WORKING PAPER
ALFRED P. SLOAN SCHOOL OF MANAGEMENT

Dynamic Scheduling of a Two-Class Queue with Setups

Martin I. Reiman
Lawrence M. Wein

#3692-94-MSA

May 1994

MASSACHUSETTS
INSTITUTE OF TECHNOLOGY
50 MEMORIAL DRIVE
CAMBRIDGE, MASSACHUSETTS 02139

Dynamic Scheduling of a Two-Class Queue with Setups

Martin I. Reiman
Lawrence M. Wein

#3692-94-MSA May 1994

Dynamic Scheduling of a Two-Class Queue with Setups

Martin I. Reiman
AT&T Bell Laboratories
Murray Hill, NJ 07974

Lawrence M. Wein
Sloan School of Management, M.I.T.
Cambridge, MA 02139

ABSTRACT

We analyze two scheduling problems for a queueing system with a single server and two customer classes. Each class has its own renewal arrival process, general service time distribution and holding cost rate. In the first problem, a setup cost is incurred when the server switches from one class to the other, and the objective is to minimize the long run expected average cost of holding customers and incurring setups. The setup cost is replaced by a setup time in the second problem, where the objective is to minimize the average holding cost. By assuming that the queueing system operates under standard heavy traffic conditions, we approximate the dynamic scheduling problems by diffusion control problems. For both problems, considerable insight is gained into the nature of the optimal policy, and the computational results show that the proposed scheduling policy is within several percent of optimal over a broad range of problem parameters.

APRIL 1994

We consider two dynamic scheduling problems for a single server queueing system with two classes of customers. In both problems, each class possesses its own renewal arrival process, general service time distribution and holding cost rate, and the server incurs a setup when switching from one class to the other. In the *setup cost* problem, a setup cost is incurred and the objective is to minimize the long run expected average setup and holding cost. In the *setup time* problem, a random setup time is incurred when the server switches class, and the objective is to minimize the long run expected average holding cost. In both problems, the server has three options at each point in time: serve a customer from the class that is currently set up, switch to the other class (and immediately begin service in the setup cost problem), or sit idle.

These scheduling problems have numerous applications, most notably for manufacturing systems and *polling systems* in computer communication networks. The setup time problem is more realistic than the setup cost problem in most situations, but is also more difficult to analyze. However, the setup cost problem is relevant for some manufacturing systems because, motivated by just-in-time (JIT) manufacturing, many facilities have *internalized* their setup times; that is, they have essentially eliminated their setup times at the expense of incurring significant material, labor and/or capital costs.

Although many studies have analyzed the performance of polling systems under various scheduling policies (see Takagi 1986, Boxma and Takagi 1992 and references therein), relatively few papers have considered the optimal scheduling of polling systems. The seminal paper in this research area is Hofri and Ross (1987), who analyze a two-class system with setup costs and times. Let c_i and μ_i denote the holding cost rate and service rate, respectively, for class i customers. When $c_1\mu_1 = c_2\mu_2$, they show that a double threshold policy, where the server serves each class until its queue is exhausted and the length of the other queue achieves a certain threshold level, minimizes the cost of setups and holding customers, under both the discounted and average cost criteria. Very little is known about the polling problem when $c_1\mu_1 \neq c_2\mu_2$, aside from the fact that the class with the larger $c\mu$ index should be served to exhaustion.

Several authors have studied the setup time problem in which more than two classes are present. Structural results for symmetric systems are derived by Liu, Nain and Towsley (1991) and references therein. Browne and Yechiali (1989) derive quasi-dynamic index policies, which allow the server to choose the sequence of classes to visit at the beginning of each cycle, that minimize or maximize the mean cycle length. Boxma, Levy and Westrate (1991) derive an efficient polling table (a predetermined fixed visit sequence) for minimizing the mean waiting cost. Bertsimas and Xu (1993) derive lower bounds and construct static policies that perform close to the bound when all classes have identical $c\mu$ indices. Van Oyen and Duenyas (1992) develop a dynamic scheduling heuristic based on myopic reward rates; Duenyas and Van Oyen (1993) also construct a dynamic policy for the setup cost problem.

Since the two-class asymmetric problem appears to be analytically intractable, heavy traffic approximations are employed in an attempt to make further headway. That is, we make the *heavy traffic* assumption that the server must be busy the great majority of the time to satisfy demand. In the setup cost problem, we also need to assume that the setup costs are very large, roughly two orders of magnitude larger than the holding cost rate. Following in the tradition of Foschini (1977) and Harrison (1988), we study the diffusion control problem that arises as a heavy traffic limit of a sequence of queueing scheduling problems. These limiting control problems tend to be more tractable than their queueing counterparts and have led to network scheduling policies (see, for example, Harrison and Wein 1990 and Wein 1990b) that have a surprisingly simple form and appear to perform well.

Using the heavy traffic averaging principle derived in Coffman, Puhalskii and Reiman (1993), we show in Section 1 that the setup cost problem simplifies rather dramatically in the limiting heavy traffic regime: the dimension of the state space collapses from three (queue length of each class and the position of the server) to one (total workload). This result also allows our analysis to naturally decompose onto two different time scales. On the very fast time scale over which individual queue lengths change, we myopically optimize a control that specifies the amount of low priority work to serve as a function of the total workload. This state-dependent control is derived in closed form and offers considerable insight. On the slower time scale over which the total workload varies, a singular control problem is solved that specifies a busy/idle policy. The solution to this control problem leads to a rather complex equation for one variable, which represents a threshold level, that can easily be solved numerically.

The setup time problem is addressed in Section 2, and the averaging principle in Coffman, Puhalskii and Reiman (1994) leads to a limiting control problem that again is one-dimensional, although here we obtain an explicit diffusion control problem. The control, which represents the amount of low priority work to serve as a function of the total workload, appears in the drift term of the diffusion process in a nonlinear fashion, and consequently the optimality equation leads to a nonlinear ordinary differential equation (ODE) that cannot be solved explicitly. However, we use asymptotics to obtain a scheduling policy; the asymptotics also reveal a substantial qualitative difference between the optimal policies in the setup cost and setup time cases.

For both problems, we use the value iteration algorithm to obtain "exact" optimal policies for a variety of test cases, and show in Section 3 that the suboptimality of the proposed policies is within several percent of optimal over a broad range of problem parameters.

Our presentation of the analysis, and indeed the analysis itself, is rather informal throughout. For example, we do not prove that the limiting control problems are the heavy traffic limit of a sequence of queueing scheduling problems. Also, several of our claims regarding the nature of the limiting control problems and their optimal solutions are not proved. Providing a rigorous presentation of our results would be extremely demanding, and would take us far afield from our two main objectives: to obtain fundamental insights into the nature of the optimal policies and to develop effective scheduling policies for these systems. However, much of our analysis relies upon observations that have been rigorously proven for simpler systems, and we have no doubt that our results are essentially correct. We hope that this approach increases the accessility of the paper without sacrificing the persuasiveness of our arguments.

1 THE SETUP COST PROBLEM

1.1 Problem Description

Customers of class $i = 1, 2$ arrive according to independent renewal processes, where λ_i and c_{ai}^2 denote respectively the arrival rate and squared coefficient of variation (variance divided by the square of the mean) of the interarrival times. Each class has its own general service time distribution with service rate μ_i and squared coefficient of variation c_{si}^2, and we define the system's traffic intensity by $\rho = \sum_{i=1}^{2}(\lambda_i/\mu_i)$. A cost c_i is incurred per unit time for holding a class i customer in the system. A setup cost $K/2$ is imposed whenever the server switches from one class to the other, so that K is the setup cost per cycle.

The server has three scheduling options at each point in time: serve the class that is currently set up, switch to the other class and initiate service, or sit idle. Since a switchover is instantaneous and costly, the option of switching to the other class and idling need not be considered. We assume that the server works in a preemptive-resume fashion, although the heavy traffic analysis is too crude to capture the effects of the nonpreemptive discipline as an alternative assumption. Let $Q_i(t)$ be the number of class i customers in queue or in service at time t, and let $J(t)$ denote the number of times the server sets up in the time interval $[0, t]$. Then our objective is to find a nonanticipating (with respect to the queue length process) scheduling policy to minimize

$$\limsup_{T\to\infty} \frac{1}{T} E\left[\int_0^T \sum_{i=1}^{2} c_i Q_i(t)dt + \frac{K}{2}J(T)\right] . \tag{1.1}$$

1.2 The Heavy Traffic Normalizations

A precise formulation of the approximating diffusion control problem requires much notation that would not be subsequently used. In addition, the limiting control problem will not be explicitly solved; rather, we optimize over a specific form of policy that is introduced in Subsection 1.4. Hence the heavy traffic control problem will not be precisely formulated, and a description of the heavy traffic conditions and normalizations will suffice for our purposes.

The approximating control problem is the limit of a sequence of scheduling problems indexed by the heavy traffic scaling parameter n, where $n \to \infty$. Since a heavy traffic limit theorem will not be proved here, we avoid unnecessary notation by considering a single large integer n satisfying $\sqrt{n}(1-\rho) = c$, where c is positive and of moderate size (that is, $O(1)$); this standard heavy traffic condition requires the server to be busy the great majority of the time over the long run. As we will see later, the scheduling policy that arises out of our heavy traffic analysis is independent of the system parameter n. Let V_i be the *unfinished workload* process for class i; $V_i(t)$ is the amount of time a continuously busy server requires to clear all of the class i customers who are present in the system at time t. The *normalized*, or scaled, queue length process is defined by $Z_i(t) = Q_i(nt)/\sqrt{n}$; similarly, $W_i(t) = V_i(nt)/\sqrt{n}$ denotes the normalized workload process. We approximate these normalized processes by the appropriate, and yet to be defined, limiting processes. Although $V_i(t)$ is not directly observable by the scheduler at time t, the normalized workload process is more convenient to employ than the normalized queue length process in the approximating heavy traffic control problem. However, we use the linear identity $Z_i = \mu_i W_i$ to translate the solution of the approximating control problem into a scheduling policy that is expressed in terms of the original queue length process (Q_1, Q_2). This linear identity is justified by extant heavy traffic limit theorems for many queueing systems.

In addition to speeding up time by a factor of n and reducing the queue lengths by a factor of $\sqrt{n}$, we also need to rescale the cost parameters c_i and K. The crux of problem (1.1) is the tradeoff between setup costs and holding costs, and hence to obtain a nontrivial solution to the approximating control problem, these two costs need to be of the same order of magnitude. Since only the ratio of these two costs matters, without loss of generality we leave the holding cost rates c_1 and c_2 unscaled at $O(1)$, and only scale the setup cost K. The following thought experiment allows us to conclude that the setup cost K needs to be divided by n in the approximating control problem. The heavy traffic condition implies that there are $O(\sqrt{n})$ customers in the original queueing system, and hence $O(1)$ scaled customers in the heavy traffic system. The holding cost rate is effectively multiplied by n because of the time scaling, so holding costs are incurred in the limiting control problem at the rate of $O(n^{3/2})$ per unit time. Since $O(\sqrt{n})$ customers are in the system, the server switches class every $O(\sqrt{n})$ unscaled time units, on average, implying that setup costs are incurred at the rate of $O(\sqrt{n})$ per unit time in the heavy traffic time scale. Since holding costs are incurred at rate $O(n^{3/2})$

and setup costs are incurred at rate $O(\sqrt{n})$, the setup cost K must be $O(n)$ for these cost rates to be of the same order, and to get an $O(1)$ limiting setup cost, we must divide the setup cost K by the heavy traffic scaling parameter n. Consequently, let $\kappa = K/n$ denote the normalized setup cost. Thus, heavy traffic conditions for the setup cost problem imply that the traffic intensity should be near one and the setup cost should be large. A canonical example is to set $n = 100$ and set c, c_1, c_2 and κ all equal to one, so that $\rho = 0.9$ and the setup cost $K = 100$.

1.3 A Preliminary Heavy Traffic Result

The starting point for the setup cost problem is a recent heavy traffic result due to Coffman, Puhalskii and Reiman (1993), which will be referred to hereafter as the CPR result. We present an informal statement of a special case of this heavy traffic limit theorem that will suffice for our purposes. As in problem (1.1), consider a queueing system with a single server and two customer classes. The CPR result is derived under a specific queue discipline: the server serves each class to exhaustion, and then switches class. The work conserving nature of the discipline implies that the total workload process $W = W_1 + W_2$ is identical to the corresponding process under the FCFS policy. It follows from the heavy traffic limit theorem of Iglehart and Whitt (1970) that this process is well approximated under heavy traffic conditions by RBM$(-c, \sigma^2)$, which is a reflected Brownian motion (see Harrison 1985 for a definition) on $[0, \infty)$ with drift $-c$ and variance

$$\sigma^2 = \sum_{i=1}^{2} \frac{\lambda_i}{\mu_i^2}(c_{ai}^2 + c_{si}^2) \ . \tag{1.2}$$

It turns out to be impossible to obtain a limit process for (W_1, W_2) in the usual sense, because in the heavy traffic limit, the two-dimensional process moves back and forth along the cross diagonal at an infinite rate, the direction being determined by which of the two queues is being served; see Figure 1. The CPR result provides an *averaging principle* that implies the following: given the normalized total workload W, the two-dimensional workload (W_1, W_2) can be treated as if it is uniformly distributed along the constant workload line from $(0, W)$ to $(W, 0)$. That is, the two-dimensional distribution is $(UW, (1-U)W)$, where U is a uniform $[0, 1]$ random variable that is independent of W.

This averaging principle is due to a *time scale decomposition.* On the time scale giving rise to reflected Brownian motion for the total workload, the two-dimensional workload process moves (asymptotically) infinitely quickly. If we slow time down so that the two-dimensional workload moves at a finite and positive rate, the total workload stays fixed, and the movement of the two-dimensional workload is deterministic. Although this result has been proved only under the exhaustive policy, we assume that it holds more generally. This has far-reaching implications for the heavy traffic analysis of our control problem. In particular, it allows us to *collapse the state space* of the control problem from three dimensions (the number of customers of each class in the system and the location of the server) to one dimension (the total workload).

1.4 The Form of the Optimal Policy

The traditional heavy traffic approach to scheduling problems is to precisely formulate the queueing system scheduling problem, find the limiting control problem that approximates the scheduling problem under heavy traffic conditions, and solve the latter problem. The approach taken here is slightly different: we first argue that the optimal policy should be of a specific form in the heavy traffic limit, and then optimize the approximating system over this class of policies.

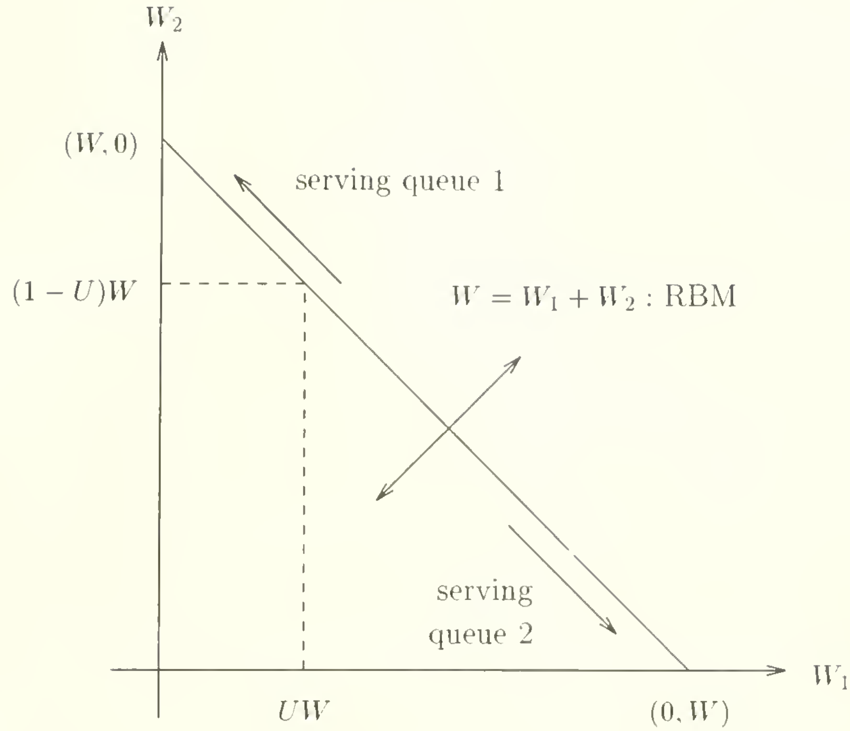

Figure 1: The heavy traffic averaging principle of Coffman, Puhalskii and Reiman (CPR).

Without loss of generality, we assume that $c_1\mu_1 \geq c_2\mu_2$ and sometimes refer to classes 1 and 2 as the high and low priority classes, respectively. Existing results (Hofri and Ross for Poisson arrivals and exponential service times, and Duenyas and Van Oyen for Poisson arrivals and general service times) as well as intuition suggest that class 1 should be served to exhaustion. (It is possible to construct examples where this policy is not optimal. Our contention is that it is *asymptotically* optimal in heavy traffic.) When the server is set up for class 1, the only other decision is to specify whether the server should idle or switch to class 2 when no class 1 customers are present. Since we work with the normalized workload process (W_1, W_2), the only reasonable form of the optimal policy is to switch when $W_2(t) \geq w_2$ for some scaled threshold level w_2.

Since switching is instantaneous, $W_1(t) = 0$ and $W_2(t) = x$ at the moment of switching, where x must be greater than or equal to the threshold w_2. Because preemption is allowed, the server should never idle at class 2 when class 2 customers are present. The CPR result implies that the total workload $W = W_1 + W_2$ remains constant in the heavy traffic time scale while the server is serving class 2 customers. Hence, our decision can be expressed as the amount $u(x)$ by which the server depletes class 2's original work. That is, class 2 is served until $W_1(t) = u(x)$ and $W_2(t) = x - u(x)$. The control $u(x)$ must be between zero and x, where $u = x$ is the exhaustive policy. Figure 2 contains a picture with $u(x) = x/3$ for a particular value of x. Since a different amount u can be chosen for each value of the total workload x, the control $u(x)$ can generate any possible switching curve in the nonnegative orthant, and so is without loss of generality.

Finally, since the server should never idle at station 2 when $W_2(t) > 0$, if $u(x) < x$ then

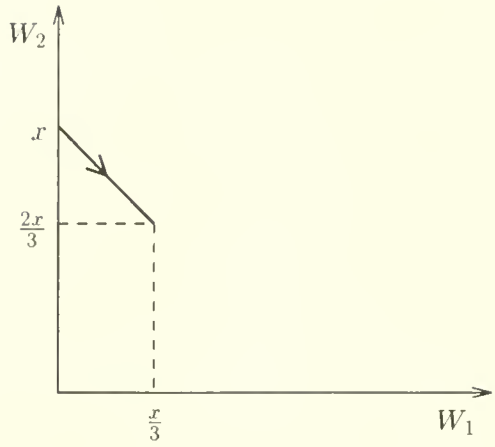

Figure 2: The control $u(x) = x/3$ for a fixed value of x.

the server immediately switches back to class 1 when $W_1(t) = u(x)$ and $W_2(t) = x - u(x)$. However, if $u(x) = x$ and hence class 2 is served exhaustively, then the server must decide whether to idle or switch back to class 1. Once again, the obvious form of the optimal policy in this case is to idle until $W_1(t)$ is greater than or equal to w_1. Notice that if the threshold levels w_1 and w_2 were both zero, then infinite setup costs would be incurred.

In summary, the controls are the function $u(x)$, which specifies the amount of class 2's work to serve, and the threshold levels w_1 and w_2, which dictate the server's busy/idle policy. The form of the optimal policy in heavy traffic is: *serve class 1 until* $W_1(t) = 0$ *and* $W_2(t) \geq w_2$; *switch to class 2. If* $W_2(t) = x$ *at the moment of switching, then serve class 2 until* $W_1(t) = u(x)$ *and* $W_2(t) = x - u(x)$. *If* $u(x) < x$, *then switch to class 1; if* $u(x) = x$, *then do not switch until* $W_1(t) \geq w_1$.

1.5 An Overview of the Analysis

The analysis hinges on the following crucial observation: *since setups are instantaneous, the total workload process is only affected by the server's busy/idle policy, not by how often the server switches class.* Hence, the control $u(x)$ only influences the total workload indirectly via the idling. However, $u(x)$ does affect the rate at which holding costs and setup costs are incurred when the total workload is x. Therefore, a two-step procedure is employed to find the optimal policy $(u(x), w_1, w_2)$ within the specified form. In the first step, the control $u(x)$ is chosen to minimize the cost rate for each state x; this minimization is performed independently for each state x. In the second step, we attempt to find the optimal threshold levels w_1 and w_2, and hence the optimal total workload process. Our heavy traffic analysis will show that the optimal total workload process is a RBM$(-c, \sigma^2)$ on $[w, \infty)$, where w is a parameter that is chosen to minimize the total expected cost. Hence, the Brownian model is too crude to distinguish between the two thresholds w_1 and w_2, and so we set both w_1 and w_2 equal to the derived value of w.

As in previous heavy traffic scheduling work (see Harrison 1988 and Wein 1990a, for example), the analysis naturally decomposes onto two time scales. On the very fast time scale, where individual queues can change instantaneously fast, we myopically optimize over $u(x)$. Then, on the slower time scale over which the total workload varies, a singular control problem is solved to find the threshold, or reflecting barrier, w that specifies the busy/idle policy.

1.6 The Optimal $u(x)$

The control $u(x)$ is chosen to minimize the cost rate that is incurred when the normalized total workload process is x. Under the policy characterized by $u(x)$, class 2's work is depleted by the amount $u(x)$ if the total workload when the server arrives to class 2 is x. The CPR result implies that, for our purposes, it is as if W_1 is uniformly distributed between 0 and $u(x)$, and W_2 is uniformly distributed between $x - u(x)$ and x. Since $Z_i = \mu_i W_i$, the holding cost rate when in state x is

$$\begin{aligned}\sum_{i=1}^{2} c_i\mu_i E[W_i] &= c_1\mu_1\frac{u(x)}{2} + c_2\mu_2\left(\frac{2x-u(x)}{2}\right) \\ &= c_2\mu_2 x + \frac{\Delta u(x)}{2}\ , \qquad (1.3)\end{aligned}$$

where

$$\Delta = c_1\mu_1 - c_2\mu_2\ . \qquad (1.4)$$

To find the setup cost rate when in state x, we need to find the cycle length. For a fixed total unfinished workload x, the two-dimensional workload process (W_1, W_2) moves back and forth deterministically at an asymptotically infinite rate along the line segment from $(0, x)$ to $(u(x), x - u(x))$; hence, the cycle length is deterministic.

We determine the deterministic cycle length, and hence the setup cost rate, as a function of the normalized workload by slowing down the time scale. If the server finds x units of work in class 2 upon arrival, then this work will be depleted at rate $1 - \rho_2$. The server works until $W_1(t) = u(x)$ and $W_2(t) = x - u(x)$, which occurs after $u(x)/(1-\rho_2)$ time units. As we will see later, the normalized total workload process W never spends any time below $\max(w_1, w_2)$, and so we need not include any unnecessary inserted idle time into the cycle length calculation. Therefore, it takes $u(x)/(1-\rho_1)$ time units to deplete class 1 and complete the cycle, resulting in a cycle of length $u(x)/(1-\rho_2) + u(x)/(1-\rho_1)$. Since the holding costs are estimated using a heavy traffic approximation and the scheduling problem essentially trades off the setup and holding costs, a more accurate analysis results if we assume that $\rho = 1$ in our cycle length expression, which simplifies the cycle length to $u(x)/\rho_1\rho_2$. Because two setups are incurred in each cycle, the setup cost rate when in state x is $\rho_1\rho_2\kappa/u(x)$.

Now we find the optimal $u(x)$ by solving:

$$\min_{u(x)\in[0,x]} c_2\mu_2 x + \frac{\Delta u(x)}{2} + \frac{\rho_1\rho_2\kappa}{u(x)}\ . \qquad (1.5)$$

If we define

$$\widehat{w} = \sqrt{\frac{2\rho_1\rho_2\kappa}{\Delta}}\ , \qquad (1.6)$$

then straightforward calculus leads to

$$u^*(x) = \min(x, \widehat{w})\ . \qquad (1.7)$$

Hence, $\widehat{w}$ is the largest value of the total workload for which class 2 is served exhaustively. Notice that $\widehat{w} = \infty$ when $\Delta = 0$, and so the optimal control in the balanced case is $u^*(x) = x$ for all x, which corresponds to exhaustive service for class 2.

1.7 The Optimal Threshold Level

In this subsection, we analyze the normalized total workload process under the form of the proposed policy, using the control $u^*(x)$ in (1.7). This analysis shows that the total workload process W is a RBM$(-c, \sigma^2)$ on $[w, \infty)$, where w is a parameter that will be optimized over.

In the balanced case, the control $u^*(x)$ implies that the form of the optimal policy is to switch from class 1 to class 2 when $W_1(t) = 0$ and $W_2(t) \geq w_2$, and switch from class 2 to class 1 when $W_2(t) = 0$ and $W_1(t) \geq w_1$. Let us begin by assuming that $w_1 < w_2$. When the two-dimensional workload process hits the point $(x, 0)$, where $x \in [w_1, w_2)$, then the server will switch to class 1 and the process instantaneously moves to the point $(0, x)$. Since $x < w_2$, the server will not immediately switch back to class 2. Rather, the server serves newly arriving class 1 customers or sits idle until class 2's workload reaches w_2. In the heavy traffic limit, time is sped up by a factor of n and the two-dimensional workload process instantaneously moves from the point $(0, x)$ to the point $(0, w_2)$. Consequently, the total workload process never spends any time below the value of w_2. A similar argument when $w_1 > w_2$ implies that the total workload process is a RBM$(-c, \sigma^2)$ on $[\max(w_1, w_2), \infty)$. Thus, the heavy traffic analysis is too crude to distinguish between the thresholds w_1 and w_2, and we follow the convention of setting them both equal to w; later in this subsection, the cost minimizing value of w will be derived. Hence, the setup cost problem *decomposes* in the balanced case, and we can optimize over a single threshold parameter w independently of $u^*(x)$.

For the imbalanced case, the total workload process needs to be investigated under four different cases, depending upon the relative values of the normalized threshold levels w_1, w_2 and $\widehat{w}$.

Case 1: $0 \leq w_1, w_2 \leq \widehat{w}$. The curves for switching from class 2 to class 1 for all four cases are pictured in Figure 3, where the vertical portion of the switching curve follows from (1.7). The argument put forth in the balanced case implies that the total workload process in this case is a RBM$(-c, \sigma^2)$ on $[\max(w_1, w_2), \infty)$. We again set w_1 and w_2 equal to the parameter w, and model the optimal total workload process as a RBM$(-c, \sigma^2)$ on $[w, \infty)$; in this case, the parameter w is optimized over the region $0 \leq w \leq \widehat{w}$.

Case 2: $\widehat{w} \leq w_1, w_2$. The state $(w_1, 0)$ is never reached, and hence the parameter w_1 does not play a role here. By a similar argument as above, W is a RBM$(-c, \sigma^2)$ on $[w_2, \infty)$. Thus, once again, we set w_1 and w_2 equal to a parameter w, let W be an RBM$(-c, \sigma^2)$ on $[w, \infty)$, and optimize w over the region $w \geq \widehat{w}$.

Case 3: $0 \leq w_1 \leq \widehat{w} \leq w_2$. The total workload W is an RBM$(-c, \sigma^2)$ on $[w_2, \infty)$, and so we set w_1 and w_2 equal to w and optimize over $w \geq \widehat{w}$. Thus, case 3 reduces to case 2.

Case 4: $0 \leq w_2 \leq \widehat{w} \leq w_1$. The parameter w_1 is not a factor, and W is an RBM$(-c, \sigma^2)$ on $[w_2, \infty)$. Hence, case 4 reduces to case 1.

In summary, it suffices to restrict our attention to cases 1 and 2; thus, as in the balanced case, the single threshold parameter $w \geq 0$ can be optimized independently of $u^*(x)$.

Now we derive the optimal value of the parameter w. Substituting the optimal control $u^*(x)$ from (1.7) into the cost rate function in (1.5) yields the optimal cost rate when the

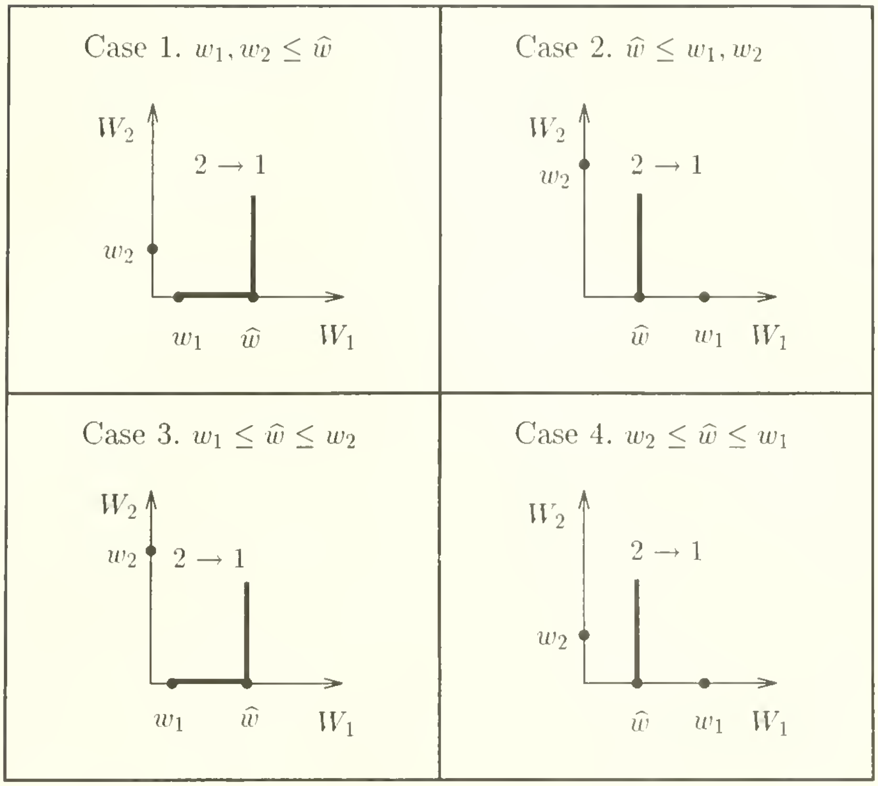

Figure 3: The total workload process for various values of w_1, w_2 and $\widehat{w}$.

normalized workload is x, which is

$$c_2\mu_2 x + \frac{\Delta x}{2} + \frac{\rho_1\rho_2\kappa}{x} \quad \text{when} \quad x \le \widehat{w} \ , \tag{1.8}$$

and

$$c_2\mu_2 x + \sqrt{2\rho_1\rho_2\Delta\kappa} \quad \text{when} \quad x \ge \widehat{w} \ . \tag{1.9}$$

To find the total expected average cost, the optimal cost rate is integrated over the steady state distribution of the total workload process. The normalized workload process is approximated by an RBM$(-c, \sigma^2)$ on $[w, \infty)$, which has stationary density function $\alpha e^{-\alpha(x-w)}$ for $x \ge w$, where $\alpha = 2c/\sigma^2$.

If $w \ge \widehat{w}$, then the total expected cost is

$$\begin{aligned} C(w) &= \int_w^\infty (c_2\mu_2 x + \sqrt{2\rho_1\rho_2\Delta\kappa})\alpha e^{-\alpha(x-w)}dx \\ &= c_2\mu_2\left(w + \frac{1}{\alpha}\right) + \sqrt{2\rho_1\rho_2\Delta\kappa} \ , \end{aligned} \tag{1.10}$$

which is increasing in w. Therefore, the optimal value of w is less than or equal to $\widehat{w}$, and case 1 of the previous subsection holds. Define the aggregate cost parameter $\bar{C} = (c_1\mu_1 + c_2\mu_2)/2$. Then the total expected cost equals

$$C(w) = \alpha e^{\alpha w}\left(\bar{C}\int_w^{\widehat{w}} xe^{-\alpha x}dx + \rho_1\rho_2\kappa\int_w^{\widehat{w}} \frac{e^{-\alpha x}}{x}dx + c_2\mu_2\int_{\widehat{w}}^\infty xe^{-\alpha x}dx\right.$$

$$+ \sqrt{2\rho_1\rho_2\Delta\kappa}\int_{\widehat{w}}^{\infty} e^{-\alpha x}dx\Bigg) . \tag{1.11}$$

Setting the derivative of the total expected cost with respect to w equal to zero yields

$$\begin{aligned} 0 &= \bar{C}\left(1-(\alpha\widehat{w}+1)e^{\alpha(w-\widehat{w})}\right)+\alpha\rho_1\rho_2\kappa\left(\alpha e^{\alpha w}(E_1(\alpha w)-E_1(\alpha\widehat{w}))-\frac{1}{w}\right) \\ &\quad + \alpha\sqrt{2\rho_1\rho_2\Delta\kappa}e^{\alpha(w-\widehat{w})}+c_2\mu_2(\alpha\widehat{w}+1)e^{\alpha(w-\widehat{w})} , \end{aligned} \tag{1.12}$$

or, upon simplification,

$$\begin{aligned} 0 &= \bar{C}+e^{\alpha(w-\widehat{w})}\left(\alpha\sqrt{2\rho_1\rho_2\Delta\kappa}-\frac{\Delta(\alpha\widehat{w}+1)}{2}\right) \\ &\quad + \alpha^2\rho_1\rho_2\kappa\left(e^{\alpha w}(E_1(\alpha w)-E_1(\alpha\widehat{w}))-\frac{1}{\alpha w}\right) , \end{aligned} \tag{1.13}$$

where

$$E_1(x)=\int_x^{\infty}\frac{e^{-t}}{t}dt, \quad x>0 \tag{1.14}$$

is the exponential integral. It turns out that $C(w)$ is not convex; however, the solution to (1.13) is well behaved numerically, and yields the global minimum of $C(w)$ for the cases we consider. We denote this solution by w^* and refer to it as the optimal threshold level. Since

$$C'(w)=\alpha C(w)-\alpha e^{\alpha w}\left(\bar{C}we^{-\alpha w}+\rho_1\rho_2\kappa\frac{e^{-\alpha w}}{w}\right) , \tag{1.15}$$

it follows that the optimal total expected cost is

$$C(w^*)=\bar{C}w^*+\frac{\rho_1\rho_2\kappa}{w^*} . \tag{1.16}$$

In the balanced case where $\Delta=0$, the first order condition (1.13) reduces to

$$\frac{1}{\alpha w}-e^{\alpha w}E_1(\alpha w)=\frac{c_1\mu_1}{\alpha^2\rho_1\rho_2\kappa} . \tag{1.17}$$

Moreover,

$$C''(w)=\alpha^3\rho_1\rho_2\kappa\left(e^{\alpha w}E_1(\alpha w)+\frac{1}{(\alpha w)^2}-\frac{1}{\alpha w}\right) , \tag{1.18}$$

and the convexity of $C(w)$ follows from the bound $e^x E_1(x)>1/(x+1)$. Replacing $e^{\alpha w}E_1(\alpha w)$ by its lower bound $1/(\alpha w+1)$ in (1.17) gives a simple approximate expression for the optimal threshold level:

$$w_{app}=\frac{1}{\alpha}\left(-\frac{1}{2}+\sqrt{\frac{1}{4}+\frac{\alpha^2\rho_1\rho_2\kappa}{c_1\mu_1}}\right) . \tag{1.19}$$

1.8 The Proposed Scheduling Policy

The heavy traffic solution is given by the control $u^*(x)$ defined in (1.7), which specifies a switching curve, and the threshold level w^* satisfying (1.13) or (1.17). We use this solution to propose a scheduling policy in terms of the three-dimensional state of the original problem, which is the two-dimensional queue length process (Q_1, Q_2), and the server location. Since both $u^*(x)$ and w^* are expressed in terms of the normalized workload W, several steps are required to translate this heavy traffic solution into a proposed policy. First, we reverse the heavy traffic scaling to express the quantities $u^*(x)$ and w^* in terms of the unscaled workload V. Since $W(t) = V(nt)/\sqrt{n}$, when the normalized workload W equals x, then the original workload V equals y, where $y = \sqrt{n}x$. The control $u^*(x)$ requires the server to serve class 2 until $W_1 = u^*(x)$, or equivalently, until $V_1/\sqrt{n} = u^*(y/\sqrt{n})$. If we substitute K/n for the normalized setup cost κ in (1.6), then when the total workload V equals y, class 2 is served until

$$\begin{aligned} V_1 &= \sqrt{n}u^*\left(\frac{y}{\sqrt{n}}\right) \\ &= \sqrt{n}\min\left(\frac{y}{\sqrt{n}}, \sqrt{\frac{2\rho_1\rho_2 K}{\Delta n}}\right) \\ &= \min\left(y, \sqrt{\frac{2\rho_1\rho_2 K}{\Delta}}\right) . \end{aligned} \tag{1.20}$$

By (1.20), class 2 is served exhaustively as long as the total workload V is less than or equal to

$$\widehat{v} = \sqrt{\frac{2\rho_1\rho_2 K}{\Delta}} , \tag{1.21}$$

which, not surprisingly, equals $\sqrt{n}\widehat{w}$. Similarly, if we define the unscaled threshold $v^* = \sqrt{n}w^*$, then substitution of $v/\sqrt{n}$ for w, K/n for κ, and $2\sqrt{n}(1-\rho)/\sigma^2$ for α in (1.13) and (1.17) yields, respectively,

$$\begin{aligned} 0 &= \bar{C} + e^{\theta(v-\widehat{w})}\left(\theta\sqrt{2\rho_1\rho_2\Delta K} - \frac{\Delta(\theta\widehat{v}+1)}{2}\right) \\ &\quad + \theta^2\rho_1\rho_2 K\left(e^{\theta v}(E_1(\theta v) - E_1(\theta\widehat{v})) - \frac{1}{\theta v}\right) \end{aligned} \tag{1.22}$$

and

$$\frac{1}{\theta v} - e^{\theta v}E_1(\theta v) = \frac{c_1\mu_1}{\theta^2\rho_1\rho_2 K} , \tag{1.23}$$

where

$$\theta = \frac{2(1-\rho)}{\sigma^2} . \tag{1.24}$$

Similar substitutions into (1.19) gives

$$v_{app} = \frac{1}{\theta}\left(-\frac{1}{2} + \sqrt{\frac{1}{4} + \frac{\theta^2\rho_1\rho_2 K}{c_1\mu_1}}\right) . \tag{1.25}$$

Finally, the predicted optimal average cost for the original scheduling problem is

$$\sqrt{n}C(w^*) = \bar{C}v^* + \frac{\rho_1\rho_2 K}{v^*} . \tag{1.26}$$

Notice that the quantities in (1.20)–(1.26) *are independent of the heavy traffic scaling parameter* n, and are expressed solely in terms of the primitive problem parameters.

Now that the optimal control has been translated into unscaled workloads, we use the simple heavy traffic relationship $\mu_i W_i = Z_i$ between workloads and queue lengths to express the switching curve and threshold level in terms of queue lengths. The only remaining hurdle is that the resulting quantities are continuous, whereas the two-dimensional queue length process resides on a lattice. We naively ignore this difference between our continuous solution and the discrete state space, which essentially amounts to rounding the threshold level up to the next highest integer, and rounding the switching curve out to the next largest lattice points. In addition to being the most natural translation of the continuous solution, it also prevents us from rounding a threshold level down to zero, where infinite setup costs would be incurred.

In the balanced case, the critical value $\widehat{v}$ in (1.21) equals infinity, which corresponds to exhaustive service. The proposed policy is: *when $Q_1(t) = 0$ and $Q_2(t) \geq \mu_2 v^*$, then switch from class 1 to class 2; when $Q_2(t) = 0$ and $Q_1(t) \geq \mu_1 v^*$, then switch from class 2 to class 1.* The parameter v^* is the solution to (1.23). This policy is a special case of the double threshold policy introduced by Hofri and Ross, who prove that the optimal policy is of this form in the balanced case when arrivals are Poisson.

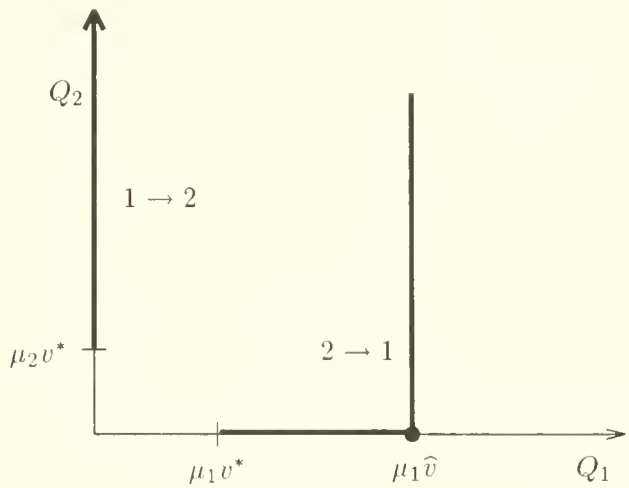

Figure 4: The proposed scheduling policy when $c_1\mu_1 > c_2\mu_2$.

By (1.20), the proposed policy for the imbalanced case has a particularly simple form, and is pictured in Figure 4: *when $Q_1(t) = 0$ and $Q_2(t) \geq \mu_2 v^*$, then switch from class 1 to class 2. When $Q_1(t) \geq \mu_1 \widehat{v}$ or ($Q_2(t) = 0$ and $Q_1(t) \geq \mu_1 v^*$), then switch from class 2 to class 1.* The parameters $\widehat{v}$ and v^* are defined in (1.21) and (1.22), respectively. Hence, the server switches to the high priority class as soon as the queue length of that class grows to the level $\mu_1 \hat{v}$. By (1.4) and (1.21), this critical level increases with the setup cost K and decreases as the $c\mu$ differential between the two classes gets larger. Although one might have expected a general nonlinear switching curve, the vertical boundary in Figure 4 is obtained. It is worth noting that the heuristic policy of Duenyas and Van Oyen is also of this general form.

2 THE SETUP TIME PROBLEM

2.1 Problem Description

The only difference between the setup time problem considered in this section and the setup cost problem is that a random setup time rather than a setup cost is incurred when the server switches from one class to the other; all relevant notation from the setup cost problem will be retained. By Coffman, Puhalskii and Reiman (1994), the performance of this system in heavy traffic depends upon the setup time distributions only through the mean setup time per cycle, which we denote by s. The server has three scheduling options at each point in time: serve a customer from the class that is currently set up, initiate a setup or sit idle. The objective is to find a preemptive-resume, nonanticipating scheduling policy to minimize

$$\limsup_{T\to\infty} \frac{1}{T} E\left[\int_0^T \sum_{i=1}^{2} c_i Q_i(t)dt\right] . \tag{2.1}$$

2.2 The Approximating Diffusion Control Problem

Unlike, for example, the server vacation times in Kella and Whitt (1990), the setup times are not rescaled as the heavy traffic limit is approached; that is, we assume that the setup times are $O(1)$. The lack of setup costs has eliminated the incentive to insert unnecessary idleness in heavy traffic; inserted idleness increases the workload, which in turn increases the holding costs. Hence, the proposed form of the optimal policy is simpler than in the setup cost problem: *serve class 1 to exhaustion and then set up for class 2. If class 2's normalized unfinished workload* $W_2(t) = x$ *at the setup completion epoch, then serve class 2 until* $W_1(t) = u(x)$ *and* $W_2(t) = x - u(x)$, *and immediately switch back to class 1.* As in the setup cost problem, the control $\{u(x), x \geq 0\}$ can generate any arbitrary switching curve in the nonnegative orthant.

Since the setup times are $O(1)$, switchovers occur instantaneously in the heavy traffic limit. Hence, the two-dimensional normalized unfinished workload process (W_1, W_2) will move at an asymptotically infinite rate back and forth between $(0, x)$ and $(u(x), x - u(x))$ when the total normalized workload $W = x$, just as in the setup cost problem. We now present a heuristic argument for the characterization of the normalized total unfinished workload process W. If setup times are zero and no unnecessary idleness is inserted, recall that the the limiting process is a RBM on the nonnegative orthant with drift $\sqrt{n}(\rho - 1)$ and variance σ^2 given by (1.2). When setup times are positive, we claim that the limiting process is a diffusion process on the nonnegative orthant with variance σ^2 and a state-dependent drift, which we denote $\mu(x)$. Since the system is heavily congested, setups are incurred relatively rarely and the mean and variance of the setup times do not appear in the variance term of the limiting diffusion process.

As explained in Harrison and Nguyen (1990), the drift of the stochastic process underlying a heavy traffic approximation equals the expected growth rate of the normalized *workload netflow process*, which is the arrival rate of work minus the potential (that is, assuming work is always available) depletion rate of work. With zero setup times, unscaled work arrives at rate ρ and is potentially depleted at rate one. With time sped up by a factor of n and workloads reduced by a factor of $\sqrt{n}$, the expected growth rate of the normalized workload netflow process is $\sqrt{n}(\rho - 1)$. When setup times are positive, the potential depletion rate of work is strictly less than one and will equal the fraction of time that the server spends doing useful work; that is, the fraction of time the server actually serves customers, rather than incurring setups. We claim that the drift when the normalized total workload is x equals

$$\mu(x) = \sqrt{n}(\rho - f(x)) , \tag{2.2}$$

where $f(x)$ is the fraction of time that the server spends doing useful work when the normalized unfinished workload W equals x. Since cycles occur rapidly in heavy traffic, only averages matter and we can carry out the calculation of $f(x)$ over one cycle. Let us begin the cycle when all $\sqrt{n}x$ units of unscaled unfinished work V is of class 2. Class 2 work is depleted at rate $1 - \rho_2$ until $V_1(t) = \sqrt{n}u(x)$ and $V_2(t) = \sqrt{n}(x - u(x))$, which takes $\sqrt{n}u(x)/(1 - \rho_2)$ time units. Similarly, $\sqrt{n}u(x)/(1-\rho_1)$ time units are required to serve class 1 customers, thereby completing the cycle. Hence, if we assume $\rho = 1$ (see Section 1 for the rationale behind this assumption), then the length of the cycle is

$$\frac{\sqrt{n}u(x)}{\rho_2} + \frac{\sqrt{n}u(x)}{\rho_1} + s \ , \tag{2.3}$$

and the fraction of time the server spends doing useful work is

$$\begin{aligned} f(x) &= \frac{\frac{\sqrt{n}u(x)}{\rho_2} + \frac{\sqrt{n}u(x)}{\rho_1}}{\frac{\sqrt{n}u(x)}{\rho_2} + \frac{\sqrt{n}u(x)}{\rho_1} + s} \\ &= \frac{\sqrt{n}u(x)}{\sqrt{n}u(x) + s\rho_1\rho_2} \ . \end{aligned} \tag{2.4}$$

By (2.2),

$$\begin{aligned} \mu(x) &= \sqrt{n}(\rho - 1) + \sqrt{n}(1 - f(x)) \\ &= -c + \sqrt{n}(1 - f(x)) \ . \end{aligned} \tag{2.5}$$

Since

$$\sqrt{n}(1 - f(x)) = \frac{\sqrt{n}\rho_1\rho_2 s}{\sqrt{n}u(x) + \rho_1\rho_2 s} \rightarrow \frac{\rho_1\rho_2 s}{u(x)} \quad \text{as} \quad n \rightarrow \infty \ , \tag{2.6}$$

we have

$$\mu(x) = \frac{\rho_1\rho_2 s}{u(x)} - c \ . \tag{2.7}$$

In summary, we approximate the normalized total unfinished workload process W by a $(\mu(x), \sigma^2)$ diffusion. In the special case of exhaustive service (that is, $u(x) = x$ for all x), Coffman, Puhalskii and Reiman (1994) show that the normalized total unfinished workload process weakly converges to this diffusion process as $\rho \rightarrow 1$. If, in addition, $c = 0$ (that is, $\rho = 1$), this diffusion process is a Bessel process.

As we mentioned earlier, given $W(t) = x$, the two-dimensional process (W_1, W_2) behaves the same with or without setup times; hence the holding cost rate when in state x is given by (1.3). Therefore, the approximating diffusion control problem is to choose $\{u(x), x \geq 0\}$ to minimize

$$\limsup_{T\rightarrow\infty} \frac{1}{T} E\left[\int_0^T \left(c_2\mu_2 X(t) + \frac{\Delta u(X(t))}{2}\right) dt\right] \ , \tag{2.8}$$

where X is a $(\mu(x), \sigma^2)$ diffusion process and $u(x) \in [0, x]$ for all $x \geq 0$.

The previous literature on heavy traffic approximations of queueing scheduling problems assumes zero setup times, and the time scale decomposition described in Section 1 leads to a deterministic pathwise optimization for the optimal queue length process and a singular control problem for the optimal cumulative idleness process. The presence of setup times destroys this simplifying structure, and (2.8) provides the first example of a scheduling problem for a queueing system that is approximated in heavy traffic by a drift control problem.

2.3 Analysis of the Diffusion Control Problem: The Balanced Case

Problem (2.8) simplifies considerably when each class has the same $c\mu$ index. Setting Δ equal to zero in (2.8) shows that the problem reduces to choosing $u(x)$ to minimize the mean of the stationary distribution of the diffusion process X. This goal is achieved by minimizing the drift $\mu(x)$ in (2.7), and hence the optimal control is $u(x) = x$ for all x; therefore, *the proposed scheduling policy for the balanced case is to serve each class to exhaustion, and immediately switch class.* The resulting diffusion process is a Bessel process with an additive drift.

The long run average cost of any stationary policy can be obtained from the stationary distribution (invariant measure) of the diffusion process 'induced' (via the resulting $\mu(x)$) by the policy. Fortunately, the subject of stationary distributions of one dimensional diffusions is old and well understood (c.f. Mandl 1968, or Karlin and Taylor 1981). Given a positive recurrent diffusion process on the nonnegative half line with drift $\mu(x)$ and variance σ^2, the stationary density satisfies the ordinary differential equation

$$\frac{\sigma^2}{2}\frac{d^2\pi(x)}{dx^2} - \frac{d}{dx}(\mu(x)\pi(x)) = 0, \quad x > 0 . \tag{2.9}$$

Associated with a reflecting boundary at zero, there is a boundary condition

$$\frac{\sigma^2}{2}\frac{d\pi(x)}{dx} = \mu(x)\pi(x), \quad x = 0 . \tag{2.10}$$

There is also the normalization condition

$$\int_0^\infty \pi(x)dx = 1 . \tag{2.11}$$

The solution of (2.9)–(2.11) can be obtained using integrating factors. For the Bessel process with an additive drift, where $\mu(x) = \rho_1\rho_2 s/x - c$, it can be shown by a bound involving Brownian motion that this process is positive recurrent when $c > 0$. The solution of (2.9)–(2.11) for this process is the gamma density

$$\pi(x) = \frac{\alpha(\alpha x)^\beta e^{-\alpha x}}{\Gamma(\beta+1)} , \quad x \geq 0 , \tag{2.12}$$

where $\alpha = 2c/\sigma^2$ is the scale parameter and $\beta = 2\rho_1\rho_2 s/\sigma^2$ is the shape parameter. (It is straightforward to verify that (2.12) solves (2.9), (2.10), and (2.11). Standard results from the theory of ordinary differential equations yield that (2.9)–(2.11) have a unique solution.) It turns out that the Bessel process reaches the origin only if $\beta < 1$; if $\beta \geq 1$ the process will never reach zero. The solution (2.12) is valid for both of these cases.

Under the exhaustive policy, the expected average cost incurred for the original system is $\sqrt{n}(c_2\mu_2 + \Delta/2)E[X(\infty)]$, where X is a $(\rho_1\rho_2 s/x - c, \sigma^2)$ diffusion. Since

$$\sqrt{n}E[X(\infty)] = \frac{\sqrt{n}\left(\frac{2\rho_1\rho_2 s}{\sigma^2}+1\right)}{\frac{2\sqrt{n}(1-\rho)}{\sigma^2}} = \frac{2\rho_1\rho_2 s + \sigma^2}{2(1-\rho)} , \tag{2.13}$$

the expected average cost is

$$\frac{\bar{C}(2\rho_1\rho_2 s + \sigma^2)}{2(1-\rho)} , \tag{2.14}$$

where the cost parameter $\bar{C}$ was defined earlier as $(c_1\mu_1 + c_2\mu_2)/2$.

For the balanced case, we can also introduce setup costs into the setup time problem without sacrificing tractability. We again let $\kappa = K/n$ denote the normalized setup cost per cycle. As in the balanced case of the setup cost problem, the proposed policy is a double threshold policy characterized by the normalized threshold level w. We now derive the optimal threshold value under the general imbalanced case, although this policy is only proposed for the balanced case. Under the threshold level w, the diffusion process with drift $\mu(x) = \rho_1\rho_2 s/x - c$ behaves as before but is not allowed to go below w. The stationary density $\pi(x)$ of the truncated process is obtained by solving equations analogous to (2.9), (2.10), and (2.11) with the reflecting barrier at $X = w$. The solution, which yields the stationary density for the normalized workload W, is

$$\pi(x) = \frac{\alpha(\alpha x)^\beta e^{-\alpha x}}{\Gamma(\beta+1, \alpha w)} \quad \text{for} \quad x \geq w \ . \tag{2.15}$$

where

$$\Gamma(\beta, \alpha) = \int_\alpha^\infty t^{\beta-1} e^{-t} dt \tag{2.16}$$

is the incomplete gamma function. Note that (2.15) reduces to (2.12) when $w = 0$.

As in the setup cost problem, setup costs are incurred at the rate $\rho_1\rho_2\kappa/u(x)$ when $W = x$. Therefore, the expected setup cost per unit time is

$$\begin{aligned} \rho_1\rho_2\kappa \int_w^\infty \frac{\alpha^{\beta+1} x^{\beta-1} e^{-\alpha x}}{\Gamma(\beta+1, \alpha w)} dx &= \rho_1\rho_2\alpha\kappa \frac{\Gamma(\beta, \alpha w)}{\Gamma(\beta+1, \alpha w)} \\ &= \frac{\rho_1\rho_2\alpha\kappa}{\beta}\left(1 - \frac{(\alpha w)^\beta e^{-\alpha w}}{\Gamma(\beta+1, \alpha w)}\right) , \end{aligned} \tag{2.17}$$

where the last equality follows from the identity $\beta\Gamma(\beta, \alpha w) = \Gamma(\beta+1, \alpha w) - (\alpha w)^\beta e^{-\alpha w}$. The expected holding cost per unit time is $\bar{C} \int_w^\infty x\pi(x)dx$, where

$$\begin{aligned} \int_w^\infty x\pi(x)dx &= \frac{\alpha^{\beta+1}}{\Gamma(\beta+1, \alpha w)} \int_w^\infty x^{\beta+1} e^{-\alpha x} dx \\ &= \frac{\Gamma(\beta+2, \alpha w)}{\alpha\Gamma(\beta+1, \alpha w)} \\ &= \frac{\beta+1}{\alpha} + \frac{\alpha^\beta w^{\beta+1} e^{-\alpha w}}{\Gamma(\beta+1, \alpha w)} \ . \end{aligned} \tag{2.18}$$

Hence, the expected total cost rate is

$$\frac{\bar{C}}{\alpha}\left(\beta + 1 + \frac{(\alpha w)^{\beta+1} e^{-\alpha w}}{\Gamma(\beta+1, \alpha w)}\right) + \frac{\rho_1\rho_2\alpha\kappa}{\beta}\left(1 - \frac{(\alpha w)^\beta e^{-\alpha w}}{\Gamma(\beta+1, \alpha w)}\right) \ . \tag{2.19}$$

If we define the constant $\bar{\kappa} = \rho_1\rho_2\alpha\kappa/\beta$, then it suffices to minimize

$$\frac{(\bar{C}w - \bar{\kappa})(\alpha w)^\beta e^{-\alpha w}}{\Gamma(\beta+1, \alpha w)} \ . \tag{2.20}$$

Using the fact that $\frac{d}{dw}\Gamma(\beta+1, \alpha w) = -\alpha e^{-\alpha w}(\alpha w)^\beta$, considerable manipulation leads to the following first order optimality condition for w^*:

$$\frac{(\alpha w)^\beta e^{-\alpha w}}{\Gamma(\beta+1, \alpha w)} = \left(1 - \frac{\beta}{\alpha w}\right) + \frac{\bar{C}}{\alpha(\bar{\kappa} - \bar{C}w)} \ . \tag{2.21}$$

Substituting $v/\sqrt{n}$ for w, K/n for κ, and $\sqrt{n}\theta$ for α (see (1.24)) into (2.21) gives

$$\frac{(\theta v)^\beta e^{-\theta v}}{\Gamma(\beta+1, \theta v)} = \left(1 - \frac{\beta}{\theta v}\right) + \frac{\beta\bar{C}}{\theta(\rho_1\rho_2\theta K - \beta\bar{C}v)} \ . \tag{2.22}$$

Although we have not been able to prove the existence of a unique positive root v^* to (2.22), the numerical solution to this equation was well behaved for our test examples.

In summary, for the balanced case with setup times and setup costs, we propose the following scheduling policy for the original problem: *when* $Q_1(t) = 0$ *and* $Q_2(t) \geq \mu_2 v^*$, *switch from class 1 to class 2; when* $Q_2(t) = 0$ *and* $Q_1(t) \geq \mu_1 v^*$, *then switch from class 2 to class 1.* The threshold v^* is found by solving (2.22).

2.4 Analysis of the Diffusion Control Problem: The Imbalanced Case

Notice that (2.8) is nonstandard, in the sense that the drift is unbounded at zero and will be unbounded whenever the control $u(x) = 0$. Nonetheless, we proceed as if standard arguments apply (see, for example, Mandl 1968), and write the Hamilton-Jacobi-Bellman optimality equation for problem (2.8) as

$$\min_{u(x)\in[0,x]} \left\{ c_2\mu_2 x + \frac{\Delta u(x)}{2} - g + \left(\frac{\rho_1\rho_2 s}{u(x)} - c\right) V'(x) + \frac{\sigma^2}{2} V''(x) \right\} = 0 \ . \qquad (2.23)$$

Hence, if we can find a constant g, which is referred to as the *gain*, and a *potential* (relative value) function $V(x)$ that solves (2.23), then the control $u^*(x)$ that minimizes the expression in brackets in (2.23) is optimal and g is the minimal average cost per unit time (independent of initial state). The resulting potential function $V(x)$ represents the cost incurred under the optimal policy when the initial state is x minus the cost incurred under the optimal policy when the initial state is zero. We assume that $V \in \mathbf{C}^2$ and, to avoid notational confusion between the potential function and the unscaled workload process, we employ the first derivative of the potential function, which is denoted by $p(x) = V'(x)$.

Rewriting (2.23) as

$$\min_{u(x)\in[0,x]} \left\{ \frac{\Delta u(x)}{2} + \frac{\rho_1\rho_2 s p(x)}{u(x)} \right\} + c_2\mu_2 x - g - cp(x) + \frac{\sigma^2}{2} p'(x) = 0 \ , \qquad (2.24)$$

we obtain the following first order optimality condition for $u(x)$:

$$u(x) = \sqrt{\frac{2\rho_1\rho_2 s p(x)}{\Delta}} \ . \qquad (2.25)$$

Since greater initial workload implies greater cost, we have $p(x) > 0$ and the function in brackets in (2.24) is convex with respect to $u(x)$. Hence, the optimal control is given by

$$u^*(x) = \min\left\{ x, \ \sqrt{\frac{2\rho_1\rho_2 s p(x)}{\Delta}} \right\} \ . \qquad (2.26)$$

It is interesting to compare (2.26) with the corresponding solution (1.6)–(1.7) in the setup cost problem. The solutions are identical except that the normalized setup cost per cycle κ in (1.6) is replaced by the expected setup time per cycle s multiplied by $p(x)$. Hence, the two optimal controls will be qualitatively similar if the potential function $V(x)$ is linear, which will turn out not to be the case. Thus, solutions to the two problems lead to fundamentally different qualitative behavior.

We assume that $2\rho_1\rho_2 sp(x)/\Delta$ is monotone enough (e.g., p is nondecreasing) and is greater than x^2 as $x \to 0$, so that

$$u^*(x) = \begin{cases} x & \text{if} \quad x \le \widehat{w} \ , \\ \sqrt{\frac{2\rho_1\rho_2 sp(x)}{\Delta}} & \text{if} \quad x \ge \widehat{w} \ , \end{cases} \tag{2.27}$$

where the normalized threshold level $\widehat{w}$ is unknown at this point and satisfies the fixed point equation

$$\widehat{w} = \sqrt{\frac{2\rho_1\rho_2 sp(\widehat{w})}{\Delta}} \ . \tag{2.28}$$

If we substitute (2.27) into (2.23), then the optimality equation reduces to two ordinary differential equations (ODE's) for $p(x)$:

$$\frac{\sigma^2}{2}p'(x) + \left(\frac{\rho_1\rho_2 s}{x} - c\right)p(x) = g - \bar{C}x \quad \text{for} \quad x \in [0, \widehat{w}] \tag{2.29}$$

and

$$\frac{\sigma^2}{2}p'(x) - cp(x) + \sqrt{2\rho_1\rho_2\Delta sp(x)} = g - c_2\mu_2 x \quad \text{for} \quad x \ge \widehat{w} \ . \tag{2.30}$$

The ODE in (2.29) is linear and possesses an explicit solution (that satisfies the properties assumed above). Unfortunately, the ODE in (2.30) is nonlinear and does not appear to admit an analytical solution. Hence, we resort to approximate analytical methods and numerical methods in the remainder of this section.

It is worth noting the similarity between problem (2.8) and the singular control problem for multidimensional Brownian motion analyzed by Cox and Karatzas (1985). Their control problem gives rise to a Bessel process with a controllable additive drift, which leads to a pair of *linear* ODE's analogous to (2.29)–(2.30), and hence to an explicit solution. Our problem can be expressed as a multiplicative, rather than additive, control of a Bessel process with drift, which leads to the intractable nonlinear ODE in (2.30).

We conclude this subsection with an asymptotic result. Although (2.30) cannot be solved analytically, first hitting time arguments can be employed to obtain the asymptotic value of $p(x)$ as $x \to \infty$. A derivation in the Appendix shows that the derivative of the potential function satisfies

$$p(x) = \frac{c_2\mu_2 x}{c} + o(x) \quad \text{as} \quad x \to \infty \ . \tag{2.31}$$

This asymptotic result allows us to see how the control $u^*(x)$ behaves as $x \to \infty$. More specifically, (2.26) and (2.31) imply that

$$\frac{u^*(x)}{\sqrt{x}} \to \sqrt{\frac{2c_2\mu_2\rho_1\rho_2 s}{c\Delta}} \quad \text{as} \quad x \to \infty \ . \tag{2.32}$$

This result is in direct contrast to the solution (1.6)–(1.7) of the setup cost problem, which implies that

$$u^*(x) \to \sqrt{\frac{2\rho_1\rho_2\kappa}{\Delta}} \quad \text{as} \quad x \to \infty \ . \tag{2.33}$$

Equations (2.32)–(2.33) summarize the contrasting qualitative behavior between the solutions to the two problems: $u^*(x)$ grows as $\sqrt{x}$ in the setup time problem and is a constant for large x in the setup cost problem.

2.5 An Approximate Analytical Solution

One of our goals is to find a scheduling policy that performs well and is relatively easy to derive. One possible approach is to derive a policy that is optimal (in heavy traffic) within a certain class of policies. Perhaps the simplest policy to consider is a *single threshold* policy that possesses a single parameter, w: *serve class 1 to exhaustion, and then switch to class 2. Switch from class 2 to class 1 whenever* $W_2(t) = 0$ *or* $W_1(t) \geq w$. The optimal policy for the setup cost problem reduces to the single threshold policy when the parameter w^* in (1.13), and hence v^*, equals zero; see Figure 4. Although it is straightforward to derive the optimal value of the parameter w in heavy traffic, we do not pursue this here, primarily because our asymptotic result (2.32) suggests that the policy is not very close to optimal.

Instead, we investigate another simple class of policies, which we refer to as *asymptotic policies*; these policies can be constructed by patching together the asymptotic result (2.31) with the first part of solution (2.27). In particular, we assume that $u(x) = x$ for x less than or equal to some unknown threshold $\widehat{w}$, and $u(x)/\sqrt{x}$ equals a constant thereafter; hence, we are assuming that the asymptotic result holds not only for very large x, but for all $x \geq \widehat{w}$. Continuity at $\widehat{w}$ gives

$$u(x) = \begin{cases} x & \text{if } x \leq \widehat{w} , \\ \sqrt{\widehat{w}x} & \text{if } x \geq \widehat{w} . \end{cases} \tag{2.34}$$

This control, and hence the resulting scheduling policy, is characterized by a single parameter, the threshold level $\widehat{w}$.

We offer two estimates for $\widehat{w}$ that are of increasing complexity. Both estimates assume that this parameter satisfies the fixed point equation (2.28), and are based on approximating the unknown function $p(x)$ in this equation. The simpler estimate for $\widehat{w}$ employs the asymptotic approximation $p(x) = c_2\mu_2 x/c$ in (2.31), and sets $\widehat{w}$ equal to the solution to the fixed point equation $x = \sqrt{2c_2\mu_2\rho_1\rho_2 s x/(c\Delta)}$, which yields $\widehat{w} = 2c_2\mu_2\rho_1\rho_2 s/(c\Delta)$. The corresponding unscaled threshold level is

$$\widehat{v} = \sqrt{n}\widehat{w} = \frac{2c_2\mu_2\rho_1\rho_2 s}{\Delta(1-\rho)} . \tag{2.35}$$

As in Section 1, when the unscaled total workload V equals y, the control $u^*(x)$ requires the server to serve class 2 until the unscaled class 1 workload $V_1 = \sqrt{n}u^*(y/\sqrt{n})$. Substituting $\widehat{v}/\sqrt{n}$ for $\widehat{w}$ in (2.34) gives

$$\sqrt{n}u^*\left(\frac{y}{\sqrt{n}}\right) = \begin{cases} y & \text{if } y \leq \widehat{v} , \\ \sqrt{\widehat{v}y} & \text{if } y \geq \widehat{v} . \end{cases} \tag{2.36}$$

Translating workloads into queue lengths gives the following scheduling policy: *serve class 1 to exhaustion and then switch to class 2; serve class 2 until*

$$\mu_1^{-1}Q_1(t) \geq \begin{cases} \mu_1^{-1}Q_1(t) + \mu_2^{-1}Q_2(t) & \textit{if} \;\; \mu_1^{-1}Q_1(t) + \mu_2^{-1}Q_2(t) \leq \widehat{v} , \\ \sqrt{\widehat{v}(\mu_1^{-1}Q_1(t) + \mu_2^{-1}Q_2(t))} & \textit{if} \;\; \mu_1^{-1}Q_1(t) + \mu_2^{-1}Q_2(t) \geq \widehat{v} , \end{cases} \tag{2.37}$$

and then switch back to class 1. This policy implies that class 2 is served to exhaustion as long as $\mu_1^{-1}Q_1(t) + \mu_2^{-1}Q_2(t) \leq \widehat{v}$. When $\widehat{v}$ is defined by (2.35), policy (2.37) will be referred to as the *crude asymptotic* policy.

A slightly more refined policy can be derived by *assuming* that $p(x) = ax + b\sqrt{x} + o(\sqrt{x})$ as $x \to \infty$. Substituting this expression into the nonlinear ODE (2.30) and ignoring all $o(\sqrt{x})$

terms leads to

$$p(x) = \frac{c_2\mu_2 x}{c} + \sqrt{\frac{2c_2\mu_2\rho_1\rho_2\Delta s x}{c^3}} + o(\sqrt{x}) \quad \text{as} \quad x \to \infty \ . \tag{2.38}$$

Substituting this expression into the fixed point equation (2.28) yields

$$\frac{2c_2\mu_2\rho_1\rho_2 s}{\Delta c x} + \sqrt{\frac{8c_2\mu_2\rho_1^3\rho_2^3 s^3}{\Delta c^3 x^3}} = 1 \ . \tag{2.39}$$

If we set $z = \sqrt{cx}$, then (2.39) becomes the cubic equation

$$\Delta z^3 - 2c_2\mu_2\rho_1\rho_2 s z - (2\rho_1\rho_2 s)^{3/2}\sqrt{c_2\mu_2\Delta} = 0 \ . \tag{2.40}$$

Since $\sqrt{c\widehat{w}} = \sqrt{(1-\rho)\widehat{v}}$, it follows that the optimal unscaled threshold level $\widehat{v}$ is $z^2/(1-\rho)$, where z solves (2.40). Substituting this quantity into (2.37) yields the *refined asymptotic* policy.

We could go one step further and analyze the heavy traffic performance of the class of asymptotic policies defined in (2.34), and then find the optimal threshold level within this class. Although the expected cost under this class of policies can be evaluated explicitly, the expression for the first derivative of the cost with respect to $\widehat{w}$ is extremely cumbersome, and a symbolic mathematics program would be required to obtain the optimal $\widehat{w}$. We did not carry out this program because our numerical results (in Section 3) indicate that the refined asymptotic policy performs extremely well.

Another possible approach to deriving an approximate analytical solution is the following. Let $p_1(x)$ denote the solution to the linear ODE (2.29), and suppose that we could obtain a solution $p_2(x)$ to the nonlinear ODE (2.30). The two solutions are expressed in terms of the unknown gain g. By (2.27)–(2.28), these two solutions lead to the following system of two equations and two unknowns, $\widehat{w}$ and g:

$$p_1(\widehat{w}) = \frac{\Delta\widehat{w}^2}{2\rho_1\rho_2 s} \quad \text{and} \quad p_2(\widehat{w}) = \frac{\Delta\widehat{w}^2}{2\rho_1\rho_2 s} \ . \tag{2.41}$$

Hence, we could derive an approximate solution to the diffusion control problem by finding an approximate solution to the nonlinear ODE (2.30), and solving (2.41) with the approximate ODE solution used in place of the unknown function $p_2(x)$. We attempted to use perturbation methods to obtain an approximate solution to (2.30), and also tried to derive a series solution, but neither approach yielded a sufficiently accurate solution to the nonlinear ODE.

2.6 An Algorithmic Solution

Since problem (2.8) cannot be solved analytically, we pursue a numerical solution. In particular, the *Markov chain approximation* technique developed by Kushner (1977) will be employed. This method systematically discretizes both time and the state space, and approximates a diffusion control problem by a control problem for a finite state Markov chain. Weak convergence methods have been developed by Kushner and his colleagues to verify that the controlled Markov chain (and its corresponding optimal cost) approximates arbitrarily closely the controlled diffusion process (and its corresponding optimal cost); we refer readers to Kushner and Dupuis (1992) for an up-to-date account of this research area, and will retain most of their notation for ease of reference.

Let h denote the *finite difference interval*, which dictates how finely both the state space and time are discretized. One can consider a sequence of controlled Markov chains indexed by the interval h, and as the value of h becomes smaller the resulting discrete time, finite state Markov chain described below becomes a better approximation of the controlled diffusion process. To numerically solve (2.8), we need to confine the one-dimensional diffusion process X to a bounded region. Since X resides on the nonnegative halfline, the state space of the controlled Markov chain will be $\{0, h, 2h, ..., N-h, N\}$, where N is an integer multiple of h. The approximating Markov chain has nonzero transition probabilities

$$P^h(x, x+h) = \frac{\sigma^2 + 2h\left(\frac{\rho_1\rho_2 s}{u(x)} - c\right)^+}{2\sigma^2 + 2h\left|\frac{\rho_1\rho_2 s}{u(x)} - c\right|} \tag{2.42}$$

and

$$P^h(x, x-h) = \frac{\sigma^2 + 2h\left(\frac{\rho_1\rho_2 s}{u(x)} - c\right)^-}{2\sigma^2 + 2h\left|\frac{\rho_1\rho_2 s}{u(x)} - c\right|} \tag{2.43}$$

on the interior of the state space, and the time intervals, or *interpolation intervals*, are of length

$$\Delta t^h = \frac{h^2}{\sigma^2 + h\left|\frac{\rho_1\rho_2 s}{u(x)} - c\right|} . \tag{2.44}$$

Two issues need to be addressed to obtain our approximating controlled Markov chain: (i) for an ergodic cost problem, the interpolation interval Δt^h needs to be independent of the state x and control $u(x)$ (see Kushner and Dupuis, page 209), and (ii) the behavior of the Markov chain at the boundary states $x = 0$ and $x = N$. To deal with the first issue, we define

$$\dot{Q}^h = \max_{x, u(x)} \sigma^2 + h\left|\frac{\rho_1\rho_2 s}{u(x)} - c\right| . \tag{2.45}$$

Since the smallest nonzero value of $u(x)$ is h, we let

$$Q^h = \sigma^2 + |\rho_1\rho_2 s - ch| , \tag{2.46}$$

and define the new nonzero interior transition probabilities

$$\bar{P}^h(x, x+h) = \frac{\sigma^2 + 2h\left(\frac{\rho_1\rho_2 s}{u(x)} - c\right)^+}{2Q^h} , \tag{2.47}$$

$$\bar{P}^h(x, x-h) = \frac{\sigma^2 + 2h\left(\frac{\rho_1\rho_2 s}{u(x)} - c\right)^-}{2Q^h} \tag{2.48}$$

and

$$\bar{P}^h(x, x) = 1 - \frac{\left(\sigma^2 + h\left|\frac{\rho_1\rho_2 s}{u(x)} - c\right|\right)}{Q^h} , \tag{2.49}$$

and the new interpolation interval

$$\Delta t^h = \frac{h^2}{Q^h} . \tag{2.50}$$

Now we consider the boundary states. A reflecting boundary is employed at the origin. However, the Markov chain approximation method assumes that $\Delta t^h = 0$ for a reflecting boundary state. Hence, since the interpolation interval Δt^h takes on a value different than

(2.50) at the origin, this boundary state must be eliminated. We define the transition probability (see page 212 of Kushner and Dupuis)

$$\tilde{P}^h(h,h) = 1 - \bar{P}^h(h,2h) \ . \tag{2.51}$$

We also impose a reflecting boundary at state N, and define the transition probability

$$\tilde{P}^h(N-h, N-h) = 1 - \bar{P}(N-h, N-2h) \ . \tag{2.52}$$

Although the reflecting barrier at N is artificial in the sense that $P(N, N+h)$ would be positive if the boundary was chosen to be larger than N, the effect of this approximation should be negligible if the boundary state N is sufficiently large, and consequently visited sufficiently infrequently. In summary, our approximating Markov chain has state space $\{h, 2h, ..., N-2h, N-h\}$, interpolation interval defined by (2.50), and nonzero transition probabilities $\tilde{P}^h(x,y)$ defined by (2.51)-(2.52) and

$$\tilde{P}^h(x,y) = \bar{P}^h(x,y) \quad \text{otherwise} \ . \tag{2.53}$$

The dynamic programming optimality equation for the controlled Markov chain is given by (see equation 5.3 on page 204 of Kushner and Dupuis)

$$V(x) = \sum_y \tilde{P}^h(x,y)V(y) + (c_2\mu_2 x + \frac{\Delta u(x)}{2} - g)\Delta t^h \quad \text{for} \quad x = h, 2h, ..., N-h \ . \tag{2.54}$$

We are now in a position to describe the policy improvement algorithm that solves the Markov chain control problem. First, an initial policy is chosen, and the natural initial policy is the exhaustive policy $u(x) = x$ for $x = h, ..., N-h$. In the policy improvement step, we solve

$$\min_{u(x)\in[0,x]} \Big[\sum_y \tilde{P}^h(x,y)V(y) + (c_2\mu_2 x + \frac{\Delta u(x)}{2})\Delta t^h\Big] \ . \tag{2.55}$$

If the drift $\rho_1\rho_2 s/u^*(x) - c$ is positive then

$$u^*(x) = \min\left\{x, \sqrt{\frac{2\rho_1\rho_2 s[V(x+h) - V(x)]}{\Delta h}}\right\} , \tag{2.56}$$

and if the drift is negative then

$$u^*(x) = \min\left\{x, \sqrt{\frac{2\rho_1\rho_2 s[V(x) - V(x-h)]}{\Delta h}}\right\} . \tag{2.57}$$

Notice that (2.56)-(2.57) converges to (2.26) as $h \to 0$, as expected. The policy improvement algorithm terminates when the new and old controls coincide.

The mapping from a numerical control $u^*(x)$ to a scheduling policy is less straightforward than when an analytical control is obtained. Since we are solving (2.8) numerically, there is no way to develop a proposed scheduling policy that is independent of the heavy traffic scaling parameter n. The drift equals $\rho_1\rho_2 s/u(x) - \sqrt{n}(1-\rho)$, and a value of n must be chosen in order to compute a numerical solution to the Markov chain control problem. In our numerical tests, we choose the integer n that makes c as close to one as possible. The numerical solution $u^*(x)$ to the Markov chain control problem is defined on the points $\{h, ..., N-h\}$, and an interpolation method must be employed to define a solution $u^*(x)$ on the interval $[0, N]$. We use a linear interpolation to obtain this continuous solution. Equation (1.20) is then used to map the continuous solution $u^*(x)$ into a policy for the unscaled workload V. Finally, we use $\mu_i V_i = Q_i$ to obtain a solution in terms of the original queue length process (Q_1, Q_2).

3 COMPUTATIONAL STUDY

A numerical experiment is undertaken in this section to investigate the effectiveness of our proposed policies. Three problems are considered: the setup cost problem addressed in Section 1, the balanced system with setup costs and setup times analyzed in Section 2.3, and the imbalanced setup time problem. For each problem, we compare the performance of the optimal policy, a straw policy, and one or more proposed policies. The straw policy for the first two problems is the *patient exhaustive* policy: switch out of a class whenever it is exhausted and at least one customer of the other class is present. The straw policy for the imbalanced setup time problem is the *exhaustive* policy: *serve each class to exhaustion and then switch class.* These straw policies are studied because they are simple to implement in practice and are commonly found in the literature. The value iteration algorithm is used to derive optimal policies and to evaluate the cost of the proposed and straw policies. We report the *suboptimality* of the proposed and straw policies, where a

$$\text{policy's suboptimality} = \frac{\text{policy's cost} - \text{optimal cost}}{\text{optimal cost}} \times 100\% \ . \tag{3.1}$$

The experiment consists of 120 test cases, including 48 cases of the setup cost problem, 45 cases of the symmetric system with setup costs and times, and 27 cases of the imbalanced setup time problem. To simplify the computational effort required to obtain the optimal policy, we assume that all interarrival times, service times and setup times are exponential. For each test case, we set the service rates $\mu_1 = \mu_2 = 1$ and the arrival rates $\lambda_1 = \lambda_2 = \rho/2$, and let the holding cost $c_2 = 1$. Hence, each test case is characterized by the holding cost c_1 of the high priority class, the setup cost per cycle K and/or the expected setup time per cycle s, and the traffic intensity ρ. This experimental design allows us to isolate the impact of three key parameters: the difference in $c\mu$ values between classes, the setup and the traffic intensity.

3.1 The Setup Cost Results

The 48 test cases are generated by considering all combinations of the parameter values in Table I. Hence, 12 cases are balanced, that is, $c_1\mu_1 = c_2\mu_2$, and 36 cases are imbalanced. Although our proposed policy, which is described in Subsection 1.8, was derived under heavy traffic conditions, the policy is tested with traffic intensities as low as 0.5, and with setup costs as small as one-tenth of the holding cost c_1.

	Holding Cost c_1	Setup Cost K	Traffic Intensity ρ
Balanced	1	–	–
Low	1.5	2	0.5
Medium	5	10	0.7
High	10	20	0.9
Very High	–	200	–

Table I: The 48 test cases for the setup cost problem.

Additional notation is required to write down the dynamic programming optimality equations from which the optimal policy is derived; we occasionally reuse earlier notation that will not be needed again, which should cause no confusion. Let x_k denote the number of customers of class k in the system, i be the class that is currently set up, and i^c be the other class. Let $x = (x_1, x_2)$, $\bar{\mu} = \max(\mu_1, \mu_2)$, $\Lambda = \lambda_1 + \lambda_2 + \bar{\mu}$, $e_1 = (1, 0)$, $e_2 = (0, 1)$, and $V(x, i)$ denote the optimal value function. Then the optimality equations are

$$\begin{aligned} V(x,i) \quad = \quad & \frac{1}{\Lambda}\left[\sum_{k=1}^{2} c_k x_k + \sum_{k=1}^{2} \lambda_k V(x+e_k, i) + \min\Big\{\mu_i V([x-e_i]^+ + i) + (\bar{\mu}-\mu_i)V(x,i)\ ,\right. \\ & \left. \bar{\mu}V(x,i), \frac{K}{2} + \mu_{i^c} V([x-e_{i^c}]^+ + i^c) + (\bar{\mu}-\mu_{i^c})V(x,i^c)\Big\}\right] . \qquad (3.2) \end{aligned}$$

The three terms inside the minimum argument represent the three respective options of serving the class that is currently set up, idling, and switching and immediately serving the other class. The state space was truncated in the value iteration algorithm, and larger and larger state spaces were tested until the results were insensitive to increasing the state space. State spaces up to 90 by 90 and up to 4000 value iterations were required to achieve three digit accuracy of the suboptimalities.

Holding Cost c_1	Setup Cost K	Traffic Intensity ρ	Suboptimality of Proposed Policy	Suboptimality of Straw Policy
1	2	0.5	0.0%	0.0%
1	2	0.7	0.0%	0.0%
1	2	0.9	0.4%	0.4%
1	10	0.5	10.8%	0.0%
1	10	0.7	0.0%	0.0%
1	10	0.9	0.4%	0.4%
1	20	0.5	12.4%	5.9%
1	20	0.7	0.0%	1.6%
1	20	0.9	0.3%	0.3%
1	200	0.5	5.0%	128.2%
1	200	0.7	1.1%	90.3%
1	200	0.9	0.0%	21.9%

Table II: Results for the setup cost problem: balanced cases.

Tables II and III provide the suboptimalities of the proposed policy and the patient exhaustive policy for the 12 balanced cases and the 36 imbalanced cases, respectively. These results are summarized in Tables IV and V to isolate the effects of the three key parameters. Each entry in Tables IV and V represents the average suboptimality of the 12 test cases (16 cases for the traffic intensity) that have a particular parameter equal to a particular value.

Holding Cost c_1	Setup Cost K	Traffic Intensity ρ	Suboptimality of Proposed Policy	Suboptimality of Straw Policy
1.5	2	0.5	0.6%	2.8%
1.5	2	0.7	0.2%	7.4%
1.5	2	0.9	0.7%	17.2%
1.5	10	0.5	1.4%	0.4%
1.5	10	0.7	0.1%	2.6%
1.5	10	0.9	0.3%	12.1%
1.5	20	0.5	3.1%	3.4%
1.5	20	0.7	0.6%	1.6%
1.5	20	0.9	0.4%	9.3%
1.5	200	0.5	13.1%	110.4%
1.5	200	0.7	3.0%	75.4%
1.5	200	0.9	0.7%	18.2%
5	2	0.5	0.0%	24.0%
5	2	0.7	0.0%	50.4%
5	2	0.9	0.3%	115.4%
5	10	0.5	0.0%	11.7%
5	10	0.7	0.0%	33.5%
5	10	0.9	4.7%	98.6%
5	20	0.5	4.6%	7.8%
5	20	0.7	0.4%	21.8%
5	20	0.9	1.0%	82.1%
5	200	0.5	27.8%	70.4%
5	200	0.7	14.3%	42.5%
5	200	0.9	1.5%	34.0%
10	2	0.5	0.0%	34.2%
10	2	0.7	0.0%	74.3%
10	2	0.9	0.2%	197.1%
10	10	0.5	0.0%	24.2%
10	10	0.7	0.0%	59.5%
10	10	0.9	0.2%	178.7%
10	20	0.5	1.5%	17.3%
10	20	0.7	0.1%	45.9%
10	20	0.9	0.2%	158.2%
10	200	0.5	30.7%	51.2%
10	200	0.7	11.9%	33.8%
10	200	0.9	1.6%	63.6%

Table III: Results for the setup cost problem: imbalanced cases.

	Holding Cost c_1	Setup Cost K	Traffic Intensity ρ
Balanced	2.5%	–	–
Low	2.0%	0.1%	6.9%
Medium	4.5%	1.4%	2.0%
High	3.8%	2.1%	0.7%
Very High	–	9.2%	–

Overall Average Suboptimality = 3.2%

Table IV: Average suboptimality of the proposed policy: setup cost problem.

	Holding Cost c_1	Setup Cost K	Traffic Intensity ρ
Balanced	1.2%	–	–
Low	21.7%	43.6%	30.8%
Medium	49.3%	35.1%	33.8%
High	78.1%	29.6%	63.0%
Very High	–	61.7%	–

Overall Average Suboptimality = 42.5%

Table V: Average suboptimalitity of the straw policy: setup cost problem.

The proposed policy performs remarkably well: although its overall average suboptimality is 3.2%, its suboptimality is less than 1% for 31 of the 48 test cases, and is less than 3.1% for 38 of the 48 test cases. Under these 38 cases, comparison of the optimal switching curves (not displayed here) with the proposed switching curves shows that the two curves differ on at most several states in the state space. In particular, the vertical boundary in Figure 4 is very close to optimal in the imbalanced case.

Recall that many of the 48 test cases grossly violate the heavy traffic conditions stated in Subsection 1.2, which requires heavy loading and much larger setup costs than holding costs. Perhaps the case that comes closest to satisfying these conditions is $c_1 = 1, K = 200$ and $\rho = 0.9$, where the proposed policy is optimal. As in previous heavy traffic work (see, for example, Chevalier and Wein 1993), the performance of the proposed policy is relatively insensitive (within a certain range) to the heavy traffic assumptions underlying the analysis: for the 12 balanced cases, the suboptimality of the proposed policy deteriorates to 5-12% when ρ drops to 0.5, and the policy performs well in all other cases. For the 36 imbalanced cases, the suboptimality increases to as high as 30% when the setup cost is very large, the traffic intensity is low, and the holding cost is high. In fact, most of the suboptimality in the 48 cases occurs when $K = 200$: the average suboptimality for the 36 cases in which $K < 200$ is 0.8%. In summary, the proposed policy performs very well over a broad range of parameter values, and then deteriorates outside of this range.

We should also point out that the derived values of $\lceil v^* \rceil$ and $\lceil v_{\text{app}} \rceil$ from (1.23) and (1.25) are identical in 10 of the 12 balanced cases (they differ by one in the other two cases), where $\lceil x \rceil$ is the smallest integer greater than or equal to x. The quantity $\lceil v^* \rceil$ ranges from one to three, and differs from the optimal threshold level by two in one case where $K = 200$, and by at most one in the other 11 balanced cases. For the 36 imbalanced cases, $\lceil v^* \rceil$ in (1.22) ranges from one to four, and $\lceil \tilde{v} \rceil$ in (1.21) averages 8/3 and varies from one to 13.

The patient exhaustive policy, with an average suboptimality of 42.5%, is clearly outperformed by the proposed policy. Not surprisingly, its performance degrades significantly as the holding cost c_1 and the traffic intensity ρ increase. Its suboptimality appears to be convex in the setup cost K. As K initially increases, holding costs play less of a role, and its suboptimality decreases; however, for very large K, the optimal policy idles much more than the patient exhaustive policy, particularly when the traffic intensity is low.

3.2 Results for the Balanced System with Setup Costs and Setup Times

Table VI describes the 45 test cases for the balanced (that is, $c_1 = 1$) system with setup costs and setup times. The proposed policy for these test cases is defined at the end of Subsection 2.3; this policy and the patient exhaustive policy, which is the straw policy for these test cases, coincide when v^* in (2.22) satisfies $\lceil v^* \rceil = 1$. The dynamic programming optimality equations for this problem are

$$\begin{aligned} V(x,i) &= \frac{1}{\Lambda}\Bigg[\sum_{k=1}^{2} c_k x_k + \sum_{k=1}^{2} \lambda_k V(x+e_k,i) + \min\Big\{\mu_i V(x-e_i,i) + (\bar{\mu}_s - \mu_i)V(x,i)\ , \\ &\qquad \bar{\mu}_s V(x,i), \frac{K}{2} + s^{-1}V(x,i^c) + (\bar{\mu}_s - s^{-1})V(x,i)\Big\}\Bigg]\ , \end{aligned} \tag{3.3}$$

where $\bar{\mu}_s$ is defined as $\max(s^{-1}, \mu_1, \mu_2)$ and $\Lambda = \lambda_1 + \lambda_2 + \bar{\mu}_s$.

	Setup Cost K	Setup Time s	Traffic Intensity ρ
Zero	0	–	–
Low	2	2	0.5
Medium	10	10	0.7
High	20	20	0.9
Very High	200	–	–

Table VI: The 45 test cases for the balanced system.

The proposed policy and the patient exhaustive policy coincide for all but one of the 45 test cases. Moreover, these policies are optimal for 36 of the test cases. Table VII displays the suboptimality of both policies for the remaining nine cases. As in the setup cost problem, the policies degrade when the setup cost is very large and the traffic intensity is low. Over the 45 test cases, the average suboptimality of the proposed policy is 3.4%, and the average suboptimality of the patient exhaustive policy is 4.9%.

Setup Cost K	Setup Time s	Traffic Intensity ρ	Suboptimality of Proposed Policy	Suboptimality of Straw Policy
20	2	0.5	4.1%	4.1%
20	2	0.7	0.5%	0.5%
200	2	0.5	41.7%	108.5%
200	2	0.7	62.2%	62.2%
200	2	0.9	7.4%	7.4%
200	10	0.5	26.7%	26.7%
200	10	0.7	5.6%	5.6%
200	20	0.5	6.2%	6.2%
200	20	0.7	0.4%	0.4%

Table VII: Results for the balanced system.

3.3 The Imbalanced Setup Time Results

Table VIII enumerates the 27 test cases for the imbalanced (that is, $c_1 > 1$) problem with setup times. Recall that we derived three scheduling policies for this problem: the crude asymptotic policy defined in (2.35) and (2.37), the refined asymptotic policy defined in (2.37) and (2.40), and the policy constructed from the algorithmic solution described in Section 2.6. We only provide detailed results of the refined asymptotic policy, and briefly summarize the results of the other two policies. The detailed results for the refined asymptotic policy and the exhaustive policy, which is the straw policy for these test cases, are given in Table IX, and summarized in Tables X and XI.

	Holding Cost c_1	Setup Time s	Traffic Intensity ρ
Low	1.5	2	0.5
Medium	5	10	0.7
High	10	20	0.9

Table VIII: The 27 test cases for the imbalanced system with setup times.

The refined asymptotic policy performs very impressively on these test cases. The suboptimality is never above 5% and the average suboptimality over the 27 test cases is 1.5%. The average suboptimality for the crude asymptotic policy is 11.8%, and hence the $\sqrt{x}$ term added to $p(x)$ in (2.38) considerably improves the asymptotic policy. The policy based on the Markov chain approximation algorithm in Section 2.6 (with heavy traffic parameter $n = 100$ and finite difference interval $h = 0.1$) also performs very well; it is very close to optimal when the traffic intensity is high, and its average suboptimality is 2.3%. In contrast, the suboptimality for the exhaustive policy averages 8.7%; not surprisingly, the policy's performance degrades when the holding cost c_1 is large and the setup times are small.

Holding Cost c_1	Setup Time s	Traffic Intensity ρ	Suboptimality of Proposed Policy	Suboptimality of Straw Policy
1.5	2	0.5	1.9%	0.1%
1.5	2	0.7	0.3%	0.2%
1.5	2	0.9	0.0%	0.3%
1.5	10	0.5	2.1%	0.9%
1.5	10	0.7	0.2%	0.1%
1.5	10	0.9	0.0%	0.1%
1.5	20	0.5	1.2%	0.4%
1.5	20	0.7	0.1%	0.0%
1.5	20	0.9	0.0%	0.6%
5	2	0.5	1.4%	8.9%
5	2	0.7	0.2%	14.4%
5	2	0.9	1.6%	19.4%
5	10	0.5	4.9%	5.6%
5	10	0.7	0.5%	3.5%
5	10	0.9	0.2%	3.9%
5	20	0.5	5.0%	4.1%
5	20	0.7	0.5%	2.0%
5	20	0.9	0.1%	4.4%
10	2	0.5	1.4%	21.5%
10	2	0.7	1.8%	29.9%
10	2	0.9	4.5%	40.3%
10	10	0.5	4.7%	12.3%
10	10	0.7	0.8%	9.8%
10	10	0.9	0.7%	12.2%
10	20	0.5	4.8%	9.7%
10	20	0.7	0.4%	6.3%
10	20	0.9	0.6%	22.9%

Table IX: Results for the imbalanced setup time problem.

	Holding Cost c_1	Setup Time s	Traffic Intensity ρ
Low	0.6%	1.5%	3.1%
Medium	1.6%	1.6%	0.5%
High	2.2%	1.4%	0.9%

Overall Average Suboptimality = 1.5%

Table X: Average suboptimalitity of the proposed policy: setup time problem.

	Holding Cost c_1	Setup Time s	Traffic Intensity ρ
Low	0.3%	15.0%	7.0%
Medium	7.4%	5.4%	7.4%
High	18.3%	5.6%	11.6%

Overall Average Suboptimality = 8.7%

Table XI: Average suboptimalitity of the straw policy: setup time problem.

4 CONCLUDING REMARKS

Using heavy traffic approximations, we analyze a dynamic scheduling problem for a two-class queue with either setup costs or setup times. As in previous heavy traffic scheduling studies, these approximations yield control problems that are more amenable to analysis than the original queueing control problems. Our analysis yields a simple two-parameter policy for the setup cost problem, where one parameter is found in closed form and the other is a solution to a specified equation. Although the diffusion control problem that approximates the setup time problem in heavy traffic is not explicitly solvable, a scheduling policy is constructed from an asymptotic result. We derive some fundamental insights into the nature of the optimal policies for these two analytically intractable problems, and computational results indicate that our proposed policies are close to optimal over a broad range of parameter values, including some cases where the heavy traffic conditions are severely violated. An interesting implication of our analysis is that setup cost and setup time problems lead to fundamentally different qualitative solutions. Setup times eat into capacity in a nonlinear fashion, and hence setup costs cannot be used as a surrogate for setup times, as is sometimes done in deterministic scheduling problems with setups (see, for example, the survey paper by Elmaghraby 1978).

Research is ongoing in two areas. A system with two classes is of limited practical interest, and we are currently analyzing the general multiclass problem. Also, a companion paper is in preparation on the *make-to-stock* version of the problem; here, the queueing system produces units in anticipation of customer arrivals, and completed units enter a finished goods inventory, which in turn services actual customer demand. This problem is a stochastic version of the classic Economic Lot Scheduling Problem (see Elmaghraby). The make-to-stock problem is more difficult to analyze than the polling problem because of the nonlinear cost structure and the lack of a natural boundary at the origin.

Acknowledgment. The second author was supported by a grant from the Leaders for Manufacturing Program at MIT and NSF grant DDM-9057297. We are grateful to David Markowitz for performing the numerical computations reported in Section 3. We would like to thank Mike Harrison for suggesting the diffusion control approach to the setup time problem, and Larry Shepp for helpful discussions related to the diffusion control problem.

APPENDIX

The goal in this appendix is to show that

$$\lim_{x\to\infty} x^{-1}p(x) = \frac{c_2\mu_2}{c} ,$$

which is equivalent to (2.31). Since

$$p(x) = \lim_{\delta\to 0} \delta^{-1}[V(x+\delta) - V(x)] , \tag{A.1}$$

we want to show that

$$\lim_{x\to\infty}\lim_{\delta\to 0} \frac{V(x+\delta) - V(x)}{x\delta} = \frac{c_2\mu_2}{c} . \tag{A.2}$$

Thus we consider the quantity $V(x+\delta) - V(x)$. We can write

$$V(x+\delta) - V(x) = E_{x+\delta}\left[\int_0^{T_x}\left(c_2\mu_2 X(t) + \frac{\Delta u^*(X(t))}{2} - g\right)dt\right] , \tag{A.3}$$

where T_x is the first hitting time of x for the $(\rho_1\rho_2 s/u^*(x) - c, \sigma^2)$ diffusion process X, and the expectation is with respect to the initial state $x+\delta$. Combining (A.1) and (A.3) yields

$$p(x) = \lim_{\delta\to 0}\frac{1}{\delta}E_{x+\delta}\left[\int_0^{T_x}\left(c_2\mu_2 X(t) + \frac{\Delta u^*(X(t))}{2} - g\right)dt\right] . \tag{A.4}$$

To obtain the desired result, we need to first show that

$$\frac{u^*(x)}{x} \to 0 \quad \text{as} \quad x\to\infty \tag{A.5}$$

and

$$u^*(x) \to \infty \quad \text{as} \quad x\to\infty . \tag{A.6}$$

These two asymptotic results will be derived in turn. Throughout this appendix we make the intuitively reasonable assumption that $u(x)$ is nondecreasing in x.

We prove (A.5) by contradiction, and hence initially assume that $\overline{\lim}_{x\to\infty} x^{-1}u^*(x) > 0$. Since $u^*(x) \in [0, x]$ for all $x \geq 0$, it follows that

$$p(x) \leq \overline{\lim_{\delta\to 0}}\left\{\frac{(c_2\mu_2 + \frac{\Delta}{2})}{\delta}E_{x+\delta}\left[\int_0^{T_x} X(t)dt\right] - \frac{g}{\delta}E_{x+\delta}[T_x]\right\} . \tag{A.7}$$

The assumed monotonicity of $u^*(x)$ yields $u^* \to \infty$, so that the drift of $X(t)$ satisfies

$$\mu(x) = \frac{\rho_1\rho_2 s}{u^*(x)} - c \to -c \quad \text{as} \quad x\to\infty . \tag{A.8}$$

Take x_0 large enough so that $\mu(x_0) \leq -\frac{c}{2}$. Note that $\mu(x) \leq -\frac{c}{2}$ for $x \geq x_0$. Let $\tilde{X}$ denote a $(-\frac{c}{2}, \sigma^2)$ Brownian motion, and $\tilde{T}_x$ its first passage time. For $x \geq x_0$, it follows that the integral in (A.7) has the bound

$$E_{x+\delta}\left[\int_0^{T_x} X(t)dt\right] \leq xE_{x+\delta}[\tilde{T}_x] + E_\delta\left[\int_0^{\tilde{T}_0}\tilde{X}(t)dt\right] , \tag{A.9}$$

where $\tilde{T}_0$ is the first passage time to zero for a $(-\frac{c}{2}, \sigma^2)$ Brownian motion.

To evaluate the last term in (A.9), let

$$h(\delta) = E_\delta \left[\int_0^{\tilde{T}_{0\wedge b}} \tilde{X}(t)dt \right] , \tag{A.10}$$

where $\tilde{T}_{0\wedge b}$ denotes the first hitting time for $\tilde{X}$ to either 0 or b. This function satisfies the ordinary differential equation (c.f. Karlin and Taylor)

$$-\frac{c}{2}h'(\delta) + \frac{\sigma^2}{2}h''(\delta) = -\delta , \tag{A.11}$$

subject to the boundary conditions $h(0) = h(b) = 0$, which yields

$$h(\delta) = \frac{2\sigma^2\delta}{c^2} + \frac{\delta^2}{c} + \frac{2(\sigma^2 b + b^2 c)(1 - e^{c\delta/\sigma^2})}{c^2(e^{cb/\sigma^2} - 1)} . \tag{A.12}$$

Therefore,

$$E_\delta \left[\int_0^{\tilde{T}_0} \tilde{X}(t)dt \right] = \lim_{b\to\infty} h(\delta) = \frac{2\sigma^2\delta}{c^2} + \frac{\delta^2}{c} . \tag{A.13}$$

Since

$$E_{x+\delta}[\tilde{T}_x] = \frac{2\delta}{c} , \tag{A.14}$$

it follows from (A.7), (A.9) and (A.13) that as $x \to \infty$,

$$p(x) \le \lim_{\delta\to 0} \left(\frac{c_2\mu_2 + \frac{\Delta}{2}}{\delta} \right) \left(\frac{2x\delta}{c} + \frac{2\sigma^2\delta}{c^2} + \frac{\delta^2}{c} \right) = \left(c_2\mu_2 + \frac{\Delta}{2} \right) \left(\frac{2x}{c} + \frac{2\sigma^2}{c^2} \right) . \tag{A.15}$$

Since $p(x)/x^2 \to 0$ as $x \to \infty$, by (2.26) we have $u^*(x)/x \to 0$ as $x \to \infty$, which is a contradiction; hence, (A.5) has been shown. An immediate consequence of (A.5) is

$$\frac{c_2\mu_2 x + \frac{\Delta u^*(x)}{2}}{x} \to c_2\mu_2 \quad \text{as} \quad x \to \infty . \tag{A.16}$$

We next show (A.6), again by contradiction. Since we have assumed $u^*(x)$ nondecreasing, assuming that (A.6) does not hold is equivalent to assuming that $u^*(x)$ approaches some finite constant as $x \to \infty$, which we denote by $u^*(\infty)$. For large x, $X(t)$ behaves as a (μ, σ^2) Brownian motion, where $\mu = \rho_1\rho_2 s/u^*(\infty) - c$ could be of either sign. From (A.4) and the fact that $\rho_1\rho_2 s/u^*(\infty) \le \rho_1\rho_2 s/u^*(x)$ we obtain

$$p(x) \ \ge \ \lim_{\delta\to 0} \frac{c_2\mu_2}{\delta} E_{x+\delta} \left[\int_0^{T_x} X(t)dt \right] - \lim_{\delta\to 0} \frac{g}{\delta} E_{x+\delta}[T_x] \tag{A.17}$$

$$\ge \ \lim_{\delta\to 0} \frac{c_2\mu_2 x - g}{\delta} E_\delta[T_0] , \tag{A.18}$$

where T_0 is the first hitting time for a Brownian motion with drift μ and variance σ^2. If $\mu \ge 0$, then $E_\delta[T_0] = \infty$, and if $\mu < 0$, then $E_\delta[T_0] = -\delta/\mu$. Hence,

$$\lim_{x\to\infty} p(x) \ge \lim_{x\to\infty} \frac{c_2\mu_2 x - g}{|\mu|} = \infty . \tag{A.19}$$

Equations (A.19) and (2.26) imply that $u^*(x) \to \infty$, which yields the desired contradiction.

Armed with (A.5) and (A.6), we can now show (A.2). Equation (A.3) can be rewritten as

$$V(x+\delta) - V(x) = E_{x+\delta}\left[\int_0^{T_x} c_2\mu_2 X(t)dt\right] + E_{x+\delta}\left[\int_0^{T_x} \frac{\Delta u^*(X(t))}{2}dt\right] - gE_{x+\delta}[T_x] . \quad (A.20)$$

Since (A.6) implies (A.8), equations (A.9) and (A.14) implies that for $x \geq x_0$,

$$\frac{gE_{x+\delta}[T_x]}{\delta x} \leq \frac{2g}{cx} ,$$

which converges to zero as $x \to \infty$. Let

$$\epsilon_x = \sup_{z \geq x} \frac{u^*(z)}{z} .$$

By (A.5), $\epsilon_x \to 0$ as $x \to \infty$. For $x \geq x_0$ we can write

$$E_{x+\delta}\left[\int_0^{T_x} u^*(X(t))dt\right] \leq \epsilon_x E_{x+\delta}\left[\int_0^{T_x} X(t)dt\right] \leq \epsilon_x \left[\frac{2\delta x}{c} + \frac{2\sigma^2\delta}{c^2} + \frac{\delta^2}{c}\right] , \quad (A.21)$$

where the last inequality follows from (A.9), (A.13) and (A.14). Since $\epsilon_x \to 0$ as $x \to \infty$, it is clear that

$$\lim_{x\to\infty}\lim_{\delta\to 0} \frac{1}{x\delta} E_{x+\delta}\left[\int_0^{T_x} \frac{\Delta u^*(X(t))}{2}dt\right] = 0 .$$

We are, finally, faced with the first term on the right-hand side of (A.20), which is the only one that does not vanish. Fix x and let $\tilde{X}^{(x)}(t)$ denote a Brownian motion with (constant) drift $\mu(x) = \rho_1\rho_2 s/u^*(x) - c$, and (constant) variance σ^2. Let $\tilde{T}^{(x)}$ denote the first passage times for this process. As in (A.9), the monotonicity of $u^*(x)$ implies that

$$\begin{aligned} &xE_{x+\delta}[\tilde{T}_x^{(\infty)}] + E_\delta\left[\int_0^{\tilde{T}_0^{(\infty)}} \tilde{X}^{(\infty)}(t)dt\right] \\ &\qquad \leq E_{x+\delta}\left[\int_0^{T_x} X(t)dt\right] \leq xE_{x+\delta}[\tilde{T}_x^{(x)}] + E_\delta\left[\int_0^{\tilde{T}_0^{(x)}} \tilde{X}^{(x)}(t)dt\right] , \end{aligned}$$

where $\tilde{X}^{(\infty)}$ is a Brownian motion with drift $-c$ and variance σ^2. Following the analysis that led to (A.13) and (A.14), we obtain

$$\frac{x\delta}{c} + \frac{\sigma^2\delta}{2c^2} + \frac{\delta^2}{2c} \leq E_{x+\delta}\left[\int_0^{T_x} X(t)dt\right] \leq \frac{x\delta}{\mu(x)} + \frac{\sigma^2\delta}{2\mu^2(x)} + \frac{\delta^2}{2\mu(x)} . \quad (A.22)$$

Since $\mu(x) \to -c$ as $x \to \infty$ by (A.6), we have

$$\lim_{x\to\infty}\lim_{\delta\to 0} \frac{E_{x+\delta}\left[\int_0^{T_x} X(t)dt\right]}{x\delta} = \frac{1}{c} ,$$

which yields

$$\lim_{x\to\infty}\lim_{\delta\to 0} \frac{V(x+\delta) - V(x)}{x\delta} = \frac{c_2\mu_2}{c}$$

by (A.20). This is what we set out to show.

REFERENCES

Bertsimas, D. and H. Xu. 1993. Optimization of Polling Systems and Dynamic Vehicle Routing Problems on Networks. Sloan School of Management, MIT, Cambridge, MA.

Boxma, O. J., H. Levy and J. A. Westrate. 1991. Efficient Visit Frequencies for Polling Tables: Minimization of Waiting Cost. *Queueing Systems* **9**, 133-162.

Boxma, O. J. and H. Takagi. 1992. Editors, Special Issue on Polling Systems, *Queueing Systems* **11** (1 and 2).

Browne, S. and U. Yechiali. 1989. Dynamic Priority Rules for Cyclic-Type Queues. *Advances in Applied Probability* **21**, 432-450.

Chevalier, P. B. and L. M. Wein. 1993. Scheduling Networks of Queues: Heavy Traffic Analysis of a Multistation Closed Network. *Operations Research*, July-August.

Coffman, E. G., Jr., A. A. Puhalskii and M. I. Reiman. 1993. Polling Systems with Zero Switchover Times: A Heavy-Traffic Averaging Principle. Submitted for publication.

Coffman, E. G., Jr., A. A. Puhalskii and M. I. Reiman. 1994. Polling Systems with Switchover Times: A Heavy-Traffic Averaging Principle. In preparation.

Cox, R. M. and I. Karatzas. 1985. Stationary Control of Brownian Motion in Several Dimensions. *Advances in Applied Probability* **17**, 531-561.

Duenyas, I. and M. P. Van Oyen. 1993. Stochastic Scheduling of Parallel Queues with Setup Costs. Technical Report 93-40, Department of Industrial and Operations Engineering, U. of Michigan, Ann Arbor, MI.

Elmaghraby, S. E. 1978. The Economic Lot Scheduling Problem (ELSP): Review and Extensions. *Management Science* **24**, 587-598.

Foschini, G. J. 1977. On Heavy Traffic Diffusion Analysis and Dynamic Routing in Packet Switched Networks. In *Computer Performance*, K. M. Chandy and M. Reiser (eds.). North-Hooland, Amsterdam.

Harrison, J. M. 1985. *Brownian Motion and Stochastic Flow Systems*. John Wiley, New York.

Harrison, J. M. 1988. Brownian Models of Queueing Networks With Heterogeneous Customer Populations. In *Stochastic Differential Systems, Stochastic Control Theory and Applications*, IMA Volume 10, W. Fleming and P. L. Lions (eds.). Springer-Verlag, New York, 147-186.

Harrison, J. M. and V. Nguyen. 1990. The QNET Method for Two-Moment Analysis of Open Queueing Networks. *Queueing Systems* **6**, 1-32.

Harrison, J. M. and L. M. Wein. 1990. Scheduling Networks of Queues: Heavy Traffic Analysis of a Two-Station Closed Network. *Operations Research* **38**, 1052-1064.

Hofri, M. and K. W. Ross. 1987. On the Optimal Control of Two Queues with Server Set-up Times and its Analysis. *SIAM Journal of Computing* **16**, 399-420.

Iglehart, D. L. and W. Whitt. 1970. Multiple Channel Queues in Heavy Traffic, I and II. *Advances in Applied Probability* **2**, 150-177 and 355-364.

Karlin, S. and H. M. Taylor. 1981. *A Second Course in Stochastic Processes.* Academic Press, New York.

Kella, O. and W. Whitt. 1990. Diffusion Approximations for Queues With Server Vacations. *Advances in Applied Probability* **22**, 706-729.

Liu, Z., P. Nain and D. Towsley. 1992. On Optimal Polling Policies. *Queueing Systems* **11**, 59-83.

Mandl, P. 1968. *Analytical Treatment of One-Dimensional Markov Processes.* Springer-Verlag, New York.

Takagi, H. 1986. *Analysis of Polling Systems.* The MIT Press, Cambridge, MA.

Van Oyen, M. P. and I. Duenyas. 1992. Control of a Queueing System with Sequence-Dependent Set-ups. Technical Report 92-51, Department of Industrial and Operations Engineering, U. of Michigan, Ann Arbor, MI.

Wein, L. M. 1990a. Optimal Control of a Two-Station Brownian Network. *Mathematics of Operations Research* **15**, 215-242.

Wein, L. M. 1990b. Scheduling Networks of Queues: Heavy Traffic Analysis of a Two-Station Network With Controllable Inputs. *Operations Research* **38**, 1065-1078.