MISTA 2011

Single server scheduling problem

Optimal policy for convex costs depends on arrival rates

Carlos F. Bispo

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 non preemptive 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.

Recently and to address the fairness issue, some authors proposed that convex costs are a better way to model the problem. Customers in the non priority queues have a highly variable cycle time as well as they may have a long average waiting time, under linear costs. This may result in customer dissatisfaction and/or high abandonment rates. The policy derived for convex costs is shown to be asymptotically optimal for heavy traffic and consists on a generalization of the optimal policy for linear costs, taking the derivative of the single stage cost function. One of the characteristics preserved in the generalization is the insensitivity to the individual loads of the classes being served.

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

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

Carlos F. Bispo
Instituto de Sistemas e Robótica – Instituto Superior Técnico
Av. Rovisco Pais, 1049-001 Lisboa, Portugal E-mail: cfb@isr.ist.utl.pt

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 non controllable and 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 on 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 there are $x_i(t)$ customers of class i, for $i = 1, 2, \ldots, K$, at the time instant t, the single stage cost for class i can be defined to be $C_i : N \to R$, such that $C_i(0) = 0$ and $\lim_{x_i \to \infty} C_i(x_i) = \infty$. 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 – infinite horizon average costs –, or take the expected integral of the discounted sum of the single stage costs – infinite horizon discounted costs. An optimal policy will be the set of decisions, function of the k-tuple $(x_1, x_2, \ldots, x_k)$ 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 non preemptive 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 – no warm up cost –, and no cost associated with switching from a class to another – 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 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 queue of the class which possesses the highest value of $\mu_i c_i$. 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 [15] 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 [6] 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 non preemptive cases. Naturally, it came with not much surprise that this rule is also optimal for stochastic and static settings as Pinedo [12] and Righter [14] 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 [16]. We only consider a sample of the ones that focus on the simpler problem, i.e., no feedback for instance. Out of those, Harrison [9] 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 on a continuous time problem, specify 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 [4], for multiclass systems under arbitrary arrival processes, geometric service times, and preemptive discipline. This followed the work of Baras, Dorsey, and Makowsky [2] 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. 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 the each class may have a higher or lower individual load is of no consequence on the optimal policy. This raises an issue of fairness, in some sense. Suppose there are only two classes of customers and the lower priority class has an individual load close to 10%, while the high priority class has an individual load close to 90% (heavy traffic), for instance costs are similar but processing rates are different. A customer of the non priority class will 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 non priority customers. Naturally, the non priority customers have more dificulty estimating when will they leave the system and, while waiting, each arrival they see occuring for the priority queue has to be a source of disapointment. The linear cost model tells us that the marginal patience of the customers is allways 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 handfull of services that it was uppon arrival. Whoever has standed in a non priority queue knows that this is not true and a natural consequence 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. This is the case addressed by Harrison and Zeevi, [10].

Another interesting feature of the problem is the fact that the optimal policy 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 [16]. There, the author considers the single stage cost to be a convex function of the delay for multiclass single server systems. Then, 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 the sum of the individual loads approaches unity. Following this work, Mandelbaum and Stolyar [11] extend the analysis to a case where the single server is replaced by a pool 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 [7,8] 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 the real numbers. Secondly, and probably the most relevant issue, suppose one formulates this as a continuous time Markov Decision Problem, assuming Poisson arrivals, exponentially distributed service times, and applies Dynamic Programming to compute the optimal policy, through a policy iteration algorithm, for instance. 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 sub-optimal policies, which is intriguing, to say the least. One exception to this is the work of [1], where the authors derive an index heuristic for convex costs, by formulating a restless-bandit problem. Their approach considers preemptive service only, and the resulting index is a function of the individual arrival rates, but only considers the cost gain of reducing the queue lenght 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, not just the cost reduction due to a departing customer, as in [1]. On this last finding, note that for linear costs they are exactly the same, thus justifying that the linear costs do hide a more interesting feature.

Naturally, these findings will have to be reconciled with [16,11,8], 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 sub-optimal policies and even achieve better results than the $Gc\mu$-rule.

In what follows we will first formulate a MDP for a two class single server with convex single stage costs in Section 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, [3]. Then, in Section 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 Section 4, we present a series of numerical examples that illustrate that the optimal policy is a function of the individual loads. Inspired by the results of Sections 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 Section 5, establishing a bridge between our work and previous work, and pointing directions for further research.

2 The model

We are going to restrict our analysis to a system serving only two classes of customers. Let λ_i be the average arrival rate for class i, for $i = 1, 2$, and assume 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 as well as 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 were 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 choses 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 customers 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. That is, a policy is defined as a function that maps the state into one of the three options for the server state, such that the decision is not a function of the time instant.

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

$$J(Z(0), \pi) = E^{\pi}_{Z(0)} \left\{ \int_0^{\infty} e^{-\beta t} C(Z(t)) dt \right\}, \tag{1}$$

where $E^{\pi}_{Z(0)} \{.\}$ 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

$$V(Z(0)) = \inf_{\pi \in \Pi} J(Z(0), \pi) \quad \text{for } Z(0) \in S, \tag{2}$$

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 non preemptive case. In what follows we will first detail the preemptive case followed by the detail of the non preemptive case. After we compare the equations and show that the standard formulation of MDP needs to be changed to accommodate systems where the server state need to be captured in the overall state description.

2.1 Detail for the non preemptive case

When preemption is not allowed, the only decision points are arrivals to an empty system or conclusion of service. So, only for $y = 0$ we have choices to make. 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, [3], for this problem becomes

$$\begin{aligned} V_{k+1}(X, 0) = & \frac{1}{\beta + \gamma} \left[C_1(x_1) + C_2(x_2) \right] + \\ & + \alpha \min \left\{ \tilde{V}_k(X, 0, u | u = 0), \tilde{V}_k(X, 0, u | u = 1), \tilde{V}_k(X, 0, u | u = 2) \right\}, \end{aligned} \tag{3}$$

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

$$\begin{aligned} \tilde{V}_k(X, 0, u | u = 0) = & \frac{\lambda_1}{\gamma} V_k(X + e_1, 0) + \frac{\lambda_2}{\gamma} V_k(X + e_2, 0) + (1 - \frac{\lambda_1 + \lambda_2}{\gamma}) V_k(X, 0), \\ \tilde{V}_k(X, 0, u | u = 1) = & \frac{\lambda_1}{\gamma} V_k(X + e_1, 1) + \frac{\lambda_2}{\gamma} V_k(X + e_2, 1) + \frac{\mu_1}{\gamma} V_k(X - e_1, 0) + \\ & + (1 - \frac{\lambda_1 + \lambda_2 + \mu_1}{\gamma}) V_k(X, 1), \\ \tilde{V}_k(X, 0, u | u = 2) = & \frac{\lambda_1}{\gamma} V_k(X + e_1, 2) + \frac{\lambda_2}{\gamma} V_k(X + e_2, 2) + \frac{\mu_2}{\gamma} V_k(X - e_2, 0) + \\ & + (1 - \frac{\lambda_1 + \lambda_2 + \mu_2}{\gamma}) V_k(X, 2), \end{aligned} \tag{4}$$

with $e_1 = [1; 0]'$ and $e_2 = [0; 1]'$. To complete the model we need to present the operator for the remaining states.

$$\begin{aligned} V_{k+1}(X, 1) = & \frac{1}{\beta + \gamma} \left[C_1(x_1) + C_2(x_2) \right] + \alpha \left\{ \frac{\lambda_1}{\gamma} V_k(X + e_1, 1) + \frac{\lambda_2}{\gamma} V_k(X + e_2, 1) + \right. \\ & \left. + \frac{\mu_1}{\gamma} V_k(X - e_1, 0) + (1 - \frac{\lambda_1 + \lambda_2 + \mu_1}{\gamma}) V_k(X, 1) \right\}, \end{aligned} \tag{5}$$

$$V_{k+1}(X,2) = \frac{1}{\beta+\gamma}\left[C_1(x_1)+C_2(x_2)\right] + \alpha\left\{\frac{\lambda_1}{\gamma}V_k(X+e_1,2)+\frac{\lambda_2}{\gamma}V_k(X+e_2,2)+\right.$$
$$\left. + \frac{\mu_2}{\gamma}V_k(X-e_2,0) + (1-\frac{\lambda_1+\lambda_2+\mu_2}{\gamma})V_k(X,2)\right\}. \qquad (6)$$

Naturally, (5) is only applicable for states where $x_1 > 0$ and (6) 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 loose manner.

Problem 1 Take a static version of the problem addressed in this paper, with $K > 1$ classes. That is, all customers are present at time zero and no arrivals will occur afterward. Assume there are x_i customers in queue i 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 the same single stage cost as defined earlier, the objective is to identify the non idling policy that minimizes the infinite horizon discounted cost.

Problem 3 Consider a closed queuing network with a single server, $K = 2$ 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 $p_i = \lambda_i/(\sum_{k=1}^{K}\lambda_k)$. 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 non idling 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 one customer of each class is allowed to enter the system. 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 non idling policy that ensures the minimum infinite horizon discounted cost.

Before analyzing each of the four problems individually we offer some remarks on each problem. The first remark on the four problems is the fact 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, [15]. 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 non idling policy chosen, the population will grow to infinity. We are looking for the non idling policy that achieves the infimum cost relative to all possible non idling policies. In other words, the policy that approaches infinity the cheapest way. Although this problem has no real life application, we hope 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

$$J(Z(0),\pi) = \frac{C(Z(0))}{\mu_i+\beta} + \frac{E^{\pi}_{Z(0)}\{C(Z(s_1))\}}{(\mu_i+\beta)(\mu_j+\beta)}\mu_i + E^{\pi}_{Z(0)}\left\{\int_{s_1+s_2}^{\infty} e^{-\beta t}C(Z(t))dt\right\}, \quad (7)$$

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 the duration of the second service.

Proof: First, note that (1) can be written as

$$\begin{aligned}
J(Z(0),\pi) &= E^{\pi}_{Z(0)}\left\{\int_0^{s_1} e^{-\beta t}C(Z(t))dt + \int_{s_1}^{s_1+s_2} e^{-\beta t}C(Z(t))dt + \right.\\
&\quad \left. + \int_{s_1+s_2}^{\infty} e^{-\beta t}C(Z(t))dt\right\}\\
&= E_{[Z(0),s_1]}\left\{\int_0^{s_1} e^{-\beta t}C(Z(t))dt\right\} + \\
&\quad + E_{[Z(0),s_1,s_2]}\left\{\int_{s_1}^{s_1+s_2} e^{-\beta t}C(Z(t))dt\right\} + \\
&\quad + E^{\pi}_{Z(0)}\left\{\int_{s_1+s_2}^{\infty} e^{-\beta t}C(Z(t))dt\right\} \qquad (8)
\end{aligned}$$

where s_1 is the duration of the first service and s_2 is the duration of the second service. We will now derive the expressions for the first two terms.

$$
\begin{aligned}
A_1 &= E_{[Z(0),s_1]}\left\{\int_0^{s_1} e^{-\beta t} C(Z(t))dt\right\} \\
&= \int_0^{\infty}\int_0^{s_1} e^{-\beta t} C(Z(t))dt\mu_i e^{-\mu_i s_1} ds_1 \\
&= C(Z(0))\int_0^{\infty}\int_0^{s_1} e^{-\beta t} dt\mu_i e^{-\mu_i s_1} ds_1 \\
&= \frac{C(Z(0))}{\mu_i+\beta}. \qquad (9)
\end{aligned}
$$

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 both queues equals those present at time zero, and the service on any of the queues is exponentially distributed.

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

$$
\begin{aligned}
A_2 &= E_{[Z(0),s_1,s_2]}\left\{\int_{s_1}^{s_1+s_2} e^{-\beta t} C(Z(t))dt\right\} \\
&= \int_0^{\infty}\int_0^{\infty}\int_{s_1}^{s_1+s_2} e^{-\beta t} C(Z(t))dt\mu_i e^{-\mu_i s_1} ds_1 \mu_j e^{-\mu_j s_2} ds_2 \\
&= E^{\pi}_{Z(0)}\{C(Z(s_1))\}\int_0^{\infty}\int_0^{\infty}\int_{s_1}^{s_1+s_2} e^{-\beta t} dt\mu_i e^{-\mu_i s_1} ds_1 \mu_j e^{-\mu_j s_2} ds_2 \\
&= \frac{E^{\pi}_{Z(0)}\{C(Z(s_1))\}}{(\mu_i+\beta)(\mu_j+\beta)}\mu_i \qquad (10)
\end{aligned}
$$

Q.E.D.

Definition 1 Let the first order difference of the single stage cost function along direction i at state Z 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 the four problems.

Theorem 1 For Problem 1, when there are x_i customers in queue i, for $i = 1, \ldots, K$, 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 the optimal policy, π, is such that the k^{th} decision choses class j and the $(k+1)^{th}$ decision choses 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 build 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 k^{th} 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 $k = 1$.

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 following decisions will follow the optimal policy π.

For policy $\pi' = [i, j, \pi]$ and from Lemma 1 we get

$$J(Z(0), \pi') = \frac{C(Z(0))}{\mu_i + \beta} + \frac{C(Z(0) - e_i)}{(\mu_i + \beta)(\mu_j + \beta)}\mu_i + A(Z(0) - e_i - e_j, \pi'). \tag{11}$$

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

$$J(Z(0), \pi) = \frac{C(Z(0))}{\mu_j + \beta} + \frac{C(Z(0) - e_j)}{(\mu_j + \beta)(\mu_i + \beta)}\mu_j + A(Z(0) - e_i - e_j, \pi), \tag{12}$$

By the nature of the problem and given the stated policies, $A(Z(0) - e_i - e_j, \pi') = A(Z(0) - e_i - e_j, \pi)$, because the state reached after the first two decisions is the same under both policies and policy π' produces the same decisions as policy π from the third decision onward. Defining $\Delta_J = J(Z(0), \pi') - J(Z(0), \pi)$

$$\begin{aligned}
\Delta_J &= \frac{C(Z(0))}{\mu_i + \beta} + \frac{C(Z(0) - e_i)}{(\mu_i + \beta)(\mu_j + \beta)}\mu_i - \frac{C(Z(0))}{\mu_j + \beta} - \frac{C(Z(0) - e_j)}{(\mu_i + \beta)(\mu_j + \beta)}\mu_j \\
&= \frac{1}{(\mu_i + \beta)(\mu_j + \beta)} \left\{ C(Z(0))(\mu_j - \mu_i) + C(Z(0) - e_i)\mu_i - C(Z(0) - e_j)\mu_j \right\} \\
&= \frac{1}{(\mu_i + \beta)(\mu_j + \beta)} \left\{ -C(Z(0))\mu_i + C(Z(0) - e_i)\mu_i + C(Z(0))\mu_j - \right. \\
&\quad \left. - \; C(Z(0) - e_j)\mu_j \right\} \\
&= \frac{1}{(\mu_i + \beta)(\mu_j + \beta)} \left\{ -\Delta_i(x_i)\mu_i + \Delta_j(x_j)\mu_j \right\}.
\end{aligned} \tag{13}$$

Given the fact that, $\frac{1}{(\mu_1+\beta)(\mu_2+\beta)} > 0$ 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^*, where costs are never worse than those achieved under policy π by enforcing the stated rule whenever it is not observed in policy π, and the result follows.

Q.E.D.

The formal results for the remaining problems will be presented without proofs, as the proof technique follows the same path as this one.

Theorem 2 For Problem 2, with x_i customers in queue i, for $i = 1, 2$, it is optimal to select for service 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.

Theorem 3 For Problem 3, with x_i customers in queue i, for $i = 1, 2$, and defining $p_i = \lambda_i / \sum_k \lambda_k$, 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.

Theorem 4 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.

Before concluding this section, the solution for Problem 4 deserves some intuitive description as it can be seen as a dual to the solution for Problem 1. For Problem 4, if both service rates are the same we should chose to serve the class that most increases cost by receiving the extra customer. If both cost increases are the same we should chose to serve the class with the lowest processing rate to delay the cost increase due to the extra customer.

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 Problem 2, the other ones are covered by the whole set of convex combinations of the two first order differences, when there are only two classes. Table 1 contains the synthesis of the results just established. The first remark we need to make concerns the fact that any choice of p_i and p_j such that each is non negative and $p_i + p_j = 1$ will result on the optimal policy for one instance of the above problems with only two classes. 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.

Table 1 The optimal multipliers.

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

Note that we are able to completely specify the optimal policies by means of a combination of first orders 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 the non preemptive 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 Cassandras, [5], pp. 492—495, for the discrete time version of the problem. However, it is the claim of this paper that there is a little more to it, as we will show in the following sections.

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 a close as possible equal to the value for unbounded state space. For all the systems we run the value iteration algorithm with $x_i \in [0; 200]$ and make $\beta = 0.001$. For results presentation we further cut the state space where we can be sure there are no errors due to the approximation. The results will be presented for $x_i \in [0; 50]$. 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 non preemptive 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, will produce the following condition.

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

which is a threshold policy on the value of x_2. Whenever x_2 is below 15, priority is given to class 1. If x_2 is above 14, class 2 has priority over class 1. Note that this policy does not depend the $\rho_1 = \lambda_1/\mu_1$ and $\rho_2 = \lambda_2/\mu_2$, nor on $\rho = \rho_1 + \rho_2$.

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 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 be.

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%.

Table 2 Optimal threshold values for System 1.

		Threshold	
ρ_1	ρ_2	$\rho = 70\%$	$\rho = 90\%$
0.05	$\rho - 0.05$	14	10
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

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 for 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. In fact, the optimal threshold for a global load of 95% and for the first line of Table 2 is 7 and for last line is still 16.

System 2 Consider a non preemptive 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 a threshold policy for class 1, according to the following.

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

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. However, for this system the threshold for equal loads does not match the level obtained through diferentiation of the single stage cost function.

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

System 3 Consider a non preemptive system with $C_1(x_1) = x_1 + 0.1x_1^2$, $C_2(x_2) = 1.2x_2 + 0.3x_2^2$, $\mu_1 = 3$, and $\mu_2 = 1$.

Applying again the $Gc\mu$-rule we get

$$u(x_1, x_2) = \begin{cases} 0 & \text{if } x_i = 0 \\ 1 & \text{if } x_1 > 0 \wedge (x_2 \leq x_1 + 3 \vee x_2 = 0) \\ 2 & \text{if } x_2 > 0 \wedge (x_1 = 0 \vee x_2 > x_1 + 3). \end{cases}$$

System 4 Consider a non preemptive system with $C_1(x_1) = x_1 + 0.1x_1^2$, $C_2(x_2) = 4.8x_2 + 0.3x_2^2$, $\mu_1 = 3$, and $\mu_2 = 1$.

For this system we get the following conditions for the $Gc\mu$-rule.

$$u(x_1, x_2) = \begin{cases} 0 & \text{if } x_i = 0 \\ 1 & \text{if } \; x_1 > 0 \wedge (x_2 \leq x_1 - 3 \vee x_2 = 0) \\ 2 & \text{if } \; x_2 > 0 \wedge (x_1 = 0 \vee x_2 > x_1 - 3). \end{cases}$$

Figure 1 presents the optimal switching curves obtained for different individual loads when the global load is 90% for Systems 3 and 4. When the state of the system at a decision point is above or over any of the curves the server choses class 2 and below choses class 1. Each plot also includes the switching curve associated with the $Gc\mu$-rule. The labels "5%" and "85%" correspond to the individual load of class 1. Taking the generalized curve as reference, we see that an increase of the individual load for class 1 shifts the curve upward, and a decrease in its individual load shifts the curve downward. This behavior is in line with what was shown for Systems 1 and 2. One could also say that the curves move left or right, given that the observed shifts are consistent with the both observations. Note that the slope of the optimal switching curves is the same as the slope of the generalized curve, which is 1. We need to clarify the little detail at the origin where we intentionally changed one of the curves to mark the square where the server is idle because there are no customers.

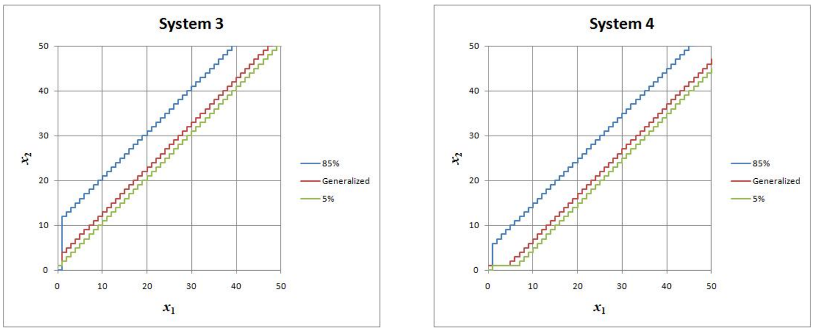

Fig. 1 Switching curves for Systems 3 and 4.

Taking these four systems as examples, one would be led to believe that the optimal switching curves correspond to some type of shift of the switching curve obtained by the $Gc\mu$-rule. The following system serves the purpose of showing that it is not so.

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

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 Figure 2, together with the generalized curve. Observing closely the optimal curves for low values of x_i we see that they are not straight lines, although

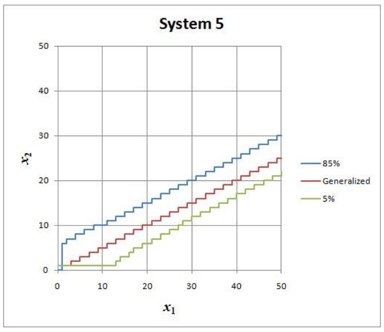

Fig. 2 Switching curves for System 5.

as $|X|$ grows they become straight lines paralel 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 of 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.

Table 4 presents $V(X, 0)$ for $X = [0; 0]'$ achieved with the optimal policy and with Mieghem's generalized rule, under "Derivative", for the first two systems. The global load is 90%. Although the threshold levels may be significantly different on both situations, as shown earlier, we see that the observed performance deviation is pratically insignificant, agreeing with the claim of asymptotic optimality.

Table 4 Policy performance for Systems 1 and 2.

	System 1			System 2		
ρ_1	Optimal	Derivative	Deviation	Optimal	Derivative	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%

A similar behavior is recorded for System 3 and 4, Table 5. However, for System 4 we note a maximum performance deviation on the sampled cases of over 5%. It appears that there would be no advantage in trying to improve performance, given that the results displayed so far are so favorable to the exhisting sub-optimal policy. However, turning again to System 5 we see in Table 6 that the performance deviation can be much higher. 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 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

Table 5 Policy performance for Systems 3 and 4.

ρ_1	System 3 Optimal	System 3 Derivative	System 3 Deviation	System 4 Optimal	System 4 Derivative	System 4 Deviation
5%	43,697.5	43,895.7	0.454%	66,859.6	66,978.5	0.178%
45%	24,681.1	24,790.5	0.443%	39,656.2	39,816.6	0.404%
85%	14,500.0	14,500.4	0.003%	21,224.8	22,383.8	5.461%

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 Section 3 and on the insights gained by the numeric examples of this section.

Table 6 Policy performance for System 5.

ρ_1	Optimal	Derivative	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%

4.1 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 $C_i(x_i) = c_i x_i$. 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.

$$\mu_i[q_i \Delta_i(x_i) + r_i \Delta_i(x_i + 1)], \tag{14}$$

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 re-write the expression above as

$$\mu_i[q_i \Delta_i(x_i) + \frac{\mu_j}{\mu_i} \hat{r}_i \Delta_i(x_i + 1)], \tag{15}$$

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

$$\mu_i q_i \Delta_i(x_i) + \mu_j \hat{r}_i \Delta_i(x_i + 1). \tag{16}$$

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 Section 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 $q_i + \hat{r}_i \neq 1$.

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 analyze the structure of the switching curves produced by this scheme for general systems.

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

$$\begin{aligned} \mu_1 c_{11} &= \mu_2[q_2(c_{21} + 2c_{22}x_2 - c_{22}) + r_2(c_{21} + 2c_{22}x_2 + c_{22})] \\ &= \mu_2[c_{21} + 2c_{22}x_2 + c_{22}(r_2 - q_2)], \end{aligned} \tag{17}$$

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 re-write (17) to explicitly account for the threshold level of x_2, as follows.

$$x_2 = \frac{\mu_1 c_{11} - \mu_2 c_{21}}{2\mu_2 c_{22}} - \frac{r_2 - q_2}{2}. \tag{18}$$

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 $q_i > 1$ 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 $q_2 = -4.495$.

Now assume we have a system such that $C_1(x_1) = c_{11}x_1 + c_{12}x_1^2$ and $C_1(x_1) = c_{11}x_1 + c_{12}x_1^2$. Using the $\Delta c\mu$-rule, the switching curve will be given by

$$\mu_1[c_{11} + 2c_{12}x_1 + c_{12}(r_1 - q_1)] = \mu_2[c_{21} + 2c_{22}x_2 + c_{22}(r_2 - q_2)], \tag{19}$$

rearranging the terms and solving in order to x_2, we get

$$x_2 = \frac{\mu_1 c_{12}}{\mu_2 c_{22}} x_1 + \frac{\mu_1 c_{11} - \mu_2 c_{22}}{2\mu_2 c_{22}} + \frac{\mu_1 c_{12}(r_1 - q_1) - \mu_2 c_{22}(r_2 - q_2)}{2\mu_2 c_{22}},$$

which is a family of straight lines with constant slope and variable intercept. The intercept variation is introduced by the parameters q_i and r_i. Again, if $q_i = r_i$ we get the equation generated by the $Gc\mu$-rule. This structure is consistent with what we observed for Systems 3 and 4. For systems like System 1 and 2, knowing the value of the threshold is enough to determine the values of the two necessary parameters. However, for systems like System 3 and 4, knowing the intercept is not enough to determine the values of the four parameters, as we are left with one more degree of freedom. As we will show in the following and much to our surprise, the existence of the extra degree of freedom seems to be apparent.

Finally, assuming pure cubic costs for both classes, that is, $C_i(x_i) = c_{i3}x_i^3$, we get

$$\mu_1[3c_{13}x_1^2 + 3c_{13}x_1(r_1 - q_1) + 1] = \mu_2[3c_{23}x_2^2 + 3c_{23}x_2(r_2 - q_2) + 1], \tag{20}$$

which can easily be solved in order to x_2. Inspecting this last expression, when $|X| \to \infty$ the quadratic terms are dominant. Thus the curve approaches a straight line. However, for low values of $|X|$ the equation above exhibits a non linear behavior. This is in line with what was observed in Figure 2 for the optimal switching curves displayed.

4.2 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 tunning the parameters introduced previously. We will focus our analysis on System 5 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.

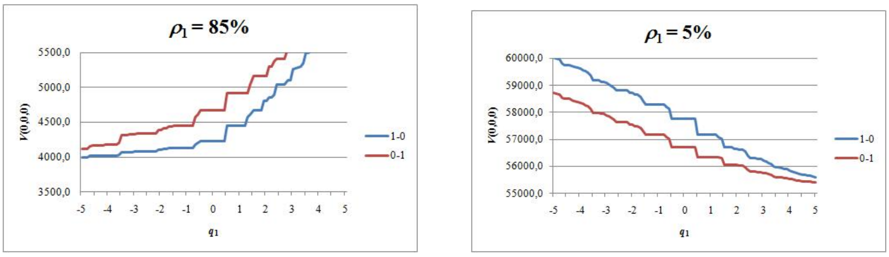

Fig. 3 Cost evolution.

Assume that for System 5 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 Figure 3 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 non decreasing with q_1 on the left and non increasing on the right.

The fact that the curves are non convex 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 non decreasing 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.

Note also that any horizontal line crosses the two displayed cost curves or none of them. For fixed values of q_2 and r_2, this means that any cost within the range of achievable costs can be reached with some choice of q_1. This naturally includes the minimal value achievable under the policy produced by the $\Delta c\mu$-rule.

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 parameters 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.

$$\begin{cases} q_1 = r_2 \\ r_1 = q_2 \end{cases} \tag{21}$$

In Figure 4 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 (21) 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. Therefore, one can conjecture that it may be possible to get even closer. However, the displayed results are already sufficiently good to make our case.

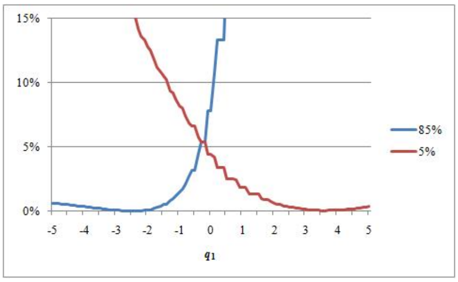

Fig. 4 Deviation to optimal value enforcing constraint (21).

Constraint (21) 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 7 presents a sample of the best achieved performances for System 5 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 tunning 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

Table 7 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%

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 $\rho_i >> \rho/K$ and the remaining classes are such that $\rho_i << \rho/K$. What we have seen for the specific case of $K = 2$ is that the $Gc\mu$-rule performs 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 $K >> 1$ 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 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 8 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

Table 8 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%

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.

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 tunning 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 $c\mu$-rule of Mieghem, [16]. 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. Our $\Delta c\mu$-rule can be fine tunned 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. However, the extension of the $\Delta c\mu$-rule to more classes needs to be investigated, together with its potential performance gains.

References

1. P. S. Ansell and K. D. Glazebrook and J. Nio-Mora and M. O'Keeffe, Whittle's index policy for a multi-class queueing system with convex holding costs, Mathematical Methods of Operations Research, Vol. 57, No. 1, 21-29 (2003)
2. J.S. Baras and A.J. Dorsey and A.M. Makowsky, Two competing queues with linear costs: the μc rule is often optimal, Advances in Applied Probability, Vol. 17, 186-209 (1985)
3. D.P. Bertsekas, Dynamic Programming and Optimal Control, Athena Scientific,(1995)
4. C. Buyukkoc and P Varaiya and J. Walrand, The $c\mu$ rule revisited, Advances in Applied Probability, Vol. 17, No. 1, 237-238, (1985)
5. C.G. Cassandras, Discrete Event Systems: Modeling and Performance Analysis, Richard D. Irwin, Inc. and Aksen Associates, Inc. (1993)
6. D.R. Cox and W.L. Smith, Queues, Chapman & Hall (1961)
7. J. G. Dai and W. Lin, Maximum pressure policies in stochastic processing networks, Operations Research, Vol. 53, No. 2, 197-218 (2005)
8. J. G. Dai and W. Lin, Asymptotic optimality of maximum pressure policies in stochastic processing networks, Annals of Applied Probability, Vol. 18, No. 6, 2239-2299 (2008)
9. J.M Harrison, Dynamic scheduling of a multiclass queue: discount optimality, IEEE Transactions on Automatic Control, Vol. 23, No. 2, 270-282 (1975)
10. J.M. Harrison and A. Zeevi, Dynamic scheduling of a multiclass queue in the Halfin-Whitt heavy traffic regime, Operations Research, Vol. 52, No. 2, 243-257 (2004)
11. A. Mandelbaum and A.L. Stolyar, Scheduling flexible servers with convex delay costs: heavy-traffic optimality of the generalized $c\mu$-rule, Operations Research, Vol. 52, No. 6, 836-855 (2004)
12. M. Pinedo, Stochastic scheduling with release dates and due dates, Operations Research, Vol. 31, 559-572 (1983)
13. M. Rebello and Z. Fernandes and C. Bispo, Simulating activity networks in JAVA, Proceedings of the 6th EUROSIM Congress on Modelling and Simulation, Ljubljana, Slovenia (2007)
14. R. Righter, *Scheduling. In* Stochastic Orders (M. Shaked and J.G. Shanthikumar eds.), Academic Press (1994)
15. W.E. Smith, Various optimizers for single-stage production, Naval Research Logistics Quarterly, Vol. 3, 59-66 (1956)
16. J.A. van Mieghem, Dynamic scheduling with convex delay costs: the generalized $c\mu$ rule, The Annals of Applied Probability, Vol. 5, No. 3, 808-833 (1995)