+	+	+	+	+
3	−	·	·	·	+	+	+	+	+	+	+	+	+	+	+	+
2	−	·	·	·	+	+	+	+	+	+	+	+	+	+	+	+
1	−	·	·	·	+	+	+	+	+	+	+	+	+	+	+	+
0	·	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+

Table 3. The policy R_T for $\lambda_1 = \lambda_2 = 1$, $\mu_1 = \mu_2 = 6$, $c_1 = 2$, $c_2 = 1$, $s_{12} = s_{21} = 20$, $\alpha = 0.95$

We compared the optimal policy and R_T as derived from the W-model, for various problem instances. We also included two other simple policies in our computations. These are the policy which serves not only queue 1, but also queue 2 exhaustively, which can be seen as R_T with threshold level ∞ (and therefore denoted with R_∞), and the list policy which gives priority to queue 1, which is R_T with threshold level 1 (denoted with R_1). Note that R_1 coincides with the μc-rule.

Our first observation is that the performance of the policies depends on the initial states. This is illustrated in table 4, where we list the discounted costs for various starting states. The computations for the optimal policy R^* were done by calculating (2.1) and (2.2), for n large enough. Also the computations for R_T, R_1 and R_∞ were done with the dp equation, by inserting the policy in (2.1) instead of taking the minimizing actions. To make computations possible we had to truncate the state space. We took the truncation levels high enough to be sure that the resulting numbers are equal to the ones for the model without truncation.

(x', y')	((0,0),1)	((0,0),2)	((10,0),1)	((10,0),2)	((0,10),1)	((0,10),2)	((10,10),1)	((10,10),2)
R^*	40.76	45.01	176.8	196.8	139.6	119.6	332.8	352.8
R_T	56.95	56.95	184.1	204.1	146.3	126.3	335.4	355.4
R_1	63.60	63.60	189.4	209.4	177.1	157.1	350.4	370.4
R_∞	56.95	56.95	184.1	204.1	146.4	126.4	335.6	420.6

Table 4. Values for different policies and initial states (x', y'), for $\lambda_1 = \lambda_2 = 1$, $\mu_1 = \mu_2 = 6$, $c_1 = 2$, $c_2 = 1$, $s_{12} = s_{21} = 20$, $\alpha = 0.95$

First we observe that the values for R_T, R_1 and R_∞ do not depend on y, if $x = (0,0)$. This can be explained by the fact that the server serves the first customer that arrives. As $\lambda_1 = \lambda_2$, this occurs at each queue with the same probability. Because also $s_{12} = s_{21}$, we find $V_R^\alpha((0,0),1) = V_R^\alpha((0,0),2)$, for $R = R_T$, R_1 and R_∞. Furthermore it is observed that the difference between entries in different columns is often equal to 20, the switching costs. The differences can easily be explained by looking at the structure of the policies involved. Of course R^* performs best, but note that R_T performs better than the other two.

Let us see what the influence of α on the results is. For α ranging from 0.5 up to 1 (representing average costs) we computed the values in state $((5,5),2)$. Note that for $\alpha = 1$, W^n is not defined. We derived R_T in this case directly from R^*. Note that taking α close to 1, reduces the dependence on the starting state. Furthermore, $\alpha = 0.95$ corresponds to a reasonable interest rate of ≈ 0.05. No low values of α are considered, as a discount rate of $\beta = 0.1$ in the continuous time model gives in the discrete time model $\alpha = \gamma/(\log(\beta^{-1})+\gamma) \approx 0.78$. The results can be found in table 5. In the table one can also find the values of T, the threshold level on which R_T is based.

α	0.5	0.75	0.8	0.85	0.9	0.95	0.98	1
R^*	29.27	56.55	69.39	87.16	114.8	164.6	267.0	2.722
R_T	29.47	57.36	69.87	88.41	118.4	170.7	283.9	3.093
T	∞	∞	∞	8	5	4	3	3
R_1	48.04	71.69	82.37	98.49	125.7	185.9	313.9	3.470
R_∞	29.47	57.36	69.87	88.39	118.6	180.9	302.1	3.088

Table 5. Values for different policies and discount factors, for $\lambda_1 = \lambda_2 = 1$, $\mu_1 = \mu_2 = 6$, $c_1 = 2$, $c_2 = 1$, $s_{12} = s_{21} = 20$, and initial state $((5,5),2)$

It is interesting to note that, as long as $\alpha \leq 0.75$, the optimal policy does not switch to the other queue. The optimal policy for the W-model has threshold level ∞ here. For $\alpha = 0.8$ R^* switches to queue 1 in states $((x_1,0),2)$ if $x_1 \geq 5$. Note that for the average cost case R_∞ performs better than R_T. As the traffic is low, this can be explained by the fact that R_∞ approximates the "nose" of the optimal policy better than R_T does. (We call the roughly triangular subset of the state space where R^* deviates from R_T, which is best illustrated in table 1, the nose of the optimal policy.)

Finally, we change the parameters of the system, keeping the discount rate and the initial state constant. First, let us change λ_2. We expect the nose to be larger if λ_2 is small, to avoid having to return to queue 2 after serving queue 1 exhaustively. Such a large nose is best approximated by R_∞. Indeed, we see in table 6 that for $\lambda_2 = 0.1$, R_∞ is slightly better than R_T. However, for λ_2 large, we see that R_T outperforms R_∞. For $\lambda = 5$, the system is unstable. As we are considering discounted costs, this does not cause problems. We also see that for larger values of λ, R_T behaves better compared to R^*. This can be explained by the fact that under high loads x_2 is relatively big. It is for these states that R_T approximates R^* best.

λ_2	0.1	0.5	1	2	4	5
R^*	133.9	150.3	164.6	190.9	248.7	278.1
R_T	138.1	155.5	170.7	195.6	249.7	278.6
T	4	4	4	4	4	3
R_1	152.9	170.4	185.9	211.6	265.9	293.7
R_∞	137.0	160.9	180.9	212.6	280.2	315.4

Table 6. Values for different policies and different values of λ_2, for $\lambda_1 = 1$, $\mu_1 = \mu_2 = 6$, $c_1 = 2$, $c_2 = 1$, $s_{12} = s_{21} = 20$, $\alpha = 0.95$, and initial state $((5,5),2)$

We now vary the value of c_1, ranging from 1 to 10. The results can be found in table 7. For $c_1 = 1$, we know by theorem 2.2 that both queue 1 and queue 2 should be served exhaustively. Therefore the values for R_T and R_∞ are equal, and lower than the value for R_1. The policy R_∞ behaves poorer and poorer if we increase c_1, and the optimal threshold value becomes 1, making R_T and R_1 equal.

c_1	1	2	3	5	10
R^*	114.1	164.6	192.7	246.4	375.0
R_T	122.7	170.7	198.3	251.9	381.1
T	∞	4	3	2	1
R_1	161.5	185.9	210.3	259.1	381.1
R_∞	122.7	180.9	239.1	355.4	646.4

Table 7. Values for different policies and different values of c_1, for $\lambda_1 = \lambda_2 = 1$, $\mu_1 = \mu_2 = 6$, $c_2 = 1$, $s_{12} = s_{21} = 20$, $\alpha = 0.95$, and initial state $((5,5),2)$

Finally we change the switching costs s_{12} and s_{21}. The results can be found in table 8. In case $s = s_{12} = s_{21} = 0$, then R_1 is optimal. This is indeed what we expect, because in this case the μc-rule, which is equal to R_1, is optimal. As the switching costs increase, R_1 gets worse and R_∞ better.

s	0	5	10	20	100
R^*	110.5	127.5	141.0	164.6	236.2
R_T	110.5	127.6	142.2	170.7	327.1
T	1	2	3	4	12
R_1	110.5	129.4	148.2	185.9	487.3
R_∞	144.3	153.5	162.6	180.9	327.1

Table 8. Values for different policies and different values of s, for $\lambda_1 = \lambda_2 = 1$, $\mu_1 = \mu_2 = 6$, $c_1 = 2$, $c_2 = 1$, $s_{12} = s_{21} = s$, $\alpha = 0.95$, and initial state $((5,5),2)$

REFERENCES

1. C. Buyukkoc, P. Varaiya, and J. Walrand. The $c\mu$ rule revisited. *Advances in Applied Probability*, 17:237–238, 1985.
2. M. Hofri and K.W. Ross. On the optimal control of two queues with server setup times and its analysis. *SIAM Journal on Computing*, 16:399–420, 1987.
3. A. Hordijk and G. Koole. On the optimality of LEPT and μc rules for parallel processors and dependent arrival processes. *Advances in Applied Probability*, 25:979–996, 1993.
4. Z. Liu, P. Nain, and D. Towsley. On optimal polling policies. *Queueing Systems*, 11:59–83, 1992.
5. F.V. Lu and R.F. Serfozo. $M|M|1$ queueing decision processes with monotone hysteretic optimal policies. *Operations Research*, 32:1116–1132, 1984.
6. R.F. Serfozo. An equivalence between continuous and discrete time Markov decision processes. *Operations Research*, 27:616–620, 1979.