OPTIMAL CONTROL OF A QUEUEING SYSTEM WITH THREE HETEROGENEOUS SERVERS

Ioannis Viniotis

Center for Communications & Signal Processing
Department of Electrical and Computer Engineering
North Carolina State University
Raleigh, NC 27695-7914

CCSP-TR-89/5

February 1989

Abstract

We study the problem of optimally controlling a three server queueing system. Arriving customers join a single queue, which is served by three servers, S_1, S_2 and S_3. The servers are exponential, but of different rates. The total expected time a customer spends in the system is to be minimized. We show that the optimal policy can be characterized by three thresholds, m_3, m_b and m_i, such that: S_1, the fastest server, should be always busy; S_3, the slowest server, should be activated only when m_3 or more customers wait in the queue, and, given that S_3 is idle (busy), S_2 should be activated only when m_i (m_b) customers wait in the queue. We use stochastic dominance arguments to establish the results.

1. Introduction

We consider the queueing system shown in figure 1. Customers arrive at an (infinite capacity) buffer in a Poisson stream of rate λ. The buffer is served by three servers S_1, S_2, S_3, of different capacities μ_1, μ_2 and μ_3 respectively. Without loss of generality, we assume that $\mu_1 > \mu_2 > \mu_3 > 0$. The service requirements are exponentially distributed, with parameter 1. Thus the time a customer spends at server S_i is exponentially distributed, with parameter μ_i. To avoid trivial cases, we assume that $0 < \lambda < \mu_1 + \mu_2 + \mu_3$.

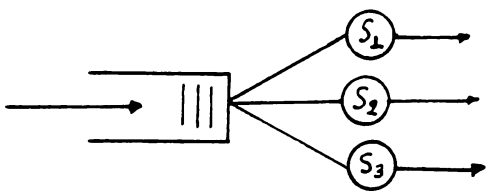

Figure 1.

The motivation for studying this model comes from applications in dynamic routing in computer networks and communication systems. The more general system with N servers would be a more appropriate model of real life applications; however, the enormity of the problem forces us to follow a moderate step–by–step approach.

This problem is a generalization of the $M|M|2$ model with unequal service rates

studied in [1]. In [1], Lin and Kumar have shown that the optimal policy (i.e., the one that minimizes the mean sojourn time of customers in the system), is of threshold type. In other words, the optimal policy keeps the faster server busy, whenever possible, and activates the slower server only when the number of waiting customers exceeds a certain threshold. Moreover, they have provided a simple formula to calculate the optimal threshold, as a function of the statistical parameters of the model. In [2], Agrawala et al considered a related problem, with an arbitrary number, N, of servers, but no arrivals. They have shown that a threshold type policy minimizes the expected total flow time (sum of all finishing times). They also provide a simple formula to calculate the threshold for each server. In [4,7,8,9] other scheduling problems with N servers have been considered.

Our goal is to minimize the expected time a customer spends in the system, by properly selecting the customer allocation strategy. Whenever one or more servers become idle, and there are customers waiting for service, one may or may not decide to forward one or more customers for service. We show in this paper that the optimal policy (among all nonanticipative, nonpreemptive policies) is of threshold type (to be precisely defined later), i.e., it may idle a server, even when there are waiting customers. We used stochastic dominance arguments to establish this result, as was done in [3] for the $M|M|2$ case. We were not able to utilize Dynamic Programming arguments.

The paper is organized as follows. In section 2 we describe the queueing model of the system and inroduce some notation. In section 3 we present the results; the proofs are provided in the appendix. Since the technique we are using is based on the arguments presented in [3], we omit lengthy details, whenever they can be readily found there. Where necessary, we emphasize the basic differences and provide only a sketch of the proof. We conclude with a conjecture regarding the $M|M|N$ model and some implementation remarks.

2. The model

The system consists of a single queue, served by three unequal speed servers. Arrivals are Poisson of rate λ. The service discipline is nonpreemptive*; service time at server S_i is

* If we allow preemptions, the optimal policy has a simple form (for the more gen-

exponential, with rate μ_i. To ensure stability of the system, we assume that $\lambda < \mu_1 + \mu_2 + \mu_3$. As mentioned above, we may assume without loss of generality that $\mu_1 > \mu_2 > \mu_3$. Let

x_{0t} denote the number of customers waiting in queue, at time t.

x_{it} denote the busy–idle condition of server S_i, $i = 1, 2, 3$.

If $x_{it} = 0$ (1), we say that the server is idle (busy).

Under the statistical assumptions we adopted, the vector

$$x_t \stackrel{\Delta}{=} (x_{0t}, x_{1t}, x_{2t}, x_{3t})$$

is a suitable state description of the system. The state space of the system is

$$X \stackrel{\Delta}{=} \{0, 1, \cdots\} \times \{0, 1\}^3$$

A *policy* γ is any (nonanticipative, nonpreemptive) rule which at every time t decides (on the basis of the history $\{x_s, s \leq t\}$), whether to activate one or more idle servers, given that the queue is nonempty. We shall refer to allocation decisions as *actions* of the policy.

Let $|x_t| \stackrel{\Delta}{=} x_{0t} + x_{1t} + x_{2t} + x_{3t}$ denote the total number of customers in the system (queue and servers) at time t. For a given discount factor $\alpha > 0$, we define the expected, α–discounted cost incurred by policy γ, when starting from initial state $x_0 = x$, at time $t = 0$, as

$$J(x; \gamma) \stackrel{\Delta}{=} E_x^\gamma \int_0^\infty e^{-\alpha t} |x_t| dt \tag{1}$$

Here E_x^γ denotes expectation with respect to the probability law of the process x_t, when the policy γ is used and the initial state is x. We are primarily interested in the average cost criterion; however, in view of the results in [5], it suffices to consider the discounted cost criterion only.

---

eral $M|M|N$ case): keep all servers busy and preempt the slowest (currently busy) server whenever a faster server becomes idle; reallocate the preempted customer to the recently available server. In computer systems preemptions may be allowed; however, in communication networks preemption of a message is typically not allowed.

A policy is *optimal* if it minimizes the cost given in equation (1). It is known that the optimal policy for this problem is Markov, deterministic and stationary [5]. Let π be the optimal policy. It can be described as follows. Whenever a transition (an arrival or a service completion) occurs, that brings the system to state y, π chooses an action that makes the state jump to the value

$$\pi(y) = (\pi(y)_0, \pi(y)_1, \pi(y)_2, \pi(y)_3)$$

To simplify the notation, we shall use

$$\pi(y)_i, \quad i = 1, 2, 3$$

to denote the actions of the optimal policy. For example, $\pi(y_0, y_1, y_2, y_3)_3 = 1$ (0) means that at state (y_0, y_1, y_2, y_3) the optimal policy decides to activate the slowest server (keep it idle).

Remark: When the optimal policy assigns customers to more than one idle servers simultaneously, we will assume, without loss of generality, that it does so in a sequential manner, from the fastest to the slowest server.

3. Characterization of the optimal policy.

Our objective in this section is to characterize the structure of the optimal policy, π. We shall make use of the following definition.

Definition: A policy γ is called a *threshold* type policy (with thresholds m_3, m_b, m_i) if:

- it keeps S_1, the fastest server, busy whenever possible, i.e., whenever the queue is nonempty.
- it allocates a customer from the queue to the slowest server, S_3, iff at least m_3 customers wait in the queue, and
- it activates server S_2 when at least m_i (m_b) customers wait in the queue and server S_3 is idle (busy).

Remark: The fact that a threshold policy will activate a server only when the queue size exceeds the corresponding threshold does not mean that the server can not be busy when the queue size falls below this threshold.

We wish to show the following theorem.

Theorem: The optimal policy for the cost criterion (1) is a threshold type policy.

The proof is is based on lemmata 1–6 below, which describe certain of the properties of the optimal policy. The following proposition, our main result, is an immediate consequence of the ergodicity assumption and the results in [5].

Proposition: The policy that minimizes the expected time a customer spends in the system is of threshold type.

The following lemmata describe the properties of the optimal policy for the discounted cost criterion. They are self–explanatory. Their proofs are presented in the appendix.

Lemma 1: The optimal policy should activate a faster rather than a slower server.

Lemma 2: The optimal policy should keep the fastest server busy whenever customers wait in queue.

An immediate consequence of lemma 2 is that for states (x_0, x_1, x_2, x_3) with $x_0 \geq 1$, we should have $x_1 = 1$, since states $(x_0, 0, x_2, x_3)$ will occur only instantaneously (i.e., just before π takes an action).

Lemma 3 (4) characterizes the actions of the optimal policy for the second server, when the system visits states in which the slowest server is idle (busy).

Lemma 3: Given that S_3 is idle, the optimal policy should activate the second server, S_2, only when at least m_i customers wait in the queue.

Lemma 4: Given that S_3 is busy, the optimal policy should activate S_2, only when at least m_b customers wait in the queue.

From the remark of the previous section and lemma 1 we see that π should consider activating S_3 only in states $(y_0, 1, 1, 0)$. The next lemma characterizes these actions.

Lemma 5: The optimal policy should activate the slowest server only when at least m_3 customers wait in the queue.

A complete characterization of the optimal policy would require computation of the three thresholds, as a function of the statistical parameters of the model. We have only been able to provide a relation between them, as the following lemma states.

Lemma 6: $m_i \leq m_3, m_b \leq m_3$.

We have not been able to provide a relationship between m_i and m_b. However, we strongly suspect that $m_i = m_b$. In this case, the optimal policy for activating the second server will be a pure threshold policy, with a threshold that depends on the queue size only, and not the condition of the third server. This conjecture is in compliance with the results in [2]. Moreover, it is backed up by extensive numerical calculations of the optimal policy, using value iteration on the Dynamic Programming Equation.

4. Conclusions

We have studied the structure of the optimal policy for a three server queueing system. We have shown that the optimal policy is of threshold type, with the threshold for the second server possibly depending on the busy/idle condition of the slowest server. Stochastic dominance arguments were used to obtain the result. They strongly rely on the number of servers; their extension to the more general case of $N > 3$ servers is not straightforward. It is possible, however, to obtain the structure for this case as well, using Linear Programming based arguments, as described for example in [6].

The proof presented here is not constructive; therefore it does not suggest how the optimal policy thresholds can be computed, for the implementation of the optimal policy. The straightforward method of calculating the cost as a function of the thresholds [1] cannot be applied here. We suggest to use the value iteration for the discounted cost as a reasonable heuristic. In all cases we tried, it converged quite rapidly. Figure 2 shows the behavior of the optimal thresholds as a function of the discount factor, for given arrival and service rates. It also "supports" the previously mentioned conjecture, $m_i = m_b$.

Appendix

In this appendix we provide the proofs of lemmata 1–6. Throughout this appendix, σ_i will denote an exponential random variable with rate μ_i. It represents the service time of a typical customer, at server S_i. Whenever the proofs are similar, the reader is refered to [3] for the missing details. The basic idea is to construct a nonstationary, nonMarkov policy, $\tilde{\pi}$, which will improve the optimal policy π, whenever the latter does not have the property that the lemma describes.

Proof of lemma 1: Assume that $i < j$ and let σ_i, σ_j be the service times of a given customer, when it is served by server S_i or S_j respectively. By stochastic dominance, we may assume that $\sigma_i < \sigma_j$.

Consider two initial states, s_i and s_j, such that at state s_i (s_j) server S_i (S_j) is active and server S_j (S_i) idle. The remaining state components are the same. Let τ be the first time that π activates server S_i when starting from initial state s_i ($\tau \leq \infty$). Clearly, if $\tau > \sigma_i$, then $J(s_i) < J(s_j)$. If $\tau < \sigma_i$, then $J(s_i) = J(s_j)$ and thus the cost when starting from s_i is always no more then the cost when starting from s_j.

Note, however, that this does not necessarily mean that it is *optimal* to activate server S_i in a state with $y_0 \geq 1, S_i = 0, S_j = 0$. Idling both servers may be optimal.

Proof of lemma 2: By lemma 1, whenever server S_1 is idle, it should be prefered to other idle servers. Since idling the fastest server is easily shown to be suboptimal, the result follows.

Proof of lemma 3: From lemma 1 we know that π will not activate server S_3 if it does not activate (at the same state) server S_2. From lemma 2, we should only consider states of the form $(y_0, 1, 0, 0)$. It is easy to show that there exists at least one state with $y_0 < \infty$, such that

$$\pi(y_0, 1, 0, 0)_2 = 1$$

The proof is essentially that of lemma 3.2b in [3], since by hypothesis if π does not utilize server S_2, it will never utilize server S_3 as well; in this case, the system behaves essentially as an $M|M|2$ system. Therefore, we will not repeat it here.

Suppose now that for some queue size y_0,we have that $\pi(y_0, 1, 0, 0)_2 = 1$ and $\pi(y_0 + 1, 1, 0, 0)_2 = 0$. Let τ_j be the first time that π activates server S_j, $j = 2, 3$, when starting from an initial state $(y_0 + 1, 1, 0, 0)$. From lemma 1, we have that $\tau_2 \leq \tau_3$ a.s.

Suppose that another policy $\tilde{\pi}$ activates S_2 at $t = 0$. If $\sigma_2 \leq \tau_2$, then $\tilde{\pi}$ improves π. If $\sigma_2 > \tau_2$, then the trajectories under $\pi, \tilde{\pi}$ are matched, since at $t = \tau_2$ we have $x_t = (y_0, 1, 1, 0)$ (or $(y_0 - 1, 1, 1, 1)$) and thus $\tilde{x}_t = x_t$. Moreover, under π, the queue size never reached 0 in $[0, \tau_2]$, qed.

Proof of lemma 4: The proof is analogous to that of lemma 3 and thus we shall only sketch it here. We have to show first that there exists a state $(y_0, 1, 0, 1)$ with $y_0 < \infty$, at

which the optimal policy utilizes server S_2. We then show that π activates the server in state $(y_0+1,1,0,1)$ as well.

To show the first part of he claim, choose $y_0 > m_i$, where m_i is defined as in lemma 3 and let the system be at state $(y_0,1,0,1)$ at time $t=0$. Let σ_3 be the service time of the customer served by server S_3.

By hypothesis, π will never activate S_2 in $[0,\sigma_3)$; let another policy $\tilde{\pi}$ activate S_2 when starting from $(y_0,1,0,1)$. Let τ be the first time to reach queue size m_i from size y_0, under π. Clearly τ increases (stochastically) as y_0 increases.

For all sample paths with $\sigma_3 \leq \tau$, π will activate S_2, since a state $(y_0,1,0,0)$ will be reached at $t=\sigma_3$, with $y_0 \geq m_i$. For those sample paths, $\tilde{\pi}$ can only improve π. For all sample paths with $\sigma_3 > \tau$, π does not activate S_2, and $\tilde{\pi}$ may incur a higher cost. However, the probability of those paths can be made arbitrarily small, by increasing y_0.

To show the second part of the claim, we may proceed along the lines of the proof of lemma 3.2(3) in [3], with minor modifications.

Proof of lemma 5: As in the previous two lemmata, again the proof has two parts, although now it is more elaborate. We first have to show that π will activate S_3 at some state $(y_0,1,1,0)$, with $y_0 < \infty$. We then show that it does so at state $(y_0+1,1,1,0)$ as well.

The proof of the first part of the claim is quite similar to that of lemma 4. Let τ denote the first time to reach an empty queue, under π. We may similarly observe that for sample paths with $\sigma_3 \leq \tau$, activating S_3 is better than idling it. As y_0 increases, the probability of those paths increases.

For the second part, observe that since the optimal policy is ergodic (all thresholds are finite), the state $(0,0,0,0)$ will be eventually visited. From this state we 'll visit state $(y_0+1,1,0,0)$, and from lemma 1 we visit state $(y_0,1,1,0)$ before we visit a state with the slowest server busy. Thus, if at a state $(y_0,1,1,0)$ it is optimal to activate S_3, at state $(y_0+1,1,0,0)$ it is optimal to activate S_2. With this observation in mind, we may now proceed as in lemma 3.2(3) of [3].

Proof of lemma 6: Let $J(x)$ denote the optimal cost function (the minimum cost in

(1)), when the initial state is x. For any state $(y_0, 1, 0, 1)$ with $y_0 < m_b$, we have

$$J(y_0 - 1, 1, 1, 1) > J(y_0, 1, 0, 1)$$

since it is better to keep S_2 idle. From lemma 1 we have

$$J(y_0, 1, 0, 1) \geq J(y_0, 1, 1, 0)$$

and thus

$$J(y_0 - 1, 1, 1, 1) > J(y_0, 1, 1, 0)$$

which implies that $y_0 < m_3$. Thus $m_b \leq m_3$.

As already mentioned in the proof of lemma 5, we visit state $(y_0, 1, 1, 0)$ before we visit a state with the slowest server busy, hence $m_i \leq m_3$, qed.

References

[1] W. Lin–P. R. Kumar, "Optimal Control of a Queueing System with Two Heterogenous Servers," IEEE Transactions on Automatic Control, vol. AC–29, pp. 696–703, Aug. 1984.

[2] A. Agrawala et al, "A Stochastic Optimization Algorithm Minimizing Expected Flow Times on Uniform Processors," IEEE Transactions on Computers, vol. C–33, pp. 351–356, 1984.

[3] J. Walrand, "A Note on Optimal Control of a Queueing System with Two Heterogenous Servers," Systems and Control Letters, Vol. 4, pp. 131–134, 1984.

[4] R. Weber, "On the Optimal Assignment of Customers to Parallel Servers," Journal Appl. Prob., vol 15, pp. 406–413, 1978.

[5] S. A. Lippman, "Semi–Markov Decision Processes with Unbounded Rewards," Management Science, Vol. 19, pp. 717–731, 1973.

[6] I. Viniotis–A. Ephremides, "Optimal Switching of Voice and Data at a Network Node," Proceedings of the 26th CDC, pp. 1504–1507, Los Angeles, CA, December 1987.

[7] E. G. Koffman et al, "Minimizing Expected Makespans on Uniform Processor Systems," Adv. Applied Prob., 19, pp. 177–201, 1987.

[8] P. R. Kumar–J. Walrand, "Individually Optimal Routing in Parallel systems," J. Applied Probability, vol. 22, pp. 989–995, 1985.

[9] G. B. Nath–E. F. Enns, "Optimal Service Rates in a Multiserver Loss System with Heterogeneous Servers," J. Applied Probability, vol. 18, pp. 776–781, 1981.

$(\lambda, \mu_1, \mu_2, \mu_3)$	$(1, 1, 0.1, 0.01)$	m_i	m_b
	β =0.1	1	1
	0.5	2	2
	0.8	3	3
	0.99	3	3
	0.9999	3	3
$(\lambda, \mu_1, \mu_2, \mu_3)$	$(1, 5, 2, 0.5)$	m_i	m_b
	β =0.1	1	1
	0.5	1	1
	0.8	1	1
	0.9	2	2
	0.99	2	2
	0.99999	2	2
$(\lambda, \mu_1, \mu_2, \mu_3)$	$(1, 10, 1, 0.1)$	m_i	m_b
	β =0.1	1	1
	0.5	3	3
	0.8	4	4
	0.9	6	6
	0.99	8	8
	0.9999	9	9
	0.99999	9	9

Figure 2.