

The Optimal Control of a Two-Stage Tandem Queueing System with Flexible Servers

Hyun-soo Ahn

*Department of Industrial Engineering and Operations Research
University of California at Berkeley, 4185 Etcheverry Hall, Berkeley, CA 94720-1777
ahn@ieor.berkeley.edu (510) 642-2497*

Izak Duenyas

*Business School, Department of Operations Management
University of Michigan, 701 Tappan Street, Ann Arbor, MI 48109-1234
duenyas@umich.edu (734) 763-1431*

Mark E. Lewis

*Department of Industrial and Operations Engineering
University of Michigan, 1205 Beal Avenue, Ann Arbor, MI 48109-2117
melewis@engin.umich.edu (734) 763-0519*

February 7, 2002

Abstract

We consider the optimal control of two parallel servers in a two-stage tandem queueing system with two flexible servers. New jobs arrive to station 1 after which a series of two operations must be performed before they leave the system. Holding costs are incurred at rate h_1 per unit time for each job in station 1 and at rate h_2 per unit time for each job at station 2.

The system is considered under two scenarios; the collaborative case and the non-collaborative case. In the prior, the servers can collaborate to work on the same job, while in the latter each server can work on a unique job though they can work on separate jobs at the same station. We provide simple conditions under which it is optimal to allocate both servers to station 1 or 2 in the collaborative case. In the non-collaborative case, we show that the same condition as in the collaborative case guarantees the existence of an optimal policy that is exhaustive at station 1. However, the condition for exhaustive service in station 2 to be optimal does not carry over. This case is examined via a numerical study.

1 Introduction

The use of a cross-trained workforce has become prevalent in the manufacturing and service industries. Despite this fact, very little work has been done to examine how to optimally allocate such flexible workers or resources to tasks on hand. This is especially true for systems with more than one flexible server. Since the set of possible policies in these models can be intractable even in small examples, finding conditions under which the structure of optimal policies is simple can be crucial.

Consider a two-stage tandem queueing system with jobs arriving only to station 1 according to a Poisson process of rate $\lambda > 0$. Each arriving job must receive service at each station before leaving the system. However, the service may be performed by either of 2 identical servers. When either

server 1 or 2 processes jobs at station 1, the service time is exponential with rate μ_1 . Similarly, when either server processes jobs at station 2 the service time is exponential with rate μ_2 . Holding costs are incurred at rate h_1 per unit time for each job in queue 1 and at rate h_2 per unit time for each job in queue 2. We further assume that preemption is permitted, and that no setup is required for the server to switch from processing jobs at one station to another. The problem then is how to allocate the servers over time to jobs in station 1 and 2. In this paper we analyze two versions of this problem. In the first version, called the *collaborative case*, the servers may collaborate to work on the same job at the same time, thereby doubling the service rate at that station, while in the second version, the *non-collaborative case*, there must be a job available for each server at a station to avoid idling a server. We present conditions in the collaborative case for the optimal policy to be exhaustive at station 1; that is, both servers process jobs at station 1 as long as there is a job to be served. By simply reversing an inequality, we also provide the condition for optimality of exhaustive service at station 2. Thus, in this case, we get a complete characterization of the optimal policy. The non-collaborative case is more subtle. We show that the same inequality provided in the collaborative case suffices to guarantee the existence of an optimal policy that is exhaustive in station 1. However, the reverse inequality no longer guarantees optimality of an exhaustive policy at station 2. Furthermore, since we discuss these issues in the context of long-run average optimality, we are also able to show how the stability region of the network can be relaxed with a flexible workforce in place.

Among the few works that have considered the optimal control of flexible servers, Duenyas et al. [4] and Iravani ([7], [8]), in parallel, consider a tandem queueing system with only one flexible server and characterize the optimal policy to be a monotone switching curve. The control of a flexible workforce in minimizing the cycle time of each job is discussed in Van Oyen et al. [17]. Andradottir et al. [2] consider the control of flexible servers to maximize throughput when the service rate is additive. Javidi et al. [9] consider two interconnected queues with identical machines and derive sufficient conditions under which the policy that prioritizes jobs in queue 1 minimizes the expected value of the first time that the system becomes empty (i.e., the dynamic version of makespan). None of these papers consider the average cost case with multiple flexible servers in a dynamic environment.

In parallel with the work described above, there has been another stream of research that analyzes the control of a flexible workforce in clearing system problems. In a clearing system, no external arrivals are permitted after time zero and the objective is to minimize the expected holding costs incurred until all jobs leave the system. Farrar [5] focuses on two-stage tandem queues with a fixed server at each queue and a single flexible server. He shows that the optimal allocation of the flexible server is transition monotone and can be characterized by a switching curve. Pandelis and Teneketzis [15] characterize the sufficient condition for the optimality of an exhaustive policy in the upstream queue in a two-stage interconnected queue. Essentially, the problem considered in Ahn et al. [1] is a clearing system version of our problem. They present necessary and sufficient conditions under which an exhaustive policy at the upstream (or downstream) queue minimizes the expected value of weighted holding costs. Although the model is similar, we focus on the case when arrivals are introduced. The reader is referred to this paper and the references therein for a more detailed literature review.

The rest of this paper is organized as follows. Section 2 provides a Markov decision process formulation of the problem under the average cost optimality criterion. We also present results on stability for both cases and show the existence of a solution to the average cost optimality equations. Section 3 contains our main results: we characterize the structure of the optimal policy under the collaborative system and give sufficient conditions under which an exhaustive policy at the upstream queue is optimal in the non-collaborative system. In Section 4, we present several examples that display the benefits of a flexible workforce.

2 Preliminaries

Throughout the rest of the paper (unless otherwise stated) we assume that *uniformization* has been applied in the spirit of Lippman [11] with uniformization constant $\Psi \equiv \lambda + 2 \max\{\mu_1, \mu_2\}$. Without loss of generality we assume $\Psi = 1$. Thus, instead of considering the continuous-time problem defined above, we consider a discrete-time equivalent. Furthermore, the arrival and service rates can be interpreted as probabilities. A non-anticipating policy π describes where each server should serve next, given the current time and the number of customers in each queue. Thus, a policy π is a sequence of decisions $\pi = \{d_1, d_2, \dots\}$, where d_k is a vector of actions to be taken depending on the current state. Let Π be the set of all such policies. The objective is to find a policy that minimizes the average cost per unit time over an infinite horizon. We next formally define the long-run average cost.

Let X_n be the state of the system at time n and d_n the decision rule at time n under a particular policy π . The n -stage expected cost is

$$V_n(\pi, x) \equiv \mathbb{E}_x^\pi \left[\sum_{k=0}^{n-1} c(X_k, d_k(X_k)) \right].$$

The optimal n -stage expected cost is

$$V_n^*(x) = \inf_{\pi \in \Pi} V_n(\pi, x). \quad (2.1)$$

Definition 2.1 *The long-run average reward or gain of a policy π , given that the initial state of the system started is x , denoted $g(\pi, x)$ is given by*

$$g(\pi, x) = \limsup_{n \rightarrow \infty} V_n(\pi, x)/n. \quad (2.2)$$

Furthermore, let the **optimal expected average cost** be denoted $g^*(x)$

$$g^*(x) = \inf_{\pi \in \Pi} g(\pi, x).$$

A policy π^* is called **long-run average or gain optimal** if $g(\pi^*, x) = g^*(x)$ for all $x \in \mathbb{X}$, where \mathbb{X} denotes the state space.

We formulate the control problem as a Markov decision process. The set of decision epochs corresponds to the set of all arrival, service completions, and dummy transitions due to uniformization. The preemptive discipline restricts the optimal policy to be non-idling since any unforced idling is suboptimal.

The state space is denoted by $\mathbb{X} = \{(i, j) \mid i, j \in \mathbb{N}\}$ where \mathbb{N} is the set of non-negative integers; the first element is the number of jobs at station 1 and the second is the number of jobs at station 2. The *average cost optimality equations* (ACOE) for the collaborative system (cf. Sennott [16, Chapter 7]) are for $i, j \geq 1$

$$g + v(i, j) = \min_{s \in \{0, 1, 2\}} [h_1 i + h_2 j + \lambda v(i+1, j) + s\mu_1 v(i-1, j+1) + (2-s)\mu_2 v(i, j-1) + (1-\lambda - s\mu_1 - (2-s)\mu_2)v(i, j)], \quad (2.3)$$

where action s corresponds to how many servers are processing jobs at station 1. Since idling is sub-optimal, the ACOE for the remaining states is

$$g + v(i, j) = \begin{cases} h_2 j + \lambda v(1, j) + 2\mu_2 v(0, j-1) + (1-\lambda - 2\mu_2)v(0, j) & \text{for } i = 0 \text{ and } j \geq 1, \\ h_1 i + \lambda v(i+1, 0) + 2\mu_1 v(i-1, 1) + (1-\lambda - 2\mu_1)v(i, 0) & \text{for } i \geq 1 \text{ and } j = 0, \\ \lambda v(1, 0) + (1-\lambda)v(0, 0) & \text{for } i = 0 \text{ and } j = 0. \end{cases} \quad (2.4)$$

In the non-collaborative case, the ACOE must consider the number of servers that can be assigned to each queue. For example, in state $(1, 1)$ the servers must be assigned to different stations. Thus, let $\mathcal{A} = \{s \in \{0, 1, 2\} \mid \max\{2 - j, 0\} \leq s \leq \min\{i, 2\}\}$ represent the allowable actions in each state. The ACOE for the non-collaborative case are given by

$$g + v(i, j) = \min_{s \in \mathcal{A}} [h_1 i + h_2 j + \lambda v(i + 1, j) + s \mu_1 v(i - 1, j + 1) + (2 - s) \mu_2 v(i, j - 1) + (1 - \lambda - s \mu_1 - (2 - s) \mu_2) v(i, j)], \quad (2.5)$$

and

$$g + v(i, j) = \begin{cases} h_1 i + \lambda v(i + 1, 0) + (\mathbf{1}_{\{i \geq 1\}} + \mathbf{1}_{\{i \geq 2\}}) \mu_1 v(i - 1, 1) \\ \quad + (1 - \lambda - (\mathbf{1}_{\{i \geq 1\}} + \mathbf{1}_{\{i \geq 2\}}) \mu_1) v(i, 0) & \text{for } i \geq 0 \text{ and } j = 0, \\ h_2 j + \lambda v(1, j) + (\mathbf{1}_{\{j \geq 1\}} + \mathbf{1}_{\{j \geq 2\}}) \mu_2 v(0, j - 1) \\ \quad + (1 - \lambda - (\mathbf{1}_{\{j \geq 1\}} + \mathbf{1}_{\{j \geq 2\}}) \mu_2) v(0, j) & \text{for } i = 0 \text{ and } j \geq 0, \end{cases} \quad (2.6)$$

where $\mathbf{1}_A$ is the indicator function of the set A .

When a solution (g, v) to the ACOE exists, $g = g^*$ and v is called a *relative value function*. Furthermore, given that (g^*, v) is a solution to the ACOE, a decision rule d^* that attains the minimum (componentwise) of the ACOE generates a stationary, deterministic policy $\pi^* = \{d^*, d^*, \dots\}$ such that π^* is gain optimal.

We next derive a condition under which there exists a solution to the ACOE. We begin by giving a stability condition for non-idling policies. Note that this condition is similar to that derived independently by Andradottir et al. [3] using alternative methods, but our focus is on showing the existence of a solution to the ACOE rather than deriving general stability conditions for this network.

First note that under any non-idling policy there is but one closed, communicating class. For example, consider the policy that allocates both servers to station 2 whenever there is work to be done at station 2. Assume that the initial state is $(0, 0)$ (i.e., no jobs in the system). Then in the non-collaborative case, there can never be more than 2 jobs at station 2, while in the collaborative case, there can never be more than 1. States with more than two customers at station 2 are not part of the communicating class and are necessarily transient.

Proposition 2.2 *Under any stationary, non-idling policy, the system is stable if and only if $\lambda(1/\mu_1 + 1/\mu_2) < 2$. That is, there exists a stationary distribution.*

Proof. To prove sufficiency we fix an arbitrary, stationary, non-idling policy π , find a *Lyapunov* function and apply *Foster's criterion* (cf. [10, Theorem 3.7]). This guarantees that all recurrent states are positive recurrent. To this end, consider the Markov chain $\{X_n, n \geq 0\}$ generated by π restricted only to those states that belong to its communicating class. Denote this chain by $\{Z_n, n \geq 0\}$, its state space by S and note that S is irreducible. Let $G = \{(i, j) \mid i \leq 1, j \leq 1\}$. Let $y(i, j)$ be the *workload process* in state (i, j) . That is,

$$y(i, j) \equiv i/\mu_1 + (i + j)/\mu_2. \quad (2.7)$$

Now note that for any action chosen and $(i, j) \notin G$

$$\begin{aligned} \mathbb{E}(y(Z_{n+1}) - y(Z_n) \mid Z_n = (i, j)) &= \lambda \left(\frac{1}{\mu_1} + \frac{1}{\mu_2} \right) - \mu_1 \left(\frac{1}{\mu_1} \right) - \mu_2 \left(\frac{1}{\mu_2} \right) \\ &= \lambda \left(\frac{1}{\mu_1} + \frac{1}{\mu_2} \right) - 2. \end{aligned} \quad (2.8)$$

Since G is finite we may now apply [10, Theorem 3.7] to $\{Z_n\}$ to get that all states in S are positive recurrent when the right hand side of (2.8) is strictly negative. Furthermore, since (2.8) holds on all states of the original chain, applying Proposition C.1.5 of Sennott [16] yields that the expected time to reach the communicating class is finite. Thus, the result holds for $\{X_n, n \geq 0\}$ and sufficiency is proven.

To show the necessity of the inequality, we note that Theorem 11.5.1 of Meyn and Tweedie [14] implies that when $\lambda(\frac{1}{\mu_1} + \frac{1}{\mu_2}) - 2 \geq 0$ the expected time to reach the set $G = \{(i, j) | i, j \leq 1\}$ from outside of this set is infinite, and thus a stationary distribution cannot exist. ■

The next result states that when the system is stable the ACOE have a solution. We find useful the fact that for a fixed policy π , if there exists a function $V : \mathbb{X} \rightarrow \mathbb{R}_+$ and a constant J such that

$$P_\pi V(x) \equiv \mathbb{E}_\pi[V(X_{n+1}) | X_n = x] \leq V(x) - c(x, d(x)) + J, \quad x \in \mathbb{X}, \quad (2.9)$$

where $\{X_n, n \geq 0\}$ is the Markov chain generated by π , then the average cost of π is finite (cf. Meyn [13, Theorem 3]).

Proposition 2.3 *If $\lambda(1/\mu_1 + 1/\mu_2) < 2$, there exists a solution (g, v) of the average cost optimality equations defined in (2.3), (2.4), (2.5) and (2.6).*

Proof. We prove the result for the collaborative case since the non-collaborative case is similar. Consider the policy that is exhaustive in queue 1. Let

$$V(i, j) = \frac{h_1}{2} \left(\frac{i^2}{2\mu_1 - \lambda} \right) + \frac{h_2}{2} \left(\frac{2\mu_1 i}{2\mu_1 - \lambda} + 2j \right) \frac{i}{2\mu_1 - \lambda} + \frac{\frac{h_2}{2} \left(\frac{2\mu_1 i}{2\mu_1 - \lambda} + j \right)^2}{2\mu_2 \left(1 - \frac{\lambda}{2\mu_1} \right) - \lambda}.$$

Intuitively, this is the total cost to empty the system starting in state (i, j) when the arrival and service processes are replaced by deterministic continuous flows with rates λ , μ_1 and μ_2 , respectively; a *fluid model*. A little algebra yields,

$$\mathbb{E}_\pi[V(X_{n+1}) - V(x) | X_n = x] = \begin{cases} -ih_1 - jh_2 + C_1 & \text{if } i \geq 2 \\ -ih_1 - jh_2 + C_2 & \text{if } i = 0 \text{ and } j \geq 2. \\ -ih_1 - jh_2 + C_3 & \text{if } i = 1 \text{ and } j \geq 2. \end{cases} \quad (2.10)$$

where C_1 , C_2 , and C_3 are finite constants. Let $C = \max\{C_1, C_2, C_3\}$. One may note that there are four states that are not covered by the cases in (2.10) (these are $(0,0)$, $(0,1)$, $(1,0)$, $(1,1)$). Except in those states, $\mathbb{E}_\pi[V(X_{n+1}) - V(x) | X_n = x] + c(x, d(x))$ is bounded above by C but since those states are finite, we can clearly choose a value of J large enough to satisfy (2.9) in all cases. Applying (2.9) yields that the cost is finite.

The result is now a direct application of Theorems 3.1, 5.1, and 5.2 of Meyn [12]. Since we have that the exhaustive policy in queue 1 has finite average cost, if we initialize policy iteration by this policy, the results of Meyn [12] state that policy iteration is guaranteed to converge to a solution to the ACOE. ■

The importance of the results of Propositions 2.2 and 2.3 are twofold. When the two servers are fixed, it is easy to establish that stability of the Markov chain requires $\lambda/\mu_1 < 1$ and $\lambda/\mu_2 < 1$. Proposition 2.2 shows that in fact the stability region is extended considerably by the use of a flexible workforce. Figure 1 illustrates this fact. Proposition 2.3 allows us to use the ACOE to find the structure of optimal policies and eliminate from consideration (in many states) the action that assigns one server to each station (see Proposition 3.1 below).

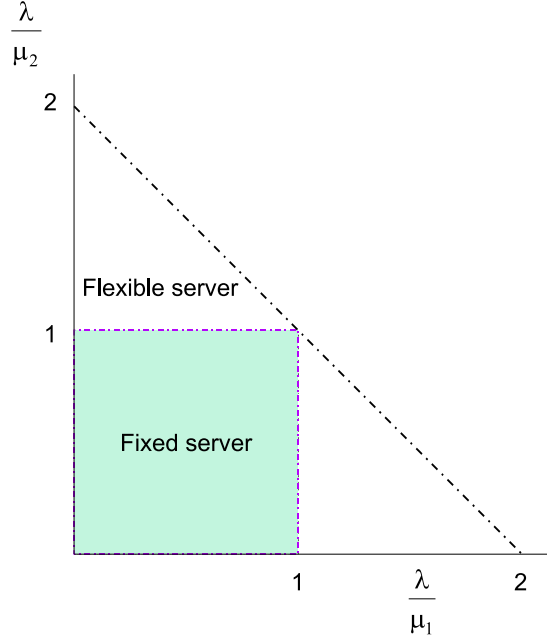


Figure 1: Stability condition in the system with fixed servers and the system with flexible servers.

3 Average Cost Optimal Policies

In this section we discuss the structure of an optimal policy and give conditions under which the policy is quite simple. To do so, consider the inequality

$$\mu_1(h_1 - h_2) \geq \mu_2 h_2. \quad (3.1)$$

Recall that we have discretized the model so that the holding cost rates can be interpreted as lump sum costs. Assume that there are customers to be served at each station and suppose we decide to assign server 1 to station 1. The left hand side of (3.1) is then the rate of *reduction* in the holding cost due to making this assignment. Similarly when we assign server 1 to station 2, the right hand side of (3.1) denotes the rate of reduction in the holding cost. Note that this intuition is independent of the number of customers currently at each station so long as there are customers available to serve. Furthermore, it may be applied to either server (since they are identical). Analogous to *steepest descent* algorithms in deterministic optimization settings, based on intuition, one might conjecture that we should choose to assign the servers so as to maximize the rate of reducing the holding costs. This leads to a policy that is exhaustive in one station or the other depending on the direction of the inequality (3.1). In the collaborative case, our intuition holds and this is in fact optimal. In the non-collaborative system the intuition fails, but only when the inequality in (3.1) is reversed. In fact, the analysis of the non-collaborative system can lead to quite unintuitive optimal policies. This is illustrated by the policy depicted in Figure 2. In the figure note that for a given number of jobs in station 2, the optimal policy assigns **fewer** servers to station 1 as the workload at station 1 increases.

We first show that in states when all three actions are feasible (both servers work at one of the stations, or each work at a separate station), we need not consider the action to assign one server to each station. Let $H_C \equiv \{(i, j) \in \mathbb{X} \mid i \geq 1 \text{ or } j \geq 1\}$ and $H_N \equiv \{(i, j) \in \mathbb{X} \mid i \geq 2 \text{ and } j \geq$

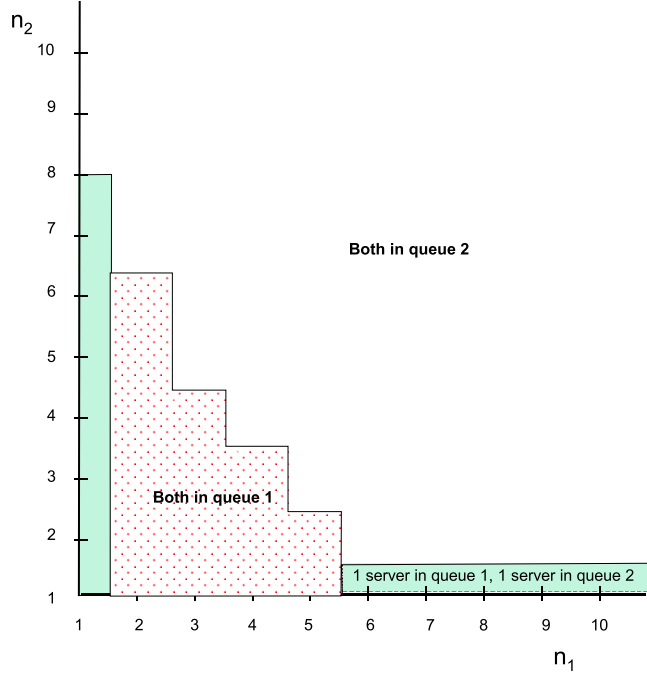


Figure 2: An example where the optimal policy is unintuitive: $\lambda = .2$, $\mu_1 = 0.4$, $\mu_2 = 0.4$, $h_1 = 1.999$ and $h_2 = 1$

2}, where $H_C(H_N)$ denotes states where all three actions are feasible for the collaborative (non-collaborative) system.

Proposition 3.1 *For $x \in H_C(H_N)$ in the collaborative (non-collaborative) system there exists an optimal policy that assigns both servers to a single station.*

Proof. In the collaborative and non-collaborative case, the minimum in the optimality equations (2.3), (2.4), (2.5) and (2.6) for the sets H_C and H_N is

$$M = \min_{s \in \{0,1,2\}} [s\mu_1 v(i-1, j+1) + (2-s)\mu_2 v(i, j-1) + (1-\lambda-s\mu_1-(2-s)\mu_2)v(i, j)].$$

A little algebra yields

$$M = 2v(i, j-1) + (1-\lambda-2\mu_2)v(i, j) + \min_{s \in \{0,1,2\}} [s\{\mu_1(v(i-1, j+1) - v(i, j)) - \mu_2(v(i, j-1) - v(i, j))\}] \quad (3.2)$$

Since the term that is to be minimized in (3.2) is linear in s the minimum must occur at the extreme points (i.e., $s = 0$ or 2). ■

The above result can be easily extended to the general case of an m -station tandem system with n servers. In this case, collaboration is optimal once again in all states in the collaborative system and assigning all servers to a single station is optimal in all states where each station has at least n jobs in the non-collaborative system. Since the proof of this result is similar to that of Proposition 4.1, we omit the details.

3.1 The Collaborative Case

In this section we characterize an optimal policy for the collaborative case. Applying Proposition 3.1 we know that we need not consider policies that assign a server to each station. The question then becomes given the number of customers in each queue where should one assign both servers? We show that in the collaborative case, the answer to this question is quite simple and easy to implement. This section is dedicated to proving the following result.

Theorem 3.2 *In the collaborative system, suppose $\lambda(\frac{1}{\mu_1} + \frac{1}{\mu_2}) < 2$ and $\mu_1(h_1 - h_2) \geq (\leq) \mu_2 h_2$. Then there exists an optimal policy with finite gain that is exhaustive in queue 1 (2).*

In order to prove Theorem 3.2 we show that the result actually holds for the n -stage total expected cost problem defined by (2.1) **for all** n . We refer to this as the *finite horizon collaborative case*. Let π^* be the exhaustive policy at queue 1. Under the assumption that $\mu_1(h_1 - h_2) \geq \mu_2 h_2$ we then have

$$V_n(\pi^*, x) \leq V_n(\pi, x) \quad (3.3)$$

for all $\pi \in \Pi$ and for all n . Dividing both sides of (3.3) by n and taking limits as n approaches infinity the average cost result follows. The finiteness of the gain follows from the results of Section 2.

To this end, it is well-known that the optimal n -stage cost satisfies the finite horizon optimality equations (FHOE). Set $v_0(i, j) = 0$ and the n -stage optimality equation for states $(i \geq 1, j \geq 1)$ is written as

$$\begin{aligned} v_n(i, j) = & h_1 i + h_2 j + \lambda v_{n-1}(i+1, j) \\ & + \min_{s \in \{0, 1, 2\}} [s \mu_1 v_{n-1}(i-1, j+1) + (2-s) \mu_2 v_{n-1}(i, j-1) + (1-\lambda-s\mu_1-(2-s)\mu_2) v_{n-1}(i, j)] \end{aligned}$$

for $i \geq 1$ and $j \geq 1$. (3.4)

It should be clear that the results of Proposition 3.1 extend to the finite horizon collaborative case. That is, we need not consider policies that assign one server to each station. Thus, from the FHOE it is optimal in the finite horizon problem to allocate both servers to station 1 (2) when $\mu_1[v_n(i-1, j+1) - v_n(i, j)] \leq (\geq) \mu_2[v_n(i, j-1) - v_n(i, j)]$. The next lemma proves that this inequality holds for all n , under the same assumptions as Theorem 3.2.

Lemma 3.3 *In the finite horizon collaborative case, suppose $\mu_1(h_1 - h_2) \geq (\leq) \mu_2 h_2$. Then $\Delta_n(i, j) \equiv \mu_1[v_n(i-1, j+1) - v_n(i, j)] - \mu_2[v_n(i, j-1) - v_n(i, j)] \leq (\geq) 0$ for $i, j \geq 1$.*

Proof. Assume $\mu_1(h_1 - h_2) \geq \mu_2 h_2$. The proof is by induction. For $n = 0$, the result holds trivially. Assume that it holds for $n - 1$. For n note that by the induction hypothesis, it is optimal for both servers to serve at station 1 whenever possible. Hence, for $i \geq 1$

$$v_n(i, j) = h_1 i + h_2 j + \lambda v_{n-1}(i+1, j) + 2\mu_1 v_{n-1}(i-1, j+1) + (1-\lambda-2\mu_1) v_{n-1}(i, j)$$

Consider the case when $i \geq 2$ and $j \geq 1$. A little algebra yields,

$$\begin{aligned} \mu_1[v_n(i-1, j+1) - v_n(i, j)] &= \mu_1\{(h_2 - h_1) + \lambda[v_{n-1}(i, j+1) - v_{n-1}(i+1, j)] \\ &\quad + 2\mu_1[v_{n-1}(i-2, j+2) - v_{n-1}(i-1, j+1)] \\ &\quad + (1-\lambda-2\mu_1)[v_{n-1}(i-1, j+1) - v_{n-1}(i, j)]\} \\ \mu_2[v_n(i, j-1) - v_n(i, j)] &= \mu_2\{-h_2 + \lambda[v_{n-1}(i+1, j-1) - v_{n-1}(i+1, j)] \\ &\quad + 2\mu_1[v_{n-1}(i-1, j) - v_{n-1}(i-1, j+1)] \\ &\quad + [1 - (\lambda + 2\mu_1)][v_{n-1}(i, j-1) - v_{n-1}(i, j)]\}. \end{aligned}$$

To examine $\Delta_n(i, j)$ we note that the difference in the expressions above can be analyzed term by term. The difference in the first terms is non-positive by the assumption that $\mu_1(h_1 - h_2) \geq \mu_2 h_2$. Each of the remaining differences is non-positive by the induction hypothesis. The algebra is similar when $i = 1$ and $j \geq 1$. Thus, the result is proven in one direction. Reversing the inequalities in the induction hypothesis yields the other direction. ■

Recall that it is optimal in the finite horizon collaborative case to allocate both servers to station 1 in state (i, j) in the n -stage problem when $\Delta_n(i, j) \leq 0$. Thus, Lemma 3.3 implies the following corollary directly.

Corollary 3.4 *Under the assumption that $\mu_1(h_1 - h_2) \geq (\leq) \mu_2 h_2$, there exists an optimal policy to the n -stage collaborative problem that is exhaustive in station 1 (2) for all n .*

Proof of Theorem 3.2: Let π^* be the non-idling policy that allocates both servers to station 1 whenever possible. Then by Corollary 3.4,

$$\frac{V_n(\pi^*, x)}{n} \leq \frac{V_n(\pi, x)}{n} \quad (3.5)$$

for all $\pi \in \Pi$. Taking the lim sup as n approaches infinity in (3.5) yields $g(\pi^*, x) \leq g(\pi, x)$ for all $\pi \in \Pi$, i.e., π^* is average cost optimal. The finiteness follows from the proof of Proposition 2.3. To show that under the assumption that $\mu_1(h_1 - h_2) \leq \mu_2 h_2$ we have existence of an optimal policy that is exhaustive in queue 2, we apply the same argument. Since the gain of the policy that is exhaustive in queue 1 is finite, the finiteness must also hold for the optimal policy. ■

3.2 The Non-collaborative Case

The non-collaborative case is more subtle than the collaborative case. The difficulty arises because we do not allow both servers to work at the same station when there is not a separate job for the server to handle. Where possible we have followed the same line of reasoning as the previous section for consistency.

Under the assumption that $\mu_1(h_1 - h_2) \geq \mu_2 h_2$ the results of the previous section continue to hold. However, as has been alluded to, when the inequality is reversed an exhaustive policy is no longer guaranteed to be optimal. The main result of this section is stated in the following theorem.

Theorem 3.5 *In the non-collaborative system, suppose $\lambda(\frac{1}{\mu_1} + \frac{1}{\mu_2}) < 2$ and $\mu_1(h_1 - h_2) \geq \mu_2 h_2$. Then there exists an optimal policy that is exhaustive in queue 1.*

Thus, we have sufficient conditions for the optimality of an exhaustive policy in queue 1 in the non-collaborative case that coincide with the collaborative case. This partially confirms a conjecture made in [1]. In Section 4 we discuss further the case when $\mu_1(h_1 - h_2) < \mu_2 h_2$.

Consider the finite horizon optimality equations for the non-collaborative case for $i, j \geq 1$.

$$\begin{aligned} v_n(i, j) = & h_1 i + h_2 j + \lambda v_{n-1}(i+1, j) \\ & + \min_{s \in \mathcal{A}} [s \mu_1 v_{n-1}(i-1, j+1) + (2-s) \mu_2 v_{n-1}(i, j-1) + (1-\lambda-s\mu_1-(2-s)\mu_2) v_{n-1}(i, j)], \end{aligned}$$

and $v_0(i, j) = 0$ for all $(i, j) \in \mathbb{X}$. The other states are defined in an analogous manner to the ACOE.

Just as in the collaborative case, when it is possible, the FHOE imply that it is optimal to assign both servers to station 1 (2) at stage $n+1$ when $\mu_1[v_n(i-1, j+1) - v_n(i, j)] - \mu_2[v_n(i, j-1) - v_n(i, j)] \leq (\geq) 0$. This leads to the following lemma.

Lemma 3.6 *In the finite horizon non-collaborative case, suppose $\mu_1(h_1 - h_2) \geq \mu_2 h_2$. For $n \geq 0$*

a. i. $v_n(i, j + 1) \geq v_n(i, j)$ for $i, j \geq 0$.

ii. $v_n(i + 1, j) \geq v_n(i, j + 1)$ for $i, j \geq 0$.

b. $\Delta_n(i, j) = \mu_1[v_n(i - 1, j + 1) - v_n(i, j)] - \mu_2[v_n(i, j - 1) - v_n(i, j)] \leq 0$ for $i, j \geq 1$.

Proof. By induction on n . For $n = 0$ the claim clearly holds. Assume it holds for $n - 1$. By the induction hypothesis (applied to Part b above) it is optimal to serve at station 1 at stage n whenever possible. For brevity, we will only show the case under which $i \geq 2$ although one needs to establish that the claim holds for all six cases - i) $i \geq 2$, ii) $i = 1, j \geq 1$, iii) $i = 1, j = 0$, iv) $i = 0, j \geq 2$, v) $i = 0, j = 1$ and vi) $i = 0, j = 0$. Note that

$$v_n(i, j) = h_1 i + h_2 j + \lambda v_{n-1}(i + 1, j) + 2\mu_1 v_{n-1}(i - 1, j + 1) + (1 - \lambda - 2\mu_1)v_{n-1}(i, j) \quad \text{if } i \geq 2.$$

A little algebra yields,

$$\begin{aligned} v_n(i, j) - v_n(i, j + 1) &= -h_2 + \lambda[v_{n-1}(i + 1, j) - v_{n-1}(i + 1, j + 1)] \\ &\quad + 2\mu_1[v_{n-1}(i - 1, j + 1) - v_{n-1}(i - 1, j + 2)] \\ &\quad + (1 - \lambda - 2\mu_1)[v_n(i, j) - v_n(i, j + 1)] \end{aligned} \quad \text{for } i \geq 2.$$

Applying the induction hypothesis for each of the terms above yields Part a. i. The other cases follow from similar algebra.

To show Part a. ii. note

$$\begin{aligned} v_n(i, j + 1) - v_n(i + 1, j) &= h_2 - h_1 + \lambda[v_{n-1}(i + 1, j + 1) - v_{n-1}(i + 2, j)] \\ &\quad + 2\mu_1[v_{n-1}(i - 1, j + 2) - v_{n-1}(i, j + 1)] \\ &\quad + (1 - \lambda - 2\mu_1)[v_{n-1}(i, j + 1) - v_{n-1}(i + 1, j)] \end{aligned} \quad \text{for } i \geq 2.$$

Applying similar algebra, one can show that the claim holds upon applying the induction hypothesis, the assumption that $h_2 - h_1 \leq 0$, and the previously proven Part a. i. for all remaining cases. Therefore, claim a.ii. holds for all n as desired.

It remains to show that Part b. holds for n . As in the previous two parts there are several cases to consider. However, the analysis requires a bit more detail. In the interest of brevity, we discuss only the state $(1, 1)$. The others follow in precisely the same manner. We must show

$$\Delta_n(1, 1) = \mu_1[v_n(0, 2) - v_n(1, 1)] - \mu_2[v_n(1, 0) - v_n(1, 1)] \leq 0. \quad (3.6)$$

We have

$$\begin{aligned} \mu_1[v_n(0, 2) - v_n(1, 1)] &= \mu_1\{(h_2 - h_1) + \lambda[v_{n-1}(1, 2) - v_{n-1}(2, 1)] + \mu_2[v_{n-1}(0, 1) - v_{n-1}(0, 2)] \\ &\quad + [1 - (\lambda + \mu_1 + \mu_2)][v_{n-1}(0, 2) - v_{n-1}(1, 1)] \\ &\quad + \mu_2[v_{n-1}(0, 1) - v_{n-1}(1, 0)]\}, \end{aligned} \quad (3.7)$$

and

$$\begin{aligned} \mu_2[v_{n-1}(1, 0) - v_{n-1}(1, 1)] &= \mu_2\{-h_2 + \lambda[v_{n-1}(2, 0) - v_{n-1}(2, 1)] + \mu_1[v_{n-1}(0, 1) - v_{n-1}(0, 2)] \\ &\quad + [1 - (\lambda + \mu_1 + \mu_2)][v_{n-1}(1, 0) - v_{n-1}(1, 1)]\}. \end{aligned} \quad (3.8)$$

From the assumption that $\mu_1(h_1 - h_2) \geq \mu_2 h_2$ and the induction hypothesis at state (2,1), the difference in each of the first 3 terms is non-positive (the second term cancels). The last term in (3.8) is non-positive by Part a. ii. Thus, each of the claims holds for n and the result is proven. ■

The next corollary is analogous to Corollary 3.4 and follows directly from Lemma 3.6, Part b.

Corollary 3.7 *Under the assumption that $\mu_1(h_1 - h_2) \geq \mu_2 h_2$, there exists an optimal policy in the n -stage non-collaborative problem that is exhaustive in station 1 for all n .*

Theorem 3.5 now follows just as Theorem 3.2. Since the proof is the same, it is omitted.

4 Discussion and Extension

In the previous section, in the collaborative case, we proved the optimality of an exhaustive policy in either station 1 or station 2 depending on the direction of the inequality (3.1). Unfortunately, the same simple policy is not optimal for the non-collaborative case. While, the optimal policy is still exhaustive under the inequality as it appears in (3.1), when the inequality is reversed the optimal policy in the non-collaborative case may be quite different. This was demonstrated in Figure 2. On the other hand, our numerical study of the non-collaborative system supports the conjecture made in Ahn et al. [1] that a sufficient condition for the optimality of an exhaustive policy in queue 2 is $h_1 \geq (1 + \frac{\mu_2}{\mu_1 + \mu_2})h_2$. Among the examples tested, no case has been discovered to contradict this conjecture. However, we have not been able to prove this conjecture to date. Furthermore upon rewriting the reverse inequality in (3.1) as $h_1 \geq (1 + \frac{\mu_2}{\mu_1})h_2$ it is interesting to note that in each example tested when $(1 + \frac{\mu_2}{\mu_1 + \mu_2})h_2 < h_1 < (1 + \frac{\mu_2}{\mu_1})h_2$, a switching curve policy was found to be optimal. That is to say that once an optimal policy assigns both servers to station 2 in state (i, j) , it will assign both to station 2 for all states (i', j') such that $i \leq i'$ and $j \leq j'$.

We have already seen in Section 2 that a flexible workforce allows for the stability region to be expanded considerably. One may wonder how important a flexible workforce can be to the overall average cost, as well as how important using an "optimal" allocation policy is to the effective use of this flexibility. We can explore the value of such flexibility by comparing the optimal policy to two candidate policies. The first is the obvious basis for comparison; where each server remains fixed at one station (i.e., no flexibility at all). We call this the *fixed tandem or static assignment* policy. A second policy, variations of which have been recently analyzed, is called a *push/pull* policy. In the push/pull policy, one server is assigned to each station until such time that there is no work available at its respective station. It then moves to the other queue to aid the other server at its station. In other words, when there is no work to be done at station 2, both servers work to "pull" work to the second station. Similarly, when there is no work at station 1, both servers work to "push" work out of station 2. Grassmann and Tavakoli [6] study a pull policy in which there is a finite buffer at the second queue while Andradottir et al. [2] discuss both a push and pull policy (separately) and show that they perform well when average throughput is the objective. Thus, comparing the optimal policy to the static one shows the value of flexibility while a comparison to the "push-pull" policy gives us a sense of the value of the "optimal" use of this flexibility. In the examples tested here, the static policy performed very poorly when compared to the optimal policy. The push/pull policy performed better than the static policy, but in some instances also performed poorly. Thus, investments in flexibility should be coordinated with sequencing policies that will optimally make use of such flexibility.

We created three sets of examples to test the optimal policy against these two candidate policies. In the first set, (shown in Table 1), we fixed $(1 + \frac{\mu_2}{\mu_1 + \mu_2})h_2 < h_1 < (1 + \frac{\mu_2}{\mu_1})h_2$. Thus, in these

λ	μ_1	μ_2	h_2	$C1$	$C2$	h_1	Optimal Cost	Fixed Tandem (%)	Push/Pull (%)
0.2	0.4	0.4	1	1.500	2.000	1.600	1.708	2.600 (52.18)	1.728 (1.17)
0.2	0.4	0.4	1	1.500	2.000	1.750	1.818	2.750 (51.30)	1.829 (0.58)
0.2	0.4	0.4	1	1.500	2.000	1.900	1.923	2.900 (50.80)	1.929 (0.32)
0.2	0.4	0.4	1	1.500	2.000	1.975	1.973	2.975 (50.77)	1.979 (0.31)
0.2	0.4	0.3	1	1.429	1.750	1.493	1.923	3.493 (81.63)	2.144 (11.47)
0.2	0.4	0.3	1	1.429	1.750	1.589	2.190	3.589 (63.88)	2.214 (1.11)
0.2	0.4	0.3	1	1.429	1.750	1.686	2.275	3.686 (62.03)	2.285 (0.45)
0.2	0.4	0.3	1	1.429	1.750	1.734	2.315	3.734 (61.32)	2.321 (0.24)
0.2	0.3	0.4	1	1.571	2.333	1.724	2.443	4.448 (82.06)	2.477 (1.37)
0.2	0.3	0.4	1	1.571	2.333	1.952	2.695	4.905 (81.97)	2.714 (0.71)
0.2	0.3	0.4	1	1.571	2.333	2.181	2.939	5.362 (82.41)	2.952 (0.43)
0.2	0.3	0.4	1	1.571	2.333	2.295	3.055	5.590 (83.02)	3.070 (0.50)
0.2	0.2	0.4	1	1.667	3.000	1.933	5.344	∞ (N/A)	5.406 (1.16)
0.2	0.2	0.4	1	1.667	3.000	2.333	6.337	∞ (N/A)	6.375 (0.60)
0.2	0.2	0.4	1	1.667	3.000	2.733	7.309	∞ (N/A)	7.344 (0.48)
0.2	0.2	0.4	1	1.667	3.000	2.933	7.779	∞ (N/A)	7.829 (0.64)
0.2	0.4	0.2	1	1.333	1.500	1.367	3.746	∞ (N/A)	3.881 (3.61)
0.2	0.4	0.2	1	1.333	1.500	1.417	3.841	∞ (N/A)	3.924 (2.16)
0.2	0.4	0.2	1	1.333	1.500	1.467	3.934	∞ (N/A)	3.967 (0.83)
0.2	0.4	0.2	1	1.333	1.500	1.492	3.979	∞ (N/A)	3.988 (0.23)

Table 1: Performance of Optimal and Two Candidate Policies When $\left(1 + \frac{\mu_2}{\mu_1 + \mu_2}\right) h_2 < h_1 < \left(1 + \frac{\mu_2}{\mu_1}\right) h_2$ The optimal policy is a switching curve policy.

examples, the optimal policy was a switching curve policy in each case. For each combination of λ , μ_1 , μ_2 , h_2 , we chose values of h_1 that were 20%, 50%, 80% and 95% of the spread between $\left(1 + \frac{\mu_2}{\mu_1 + \mu_2}\right) h_2$ and $\left(1 + \frac{\mu_2}{\mu_1}\right) h_2$. In the table, we report the optimal cost as well as the cost achieved by the two polices (and the percentage difference from the optimal cost in parantheses). The benefit of a flexible workforce is substantial when compared to the fixed server system. In fact, the average sub-optimality of the fixed server system is greater than 60% in the tested examples, and this is an average that excludes those instances where the stability condition is violated for the fixed tandem case resulting in an average cost of infinity (as expected). By comparison the average sub-optimality of the push/pull policy is approximately 1.4%. The highest sub-optimality for the push/pull was 11.47%. These results are detailed in Table 1.

We next considered examples when $\left(1 + \frac{\mu_2}{\mu_1}\right) h_2 \leq h_1$. As was shown in Theorem 3.5 an exhaustive policy in queue 1 is optimal. Values of h_1 that are 100%, 150% and 200% of $\left(1 + \frac{\mu_2}{\mu_1}\right) h_2$ were considered. Once again, the system with flexible servers considerably outperforms the fixed assignment server system (on average approximately 64.9%). In this case the push/pull policy under-performed on average by approximately 7.8%. Furthermore, there were several cases larger than 10% with one case reaching 27.97%. Finally, we considered $\left(1 + \frac{\mu_2}{\mu_1 + \mu_2}\right) h_2 \geq h_1$. This is the region where Ahn et al. [1] conjectured, the optimal policy serves queue 2 with both servers exhaustively. Values of h_1 were chosen that were 50%, 75 %, 100 % of $\left(1 + \frac{\mu_2}{\mu_1 + \mu_2}\right) h_2$. In every example tested, the optimal policy is in fact exhaustive in queue 2. The average sub-optimality of the fixed tandem policy is approximately 73.1% while that of the push/pull policy is approximately 10.73% with the worst resulting in a significant sub-optimality of 36.26%. We have shown that the optimal policy in the collaborative case is quite simple and easy to implement. We were unable to completely characterize an optimal policy in the non-collaborative case, however one direction is

complete, while the other direction is supported by the numerical study. Furthermore, this study suggests that the use of the flexible workforce can lead to significant improvements in the average cost if the flexible workforce is in fact allocated to tasks optimally. Even when the policy is simplified in the form of a push/pull policy, a workforce that has the freedom to work at either station can lead to significant improvements in certain cases but not in all. We conjecture that a similar phenomenon will hold when the two-server model in this paper is extended to several servers but further research is required to characterize optimal policies in that more complicated case.

5 Acknowledgements

The authors would like to thank Sheldon Ross of the University of California, Berkeley, Shane G. Henderson of Cornell University and Sean P. Meyn of the University of Illinois at Urbana-Champaign for their helpful comments on earlier versions of this paper.

References

- [1] Ahn, H-S, Duenyas, I. and Zhang, R. (1999), “Optimal Stochastic Scheduling of a Two-Stage Tandem Queue With Parallel Servers”, *Advances in Applied Probability*, **31**, 1095-1117.
- [2] Andradottir, S., Ayhan, H. and D.G. Down, (2001) “ Server Assignment Policies for Maximizing the Steady-State Throughput of Finite Queueing Aystems”, To appear in *Management Science*.
- [3] Andradottir, S., Ayhan, H. and D.G. Down, (2001) “Dynamic Server Allocation for Queueing Networks with Flexible Servers”, *preprint*.
- [4] Duenyas, I., Gupta, D. and Olsen, T. (1998), “Control of a Single Server Tandem Queueing System with Setups,” Technical Report No. 95-19, University of Michigan, to appear in *Operations Research*.
- [5] Farrar, T. (1993), “Optimal Use of an Extra Server in a Two Station Tandem Queueing Network,” (1993), *IEEE Transactions on Automatic Control*, **38**, 1296-1299.
- [6] Grassmann, W. and Tavakoli, J. (2001), “A Tandem Queue with a Movable Server: an Eigenvalue Approach,” *preprint*.
- [7] Iravani, S.M.R., Posner, M.J.M., and J.A. Buzacott, (1997), “Two-Stage Tandem Queue Attended by a Moving Server with Holding and Switching costs; static and semi-dynamic policy,” *Queueing Systems***26** 203-228.
- [8] Iravani, S.M.R. and J.A. Buzacott, (1996), “An N-stage Tandem Queueing System Attended by a Moving Server with Holding and Switching costs,” Technical Report., 96-09, University of Toronto, Department of Industrial Engineering, Toronto, Canada.
- [9] Javidi, T., Song, N. and Teneketzis, D. (2001), “Expected Makespan Minimization of Identical Machines in Two Interconnected Queues” To appear in *Probability in Engineering and Informational Science*
- [10] Kulkarni, Vidyadhar G., (1995), *Modeling and Analysis of Stochastic Systems*, Chapman and Hall, London.

- [11] Lippman, S., (1975), “Applying a New Device in the Optimization of Exponential Queueing System”, *Operations Research*, **23**, 687-710.
- [12] Meyn, S. (1997), “The Policy Improvement Algorithm for Markov Decision Processes”, *IEEE Transactions on Automatic Control*, **42:12**, 1663 -1680.
- [13] Meyn, S. (2001), Stability, Performance Evaluation, and Optimization. In Eugene Feinberg and Adam Schwartz, editors, *Handbook of Markov Decision Processes: Methods and Applications*. Kluwer Academic Publishers, 2001. in press.
- [14] Meyn, S. and Tweedie, R. (1993). *Markov Chains and Stochastic Stability*, Springer-Verlag, New York.
- [15] Pandelis, D. and Teneketzis, D., (1994a), “Optimal Multiserver Stochastic Scheduling of Two Interconnected Priority Queues,” *Advances in Applied Probability*, **26**, 258-279.
- [16] Sennott, L. (1999) *Stochastic Dynamic Programming and the Control of Queueing Systems*.
- [17] Van Oyen, M.P., E. Senturk-Gel, and W.J. Hopp, “Performance Opportunity for Workforce Agility in Collaborative and Noncollaborative Work Systems” Forthcoming in IIE Transactions,