Queueing Syst
DOI 10.1007/s11134-012-9316-8

The single-server scheduling problem with convex costs

Carlos F. Bispo

Received: 3 January 2011 / Revised: 30 April 2012

Abstract Being probably one of the oldest decision problems in queuing theory, the single-server scheduling problem continues to be a challenging one. The original formulations considered linear costs, and the resulting policy is puzzling in many ways. The main one is that, either for preemptive or nonpreemptive problems, it results in a priority ordering of the different classes of customers being served that is insensitive to the individual load each class imposes on the server and insensitive to the overall load the server experiences. This policy is known as the $c\mu$-rule.

We claim and show that for convex costs, the optimal policy depends on the individual loads. Therefore, there is a need for an alternative generalization of the $c\mu$-rule. The main feature of our generalization consists on first-order differences of the single stage cost function, rather than on its derivatives. The resulting policy is able to reach near optimal performances and is a function of the individual loads.

Keywords Scheduling · Production control · Queuing systems · Dynamic priorities · $c\mu$-Rule

Mathematics Subject Classification 90B22 · 90C39 · 90C40

1 Introduction

The setting for the problem we address consists of a single server that can process different classes of customers, which arrive from the outside world and queue up in front of it, waiting for service. There will be one queue per class, and each queue is served on a first-come-first-serve basis. The arrival process is noncontrollable, and

C.F. Bispo (✉)
Instituto de Sistemas e Robótica, Instituto Superior Técnico, Av. Rovisco Pais, 2049-001 Lisbon, Portugal
e-mail: cfb@isr.ist.utl.pt

each class requires different processing times that, in general, are assumed random and a priori unknown to the server. Whenever the server concludes a service, it will have to decide which of the classes to serve next, out of the ones which have customers present. It is assumed that there is a cost associated with each queue that is proportional to the number of customers in that queue or, conversely, proportional to the waiting time of the customers. So, the growing of the queues constitutes the incentive for the server to work.

In general, considering that there are $x_i(t)$ customers of class i for $i = 1, 2, \dots, K$, at the time instant t, the single-stage cost for class i can be defined to be $C_i : N \to R$ such that $C_i(x_i)$ is nondecreasing in x and convex for $x \in R$. Here we dropped the explicit time dependence for convenience. Furthermore, we will be interested in the cases where these functions are convex. For a finite-time problem, under some decision policy, one may take the expected value over all possible trajectories of the integral over time of the single-stage cost functions sum for all classes. If the length of the trajectory is unbounded, one may choose to take a series of fixed-length time average of that expected integral with growing length, i.e., infinite-horizon average costs, or take the expected integral of the discounted sum of the single-stage costs, i.e., infinite-horizon discounted costs. That is,

$$\lim_{T\to\infty} \frac{1}{T}\mathrm{E}\left[\int_0^T C(x)\,dx\right] \tag{1}$$

or

$$\mathrm{E}\left[\int_0^\infty e^{-\beta t} C(x)\,dx\right]. \tag{2}$$

An optimal policy will be the set of decisions, a function of the k-tuple $(x_1, x_2, \dots, x_k)$ at any decision point, that for each of the above cases will achieve the minimal cost.

If the service can be preempted and later resumed for any class, there will be two decision points, the completion of a service and the occurrence of an arrival. For the nonpreemptive case, only a service conclusion is a decision point. The only exception to this is when an arrival occurs while the server is idle because, prior to the arrival and upon conclusion of the last service, there were no customers waiting for service. We also focus on the case where there is no cost associated with activating the server, i.e., no warm up cost, and no cost associated with switching from a class to another, i.e., no set up, or change over, cost.

The classic approach to this problem assumed that the single-stage costs are linear, e.g., $C_i(x_i) = c_i x_i$. When such is the case, for both infinite cost versions, in general, the optimal policy is known as the $c\mu$-rule. That is, assuming that the average processing time for class i is $1/\mu_i$, the optimal policy is such that at any decision point, the server will engage service with the head of the nonempty queue of the class which possesses the highest value of $c_i\mu_i$. An easy way to interpret intuitively this rule is to consider that if all the processing rates are the same, priority should be given to the most costly queue, or to consider that if all the single-stage costs are the same, priority should be awarded to the queue with the shortest average processing time.

The oldest known reference to a version of this problem dates back to 1956. It is considered that Smith [19] was the first to suggest the optimality of the $c\mu$-rule. His setting was deterministic and static. That is, the processing times are fixed for each class (deterministic), and all the customers are present at time 0, and no arrivals are allowed after that (static). Outside the queuing theory community, in the scheduling theory community, this is also referred to as the WSPT (Weighted Shortest Processing Time) rule. Later, Cox and Smith [8] showed the $c\mu$-rule to be optimal for a stochastic, dynamic environment with arbitrary time horizon. Their setting was that of a multiclass $M/G/1$ queue. They considered both preemptive and nonpreemptive cases. Naturally, it came with not much surprise that this rule is also optimal for stochastic and static settings; Pinedo [17] and Righter [18] are examples where such result can be found.

The amount of extensions and variants of the problem that have been considered after Cox and Smith is quite significant. For more references on related work, we refer the reader to the literature review presented in [20]. We only consider a sample of the ones that focus on the simpler problem, i.e., no feedback for instance. Out of those, Harrison [13] considered a multiclass $M/G/1$ with the added feature that there are also rewards for each service completion. His policy is slightly more complex than the $c\mu$-rule, as his β-optimal, β being the discount parameter of a continuous-time problem, specifies a priority ranking also, but some classes may never be served. The ranking is a function of β, which is not the case of the original problem. Also, the ranking is not defined by the simple $c\mu$-rule. We believe that these differences are explained by the inclusion of rewards, which distorts the original problem significantly. For the case of discrete-time problems, one example of optimality of the $c\mu$-rule was presented by Buyukkoc, Varaiya, and Walrand [6] for multiclass systems under arbitrary arrival processes, geometric service times, and preemptive discipline. This followed the work of Baras, Dorsey, and Makowski [4], which established the optimality of the $c\mu$-rule, considering only two classes of customers, with arbitrary arrival processes and service completions generated by independent Bernoulli streams.

One of the most intriguing features of this problem is the fact that the arrival rates play no role on the optimal policy structure in all the above-mentioned variants under linear costs. Defining as λ_i the average arrival rate for class i and defining as individual load of class i the ratio λ_i/μ_i, the fact that each class may have a higher or lower individual load is of no consequence on the optimal policy. This raises, in some sense, an issue of fairness. Suppose that there are only two classes of customers and the lower priority class has an individual load close to 10 %, say $\lambda_1 = 1$ and $\mu_1 = 10$, while the high priority class has an individual load close to 90 % (heavy traffic), say $\lambda_2 = 90$ and $\mu_2 = 100$, for instance costs are similar, but processing rates are different. A customer of the nonpriority class may see many customers of the priority class being served first, despite the fact that they may have arrived after. A consequence of this is a high variance of the waiting time for the nonpriority customers. Naturally, the nonpriority customers have more difficulty estimating when will they leave the system, and, while waiting, each arrival they see occurring for the priority queue has to be a source of disappointment. The linear cost model tells us that the marginal patience of the customers is always the same no matter how much time they have been waiting. That is, the willingness to wait an extra time unit is the same after a handful of services that it was upon arrival. Whoever has stood in a nonpriority queue knows that this is not true.

One natural consequence of customer's impatience is a high abandonment rate, or worse. Naturally, one could argue that the abandonment behavior could be incorporated in the model and the appropriate policy could be afterwards derived. Although our paper will take a different approach by considering convex costs, an instance of work incorporating abandonments is [14], where an asymptotically optimal policy is derived. More recent examples of asymptotically optimal policies where abandonment is included in the model are presented in [2] and [3]. Optimal policies are also studied in [3], e.g., a two-user optimal policy is given, where indices are not separable anymore, and in [11], where a sufficient condition for optimality of the $c\mu$-rule is given.

Another interesting feature of the linear cost problem is the fact that the optimal policy, $c\mu$-rule, is intrinsically myopic. That is, what appears to be the best short-term decision agrees with the long-term best decision. Associated with this, given its simple structure, it appears that the processing rate should be multiplied by the derivative of the cost function when the single-stage costs are linear.

The first work on this problem that addresses the concern of fairness is that of Van Mieghem [20], who considers the single-stage cost to be a convex function of the delay for multiclass single-server systems. Then, he proposes to use the generalized $c\mu$-rule, where c is replaced by the first derivative with respect to delay of the single-stage cost function. By performing a heavy traffic analysis, the author shows that this generalized rule is asymptotically optimal, in the sense that the cost achieved under this policy approaches the cost of the optimal policy as the sum of the individual loads approaches unity. Following this work, Mandelbaum and Stolyar [16] extend the analysis to a case where the single server is replaced by a pool of multiskilled servers that work in parallel, considering convex single-stage costs as functions of the individual queue lengths. They also establish the asymptotic optimality of the generalized $c\mu$-rule by means of conducting a heavy traffic analysis. The maximum pressure policies of [9, 10] for general stochastic processing networks produce exactly Van Mieghem's generalized $c\mu$-rule for single-server problems.

While agreeing with the inclusion of convex costs to better reflect the marginal patience of the waiting customers, we believe that there are two points on the generalization that deserve further discussion. The first point concerns using the derivative of the single-stage cost function to generalize the $c\mu$-rule. Firstly, we stress that each $C_i : N \to R$ and one can construct many convex such functions which have no derivative when assuming their domain to be the set of real numbers. Second and probably the most relevant issue is that one can formulate this as a continuous-time Markov Decision Problem, assuming Poisson arrivals, exponentially distributed service times, and apply Dynamic Programming to compute the optimal policy, for instance, through a policy iteration algorithm. Given the fact that the state space is a k-tuple of integers and that through its successive iterations the algorithm only produces valid state space transitions, one should wonder how would it be possible to converge to derivatives. In other words, is the simplicity of the linear costs hiding something else?

The second point concerns the fact that the individual loads are still not playing any role on the structure of the optimal or suboptimal policies, which is intriguing, to say the least. One exception to this is the work of [1, 12], where the authors derive an index heuristic for convex costs by formulating a restless-bandit problem. Their approach considers preemptive service [1] or nonpreemptive service [12], and the resulting index is a function of the individual arrival rates. In both cases it only considers the cost gain of reducing the queue length of the served class.

It is the purpose of the work presented here to further our knowledge on this problem, and to accomplish this, we will show that the optimal policy does depend on the individual loads and that a better generalization of the $c\mu$-rule relies on first-order differences of the single-stage cost function. Our generalization includes also the influence of cost increases when a queue gets an extra customer while being served, not just the cost reduction due to a departing customer, as in [1, 12]. On this last finding, note that for linear costs, they are exactly the same, thus justifying that the linear costs may be hiding a more interesting feature.

Naturally, these findings will have to be reconciled with [10, 16, 20], as our work does not question the validity of the results there reported. In fact, the asymptotical optimality of their generalized $c\mu$-rule, which we will term as the $Gc\mu$-rule, does not conflict with the fact that, in general, we get costs no further from the optimal costs with our proposed suboptimal policies and even achieve better results than the $Gc\mu$-rule.

In what follows, we will first formulate an MDP for a two-class single server with convex single-stage costs in Sect. 2. The restriction to two classes is done due to the fact that we intend to numerically compute the optimal policy and do not want to be overwhelmed by the curse of dimensionality [5]. Also, in Sect. 2, we will address the issue of state representation of MDPs that constitutes a generalization of the commonly accepted standard form. Then, in Sect. 3, we establish a set of very interesting results for particular instances of the single-server scheduling problem that will help us identifying how should the $c\mu$-rule be generalized. These results are valid per se, as some of the systems considered can occur in real life. Following this, in Sect. 4, we present some numerical examples that illustrate that the optimal policy is a function of the individual loads. Inspired by the results of Sect. 3, we propose a generalization of the $c\mu$-rule and present numerical data to support our claim that it is possible to have a better generalization than the existing ones. Finally, we conclude in Sect. 5, establishing a bridge between our work and previous work, and pointing directions for further research.

2 The model

To avoid excessive clutter, we are going to restrict the derivation to a system serving only two classes of customers. The extension of the model to more queues is straightforward. Let λ_i be the average arrival rate for class i for $i = 1, 2$, and assume that customers arrive according to independent Poisson processes. The processing requirements of each customer are assumed to be statistically the same within each class, with service times being exponentially distributed with mean $1/\mu_i$. Each service duration is independent of previous service durations and independent of the number of customers waiting in the system. Once started, a service may or may not be preempted and later resumed with no penalty. We will address both cases where preemption is and is not allowed, because there are some issues worth discussion concerning the later. Upon conclusion of a service, the customer being served leaves the system.

We define as $X(t) = [x_1(t)\ \ x_2(t)]'$ the amount of customers of both classes present in the system at time t. Given that there is only one server, it may be the case that either a customer of class 1 or of class 2 is being served when $X(t)$ is in the positive quadrant, while all the others are waiting. Also, we assume idleness as a possible decision for the server, although it will be seen later that the server never chooses to remain idle if there is at least one customer in one of the two queues. Given the fact that customers in the same queue are undistinguishable, each queue is served by the order of their arrival to the system, although customers of a given queue may be served prior to customers of the other queue that arrived earlier to the system.

Our state description will also have to include the state of the server when we consider the no-preemption model. Therefore, we define as $Z(t) = [X'(t)\ \ y(t)]'$ the state of the system, where $y(t) \in \{0, 1, 2\}$ is the server state at time t. If $y(t) = 0$, the server is idle, or serving a customer of class i if $y(t) = i$.

We will consider an infinite-horizon discounted cost criterion with discount parameter $\beta > 0$ and will be interested in obtaining a stationary Markov policy. A policy is defined as a function that maps the state into one of the three options for the server state. If the decision is not a function of the time instant, the policy is said to be stationary.

With the instantaneous cost rate defined earlier, we can define the expected present value of future costs, under a policy π, as follows:

$$J\big(Z(0), \pi\big) = E^{\pi}_{Z(0)}\left\{\int_0^{\infty} e^{-\beta t} C\big(Z(t)\big)\, dt\right\}, \tag{3}$$

where $E^{\pi}_{Z(0)}\{\cdot\}$ denotes the expectation with respect to the probability distribution of the path space of Z that corresponds to initial state $Z(0)$ and control policy π, and $C(Z(t)) = C_1(x_1(t)) + C_2(x_2(t))$. We then define the value function as

$$V\big(Z(0)\big) = \inf_{\pi \in \Pi} J\big(Z(0), \pi\big) \quad \text{for } Z(0) \in S, \tag{4}$$

where S defines the set of all possible states, and Π defines the set of all stationary policies. We will use $V(X(0))$ in the preemptive case and $V(X(0), y(0))$ in the nonpreemptive case. In what follows, we will first detail the preemptive case followed by the detail of the nonpreemptive case. Afterwards, we compare the equations and show that the standard formulation of MDP needs to be changed to accommodate systems where the server state needs to be captured in the overall state description.

2.1 Detail for the preemptive case

When preemption is allowed, any service conclusion event and any arrival event constitute decision epochs, i.e., will trigger the decision maker. For the first type of events the server will have to decide which of the queues to serve if both of them have customers or to remain idle. In the event of an arrival while the server is busy, it has to decide if it should switch to the class of the newly arrived customer or continue with the customer with which it has engaged previously. Under these circumstances, there is no need to explicitly include the server state for stationary policies. Given the fact that the policy produces always the same decision for the same values of x_i, knowing the queue lengths is enough to know what are the feasible transitions out of that state.

Because we are dealing with a continuous-time Markov process, we resort to the uniformization procedure to convert it into a discrete-time problem. Defining the uniform rate as $\gamma \geq \lambda_1 + \lambda_2 + \mu_1 + \mu_2 \geq 0$, $\alpha = \gamma/(\beta + \gamma)$ and omitting the explicit time dependency to avoid an excessively cumbersome notation, the value iteration algorithm [5] for this problem becomes

$$\begin{aligned} V_{k+1}(X) = {} & \frac{1}{\beta+\gamma}\big[C_1(x_1) + C_2(x_2)\big] \\ & + \alpha \min\big\{\tilde{V}_k\big(X, u \mid u=0\big), \tilde{V}_k\big(X, u \mid u=1\big), \tilde{V}_k\big(X, u \mid u=2\big)\big\}, \end{aligned} \tag{5}$$

where u represents the control decision, $V_0(X) = 0$ for all $X \in S$, with

$$\begin{aligned} \tilde{V}_k\big(X, u \mid u=0\big) &= \frac{\lambda_1}{\gamma} V_k(X+e_1) + \frac{\lambda_2}{\gamma} V_k(X+e_2) + \left(1 - \frac{\lambda_1+\lambda_2}{\gamma}\right) V_k(X), \\ \tilde{V}_k\big(X, u \mid u=1\big) &= \frac{\lambda_1}{\gamma} V_k(X+e_1) + \frac{\lambda_2}{\gamma} V_k(X+e_2) + \frac{\mu_1}{\gamma} V_k(X-e_1) \\ &\quad + \left(1 - \frac{\lambda_1+\lambda_2+\mu_1}{\gamma}\right) V_k(X), \\ \tilde{V}_k\big(X, u \mid u=2\big) &= \frac{\lambda_1}{\gamma} V_k(X+e_1) + \frac{\lambda_2}{\gamma} V_k(X+e_2) + \frac{\mu_2}{\gamma} V_k(X-e_2) \\ &\quad + \left(1 - \frac{\lambda_1+\lambda_2+\mu_2}{\gamma}\right) V_k(X), \end{aligned} \tag{6}$$

with e_i the unit vector along direction i. We omit the details concerning transforming (3) into (5) and (6). The interested reader may find this in [5] for general cases. Note that the above set of equations is only valid when both x_1 and x_2 are nonzero. If one or both are zero, then the min operator will not have the corresponding term.

We can rewrite (5) as follows:

$$\begin{aligned} V_{k+1}(X) = {} & \frac{1}{\beta+\gamma}\big[C_1(x_1) + C_2(x_2)\big] \\ & + \alpha \left\{ \frac{\lambda_1}{\gamma} V_k(X+e_1) + \frac{\lambda_2}{\gamma} V_k(X+e_2) + \left(1 - \frac{\lambda_1+\lambda_2}{\gamma}\right) V_k(X) \right\} \end{aligned}$$

$$+\alpha\min\left\{0, \frac{\mu_1}{\gamma}\left[V_k(X-e_1)-V_k(X)\right], \frac{\mu_2}{\gamma}\left[V_k(X-e_2)-V_k(X)\right]\right\}. \tag{7}$$

Letting $k \to \infty$, it is known that the value function is the fixed point of the procedure defined by (7), [5]. So, the following holds:

$$\begin{aligned} V(X) = {} & \frac{1}{\beta+\gamma}\left[C_1(x_1)+C_2(x_2)\right] \\ & + \alpha\left\{\frac{\lambda_1}{\gamma}V(X+e_1)+\frac{\lambda_2}{\gamma}V(X+e_2)+\left(1-\frac{\lambda_1+\lambda_2}{\gamma}\right)V(X)\right\} \\ & + \alpha\min\left\{0, \frac{\mu_1}{\gamma}\left[V(X-e_1)-V(X)\right], \frac{\mu_2}{\gamma}\left[V(X-e_2)-V(X)\right]\right\}. \end{aligned} \tag{8}$$

This last equation allows us to conclude easily that idleness is never the optimal decision when one or both queues are not empty, due to the following theorem.

Theorem 1 *If the single-stage cost is nondecreasing in x, then $V(X)$ is also nondecreasing.*

Proof The proof goes by induction on $V_k(\cdot)$. Given $C(X+e_i) \geq C(X)$ and $V_0(X)=0$, it follows trivially that $V_1(X+e_i) \geq V_1(X)$ for all $X \in S$. Assuming that the result holds for all $n = 1, 2, \ldots, k$, let us compute $V_{k+1}(\cdot)$.

By the induction assumption the following holds:

$$\begin{aligned} \tilde{V}_k\bigl(X+e_i, u \mid u=0\bigr) &= \frac{\lambda_1}{\gamma}V_k(X+e_i+e_1)+\frac{\lambda_2}{\gamma}V_k(X+e_i+e_2) \\ &\quad + \left(1-\frac{\lambda_1+\lambda_2}{\gamma}\right)V_k(X+e_i) \\ &\geq \frac{\lambda_1}{\gamma}V_k(X+e_1)+\frac{\lambda_2}{\gamma}V_k(X+e_2)+\left(1-\frac{\lambda_1+\lambda_2}{\gamma}\right)V_k(X) \\ &= \tilde{V}_k\bigl(X, u \mid u=0\bigr), \\ \tilde{V}_k\bigl(X+e_i, u \mid u=1\bigr) &= \frac{\lambda_1}{\gamma}V_k(X+e_i+e_1)+\frac{\lambda_2}{\gamma}V_k(X+e_i+e_2) \\ &\quad + \frac{\mu_1}{\gamma}V_k(X+e_i-e_1)+\left(1-\frac{\lambda_1+\lambda_2+\mu_1}{\gamma}\right)V_k(X+e_i) \\ &\geq \frac{\lambda_1}{\gamma}V_k(X+e_1)+\frac{\lambda_2}{\gamma}V_k(X+e_2)+\frac{\mu_1}{\gamma}V_k(X-e_1) \\ &\quad + \left(1-\frac{\lambda_1+\lambda_2+\mu_1}{\gamma}\right)V_k(X) \\ &= \tilde{V}_k\bigl(X, u \mid u=1\bigr), \end{aligned}$$

$$\begin{aligned}\tilde{V}_k\big(X+e_i,u\mid u=2\big) &= \frac{\lambda_1}{\gamma}V_k(X+e_i+e_1)+\frac{\lambda_2}{\gamma}V_k(X+e_i+e_2)\\ &\quad+\frac{\mu_2}{\gamma}V_k(X+e_i-e_2)+\left(1-\frac{\lambda_1+\lambda_2+\mu_2}{\gamma}\right)V_k(X+e_i)\\ &\geq \frac{\lambda_1}{\gamma}V_k(X+e_1)+\frac{\lambda_2}{\gamma}V_k(X+e_2)+\frac{\mu_2}{\gamma}V_k(X-e_2)\\ &\quad+\left(1-\frac{\lambda_1+\lambda_2+\mu_2}{\gamma}\right)V_k(X)\\ &= \tilde{V}_k\big(X,u\mid u=2\big).\end{aligned}$$

Given that $\min\{a,b,c\} \geq \min\{a',b',c'\}$ when $a \geq a'$, $b \geq b'$, and $c \geq c'$, it follows from (5) and from the nondecreasing nature of $C(X)$ that $V_{k+1}(X+e_i) \geq V_{k+1}(X)$ for all $X \in S$.

Since $V(X) = \lim_{k\to\infty} V_k(X)$, the result holds. □

Therefore, due to Theorem 1, the second and third terms of the min operator in (8) are negative, implying that the first term is never the lowest of the three. That is, under the optimal policy, the server is never idle in the presence of customers.

Another observation on the nature of the optimal policy, taken from (8), is that when both queues are nonempty, one chooses to serve class 1 if

$$\mu_1\big[V(X)-V(X-e_1)\big] \geq \mu_2\big[V(X)-V(X-e_2)\big] \tag{9}$$

and to serve class 2 otherwise. As a final note, we should stress that these equations refer to decision points. In the case of the model being addressed here, those are all arrival and departure instants.

2.2 Detail for the nonpreemptive case

When preemption is not allowed, the only decision points are arrivals at an empty system or conclusions of service. So, only for $y = 0$, we have choices to make, which means that we cannot drop the explicit dependence on the server state. Therefore, after the uniformization procedure we get

$$\begin{aligned}&V_{k+1}(X,0)\\ &\quad=\frac{1}{\beta+\gamma}\big[C_1(x_1)+C_2(x_2)\big]\\ &\qquad+\alpha\min\big\{\tilde{V}_k\big(X,0,u\mid u=0\big),\tilde{V}_k\big(X,0,u\mid u=1\big),\tilde{V}_k\big(X,0,u\mid u=2\big)\big\},\end{aligned} \tag{10}$$

where

$$\begin{aligned}\tilde{V}_k\big(X,0,u\mid u=0\big) &= \frac{\lambda_1}{\gamma}V_k(X+e_1,0)+\frac{\lambda_2}{\gamma}V_k(X+e_2,0)\\ &\quad+\left(1-\frac{\lambda_1+\lambda_2}{\gamma}\right)V_k(X,0),\end{aligned}$$

$$\tilde{V}_k\big(X,0,u \mid u=1\big) = \frac{\lambda_1}{\gamma} V_k(X+e_1,1) + \frac{\lambda_2}{\gamma} V_k(X+e_2,1) + \frac{\mu_1}{\gamma} V_k(X-e_1,0) + \left(1 - \frac{\lambda_1+\lambda_2+\mu_1}{\gamma}\right) V_k(X,1), \tag{11}$$

$$\tilde{V}_k\big(X,0,u \mid u=2\big) = \frac{\lambda_1}{\gamma} V_k(X+e_1,2) + \frac{\lambda_2}{\gamma} V_k(X+e_2,2) + \frac{\mu_2}{\gamma} V_k(X-e_2,0) + \left(1 - \frac{\lambda_1+\lambda_2+\mu_2}{\gamma}\right) V_k(X,2).$$

Again, note that the number of terms in the min operator depends on the number of nonempty queues. Comparing expressions (6) and (11) should shed a light into the impact the nonpreemptive assumption has on the dynamic programming recursions. The last term of each equation in (6) represents no transition due to lack of arrivals or service conclusion. The same is the case for the last term of each equation in (11). Let us call that term the self-loop. In the preemptive model, the self-loop keeps the system in the same state. However, in the nonpreemptive model, the self-loop out of a decision point sends the system to a state which is different from the state before the transition. That is, if for $Z=[X,0]$, the system decides to serve class i, the self-loop has to account for the fact that the server is still busy with a customer of class i. Thus the self-loop represents transitions to state $Z=[X,i]$. Therefore, given that the decision epochs coincide with service conclusions or arrivals while the server is idle, it is the case that such state transitions will include a second instantaneous state transition due to the decision that is made. This state transition reflects the fact that the server is no longer idle.

For the nonpreemptive case, the model needs to capture the fact that a decision to serve a given class will remain until the service is concluded. Therefore, the conversion from continuous to discrete time needs to account for the fact that when a service is initiated and there are no immediate transitions, due to service or arrivals, the state has nevertheless changed due to the earlier decision to initiate service. Whereas in the preemptive case knowing the queue length is enough to know which class is being served, for the nonpreemptive case, it is possible for the server to be working on different classes for states where the customers in all queues are the same. The majority of textbooks on dynamic programming for MDPs fail to address this fact. Markov models build on the notion that there are no simultaneous events. However, in the context of nonpreemptive queuing models, every time a state changes to a decision point, there will be a second state change due to the decision being taken. The general value iteration recursions have to account for the instantaneous state transitions due to the decision maker. We have to take into account that there are states $Z(t_k^-)$ and $Z(t_k^+)$ representing the state immediately before the kth decision epoch and immediately following that same decision epoch, respectively. Naturally, $Z(t_k^+)$ is a function of the state Z and of the control decision taken, that is, $Z(t_k^+) = f(Z(t_k^-), u_k)$. One first consequence of this is that we no longer can simplify (11) like we did transform (6) into (7).

To complete the model, we need to present the operator for the states which do not correspond to decision epochs. These recursions follow the standard MDP formulations, except for the fact that there is no decision to be made that affects the immediate transition probabilities, because they do not refer to decision epochs:

$$V_{k+1}(X,1) = \frac{1}{\beta+\gamma}\big[C_1(x_1)+C_2(x_2)\big] + \alpha\left\{\frac{\lambda_1}{\gamma}V_k(X+e_1,1) + \frac{\lambda_2}{\gamma}V_k(X+e_2,1)\right.$$
$$\left. + \frac{\mu_1}{\gamma}V_k(X-e_1,0) + \left(1-\frac{\lambda_1+\lambda_2+\mu_1}{\gamma}\right)V_k(X,1)\right\}, \qquad (12)$$

$$V_{k+1}(X,2) = \frac{1}{\beta+\gamma}\big[C_1(x_1)+C_2(x_2)\big] + \alpha\left\{\frac{\lambda_1}{\gamma}V_k(X+e_1,2) + \frac{\lambda_2}{\gamma}V_k(X+e_2,2)\right.$$
$$\left. + \frac{\mu_2}{\gamma}V_k(X-e_2,0) + \left(1-\frac{\lambda_1+\lambda_2+\mu_2}{\gamma}\right)V_k(X,2)\right\}. \qquad (13)$$

Naturally, (12) is only applicable for states where $x_1 > 0$ and (13) for states where $x_2 > 0$.

3 Exact results on specific systems

In an effort to better understand the nature of the optimal policy for the problem addressed in this paper, we are now going to analyze four particular problems that have some connection with it. We start by defining the problems in a somewhat lose manner. All considered cases will be assumed to be nonpreemptive.

Problem 1 Take a static version of the problem addressed in this paper, with K classes of customers. That is, all customers are present at time zero, and no arrivals will occur afterward. Assume there are x_i customers in queue i with $i = 1, \ldots, K$ and that the single-stage cost is convex as defined earlier. The objective is to clear the system of customers with the lowest cost possible.

Problem 2 Consider a closed queuing network with a single server, two classes of customers, and fixed population. At the conclusion of a service on a given class, a new customer of the other class is allowed to enter. Initially, there are x_i customers of class i with $i = 1, 2$. The objective is to identify the stationary policy that minimizes the infinite-horizon discounted cost.

Problem 3 Consider a closed queuing network with a single server, two classes of customers, and fixed population. At the conclusion of a service on any given class, a new customer will be allowed to enter the system. The new customer is of class i according to the ratio $p_i = \lambda_i/(\sum_{i=1}^{2}\lambda_i)$. With the single-state cost defined earlier and x_i customers of class i present in the system at time zero, the objective is to identify the policy that minimizes the infinite-horizon discounted cost.

Problem 4 Consider an open queuing network with a single server and two classes of customers. At the conclusion of a service, two customers enter the system, one for each class. Assuming there are x_i customers of class i at time zero and using the single-stage cost defined earlier, the objective is to identify the policy that ensures the minimum infinite-horizon discounted cost.

Before analyzing each of the four problems individually, we offer some remarks on each problem. Firstly note that the arrival process is no longer uncontrollable. Naturally, knowing that no customers will arrive or that they only arrive when a service is concluded drastically changes the nature of the problem. An intrinsic feature of the single-server scheduling problem we are addressing is the fact that only the stochastic nature of the arrival process is known, not the specific arrival instants.

Problem 1 can be seen as the convex cost successor, with stochastic services, of the original problem addressed by Smith [19]. Also, in many service contexts, there is such a thing as the closing hours, after which only the customers already inside the system will be served. At that point in time, when the doors are closed, the problem to be solved no longer is an infinite-horizon dynamic problem, becoming static as Problem 1. Problems 2 and 3 are examples of manufacturing contexts where there is a fixed number of pallets where parts are mounted on for processing. So, only when a part is completed, another one will use the available pallet. Problem 4 is naturally the oddest of them all, given the fact that it is unstable, whereas Problems 2 and 3 are marginally stable. Therefore, the concept of minimal cost needs to be clarified here. No matter what customer is served, two new customers will enter the system. Therefore, for any policy chosen, the population will grow to infinity. We are looking for the policy that achieves the infimum of cost relative to all possible policies, in other words, the policy that approaches infinity the cheapest way. Although this problem has no real-life application, we hope that its usefulness for our discussion will become clear by the end of this section. For the four problems, we are able to characterize the structure of the optimal policy.

Lemma 1 *In a situation where there are no arrivals during service, either because arrivals are switched off or because they only occur at the conclusion of a service, and assuming the first and second services will serve different queues, the value function for a given policy* π *can be written as*

$$J\big(Z(0),\pi\big) = \frac{C(Z(0))}{\mu_i+\beta} + \frac{E^{\pi}_{Z(0)}\{C(Z(s_1))\}}{(\mu_i+\beta)(\mu_j+\beta)}\mu_i + E^{\pi}_{Z(0)}\left\{\int_{s_1+s_2}^{\infty} e^{-\beta t} C\big(Z(t)\big)\,dt\right\}, \tag{14}$$

where $Z(0)$ *is the initial state,* μ_i *is the processing rate of the first class served,* μ_j *is the processing rate of the second class served,* s_1 *is the duration of the first service, and* s_2 *is the duration of the second service.*

Proof First, note that (3) can be written as

$$\begin{aligned} J\big(Z(0),\pi\big) &= E^{\pi}_{Z(0)}\left\{\int_0^{s_1} e^{-\beta t} C\big(Z(t)\big)\,dt + \int_{s_1}^{s_1+s_2} e^{-\beta t} C\big(Z(t)\big)\,dt \right.\\ &\quad \left. + \int_{s_1+s_2}^{\infty} e^{-\beta t} C\big(Z(t)\big)\,dt\right\} \\ &= E_{[Z(0),s_1]}\left\{\int_0^{s_1} e^{-\beta t} C\big(Z(t)\big)\,dt\right\} \end{aligned}$$

$$+ E_{[Z(0),s_1,s_2]}\left\{\int_{s_1}^{s_1+s_2} e^{-\beta t} C\bigl(Z(t)\bigr)\,dt\right\}$$
$$+ E^{\pi}_{Z(0)}\left\{\int_{s_1+s_2}^{\infty} e^{-\beta t} C\bigl(Z(t)\bigr)\,dt\right\}. \tag{15}$$

We will now derive the expressions for the first two terms.

$$\begin{aligned}
A_1 &= E_{[Z(0),s_1]}\left\{\int_0^{s_1} e^{-\beta t} C\bigl(Z(t)\bigr)\,dt\right\}\\
&= \int_0^{\infty}\int_0^{s_1} e^{-\beta t} C\bigl(Z(t)\bigr)\,dt\,\mu_i e^{-\mu_i s_1}\,ds_1\\
&= C\bigl(Z(0)\bigr)\int_0^{\infty}\int_0^{s_1} e^{-\beta t}\,dt\,\mu_i e^{-\mu_i s_1}\,ds_1\\
&= \frac{C(Z(0))}{\mu_i+\beta}.
\end{aligned} \tag{16}$$

The above relation is valid if and only if there are no arrivals during service, which is the case of any of the four problems presented. Moreover, the number of customers on all queues equals those present at time zero, and the service on any of the queues is exponentially distributed.

For the second term, assuming that the first and second services are on different classes and that there are no arrivals during the second service, we get

$$\begin{aligned}
A_2 &= E_{[Z(0),s_1,s_2]}\left\{\int_{s_1}^{s_1+s_2} e^{-\beta t} C\bigl(Z(t)\bigr)\,dt\right\}\\
&= \int_0^{\infty}\int_0^{\infty}\int_{s_1}^{s_1+s_2} e^{-\beta t} C\bigl(Z(t)\bigr)\,dt\,\mu_i e^{-\mu_i s_1}\,ds_1\,\mu_j e^{-\mu_j s_2}\,ds_2\\
&= C\bigl(Z(s_1)\bigr)\int_0^{\infty}\int_0^{\infty}\int_{s_1}^{s_1+s_2} e^{-\beta t}\,dt\,\mu_i e^{-\mu_i s_1}\,ds_1\,\mu_j e^{-\mu_j s_2}\,ds_2\\
&= \frac{C(Z(s_1))}{(\mu_i+\beta)(\mu_j+\beta)}\mu_i.
\end{aligned} \tag{17}$$

□

Definition 1 Let the first-order difference of the single-stage cost function along direction i at state $Z=(x_1,x_2,y)$ be defined as $\Delta_i(x_i)=C(Z)-C(Z-e_i)=C_i(x_i)-C_i(x_i-1)$.

Now we are in a position to characterize the optimal policies for these four problems.

Theorem 2 *For Problem 1, when there are x_i customers in queue i for $i=1,\dots,K$, it is optimal to select for service the class for which $\mu_i\Delta_i(x_i)$ is maximum.*

Proof We use a pairwise interchange argument. Assume that the optimal policy, π, is such that the kth decision chooses class j, the $(k+1)$th decision chooses class i, and the condition of the theorem is violated. That is, $\mu_j \Delta_j(x_j) < \mu_i \Delta_i(x_i)$. We construct an alternative policy π' where only those two decisions are altered, serving first class i followed by class j.

Under both policies, all decisions taken prior to the kth decision and after the $(k+1)$th decision will incur the same cost. Given the nature of the problem, we can assume with no loss of generality that $k = 1$ and make use of Lemma 1.

So, we can compare the costs of serving first class j followed by class i with the costs of serving first class i followed by class j, and the remaining decisions taken according to policy π.

For policy $\pi' = [i, j, \pi]$, we get

$$J\big(Z(0), \pi'\big) = \frac{C(Z(0))}{\mu_i + \beta} + \frac{C(Z(0) - e_i)}{(\mu_i + \beta)(\mu_j + \beta)}\mu_i + A\big(Z(0) - e_i - e_j, \pi'\big). \quad (18)$$

The term $A(Z(0) - e_i - e_j, \pi')$ represents the cost to go after the second service is concluded, and policy π is used from then on, taking into account that a class i customer was served followed by a class j customer in the first two services.

For policy $\pi = [j, i, \pi]$, we get

$$J\big(Z(0), \pi\big) = \frac{C(Z(0))}{\mu_j + \beta} + \frac{C(Z(0) - e_j)}{(\mu_j + \beta)(\mu_i + \beta)}\mu_j + A\big(Z(0) - e_i - e_j, \pi\big). \quad (19)$$

By the nature of the problem, $A(Z(0) - e_i - e_j, \pi') = A(Z(0) - e_i - e_j, \pi)$. Defining $\Delta_J = J(Z(0), \pi') - J(Z(0), \pi)$, we have

$$\begin{aligned}
\Delta_J &= \frac{C(Z(0))}{\mu_i + \beta} + \frac{C(Z(0) - e_i)}{(\mu_i + \beta)(\mu_j + \beta)}\mu_i - \frac{C(Z(0))}{\mu_j + \beta} - \frac{C(Z(0) - e_j)}{(\mu_j + \beta)(\mu_i + \beta)}\mu_j \\
&= \frac{1}{(\mu_i + \beta)(\mu_j + \beta)}\big\{C\big(Z(0)\big)(\mu_j - \mu_i)C\big(Z(0) - e_i\big)\mu_i - C\big(Z(0) - e_j\big)\mu_j\big\} \\
&= \frac{1}{(\mu_i + \beta)(\mu_j + \beta)}\big\{-\Delta_i(x_i)\mu_i + \Delta_j(x_j)\mu_j\big\}. \qquad (20)
\end{aligned}$$

Given that $\frac{1}{(\mu_i+\beta)(\mu_j+\beta)} > 0$ and $\mu_j \Delta_j(x_j) < \mu_i \Delta_i(x_i)$, it follows that $\Delta_J < 0$, which contradicts the optimality assumption for policy π.

We can apply the same argument for all consecutive decisions where different classes are served. Therefore, from the optimal policy π we can construct an alternative policy, π^*, for which costs are never worse that those achieved under policy π, by enforcing the stated rule whenever policy π fails to adhere to it, and the result follows. □

Theorem 3 *For Problem 2 with x_i customers in queue i for $i = 1, 2$, it is optimal to select for service the class i if $\mu_i \Delta_i(x_i) + \mu_j \Delta_i(x_i + 1)$ is maximum, where j is the index for the other class.*

Proof We use the same proof strategy as for the previous theorem. Assume that policy π is optimal and look for the first decision where the rule is violated. That is, class j is served immediately before class i, with class i satisfying the condition of the theorem. Next, assume with no loss of generality that to be the first decision, construct a new policy where those two decisions are reversed, $\pi' = [i, j, \pi]$, and use Lemma 1 to obtain

$$J\bigl(Z(0), \pi'\bigr) = \frac{C(Z(0))}{\mu_i + \beta} + \frac{C(Z(0) - e_i + e_j)}{(\mu_i + \beta)(\mu_j + \beta)}\mu_i + A\bigl(Z(0), \pi'\bigr), \tag{21}$$

and for policy $\pi = [j, i, \pi]$, we get

$$J\bigl(Z(0), \pi\bigr) = \frac{C(Z(0))}{\mu_j + \beta} + \frac{C(Z(0) - e_j + e_i)}{(\mu_i + \beta)(\mu_j + \beta)}\mu_j + A\bigl(Z(0), \pi\bigr). \tag{22}$$

Taking the difference of costs for both policies and noting that $A(Z(0), \pi') = A(Z(0), \pi)$, we have

$$\begin{aligned}
\Delta_J &= \frac{C(Z(0))}{\mu_i + \beta} - \frac{C(Z(0))}{\mu_j + \beta} + \frac{C(Z(0) - e_i + e_j)}{(\mu_i + \beta)(\mu_j + \beta)}\mu_i - \frac{C(Z(0) - e_j + e_i)}{(\mu_i + \beta)(\mu_j + \beta)}\mu_j \\
&= \frac{C_i(x_i) + C_j(x_j)}{\mu_i + \beta} - \frac{C_i(x_i) + C_j(x_j)}{\mu_j + \beta} + \frac{C_i(x_i - e_i) + C_j(x_j + e_j)}{(\mu_i + \beta)(\mu_j + \beta)}\mu_i \\
&\quad - \frac{C_i(x_i + e_i) + C_j(x_j - e_j)}{(\mu_i + \beta)(\mu_j + \beta)}\mu_j \\
&= \frac{1}{(\mu_i + \beta)(\mu_j + \beta)}\bigl\{C_i(x_i)(\mu_j + \beta - \mu_i - \beta) + C_i(x_i - 1)\mu_i + C_j(x_j + 1)\mu_i \\
&\quad + C_j(x_j)(\mu_j + \beta - \mu_i - \beta) - C_i(x_i + 1)\mu_j - C_j(x_j - 1)\mu_j\bigr\} \\
&= \frac{1}{(\mu_i + \beta)(\mu_j + \beta)}\bigl\{C_i(x_i)(\mu_j - \mu_i) + C_i(x_i - 1)\mu_i + C_j(x_j + 1)\mu_1 \\
&\quad + C_j(x_j)(\mu_j - \mu_i) - C_i(x_i + 1)\mu_j - C_j(x_j - 1)\mu_j\bigr\} \\
&= \frac{1}{(\mu_i + \beta)(\mu_j + \beta)}\bigl\{-\Delta_i(x_i + 1)\mu_j - \Delta_i(x_i)\mu_i \\
&\quad + \Delta_j(x_j)\mu_j + \Delta_j(x_j + 1)\mu_i\bigr\}.
\end{aligned} \tag{23}$$

Therefore, $\Delta_J \le 0$, contradicting the optimality assumption for policy π. The rest of the proof follows the same reasoning presented earlier, and the result follows. □

Theorem 4 *For Problem 3 with x_i customers in queue i for $i = 1, 2$, defining $p_i = \lambda_i / \sum_{k=1}^{2}(\lambda_k)$, it is optimal to select for service the class i if $p_j\mu_i\Delta_i(x_i) + p_i\mu_j\Delta_i(x_i + 1)$ is maximum, where j is the index for the other class.*

Proof As previously, we will use a pairwise interchange argument. For policy $\pi' = [i, j, \pi]$, we get

$$J\big(Z(0), \pi'\big) = \frac{C(Z(0))}{\mu_i + \beta} + \frac{p_i C(Z(0)) + p_j C(Z(0) - e_i + e_j)}{(\mu_i + \beta)(\mu_j + \beta)} \mu_i + A\big(Z(0) - e_i - e_j + \zeta, \pi'\big), \tag{24}$$

where

$$\zeta = \begin{cases} 2e_1 & \text{with probability } p_1^2, \\ e_1 + e_2 & \text{with probability } 2p_1 p_2, \\ 2e_2 & \text{with probability } p_2^2. \end{cases}$$

For policy $\pi = [j, i, \pi]$, we get

$$J\big(Z(0), \pi\big) = \frac{C(Z(0))}{\mu_j + \beta} + \frac{p_i C(Z(0) - e_j + e_i) + p_j C(Z(0))}{(\mu_i + \beta)(\mu_j + \beta)} \mu_j + A\big(Z(0) - e_i - e_j + \zeta, \pi\big). \tag{25}$$

Taking the difference of costs for both policies, note that $A(Z(0) - e_i - e_j + \zeta, \pi') = A(Z(0) - e_i - e_j + \zeta, \pi)$, because ζ is independent of the processing order and we are assuming that the choice of customers to enter will be the same for the sample path under both policies. Therefore,

$$\begin{aligned}
\Delta_J ={}& \frac{C_i(x_i) + C_j(x_j)}{\mu_i + \beta} + p_i \frac{C_i(x_i) + C_j(x_j)}{(\mu_i + \beta)(\mu_j + \beta)} \mu_i + p_j \frac{C_1(x_i - 1) + C_j(x_j + 1)}{(\mu_i + \beta)(\mu_j + \beta)} \mu_i \\
& - \frac{C_i(x_i) + C_j(x_j)}{\mu_j + \beta} - p_i \frac{C_i(x_i + 1) + C_j(x_j - 1)}{(\mu_i + \beta)(\mu_j + \beta)} \mu_j \\
& - p_j \frac{C_i(x_i) + C_j(x_j)}{(\mu_i + \beta)(\mu_j + \beta)} \mu_j \\
={}& \frac{1}{(\mu_i + \beta)(\mu_j + \beta)} \big\{ C_i(x_i)(\mu_j + \beta + p_i \mu_i - \mu_i - \beta - p_j \mu_j) \\
& + p_j C_i(x_i - 1)\mu_i + p_j C_j(x_j + 1)\mu_i \\
& + C_j(x_j)(\mu_j + \beta + p_i \mu_i - \mu_i - \beta - p_j \mu_j) - p_i C_i(x_i + 1)\mu_j \\
& - p_i C_j(x_j - 1)\mu_j \big\} \\
={}& \frac{1}{(\mu_i + \beta)(\mu_j + \beta)} \big\{ C_i(x_i)(p_i \mu_j - p_j \mu_i) + p_j C_i(x_i - 1)\mu_i \\
& + p_j C_j(x_j + 1)\mu_i + C_j(x_j)(p_i \mu_j - p_j \mu_i) - p_i C_i(x_i + 1)\mu_j \\
& - p_i C_j(x_j - 1)\mu_j \big\} \\
={}& \frac{1}{(\mu_i + \beta)(\mu_j + \beta)} \big\{ -\Delta_i(x_i + 1) p_i \mu_j - \Delta_i(x_i) p_j \mu_i + \Delta_j(x_j) p_i \mu_j
\end{aligned}$$

$$
\begin{aligned}
&\quad + \Delta_j(x_j+1)p_j\mu_i\bigr\} \\
&\le 0. \qquad (26)
\end{aligned}
$$

This contradicts the optimality assumption of policy π, and the result follows. □

Theorem 5 *For Problem 4 with x_i customers in queue i for $i = 1, 2$, it is optimal to select for service the class i if $\mu_j \Delta_i(x_i + 1)$ is maximum, where j is the index for the other class.*

Proof Making use of the pairwise interchange argument once more, for policy $\pi' = [i, j, \pi]$, we get

$$
J\bigl(Z(0),\pi'\bigr) = \frac{C(Z(0))}{\mu_i+\beta} + \frac{C(Z(0)+e_j)}{(\mu_i+\beta)(\mu_j+\beta)}\mu_i + A\bigl(Z(0)+e_i+e_j,\pi'\bigr), \qquad (27)
$$

and for the optimal policy $\pi = [j, i, \pi]$, we get

$$
J\bigl(Z(0),\pi\bigr) = \frac{C(Z(0))}{\mu_j+\beta} + \frac{C(Z(0)+e_i)}{(\mu_i+\beta)(\mu_j+\beta)}\mu_j + A\bigl(Z(0)+e_i+e_j,\pi\bigr). \qquad (28)
$$

Taking the difference of costs for both policies and noting that $A(Z(0) + e_i + e_j, \pi') = A(Z(0) + e_i + e_j, \pi)$, we obtain

$$
\begin{aligned}
\Delta_J &= \frac{C_i(x_i)}{\mu_i+\beta} + \frac{C_j(x_j)}{\mu_i+\beta} + \frac{C_i(x_i)}{(\mu_i+\beta)(\mu_j+\beta)}\mu_i + \frac{C_j(x_j+1)}{(\mu_i+\beta)(\mu_j+\beta)}\mu_i \\
&\quad - \frac{C_i(x_i)}{\mu_j+\beta} - \frac{C_j(x_j)}{\mu_j+\beta} - \frac{C_i(x_i+1)}{(\mu_i+\beta)(\mu_j+\beta)}\mu_j - \frac{C_j(x_j)}{(\mu_i+\beta)(\mu_j+\beta)}\mu_j \\
&= \frac{1}{(\mu_i+\beta)(\mu_j+\beta)}\bigl\{C_i(x_i)(\mu_j+\beta+\mu_i-\mu_i-\beta) + C_j(x_j+1)\mu_i \\
&\quad + C_j(x_j)(\mu_j+\beta-\mu_i-\beta-\mu_j) - C_i(x_i+1)\mu_j\bigr\} \\
&= \frac{1}{(\mu_i+\beta)(\mu_j+\beta)}\bigl\{C_i(x_i)\mu_j + C_j(x_j+1)\mu_i - C_j(x_j)\mu_i - C_i(x_i+1)\mu_j\bigr\} \\
&= \frac{1}{(\mu_i+\beta)(\mu_j+\beta)}\bigl\{-\Delta_i(x_i+1)\mu_j + \Delta_j(x_j+1)\mu_i\bigr\}. \qquad (29)
\end{aligned}
$$

Therefore, if class j violates the condition, it follows that $\Delta_J < 0$, contradicting the optimality assumption of policy π, and the result follows. □

The relevance of Problem 4 to our discussion should now be obvious. For any of the problems, we use a combination of the term $\mu_i \Delta_i(x_i)$ with the term $\mu_j \Delta_i(x_i + 1)$. Apart from Problem 2, the other ones are covered by the whole set of convex combinations of the two first-order differences. Table 1 contains the synthesis of the results just established. The first remark we need to make concerns the

Table 1 The optimality conditions for the four problems presented

Problem	$\mu_i \Delta_i(x_i)$	$\mu_j \Delta_i(x_i+1)$
1	1	0
2	1	1
3	p_j	p_i
4	0	1

fact that any choice of p_i and p_j such that each is nonnegative and $p_i + p_j = 1$ will result on the optimal policy for one instance of the above problems. If both multipliers are equal to 1, then we are producing the optimal policy for Problem 2. Therefore, if the optimal policy for the problem we address in this paper was also to be produced by means of a combination of first-order differences, the multipliers to be used would have to be such that one or both would take values outside the [0; 1] interval. Although one could argue that even for Problem 2, the optimal choice of multipliers also adds up to one, given that their sum can be normalized without changing the nature of the decisions, what we intend to stress here is that one of the multipliers may have to assume negative values for other problems. We will provide numeric evidence for this claim in what follows (Sect. 4).

Note that we are able to completely specify the optimal policies by means of a combination of first-order differences of the single-stage cost function, which can be easily computed, and the optimal policies do not depend on the discount parameter being used. It is also relatively easy to prove the above results for the infinite-horizon average costs. The proof scheme for these results is also based on a pairwise interchange argument. We do not present those to avoid redundancy of the proofs and expect any interested reader to be able to produce them, as they are simpler because there is no discount term to clutter the equations.

Finally, note that the convexity assumption for the single-stage cost function does include the case of linear costs, meaning the above results to be valid also for those. Also, the above results are trivially valid for the preemptive and nonpreemptive cases. Since we know that the optimal policy for linear costs is the $c\mu$-rule, it would appear that the problem with linear costs is equivalent to our Problem 1. In a sense it is, because the arrivals during service are irrelevant in a pairwise interchange argument, as can be seen in [7, pp. 492–495], for the discrete-time preemptive version of the problem. However, it is the claim of this paper that there is probably a little more to it, as we will show in the following sections.

It is relatively trivial to show optimality of the $c\mu$-rule for the nonpreemptive case when costs are linear.

Theorem 6 *Consider the single-server multiple-queue nonpreemptive problem with linear costs, $C(Z(t)) = \sum_{k=1}^{K} c_k x_k(t)$. The stationary optimal policy for infinite-horizon discounted costs is an index policy such that whenever there is a choice between more than a queue, service is given to the queue i with the highest index $\mu_i \Delta_i(x_i) = \max_{k, x_k(t) \neq 0}\{\mu_k c_k\}$.*

Proof Consider the optimal policy π and assume that there is a decision point where both classes i and j are present, $\mu_j c_j < \mu_i c_i$, and class j is served followed by a service on class i. Without loss of generality, assume that decision point to be at time zero. Therefore, the cost incurred under policy π is given by

$$\begin{aligned} J\big(Z(0),\pi\big) &= E^{\pi}_{Z(0)}\left\{\int_0^{\infty} e^{-\beta t}\left(\sum_{k=1}^{K} c_k x_k(t)\right) dt\right\} \\ &= E^{\pi}_{Z(0)}\left\{\int_0^{s_j} c_i x_i(t) e^{-\beta t}\, dt + \int_{s_j}^{s_j+s_i} c_i x_i(t) e^{-\beta t}\, dt \right. \\ &\quad \left. + \int_{s_j+s_i}^{\infty} c_i x_i(t) e^{-\beta t}\, dt\right\} \\ &\quad + E^{\pi}_{Z(0)}\left\{\int_0^{s_j} c_j x_j(t) e^{-\beta t}\, dt + \int_{s_j}^{s_j+s_i} c_j x_j(t) e^{-\beta t}\, dt \right. \\ &\quad \left. + \int_{s_j+s_i}^{\infty} c_j x_j(t) e^{-\beta t}\, dt\right\} \\ &\quad + E^{\pi}_{Z(0)}\left\{\int_0^{\infty} \sum_{k\neq i,j} c_k x_k(t) e^{-\beta t}\, dt\right\}, \end{aligned} \tag{30}$$

where s_i denotes the service duration on a customer of class i, and s_j the same for a customer of class j. Now consider the alternative policy π' obtained from policy π just by reversing the order of the first two services and keeping everything else unchanged. The cost under policy π' is given by

$$\begin{aligned} J'\big(Z(0),\pi'\big) &= E^{\pi'}_{Z(0)}\left\{\int_0^{\infty} e^{-\beta t}\left(\sum_{k=1}^{K} c_k x'_k(t)\right) dt\right\} \\ &= E^{\pi'}_{Z(0)}\left\{\int_0^{s_i} c_i x'_i(t) e^{-\beta t}\, dt + \int_{s_i}^{s_j+s_i} c_i x'_i(t) e^{-\beta t}\, dt \right. \\ &\quad \left. + \int_{s_j+s_i}^{\infty} c_i x'_i(t) e^{-\beta t}\, dt\right\} \\ &\quad + E^{\pi'}_{Z(0)}\left\{\int_0^{s_i} c_j x'_j(t) e^{-\beta t}\, dt + \int_{s_i}^{s_j+s_i} c_j x'_j(t) e^{-\beta t}\, dt \right. \\ &\quad \left. + \int_{s_j+s_i}^{\infty} c_j x'_j(t) e^{-\beta t}\, dt\right\} \\ &\quad + E^{\pi'}_{Z(0)}\left\{\int_0^{\infty} \sum_{k\neq i,j} c_k x'_k(t) e^{-\beta t}\, dt\right\}. \end{aligned} \tag{31}$$

Clearly, $x'_k(t) = x_k(t)$ for all $t \geq 0$ and $k \neq i, j$. Also, $x'_i(t) = x_i(t)$ for all $t \geq s_i + s_j$. The same holds for $x'_j(t)$ and $x_j(t)$. Therefore, we only have to consider four terms in each of the two cost functions above. For policy π, let us define the following terms:

$$
\begin{aligned}
A_1 &= E^{\pi}_{Z(0)}\left\{\int_0^{s_j} c_i x_i(t)e^{-\beta t}\,dt\right\},\\
A_2 &= E^{\pi}_{Z(0)}\left\{\int_{s_j}^{s_j+s_i} c_i x_i(t)e^{-\beta t}\,dt\right\},\\
B_1 &= E^{\pi}_{Z(0)}\left\{\int_0^{s_j} c_j x_j(t)e^{-\beta t}\,dt\right\},\\
B_2 &= E^{\pi}_{Z(0)}\left\{\int_{s_j}^{s_j+s_i} c_j x_j(t)e^{-\beta t}\,dt\right\}.
\end{aligned}
\tag{32}
$$

We define similar counterparts for policy π', denoting them as A'_1, A'_2, B'_1, and B'_2, respectively. Determining a closed-form expression for each of these eight terms is impossible in general due to the fact that arrivals are uncontrollable. However, given that $x'_i(t) = x_i(t) - 1$ for $t \in [s_i, s_j + s_i]$ and $x'_j(t) = x_j(t) + 1$ for $t \in [s_j, s_j + s_i]$, the change in service order does not affect the service duration, and costs are linear, it is possible to determine the following:

$$
(A_1 + A_2) - \big(A'_1 + A'_2\big) = \int_0^{\infty}\int_0^{\infty}\left[\int_{s_i}^{s_j+s_i} (+c_i)e^{-\beta t}\,dt\right]\mu_j e^{-\mu_j s_j}\,ds_j\,\mu_i e^{-\mu_i s_i}\,ds_i
\tag{33}
$$

and

$$
(B_1 + B_2) - \big(B'_1 + B'_2\big) = \int_0^{\infty}\int_0^{\infty}\left[\int_{s_j}^{s_j+s_i} (-c_j)e^{-\beta t}\,dt\right]\mu_j e^{-\mu_j s_j}\,ds_j\,\mu_i e^{-\mu_i s_i}\,ds_i.
\tag{34}
$$

Both (33) and (34) are trivially solved to produce

$$
\begin{aligned}
(A_1 + A_2) - \big(A'_1 + A'_2\big) &= -\frac{c_i\mu_i\mu_j}{\beta(\beta+\mu_i)(\beta+\mu_j)} + \frac{c_i\mu_i}{\beta(\beta+\mu_i)},\\
(B_1 + B_2) - \big(B'_1 + B'_2\big) &= \frac{c_j\mu_i\mu_j}{\beta(\beta+\mu_i)(\beta+\mu_j)} - \frac{c_j\mu_j}{\beta(\beta+\mu_j)}.
\end{aligned}
\tag{35}
$$

Therefore,

$$
\begin{aligned}
J\big(Z(0),\pi\big) - J'\big(Z(0),\pi'\big) &= (A_1 + A_2) - \big(A'_1 + A'_2\big) + (B_1 + B_2) - \big(B'_1 + B'_2\big)\\
&= \frac{-c_i\mu_i\mu_j + c_i\mu_i(\beta+\mu_j) + c_j\mu_i\mu_j - c_j\mu_j(\beta+\mu_i)}{\beta(\beta+\mu_i)(\beta+\mu_j)}\\
&= \frac{\beta(c_i\mu_i - c_j\mu_j)}{\beta(\beta+\mu_i)(\beta+\mu_j)}\\
&> 0,
\end{aligned}
\tag{36}
$$

which contradicts the optimality assumption for policy π and establishes the optimality of the $c\mu$-rule for the nonpreemptive case. □

Note that the key feature to establish the optimality of the $c\mu$-rule above is the fact that the costs are linear, (33)–(34). Also, the derivation of this result for average costs follows similar arguments and produces the same policy. This is a special case of the model in [15], where there is feedback. That is, customers can enter the queue of another class after completing service. The result in [15] was established for infinite-horizon average costs, and the proof presented here differs significantly from that one.

4 Toward a generalization of the $c\mu$-rule

In this section we present numeric evidence of the fact that the optimal policy for the convex cost version of the scheduling problem depends on the individual load. Given that we assume the buffers to be unbounded and we have to run the value iteration algorithm for bounded state space, we ran a series of tests to define an acceptable cutting point, so that the value of the optimal cost is as close as possible to the value for unbounded state space. For all the systems, we run the value iteration algorithm with $x_i \in [0 \quad 200]$ and make $\beta = 0.001$. For presentation of results, we further cut the state space where we can be sure that the errors due to the approximation are negligible. The results will be presented for $x_i \in [0 \quad 50]$. After these results we will propose an alternative generalization of the $c\mu$-rule, discuss its structure, and will present numeric evidence supporting its adequacy in terms of performance.

System 1 Consider a nonpreemptive system with $C_1(x_1) = 2x_1$, $C_2(x_2) = 1.001x_2 + 0.1x_2^2$, $\mu_1 = 2$, and $\mu_2 = 1$.

Mieghem's generalized $c\mu$-rule, which we will refer to as the $Gc\mu$-rule, takes the derivative of the single-stage cost function. For this case, the indexes for both classes are $\mu_1 \frac{\partial C(x)}{\partial x_1} = 2\mu_1$ and $\mu_2 \frac{\partial C(x)}{\partial x_2} = (1.001 + 0.2x_2)\mu_2$, respectively. The class to be served is the one that has the highest index. Combining the two indexes will generate the following policy:

$$u(x_1, x_2) = \begin{cases} 0 & \text{if } x_i = 0, \\ 1 & \text{if } x_1 > 0 \wedge x_2 \leq 14.995, \\ 2 & \text{if } x_2 > 0 \wedge (x_1 = 0 \vee x_2 > 14.995), \end{cases}$$

which is a threshold policy on the value of x_2, with threshold value of 14.995. Given that the state space is discrete, whenever x_2 is below 15, priority is given to class 1. If $x_2 \geq 15$, class 2 has priority over class 1. For all practical purposes, the real threshold for this system is 15. Note that this policy does not depend on the $\rho_1 = \lambda_1/\mu_1$ and $\rho_2 = \lambda_2/\mu_2$, nor on $\rho = \rho_1 + \rho_2$. In terms of the queue lengths for both classes, the policy can also be described as being of the switching type. That is, there is a switching curve above which it is optimal to serve one class and below which it is optimal to serve the other. For state space values on the switching curve, it is indifferent which class to serve, as the indexes for both have the same value.

If the quadratic term of the single-stage cost of class 2 were zero, then the optimal policy would be to give priority to class 1 at all times. The existence of the quadratic

Table 2 Optimal threshold values for System 1

ρ_1	ρ_2	Threshold		
		$\rho = 70\ \%$	$\rho = 90\ \%$	$\rho = 95\ \%$
0.05	$\rho - 0.05$	14	10	7
0.10	$\rho - 0.10$	14	12	–
$\rho/2$	$\rho/2$	15	15	–
$\rho - 0.10$	0.10	16	16	–
$\rho - 0.05$	0.05	16	16	16

term for class 2 forces the decision maker to change priority when the population of class 2 becomes higher than some amount. The higher the quadratic term, the lower will the threshold.

If we compute the optimal policy, using the value iteration algorithm, we also get a threshold-type policy as a function of x_2. Table 2 presents the optimal threshold values for a sample of values of ρ_i and a global load of 70 % and 90 %.

When the individual loads are the same, the optimal threshold equals the one obtained by the $Gc\mu$-rule. However, in the case of fixed global load, if the individual load of, say, class 2 grows, the optimal threshold decreases. This behavior is consistent with intuition in the following sense. There should be a threshold for x_2 above which class 2 gets priority over class 1. If the traffic for class 2 is heavier, then the threshold should drop because there will be more customers of class 2. This threshold drop does not affect the performance of class 1, given that their input rate is lower and their queue does not get as many customers as often as queue 2 gets. Moreover, we see that as the global load increases, there is a tendency for the span of the optimal threshold to get wider as a function of the individual loads. We show this just by presenting the optimal thresholds for the two extreme cases where the global load is 95 %.

System 2 Consider a nonpreemptive system with $C_1(x_1) = 1.001x_1 + 0.05x_1^2$, $C_2(x_2) = 5x_2$, $\mu_1 = 2$, and $\mu_2 = 1$.

Given that the quadratic term is present on the cost associated with class 1, the $Gc\mu$-rule will produce the following threshold policy for class 1:

$$u(x_1, x_2) = \begin{cases} 0 & \text{if } x_i = 0, \\ 1 & \text{if } x_1 > 0 \wedge (x_1 \geq 14.99 \vee x_2 = 0), \\ 2 & \text{if } x_2 > 0 \wedge x_1 < 14.99. \end{cases}$$

Under this rule, the practical threshold value for this system is 15. That is, priority is given to class 2 when x_1 in under 15. Table 3 presents the threshold results for a global load of 90 %. The behavior for this system follows the same principle as for System 1, although the optimal threshold for equal loads is slightly off the level obtained by the $Gc\mu$-rule.

System 3 Consider a nonpreemptive system with $C_1(x_1) = 2^{-9}x_1^3$, $C_2(x_2) = 2^{-5}x_2^3$, $\mu_1 = 4$, and $\mu_2 = 1$.

Table 3 Optimal threshold values for System 2

ρ_1	ρ_2	Threshold
0.05	$\rho - 0.05$	16
0.10	$\rho - 0.10$	16
$\rho/2$	$\rho/2$	14
$\rho - 0.10$	0.10	10
$\rho - 0.05$	0.05	9

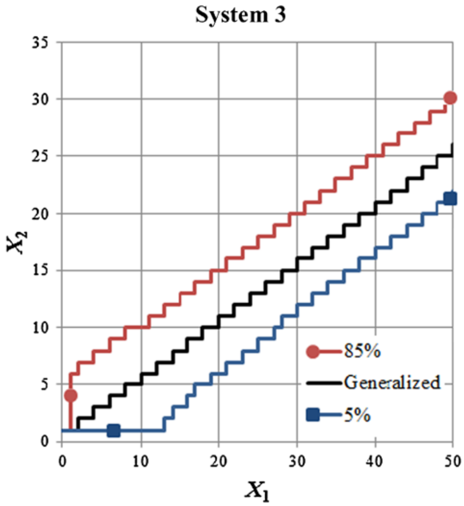

Fig. 1 Switching curves for System 3

Applying the $Gc\mu$-rule we get the following:

$$u(x_1, x_2) = \begin{cases} 0 & \text{if } x_i = 0, \\ 1 & \text{if } x_1 > 0 \wedge x_2 \leq 0.5x_1, \\ 2 & \text{if } x_2 > 0 \wedge (x_1 = 0 \vee x_2 > 0.5x_1). \end{cases}$$

The optimal switching curves for $\rho_1 = 0.05$ and $\rho_1 = 0.85$, when the global load is 90 %, are displayed in Fig. 1, together with the generalized $Gc\mu$-rule switching curve. Observing closely the optimal curves for low values of x_i, we see that they are not straight lines, although as $|X|$ grows, they become straight lines parallel to the generalized curve. This numeric evidence on the structure of the optimal policy and its structural differences relative to the switching curve generated through taking the derivative of the single-stage cost function calls for an attempt to propose a different manner on how to generalize the $c\mu$-rule. Before we move on to propose a new generalization, we need to analyze the performance achieved under the optimal policies versus the performance achieved under the $Gc\mu$-rule.

4.1 Suboptimality of the $Gc\mu$-rule

Table 4 presents $V(X, 0)$ for $X = [0 \;\; 0]'$ achieved with the optimal policy and with Mieghem's generalized rule, under $Gc\mu$-rule, for the first two systems. The global

Table 4 Policy performance for Systems 1 and 2

ρ_1	System 1			System 2		
	Optimal	$Gc\mu$-rule	Deviation	Optimal	$Gc\mu$-rule	Deviation
5 %	20 371.2	20 530.4	0.781 %	36 184.6	36 187.8	0.009 %
10 %	19 290.6	19 391.1	0.521 %	34 076.4	34 079.3	0.009 %
45 %	15 119.5	15 119.5	0.000 %	25 028.0	25 033.5	0.022 %
80 %	13 660.4	13 661.5	0.008 %	17 429.7	17 563.7	0.769 %
85 %	14 500.0	14 500.4	0.003 %	16 585.5	16 754.5	1.019 %

Table 5 Policy performance for System 3

ρ_1	Optimal	$Gc\mu$-rule	Deviation
5 %	55 319.6	56 721.2	2.534 %
45 %	14 933.7	15 082.9	0.999 %
85 %	3 920.7	4 673.1	19.189 %

load is 90 %. Although the threshold levels may be significantly different in both situations, as shown earlier, we see that the observed performance deviation is practically insignificant, agreeing with the claim of asymptotic optimality.

Turning again to System 3, we see in Table 5 that the performance deviation can be significant. We recorded a little over 19 % deviation in performance for this particular system, which is very significant.

Therefore, given the small sample of systems presented and the potential performance deviation that may occur, we believe that there is room for improvement in an attempt to produce alternative approximations to the optimal policy for convex costs. Naturally, even if the performance deviations were not significant, the simple fact that the optimal policy is a function of the individual loads is in itself an interesting aspect of the problem we are addressing. This fact alone has escaped a long series of work in the area for over half a century. Our challenge is to identify a better approximation of the optimal policy that at the same time is compatible with the optimal policy for linear costs. In order to achieve that, we will build on the results of Sect. 3 and on the insights gained by the numeric examples of this section.

4.2 Alternative generalization

The motivation that led to the generalized $c\mu$-rule based on derivatives has been the coincidence between the fact that c_i is the derivative of cost for class i, assuming that $C_i(x_i) = c_i x_i$. Also, the heavy traffic analysis is partially responsible for this. In fact, by scaling time and state space to convert a discrete problem into a continuous problem, the emergence of derivatives is a natural consequence.

We propose to alternatively consider a combination of first-order differences, designated as the $\Delta c\mu$-rule, according to the following:

$$\mu_i \left[q_i \Delta_i(x_i) + r_i \Delta_i(x_i + 1) \right], \tag{37}$$

where $q_i + r_i = 1$, and both are real numbers. The fact that $\Delta_i(x_i) = \Delta_i(x_i + 1) = c_i$ when costs are linear ensures the desired compatibility with the $c\mu$-rule. Recalling Table 1, we could rewrite the expression above as

$$\mu_i \left[q_i \Delta_i(x_i) + \frac{\mu_j}{\mu_i} \hat{r}_i \Delta_i(x_i + 1) \right], \tag{38}$$

defining $r_i = \hat{r}_i \mu_j / \mu_i$, which leads to

$$\mu_i q_i \Delta_i(x_i) + \mu_j \hat{r}_i \Delta_i(x_i + 1). \tag{39}$$

With q_i taking the place of p_j and $\hat{r}_i$ taking the place of p_i in Table 1, at the end of Sect. 3, we observed that if $q_i + \hat{r}_i = 1$ and both parameters are in [0; 1], we produce the optimal policy for some instance of Problems 1, 3 or 4. Then we claimed that to address other problems, one would have to lift those constraints. Saying that $q_i + r_i = 1$ is equivalent to saying that $q_i + \hat{r}_i \mu_j / \mu_i = 1$, which in general means that $q_i + \hat{r}_i \neq 1$. We should stress that the only reason we are enforcing the constraint $q_i + r_i = 1$ is to ensure compatibility with the index policy for linear costs.

Given the fact that the optimal policy depends on the individual loads, both q_i and r_i will have to depend on the individual loads too. That is, we need to write $q_i(\rho_1, \rho_2)$ and $r_i(\rho_1, \rho_2)$ in general. We omitted that dependence earlier to simplify the presentation. We will now illustrate the structure of the switching curves produced by this scheme for System 1.

Consider a system where $C_1(x_1) = c_{11} x_1$ and $C_2(x_2) = c_{21} x_2 + c_{22} x_2^2$. The equation defining the switching curve is given by

$$\begin{aligned} \mu_1 c_{11} &= \mu_2 \left[q_2 (c_{21} + 2c_{22} x_2 - c_{22}) + r_2 (c_{21} + 2c_{22} x_2 + c_{22}) \right] \\ &= \mu_2 \left[c_{21} + 2c_{22} x_2 + c_{22}(r_2 - q_2) \right], \end{aligned} \tag{40}$$

because $\Delta_1(x_1) = \Delta_1(x_1 + 1) = c_{11}$, and taking arbitrary values for q_i and r_i such that $q_i + r_i = 1$. Therefore, we can rewrite (40) to explicitly account for the threshold level of x_2 as follows:

$$x_2 = \frac{\mu_1 c_{11} - \mu_2 c_{21}}{2\mu_2 c_{22}} - \frac{r_2 - q_2}{2}. \tag{41}$$

If $q_2 = r_2$, we get the same threshold as the $Gc\mu$-rule. If $q_2 < r_2$, the threshold will go down, and it will go up when $q_2 > r_2$. The essential feature we want to stress here is the fact that this formulation produces a threshold that can be shifted up or down, depending on the individual loads. To achieve the range displayed in Table 2, we need to make $q_2 < 0$ and $q_1 > 1$ when ρ_1 is low. More specifically, if $r_2 - q_2 = 9.99$, we achieve a threshold of 10 for x_2. With the assumption that $q_2 + r_2 = 1$, this is accomplished with $r_2 = 5.495$ and $q_2 = -4.495$.

4.3 Numeric results

Once we have proposed a new generalization for the $c\mu$-rule, it is necessary to evaluate how close to the optimal performance is it possible to get by tuning the parameters

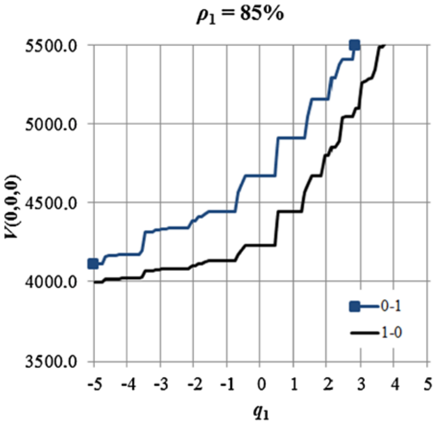

Fig. 2 Cost evolution

introduced previously. We will focus our analysis on System 3 exclusively. First, we will take a fixed value for ρ_1 and ρ_2 and will show how the performance depends on the parameters. After, we will present a map of the best achieved performances for a range of individual loads, comparing with the $Gc\mu$-rule. At the end of this section, we will also compare the achieved performances with the index heuristic proposed in [12]. All results are for infinite-horizon discounted costs.

Assume that for System 3, we have $\rho = 0.90$ and consider two cases for ρ_1, 85 % and 5 %, which are the cases of maximum observed deviation between the optimal cost and the cost achieved under the $Gc\mu$-rule. In Fig. 2 we present the evolution of $V(0,0,0)$ as a function of q_1 for two pairs of values for q_2 and r_2. Values were computed for steps of 0.1 for q_1. The left plot displays the evolution for $\rho_1 = 85$ % and on the right for $\rho_1 = 5$ %. The curve labeled "1-0" represents the case where $q_2 = 1$ and $r_2 = 0$, whereas the curve labeled "0-1" refers to $q_2 = 0$ and $r_2 = 1$. The first observation on these two plots concerns the dual nature of the behavior. In the left plot, for fixed q_1, the lower costs are achieved for the "1-0" case, while on the right plot the lower costs are achieved for the "0-1" case. Also, cost is nondecreasing with q_1 on the left and nonincreasing on the right.

The fact that the curves are nonconvex should not be a surprise given the discrete nature of the problem, since some minor change on q_1 may not produce any significant difference on the switching curve for integer values of the state space.

The behavior here displayed has been observed in all systems we tested. Taking the left plot as an instance, one should expect the cost to keep dropping as q_1 decreases down to some point, after which it should start increasing again. What we need to stress here is the fact that for high values of ρ_1 and fixing q_2 and r_2, there should be a value for q_1 where the curve reaches its minimum value, and the value of q_1 which achieves it is negative. On the right plot, there should also be a value of q_1 after which the cost should become nondecreasing and that turning point is reached for positive values of q_1. This behavior is consistent with the shifts observed for the optimal switching curves, presented earlier.

In an effort to investigate how close to the optimal one could get with the $\Delta c\mu$-rule, we conducted a series of line searches for different values of the param-

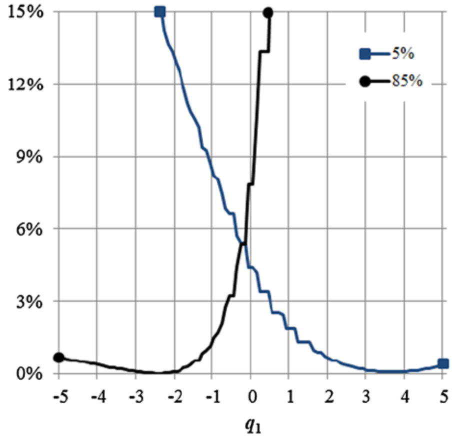

Fig. 3 Deviation to optimal value with constraint (42)

eters and came to a striking and elegant numeric coincidence. Maintaining the constraint that $q_1 + r_1 = 1$, we can restrict the search effort by imposing the following constraints:

$$\begin{cases} q_1 = r_2, \\ r_1 = q_2. \end{cases} \tag{42}$$

In Fig. 3 we present the evolution of the percent deviation of $V(0,0,0)$ achieved relative to the optimal value as a function of q_1 when constraint (42) is enforced. We get very close to the optimal value on both situations. More specifically, for $\rho_1 = 85\ \%$ and with $q_1 = -2.5$, we achieve a cost of 3 920.74, while the optimal value is 3 920.73. When $\rho_1 = 5\ \%$ and $q_1 = 3.6$, we a get a cost of 55 346.6, and the optimal value is 55 319.6. We limited the line search to steps of 0.1.

Constraint (42) was unexpected when we initiated the study, but the fact that it holds is a strong mark of elegance. Given the fact that these parameters are functions of the individual loads, when we are dealing with only two classes of customers, it makes a lot of intuitive sense that it should be this way. Therefore, the problem of identifying the optimal parameters for the $\Delta c\mu$-rule reduces to a pure line search. Table 6 presents a sample of the best achieved performances for System 3 with varying ρ_1. For each value of ρ_1, we display the value of q_1 which achieves the best performance and the percent deviation of the $Gc\mu$ and $\Delta c\mu$-rules relative to the optimal value of $V(0,0,0)$. Although there is a range of loads for which the $Gc\mu$-rule achieves highly acceptable performances, the table shows that it is always possible to do better by tuning the parameters of the $\Delta c\mu$-rule.

Before we move on there are a couple of issues that deserve discussion. Firstly, given the fact that the $Gc\mu$-rule has been proved to be asymptotically optimal in heavy traffic and given the numeric evidence here presented, there is a need to interpret this inconsistency. In the context of a multiclass system, we define loosely the concepts of *biased* and *unbiased* heavy traffic. We term as unbiased heavy traffic for K classes a situation where $\rho_i \approx \rho/K$ for $i = 1, 2, \ldots, K$ and define as biased heavy traffic a situation where one or more classes are such that $\rho_i \gg \rho/K$ and the

Table 6 Performance comparison

ρ_1	q_1	$Gc\mu$	$\Delta c\mu$
85 %	−2.5	19.19 %	≪0.001 %
75 %	−1.4	17.95 %	≪0.001 %
65 %	−0.9	7.68 %	≪0.001 %
55 %	−0.6	2.99 %	0.01 %
45 %	−0.1	1.00 %	0.019 %
35 %	0.2	0.20 %	≪0.001 %
25 %	0.8	0.03 %	≪0.001 %
15 %	1.2	0.40 %	0.0092 %
5 %	3.6	2.53 %	0.049 %

Table 7 Deviation from optimal for a subset of policies

q_1	r_1	q_2	r_2	Code	ρ_1	
					85 %	5 %
1	0	1	0	LL	13.35 %	3.37 %
1	0	0	1	LH	25.27 %	1.86 %
0	1	1	0	HL	7.83 %	4.25 %
0	1	0	1	HH	19.19 %	2.53 %

remaining classes are such that $\rho_i \ll \rho/K$. What we have seen for the specific case of $K=2$ is that the $Gc\mu$-rule appears to perform best in unbiased heavy traffic. The traditional heavy traffic analysis methodology does not account for this difference between biased and unbiased heavy traffic. In fact, in the absence of a specifically stated characterization on the nature of the heavy traffic, one can only assume that for $K \gg 1$, the results derived are specific for unbiased heavy traffic. While in general the results may be valid in the majority of the contexts irrespective of the heavy traffic characterization, we believe that this particular example suggests that future heavy traffic analysis will have to include a validation that takes into account a possible variation of the derived policies when the traffic is biased.

Secondly, although we have shown that the optimal policy is sensitive to the individual loads and that using the $\Delta c\mu$-rule is a way to get very close to the optimal performance for any load, there is no simple way to determine the adequate parameters to achieve those performances. Because we have been unable to identify the exact relation between q_1 and ρ_i, determining the optimal value for q_1 is more complex than determining the optimal policy alone. To do the line search described above implies running the value iteration algorithm for a choice of q_1 while the cost obtained for each keeps going down. On the other hand, there is no such problem with the $Gc\mu$-rule. To overcome this drawback, we propose to analyze a subset of choices for q_1.

Table 7 presents the percent deviation of cost for the subset of interest. We code each of the entries to facilitate a reference to them in the discussion. Although the results presented refer to System 5, the qualitative behavior presented has been verified across all systems tested. What we see is that when ρ_1 is high, the best cost in the subset is achieved by the entry HL, while when ρ_1 is low, the best cost is achieved with entry LH. If we assign to each character of the code the meaning H as high and L as low, then the results displayed have an easy and interesting interpretation. That is, when ρ_1 is high, the first cut solution that improves over the $Gc\mu$-rule with no computational burden is High for class 1 and Low for class 2, which relates to their relative position in terms of individual loads. Conversely, if ρ_1 is low, then the solution is Low for class 1 and High for class 2.

Therefore, for the general case, one would expect the definition of three regions for ρ_1: high, intermediate, and low. If the individual load of class 1 falls into the high region, the choice should be HL; if it falls in the low region, the choice should be LH; and if it falls in the intermediate region, the choice could either be LL or the $Gc\mu$-rule. When traffic is unbiased, one can say that both classes have a low individual load, thus justifying the choice for the intermediate region. Determining the specific cutoff values to define the three regions can be done in a qualitative and loose manner. The reason for this is as follows. Even if range limits are slightly off, the resulting policy for the whole spectrum of values for ρ_1 is definitely better than just using the $Gc\mu$-rule all the time.

Although we are not presenting any specific numeric evidence for preemptive systems, the results follow the same general structure just discussed. The only notable issue to remark here concerns the fact that the optimal switching curves differ for each system, depending if we allow preemption or not. Recall that for linear costs, the switching curves are exactly the same.

We now move on to compare the $\Delta c\mu$-rule with the index heuristic proposed in [12] for nonpreemptive systems and will still be using System 3 as the basis for comparison. According to [12], for a multiclass $M/G/1$ nonpreemptive queue with cost rate $C(x)$, where there are x customers in the queue and S is the single service time, the index that emerges from the Lagrangian Relaxation approach has the form

$$W(x) = \frac{\mathrm{E}[C(x+\chi) - C(x-1+\chi)]}{\mathrm{E}(S)} \quad \text{for } x \geq 1, \tag{43}$$

where χ has the distribution of the customers in the system of the single class $M/G/1$ queue under nonidling service. Assuming that the individual costs are expressed as $C_i(x_i) = c_{1i}x_i + c_{2i}x_i^2 + c_{3i}x_i^3$, we get

$$\begin{aligned} W_i(x_i) = \frac{1}{\mathrm{E}[S]}\mathrm{E}\big\{&c_{1i}\big[(x+\chi) - (x-1+\chi)\big] + c_{2i}\big[(x+\chi)^2 - (x-1+\chi)^2\big] \\ &+ c_{3i}\big[(x+\chi)^3 - (x-1+\chi)^3\big]\big\}. \end{aligned}$$

Given that

$$\begin{aligned} &\mathrm{E}\big\{c_{1i}\big[(x+\chi) - (x-1+\chi)\big]\big\} = c_{1i}, \\ &\mathrm{E}\big\{c_{2i}\big[(x+\chi)^2 - (x-1+\chi)^2\big]\big\} = c_{2i}\big\{x^2 - (x-1)^2 + 2\mathrm{E}[\chi]\big\}, \\ &\mathrm{E}\big\{c_{3i}\big[(x+\chi)^3 - (x-1+\chi)^3\big]\big\} \\ &\quad = c_{3i}\big\{x^3 - (x-1)^3 + 3\big[x^2 - (x-1)^2\big]\mathrm{E}[\chi] + 3\mathrm{E}\big[\chi^2\big]\big\}, \end{aligned}$$

assuming Poisson arrivals and exponentially distributed service, and recalling that, for the resulting $M/M/1$ queue, $\mathrm{E}[S]=1/\mu$, $\mathrm{E}[\chi]=\rho/(1-\rho)$, $\mathrm{E}[\chi^2]=(\rho+\rho^2)/(1-\rho)^2$, and $\mu\rho=\lambda$, results in the following index:

$$W(x_i)=\mu_i\bigl(3c_{3i}x_i^2+2c_{2i}x_i-3c_{3i}x_i+c_{3i}-c_{2i}+c_{1i}\bigr)\\+\lambda_i\frac{6c_{3i}x_i+2c_{2i}-3c_{3i}}{1-\rho_i}+3c_{3i}\lambda_i\frac{1+\rho_i}{(1-\rho_i)^2}. \tag{44}$$

At decision points one should choose to serve the class which has the highest value for such index. For the particular case of System 3, the above index assumes the form

$$W(x_i)=\mu_i\bigl(3c_{3i}x_i^2-3c_{3i}x_i+c_{3i}\bigr)+\lambda_i\frac{6c_{3i}x_i-3c_{3i}}{1-\rho_i}+3c_{3i}\lambda_i\frac{1+\rho_i}{(1-\rho_i)^2}, \tag{45}$$

whereas the index obtained by the $\Delta c\mu$-rule, say $D(x_i)$, assumes the following expression:

$$D(x_i)=\mu_i\bigl[q_i\bigl(3c_{3i}x_i^2-3c_{3i}x_i+c_{3i}\bigr)+r_i\bigl(3c_{3i}x_i^2+3c_{3i}x_i+c_{3i}\bigr)\bigr]\\=\mu_i\bigl[3c_{3i}x_i^2+(r_i-q_i)3c_{3i}x_i+c_{3i}\bigr], \tag{46}$$

because $q_i+r_i=1$.

The experimental setting is the following. Relative to System 3, we fix $\mu_2=1$ for all cases and consider four variants for the value of μ_1, which are in the set $\{4, 8, 12, 16\}$. We term the case where $\mu_1=4$ as the *simple* case. The other three will be termed as *double, triple*, and *quadruple* cases, respectively. In all four cases we will change the values of λ_i to incur a load of 5 %, 45 %, and 85 % for class 1 and a 90 % load for the ensemble of the two classes. The discount parameter is fixed and now set to be 5E–4. We change this relative to the results presented before to illustrate that the relative order of the several heuristics tested is invariant to the discount value.

In Table 8 we present the achieved performances, where the costs have been normalized with the discount parameter.

The first line corresponds to the optimal policy, and the last line presents the best achievable performance for the $\Delta c\mu$-rule, upon the line search described earlier (in parenthesis we display the optimal value of q_1). The boldfaced values indicate the best heuristic value, excluding the last line of the table. The four versions of the $\Delta c\mu$-rule retain the relative order discussed in Table 7. In the 12 sets presented, the Whittle index heuristic achieves the best results five times, whereas the four versions of the $\Delta c\mu$-rule achieve the best results seven times. However, the $\Delta c\mu$-rule is tunable, and the last line shows that it can get closer to the optimal performance than any of the other six heuristics.

This last set of numerical results shows that the optimal policy is also sensitive to the relative magnitude of the arrival and service processes when costs are nonlinear.

A quick note on the differences between Table 8 under *"simple"* and Table 5 relative to the normalized and nonnormalized costs for the optimal policy and the $Gc\mu$-rule. The discount factor used in the two sets of experiments are different, and hence the costs are different. Discounted costs converge to the infinite-horizon average costs as the discount factor drops to zero. In general, a similar behavior occurs

Table 8 Performance for a subset of policies

Policy	*Simple*			*Double*			*Triple*			*Quadruple*		
	ρ_1			ρ_1			ρ_1			ρ_1		
	5 %	45 %	85 %	5 %	45 %	85 %	5 %	45 %	85 %	5 %	45 %	85 %
OPT	61.39	16.19	4.02	85.69	20.01	6.08	96.63	21.95	10.89	102.01	24.40	19.07
$Gc\mu$	62.94	16.34	4.80	86.11	20.63	6.83	96.76	23.36	11.91	**102.14**	27.09	20.53
Whittle	**61.81**	16.35	**4.03**	87.20	20.69	**6.09**	98.50	23.40	**10.89**	103.82	27.09	**19.09**
HL	64.06	**16.19**	4.34	86.45	**20.33**	6.39	96.91	**22.87**	11.21	102.18	**26.33**	19.50
HH	62.94	16.34	4.80	86.05	20.70	6.91	96.75	23.40	11.91	**102.14**	27.09	20.53
LL	63.43	16.24	4.56	86.15	20.59	6.73	96.77	23.34	11.91	**102.14**	27.09	20.53
LH	62.53	16.50	5.04	**85.87**	21.05	7.25	**96.70**	23.95	12.60	102.18	27.92	21.56
$\Delta c\mu$	61.42	16.19	4.02	85.74	20.02	6.08	96.70	21.96	10.89	102.14	24.40	19.09
	(3.56)	(0)	(−2.5)	(1.7)	(−1.3)	(−5)	(1.5)	(−2.8)	(−8)	(1)	(−4.8)	(−11.8)

for the discounted optimal policies. However, given the discrete nature of the decision space, convergence of the optimal policies occurs while the discount factor is still nonzero [5]. The optimal switching curves obtained for Tables 5 and 8 for the simple case are exactly the same.

Out of the 12 cases presented in Table 8, we present a sample of the switching curves for the optimal policy, as well as for the Whittle-index heuristic and the $\Delta c\mu$-rule. We will be focusing only on the 45 % load for each class. The qualitative behavior for the remaining cases does not change significantly. In Fig. 4 we present those curves. In all cases the curve produced by the Whittle-index heuristic does not match the optimal switch curve and remains below it. On the other hand, the $\Delta c\mu$-rule has a very interesting proximity to the optimal curve. As the absolute value of μ_1 increases from the *simple* to the *quadruple* case, the curves produced by the Whittle-index heuristic fail to accompany the movement of the optimal curve, drifting away from it, although they move in the right direction. The percent deviation observed in Table 8 to the optimal cost by the Whittle-index heuristic is consistent with this drift. This behavior has been observed for all cases tested. It is as if there is not enough elasticity for the Whittle-index heuristic.

Reasons for this may be explained from Eq. (43). As the $Gc\mu$-rule generalizes the $c\mu$-rule interpreting it as the derivative of the linear single stage cost, the Whittle-index derivation interprets the $c\mu$-rule from the point of view of the savings. That is, what is the class that saves the most and/or faster if it sees its queue reduced by one customer at the end of service, in line also with the arguments for the proof of Theorem 6. However, one may need to take into account the case that saves the most by not having extra customers at the end of service. This is exactly what the $\Delta c\mu$-rule does by considering both $\Delta_i(x_i)$ and $\Delta_i(x_i + 1)$. Therefore, the Whittle-index heuristic produces a biased estimate of the optimal switching curve. Eventually, χ should be replaced by the distribution of arriving customers during a service.

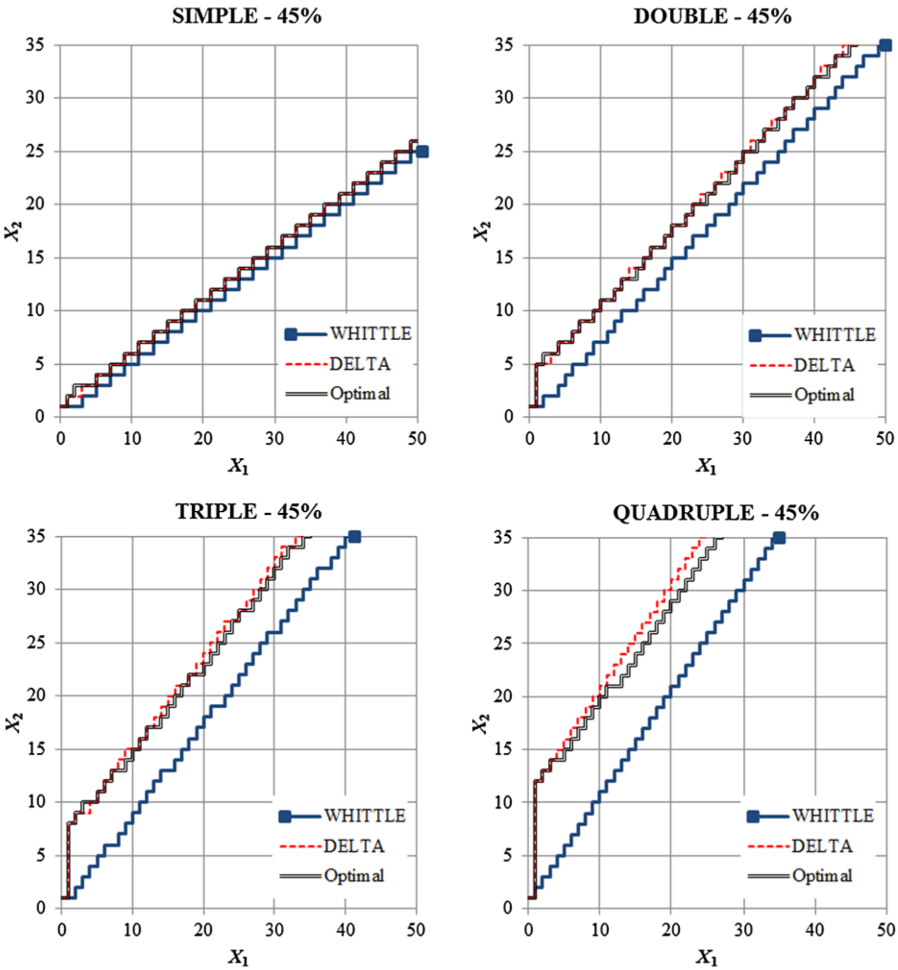

Fig. 4 Sample of switching curves obtained by the $\Delta c\mu$-rule (DELTA) and the Whittle-index heuristic (WHITTLE) compared with the optimal switching curves (Optimal)

5 Conclusions

In this paper, we have provided numeric evidence that the optimal policy for the single-server scheduling problem, when costs are convex, depends on the individual load each class imposes on the server. We restricted our analysis to systems serving only two classes of customers. We formulated a set of related problems for which we were able to derive the optimal policy and used the knowledge these problems provided to propose an alternative generalization of the $c\mu$-rule. This new generalization, designated as the $\Delta c\mu$-rule, relies on a composition of first-order differences of the single-stage cost function and is a function of the individual loads. We provided numeric evidence of near optimality for the $\Delta c\mu$-rule. Given that tuning the parameters for this rule is more time consuming than finding the optimal policy, we proposed an approximation to it that can be obtained without any computational effort. This works by dividing the load space into three regions: High, Intermediate, and Low. Although we lose the near optimality, we still obtain better performances than the generalized $Gc\mu$-rule of Mieghem [20] and than the Whittle index for many cases. If traffic is biased, the performance deviations tend to be higher in Mieghem's generalized rule. The performance deviations to the optimal are very small for unbiased traffic and when classes have similar processing rates. Our $\Delta c\mu$-rule can be fine tuned for any traffic condition, where it achieves near optimal performance in all cases tested.

Several different directions for future research can be foreseen. We focused on single server and two classes. So, a natural development would be to consider pools of servers and more classes of customers. We believe that the optimal policies should keep a similar structure to what was here presented, in terms of their dependence on the individual loads and relative magnitude of the service rates. However, the extension of the $\Delta c\mu$-rule to more classes needs to be investigated, together with its potential performance gains. The best educated guess in terms of generalizing the $\Delta c\mu$-rule to more classes of customers is that there should be a fixed set of weights for each class, q_i and r_i. Given that these need to add up to one to maintain consistency with the optimal policy for linear costs, it turns out that there is only one parameter to determine for each class. One possible avenue to explore is to formulate a Dynamic Programming problem with fixed policy and optimize the policy parameters, which are the weights. For fixed policies, it may be possible to formulate a nonlinear programming problem as discussed in [5]. Alternatively, the χ in Eq. (43) could be reinterpreted as discussed at the end of Sect. 4.

Acknowledgements The author wishes to express his gratitude to Prof. Kevin D. Glazebrook for the help provided in deriving the index expression for cubic costs as presented in Eq. (44) and also to the anonymous reviewer, whose constructive and pertinent observations led to a much improved, and more interesting, version of the original submission.

This work has been supported by the FCT (ISR/IST plurianual funding) through the PIDDAC Program funds.

References

1. Ansell, P.S., Glazebrook, K.D., Niño-Mora, J., O'Keeffe, M.: Whittle's index policy for a multi-class queueing system with convex holding costs. Math. Methods Oper. Res. **57**(1), 21–29 (2003)
2. Atar, R., Giat, C., Shimpkin, N.: On the asymptotic optimality of the $c\mu/\theta$ rule under ergodic cost. Queueing Syst. **67**(2), 127–144 (2011)
3. Ayesta, U., Jacko, P., Novak, V.: A nearly-optimal index rule for scheduling of users with abandonment. In: Proceedings IEEE INFOCOM, 2011 pp. 2849–2857 (2011)
4. Baras, J.S., Dorsey, A.J., Makowski, A.M.: Two competing queues with linear costs: the μc rule is often optimal. Adv. Appl. Probab. **17**, 186–209 (1985)
5. Bertsekas, D.P.: Dynamic Programming and Optimal Control. Athena Scientific, Nashua (1995)
6. Buyukkoc, C., Varaiya, P., Walrand, J.: The $c\mu$ rule revisited. Adv. Appl. Probab. **17**(1), 237–238 (1985)
7. Cassandras, C.G.: Discrete Event Systems: Modeling and Performance Analysis. Richard D. Irwin/Aksen Associates, Homewood/Pacific Palisades (1993)
8. Cox, D.R., Smith, W.L.: Queues. Chapman & Hall, London (1961)
9. Dai, J.G., Lin, W.: Maximum pressure policies in stochastic processing networks. Oper. Res. **53**(2), 197–218 (2005)

10. Dai, J.G., Lin, W.: Asymptotic optimality of maximum pressure policies in stochastic processing networks. Ann. Appl. Probab. **18**(6), 2239–2299 (2008)
11. Down, D.G., Koole, G., Lewis, M.E.: Dynamic control of a single-server system with abandonments. Queueing Syst. **67**(1), 63–90 (2011)
12. Glazebrook, K.D., Lumley, R.R., Ansell, P.S.: Index heuristics for multiclass $M/G/1$ systems with nonpreemptive service and convex holding costs. Queueing Syst. **45**(2), 81–111 (2003)
13. Harrison, J.M.: Dynamic scheduling of a multiclass queue: discount optimality. IEEE Trans. Autom. Control **23**(2), 270–282 (1975)
14. Harrison, J.M., Zeevi, A.: Dynamic scheduling of a multiclass queue in the Halfin–Whitt heavy traffic regime. Oper. Res. **52**(2), 243–257 (2004)
15. Klimov, G.P.: Time-sharing service systems I. Theory Probab. Appl. **19**, 532–551 (1974)
16. Mandelbaum, A., Stolyar, A.L.: Scheduling flexible servers with convex delay costs: heavy-traffic optimality of the generalized $c\mu$-rule. Oper. Res. **52**(6), 836–855 (2004)
17. Pinedo, M.: Stochastic scheduling with release dates and due dates. Oper. Res. **31**, 559–572 (1983)
18. Righter, R.: Scheduling. In: Shaked, M., Shanthikumar, J.G. (eds.) Stochastic Orders. Academic Press, San Diego (1994)
19. Smith, W.E.: Various optimizers for single-stage production. Nav. Res. Logist. Q. **3**, 59–66 (1956)
20. van Mieghem, J.A.: Dynamic scheduling with convex delay costs: the generalized $c\mu$ rule. Ann. Appl. Probab. **5**(3), 808–833 (1995)