REPRINTED FROM:

83-05

SYSTEMS
& CONTROL
LETTERS

Volume 6, No. 3, August 1985

K competing queues with geometric service requirements and linear costs: The μc-rule is always optimal *

J.S. BARAS, D.-J. MA and A.M. MAKOWSKI
Electrical Engineering Department, University of Maryland, College Park, MD 20742, USA

pp. 173–180

NORTH-HOLLAND PUBLISHING COMPANY – AMSTERDAM

Systems & Control Letters 6 (1985) 173–180
North-Holland
August 1985

K competing queues with geometric service requirements and linear costs: The μc-rule is always optimal *

J.S. BARAS, D.-J. MA and A.M. MAKOWSKI

Electrical Engineering Department, University of Maryland, College Park, MD 20742, USA

Received 15 June 1984

Abstract: In this note, a discrete-time system of K competing queues with geometric service requirements and arbitrary arrival patterns is studied. When the cost per slot is linear in the queue sizes, it is shown that the μc-rule minimizes the expected discounted cost over the infinite horizon.

Keywords: Queues, μc-rule, Competing queues.

1. Introduction

In the context of resource sharing environments, a great many situations can be modelled by the following scenario: A natural time unit exists and is used to divide the time horizon into consecutive slots of unit length. The system is composed of a single server with the capability of providing several grades of service, each such grade of service being characteristic of a customer class (or equivalently of a queue). New customers arrive on a slot-per-slot basis and await service in an infinite capacity waiting room. At the beginning of a time slot, a customer class (or equivalently its queue) is selected to receive service attention during that time slot. The service requirements are distributed as geometric random variables with class dependent parameters, and are statistically independent from customer to customer. Note that this assignment of service attention may be *pre-emptive* as a customer is denied service attention before completion of its service requirement. Moreover the allocation of effort may fail to be *work-conserving* for the server may give service attention to a customer class with an empty queue.

Service attention to be given in a time slot is decided on the basis of past decisions and service completions, as well as of past and present arrival data.

A cost, linear in the queue sizes, is incurred for operating the system over one time slot and one seeks to select the service discipline so as to minimize the total expected discounted cost over an infinite horizon. The policy that allocates service attention to the non-empty queue with the largest expected cost decrease per slot is called the *μc-rule*. It is shown here that the μc-rule is optimal among all admissible allocation policies when the arrival streams have *arbitrary* statistics but are statistically *independent* of the service requirements. This work generalizes and extends the one presented by Baras, Dorsey and Makowski in [1] for the case of *two* competing queues under the more restrictive assumption that the arrivals are *independent* and *identically* distributed over time slots; there the problem was solved by Dynamic Programming as only feedback policies based on past queue *sizes* and *control* decisions were considered.

In this paper, the cost transformation of [1] is used to recast the problem as an *arm-acquiring bandit problem* [5]. Taking advantage of the very special structure of this auxiliary problem, one provides a direct argument for establishing the optimality of the μc-rule. This simple argument is a very natural extension of the one given in [5] to prove the optimality of the index rule for the simplest bandit problem.

* This work was supported in part by the Department of Energy under Grant DOE-AC01-78ET29244, A5 and the National Science Foundation under Grant ECS-82-04451.

A similar optimality result was established by Buyukkoc, Varaiya and Walrand in [2] and obtained via yet another line of argumentation.

2. The model

In this section a simple model is formulated that captures the evolution of the multi-queue system loosely described in the introduction. Throughout this paper, the number of competing queues is denoted by a fixed positive integer K.

To describe the model, one starts with a probability triple $(\Omega, \mathscr{F}, P)$ that simultaneously carries an N^K-valued random variable (RV) ξ, a sequence $\{A(t)\}_1^\infty$ of N^K-valued RV's and a sequence $\{B(n)\}_1^\infty$ of $\{0,1\}^K$-valued RV's. As a rule, the k-th component of a K-dimensional RV is always denoted by the same symbol subscripted by k.

The initial size (at time $t = 1$) of the k-th queue is represented by ξ_k, the RV $A_k(t)$ quantifies the arrivals to the queue during the time slot $[t, t+1)$ whereas $B_k(n)$ records service completion in the slot during which the k-th queue is non-empty and is given service attention for the n-th time.

Some technical assumptions are required; they are motivated on one hand by the need to capture the fact that the service requirements should be geometrically distributed and independent from customer to customer, and on the other hand by the desire to validate a useful cost transformation forthcoming in Section 3. Thus from now on the following assumption (A) is enforced:

(A) *The sequences $\{B_k(n)\}_1^\infty$ are Bernoulli sequences with parameter μ_k, $1 \leq k \leq K$, which are mutually independent of each other as well as of the RV ξ and of the arrival sequence $\{A(t)\}_1^\infty$.*

The assignment of service attention in the slot $[t, t+1)$ will be based on knowledge of the initial queue sizes, past and present arrival data over the horizon $[1, t+1)$, and past control values and service completion data over the horizon $[1, t)$. Here, an *admissible policy* π is any sequence $\{\pi_t\}_1^\infty$ of mappings π_t from $N^K \times N^K \times (\{1,\ldots,K\} \times \{0,1\}^K \times N^K)^{t-1}$ into $\{1,\ldots,K\}$, with the convention that the domain of definition of π_1 is simply $N^K \times N^K$. The collection of all such policies is denoted by $\mathscr{P}$ in the sequel.

For every policy π in $\mathscr{P}$ one recursively generates four sequences $\{X^\pi(t)\}_1^\infty$, $\{U^\pi(t)\}_1^\infty$, $\{N^\pi(t)\}_1^\infty$ and $\{B^\pi(t)\}_1^\infty$ of RV's taking values in N^K, $\{1,\ldots,K\}$, N^K and $\{0,1\}^K$ respectively. The RV $X_k^\pi(t)$ represents the number of customers present in the k-th queue at the end of the horizon $[1, t)$, $N_k^\pi(t)$ counts the number of slots over this same horizon during which the k-th queue was non-empty and received service attention, while $B_k^\pi(t)$ encodes the completion of service requirement in the slot $[t, t+1)$. To initialize the recursion, define $X^\pi(1) := \xi$ and $U^\pi(1) := \pi_1[\xi, A(1)]$, and finally for $1 \leq k \leq K$, set $N_k^\pi(1) := 0$ and $B_k^\pi(1) := 1[\xi_k]\delta[k, U^\pi(1)]B_k(N_k^\pi(1)+1)$. Here the expression $1[\ ]$ is defined for all n in N to be $1[n] = 0$ for $n = 0$ and $1[n] = 1$ for $n \neq 0$, and $\delta[\ ,\ ]$ stands for the standard Kronecker symbol. For $t = 1,2,\ldots$ set

$$X_k^\pi(t+1) := X_k^\pi(t) + A_k(t) - \delta[k, U^\pi(t)]\,B_k^\pi(t), \quad 1 \leq k \leq K, \tag{2.1}$$

$$U^\pi(t+1) := \pi_{t+1}[\xi, A(1);\ U^\pi(s), B^\pi(s), A(s+1), 1 \leq s \leq t], \tag{2.2}$$

and finally for $1 \leq k \leq K$,

$$B_k^\pi(t+1) := 1[X_k^\pi(t+1)]\,\delta[k, U^\pi(t+1)]\,B_k(N_k^\pi(t+1)+1), \tag{2.3}$$

$$N_k^\pi(t+1) := N_k^\pi(t) + 1[X_k^\pi(t)]\,\delta[k, U^\pi(t)]. \tag{2.4}$$

An admissible policy π in $\mathscr{P}$ (or equivalently the corresponding control sequence $\{U^\pi(t)\}_1^\infty$) is *idling at time t* if one can find k and l, $1 \leq k \neq l \leq K$, such that for some ω in Ω, $X_k^\pi(t, \omega) > 0$, $X_l^\pi(t, \omega) = 0$ but $U^\pi(t, \omega) = l$. An admissible policy π in $\mathscr{P}$ which is not idling at any time is called a *non-idling* or *work conserving* policy; the collection of all such policies is denoted by $\mathscr{N}$.

3. The optimization problems

A simple measure of performance is associated with the operation of this queueing system by imposing a cost per slot proportional to queue lengths. To that end, a set of K non-negative constants c_k, $1 \leqslant k \leqslant K$, is introduced and held fixed throughout the discussion. For every β in (0, 1), the *β-discounted expected cost $J_\beta(\pi)$ over the infinite horizon* associated with an admissible policy π in $\mathscr{P}$ is defined by

$$J_\beta(\pi) := E\left[\sum_{t=1}^{\infty} \beta^t \left[\sum_{k=1}^{K} c_k X_k^\pi(t)\right]\right], \tag{3.1}$$

with $\{X^\pi(t)\}_1^\infty$ generated by the dynamics (2.1)–(2.4) under the action of π. This note is devoted to studying the problems (P_β) and (N_β) of *minimizing $J_\beta(\pi)$ over the classes $\mathscr{P}$ and $\mathscr{N}$ of admissible policies*, respectively.

As in [1], it will be useful to transform the cost function (3.1) by injecting the queue size dynamics (2.1) into it. Anticipating the result, for every β in (0, 1) and every policy π in $\mathscr{P}$, one defines the quantity $\tilde{J}_\beta(\pi)$ as

$$\tilde{J}_\beta(\pi) := E\left[\sum_{t=1}^{\infty} \beta^t \left[\sum_{k=1}^{K} \mu_k c_k 1[X_k^\pi(t)]\,\delta[k, U^\pi(t)]\right]\right]. \tag{3.2}$$

A simple relationship exists between the two cost functions:

Proposition 3.1. *Under assumption* (A), *for every β in* (0, 1) *and every policy π in $\mathscr{P}$, the relation*

$$(1-\beta)J_\beta(\pi) = \beta E\left[\sum_{k=1}^{K} c_k \xi_k\right] + \beta E\left[\sum_{t=1}^{\infty} \beta^t \left[\sum_{k=1}^{K} c_k A_k(t)\right]\right] - \beta \tilde{J}_\beta(\pi) \tag{3.3}$$

holds true.

A few remarks are now in order at this stage of the discussion.

It is easy to see that the problems (P_β) and (N_β) are equivalent to the problems $(\tilde{P}_\beta)$ and $(\tilde{N}_\beta)$ of maximizing (3.2) over $\mathscr{P}$ and $\mathscr{N}$ respectively; this cost function (3.2) together with the dynamics (2.1)–(2.4) naturally defines $(\tilde{P}_\beta)$ and $(\tilde{N}_\beta)$ as *arm-acquiring bandit* problems of the type described in [5] and the results therein could thus be applied. However, the simple structure of the problem at hand allows for a simpler and more direct treatment which is presented in the next two sections. The elementary optimality argument given in [5, Theorem 2.1] for the basic bandit problem is adapted to accomodate the arrivals of new customers; the concepts of *superprocesses* and *machine domination* [5, Section 3], [6] need not be used for this very special situation.

In addition to providing insight into the structure of the problem [1], the cost transformation stated in Proposition 3.1 naturally singles out the very simple *μc-rule* as a candidate for optimality. The μc-rule is the policy π^* that gives service attention to the non-empty queue with the largest value for the products $\mu_k c_k$, $1 \leqslant k \leqslant K$, or equivalently, with the largest *expected cost decrease per slot*. For ease of presentation, one can assume without loss of generality that the queues are labelled so that

$$\mu_K c_K \leqslant \mu_{K-1} c_{K-1} \leqslant \cdots \leqslant \mu_2 c_2 \leqslant \mu_1 c_1. \tag{3.4}$$

In that event, the μc-rule is the policy π^* that selects the queue with the *smallest* index among all non-empty queues.

The special and simpler case $K = 2$ is discussed separately in the next section. This is done so as to better illustrate the basic ideas which in the general case may be masked by notational and technical difficulties. At any rate, the key idea in both cases can be loosely discribed as follows: Start with a non-idling policy π and a possibly random time σ. As in [5, Theorem 2.1], one seeks to construct a non-idling policy $\tilde{\pi}(\sigma)$ and another random time $\tau^\pi(\sigma)$ with the property that the policy $\tilde{\pi}(\sigma)$: (i) agrees with the policy π on $[1, \sigma)$, (ii) behaves like the μc-rule π^* on $[\sigma, \tau^\pi(\sigma))$ with $\sigma + 1 \leqslant \tau^\pi(\sigma)$ and (iii) improves upon the policy π in the sense that $\tilde{J}_\beta(\pi) \leqslant \tilde{J}_\beta(\tilde{\pi}(\sigma))$.

For future reference one now introduces the sequence $\{T^\pi(n)\}_1^\infty$ of N^K-valued RV's associated with a policy π in $\mathscr{P}$: for $1 \leqslant k \leqslant K$, the RV $T_k^\pi(n)$ represents the left boundary of the slot during which the k-th queue is non-empty and is given service attention for the n-th time under the action of π. It is worth noticing that for any non-idling policy π in $\mathscr{N}$, the cost (3.1) can now be expressed in the form

$$\tilde{J}_\beta(\pi) = E\left[\sum_{k=1}^{K} \mu_k c_k \left[\sum_{n=1}^{\infty} \beta^{T_k^\pi(n)}\right]\right]. \tag{3.5}$$

In the sequel, the time sequences that correspond to the policies π and $\tilde{\pi}(\sigma)$ will be denoted simply by $\{T(n)\}_1^\infty$ and $\{\tilde{T}(n)\}_1^\infty$, respectively, for ease of notation.

4. The case $K = 2$

The flow of information that corresponds to an admissible policy π in $\mathscr{P}$ is described by the filtration $(\mathscr{F}_t^\pi,\ t = 1,2,\dots)$ which is recursively defined by

$$\mathscr{F}_{t+1}^\pi := \mathscr{F}_t^\pi \vee \sigma\{U^\pi(t),\ B^\pi(t),\ A(t+1)\}, \quad t = 1,2,\dots, \tag{4.1}$$

with $\mathscr{F}_1^\pi := \sigma\{\xi,\ A(1)\}$. The admissibility of π is easily seen to be equivalent to the $\mathscr{F}_t^\pi$-measurability of the RV $U^\pi(t)$ for all $t = 1,2,\dots$.

To study the case $K = 2$, start with a non-idling policy π in $\mathscr{N}$ and an $\mathscr{F}_t^\pi$-stopping time σ, and define the $\mathscr{F}_\sigma^\pi$-measurable set $A^\pi(\sigma)$ to be $A^\pi(\sigma) := [X_1^\pi(\sigma) > 0]$. An auxiliary policy $\pi^*(\sigma)$ is obtained by first following π on the horizon $[1, \sigma)$ and then switching to the μc-rule from time σ onward. The N-valued RV's $\varepsilon^\pi(\sigma)$ and $\gamma^\pi(\sigma)$ are defined now as

$$\varepsilon^\pi(\sigma) := \inf\{t \geqslant 0:\ X_1^{\pi^*(\sigma)}(\sigma + t) = 0\}, \qquad \gamma^\pi(\sigma) := \begin{cases} \varepsilon^\pi(\sigma) & \text{on } A^\pi(\sigma), \\ 1 & \text{on } \Omega \backslash A^\pi(\sigma), \end{cases} \tag{4.2}$$

and obviously $1 \leqslant \gamma^\pi(\sigma)$. Finally, define the RV $\nu^\pi(\sigma)$ to be the *total* number of slots it will take under the action of π for the first queue to be given service attention *exactly* $\gamma^\pi(\sigma)$ times on the horizon $[\sigma, +\infty)$.

To define $\tilde{\pi}(\sigma)$, generate the following actions: First use the policy π on the horizon $[1, \sigma)$ and then

(i) on the set $\Omega \backslash A^\pi(\sigma)$, keep on using π from time σ onward; observe that $\tilde{\pi}(\sigma)$ is non-idling and behaves like the μc-rule on $[\sigma, \sigma + \gamma^\pi(\sigma))$, owing to (4.2) and to the fact that π itself is non-idling,

(ii) on the set $A^\pi(\sigma)$,

(a) use the μc-rule on $[\sigma, \sigma + \gamma^\pi(\sigma))$ to give service attention to the first queue,

(b) give service attention to the second queue on $[\sigma + \gamma^\pi(\sigma), \sigma + \nu^\pi(\sigma))$,

(c) generate the *same* actions as the policy π from time $\sigma + \nu^\pi(\sigma)$ onward.

Owing to the special structure of the dynamics (2.1)–(2.4), it follows that $\tilde{\pi}(\sigma)$ is an admissible policy and thus in $\mathscr{N}$ since non-idling. Moreover, under both policies π and $\tilde{\pi}(\sigma)$, each queue is given service attention during the exact same number of slots over the horizon $[\sigma, \sigma + \nu^\pi(\sigma))$. Using the convention adopted in Section 3, one first observes that

$$T_1(n) \begin{cases} \geqslant \tilde{T}_1(n) := [\sigma - 1] + [n - N_1^\pi(\sigma)] & \text{on } A^\pi(\sigma) \text{ if } N_1^\pi(\sigma) < n \leqslant N_1^\pi(\sigma) + \gamma^\pi(\sigma), \\ = \tilde{T}_1(n) & \text{otherwise}, \end{cases} \tag{4.3}$$

while upon setting $\tau^\pi(\sigma) := \sigma + \gamma^\pi(\sigma)$, one concludes that

$$T_2(n) \begin{cases} \leqslant \tilde{T}_2(n) := [\tau^\pi(\sigma) - 1] + [n - N_2^\pi(\sigma)] & \text{on } A^\pi(\sigma) \text{ if } N_2^\pi(\sigma) < n \leqslant N_2^\pi(\sigma + \nu^\pi(\sigma)), \\ = \tilde{T}_2(n) & \text{otherwise}. \end{cases} \tag{4.4}$$

It is easy to see that $\gamma^{\pi}(\sigma)$ is an $\mathscr{F}_{\sigma+t}^{\tilde{\pi}(\sigma)}$-stopping time and therefore that $\tau^{\pi}(\sigma)$ is an $\mathscr{F}_t^{\tilde{\pi}(\sigma)}$-stopping time, while from (4.3) and (4.4) one concludes that

$$\sum_{k=1}^{2}\sum_{n=1}^{\infty}\beta^{\tilde{T}_k(n)}=\sum_{k=1}^{2}\sum_{n=1}^{\infty}\beta^{T_k(n)}. \tag{4.5}$$

Invoking (4.4) again, one readily obtains the inequality

$$\sum_{n=1}^{\infty}\beta^{\tilde{T}_2(n)}-\sum_{n=1}^{\infty}\beta^{T_2(n)}=\sum_{N_2^{\pi}(\sigma)<n\leqslant N_2^{\pi}(\sigma+\nu^{\pi}(\sigma))}\left[\beta^{\tilde{T}_2(n)}-\beta^{T_2(n)}\right]\leqslant 0 \tag{4.6}$$

and its immediate corrolary

$$\mu_1c_1\left[\sum_{n=1}^{\infty}\beta^{\tilde{T}_2(n)}-\sum_{n=1}^{\infty}\beta^{T_2(n)}\right]\leqslant\mu_2c_2\left[\sum_{n=1}^{\infty}\beta^{\tilde{T}_2(n)}-\sum_{n=1}^{\infty}\beta^{T_2(n)}\right], \tag{4.7}$$

where the passage from (4.6) to (4.7) is justified by (3.4). Combining (4.5) and (4.7) with (3.5), one finally gets that

$$\tilde{J}_{\beta}(\tilde{\pi}(\sigma))-\tilde{J}_{\beta}(\pi)\geqslant\mu_1c_1E\left[\sum_{k=1}^{2}\left[\sum_{n=1}^{\infty}\beta^{\tilde{T}_k(n)}-\beta^{T_k(n)}\right]\right]=0, \tag{4.8}$$

i.e. the policy $\tilde{\pi}(\sigma)$ does better than π!

The discussion that leads to (4.8) can now be summarized as follows:

Theorem 4.1. *For every non-idling policy π in $\mathscr{N}$ and every $\mathscr{F}_t^{\pi}$-stopping time σ, there always exists a non-idling policy $\tilde{\pi}(\sigma)$ that agrees with the policy π on* $[1, \sigma)$, *behaves like the μc-rule π^* on* $[\sigma, \tau^{\pi}(\sigma))$ *where $\tau^{\pi}(\sigma)$ is an $\mathscr{F}_t^{\tilde{\pi}(\sigma)}$-stopping time with $\sigma+1\leqslant\tau^{\pi}(\sigma)$, and improves upon the policy π.*

The following result is now straightforward by a simple induction argument:

Theorem 4.2. *For every β in* $(0, 1)$, *the μc-rule π^* solves the two-competing queue problem* $(\tilde{N}_{\beta})$ *as it maximizes the cost* (3.4) *over the class $\mathscr{N}$ of all non-idling policies.*

Leaving the details of the proof to the interested reader, one now moves on to the general case in the next section, where the main optimality results of this work are stated and derived.

5. The general case

The argument given in the previous section for $K=2$ can be adapted to the general case. However, some care should be exercised in the forthcoming discussion as the constructed policy $\tilde{\pi}(\sigma)$ will turn out in general *not* to be admissible in $\mathscr{N}$ and the line of argumentation thus seemingly breaks down. Fortunately, the very special structure of the problem will help remedy to this difficulty by introducing an enlarged collection of 'policies': an *assignment* π is any sequence $\{\pi_t\}_1^{\infty}$ of mappings π_t from $N^K\times N^K\times(\{1,\ldots,K\}\times\{0,1\}^K\times N^K)^{t-1}\times(\{0,1\}^K)^{\infty}$ into $\{1,\ldots,K\}$, with the convention that the domain of definition of π_1 is simply $N^K\times N^K\times(\{0,1\}^K)^{\infty}$. The collection of all such assignments is denoted by $\mathscr{P}_{\text{ext}}$ in the sequel. The notion of *idling assignment* parallels the one of idling policy given in Section 2 and the collection of all non-idling assignments is denoted by $\mathscr{N}_{\text{ext}}$. For every assignment π in $\mathscr{P}_{\text{ext}}$, one can generate the exact same quantities that were generated in previous sections for an admissible policy in $\mathscr{P}$; the only difference lies now in the fact that the control sequence $\{U^{\pi}(t)\}_1^{\infty}$ is given by

$$U^{\pi}(t+1):=\pi_{t+1}[\xi,\ A(1);\ U^{\pi}(s),\ B^{\pi}(s),\ A(s+1),\ 1\leqslant s\leqslant t;\ B(r),\ 1\leqslant r] \tag{5.1}$$

for $t = 1,2,\ldots$ with $U^{\pi}(1) := \pi_1[\xi, A(1); B(r), 1 \leqslant r]$. In other words, at each moment, the server has knowledge of *all possible* past, present and future service completion data! Obviously, the inclusions $\mathscr{P} \subseteq \mathscr{P}_{\text{ext}}$ and $\mathscr{N} \subseteq \mathscr{N}_{\text{ext}}$ must hold true.

The discussion can now proceed. Consider a non-idling assignment π in $\mathscr{N}_{\text{ext}}$ and an arbitrary N-valued RV σ, and define the $\{0,1,\ldots, K\}$-valued RV $k^{\pi}(\sigma)$ as

$$k^{\pi}(\sigma) := \begin{cases} \min\{k: 1 \leqslant k \leqslant K \text{ and } X_k^{\pi}(\sigma) > 0\} & \text{if this set is non-empty,} \\ 0 & \text{otherwise.} \end{cases} \tag{5.2}$$

Now, denote by $\pi^*(\sigma)$ the auxiliary assignment in $\mathscr{N}_{\text{ext}}$ obtained by first following the assignment π on the horizon $[1, \sigma)$ and then switching to the μc-rule π^* from time σ onward. The N-valued RV's $\varepsilon^{\pi}(\sigma)$, $\alpha_k(\sigma)$, $1 \leqslant k \leqslant K$, and $\alpha^{\pi}(\sigma)$ are then defined as follows:

$$\varepsilon^{\pi}(\sigma) := \begin{cases} \inf\{t \geqslant 0: X_{k^{\pi}(\sigma)}^{\pi^*(\sigma)}(\sigma + t) = 0\} & \text{if } 0 < k^{\pi}(\sigma), \\ 1 & \text{otherwise,} \end{cases} \tag{5.3}$$

$$\alpha_k(\sigma) := \inf\{t \geqslant 0: A_k(\sigma + t) \geqslant 0\}, \qquad \alpha(\sigma) := 1 + \min\{\alpha_k(\sigma), 1 \leqslant k \leqslant k^{\pi}(\sigma)\}. \tag{5.4}$$

In all cases one adopts the usual convention that the RV takes on the value ∞ when the defining set is empty. The RV $\gamma^{\pi}(\sigma)$ is now introduced as $\gamma^{\pi}(\sigma) := \min\{\varepsilon^{\pi}(\sigma), \alpha^{\pi}(\sigma)\}$. Obviously $\gamma^{\pi}(\sigma) \geqslant 1$ and a moment of reflection should convince the reader that when the μc-rule operates over the horizon $[\sigma, \sigma + \gamma^{\pi}(\sigma))$, service attention is constantly given out to the $k^{\pi}(\sigma)$-th queue, provided $k^{\pi}(\sigma) > 0$. On the set $[k^{\pi}(\sigma) > 0]$, one concludes that under both policies π and $\pi^*(\sigma)$,

$$X_k^{\pi}(\sigma + t) = X_k^{\pi^*(\sigma)}(\sigma + t) = 0 \tag{5.5}$$

for $1 \leqslant k < k^{\pi}(\sigma)$ and $0 \leqslant t < \gamma^{\pi}(\sigma)$; since π is a *non*-idling assignment, it follows from the definition of $\gamma^{\pi}(\sigma)$ that on the horizon $[\sigma, \sigma + \gamma^{\pi}(\sigma))$, the $k^{\pi}(\sigma)$-th queue is never empty and therefore $k^{\pi}(\sigma) \leqslant U^{\pi}(\sigma + t)$ for $0 \leqslant t < \gamma^{\pi}(\sigma)$ by virtue of (5.5).

Define the RV $\nu^{\pi}(\sigma)$ to be the *total* number of slots it will take under assignment π for the $k^{\pi}(\sigma)$-th queue to be given service attention *exactly* $\gamma^{\pi}(\sigma)$ times on the horizon $[\sigma, +\infty)$, provided $k^{\pi}(\sigma) > 0$; otherwise set $\nu^{\pi}(\sigma) = 1$. To define $\tilde{\pi}(\sigma)$ generate the following actions: First use the assignment π on the horizon $[1, \sigma)$ and then

(i) if $k^{\pi}(\sigma) = 0$, keep on using the assignment π from time σ onward,

(ii) if $k^{\pi}(\sigma) > 0$,

(a) on the horizon $[\sigma, \sigma + \gamma^{\pi}(\sigma))$, use the μc-rule to give service attention to the $k^{\pi}(\sigma)$-th queue,

(b) on the horizon $[\sigma + \gamma^{\pi}(\sigma), \sigma + \nu^{\pi}(\sigma))$, give service attention to the queues with label $1 \leqslant k < k^{\pi}(\sigma)$ in the *same slots* as if the assignment π were used from time σ onward and give service attention to the queues with label $k^{\pi}(\sigma) < k \leqslant K$ in the remaining slots of the horizon $[\sigma + \gamma^{\pi}(\sigma), \sigma + \nu^{\pi}(\sigma))$ but in the *same order* as if the assignment π were used from time σ onward,

(c) on the horizon $[\sigma + \nu^{\pi}(\sigma), +\infty)$, generate the *same* actions as the assignment $\tilde{\pi}$.

Owing to the special structure of the system dynamics and to the definition (5.1) of assignment, it follows that $\tilde{\pi}(\sigma)$ is a non-idling assignment in $\mathscr{N}_{\text{ext}}$ for π is an assignment in $\mathscr{N}_{\text{ext}}$. It should be pointed out that even if π were an *admissible policy* in $\mathscr{N}$, the assignment $\tilde{\pi}(\sigma)$ would *not* be in $\mathscr{N}$ in general; this follows from the fact that on the horizon $[\sigma + \gamma^{\pi}(\sigma), \sigma + \nu^{\pi}(\sigma))$, the assignment $\tilde{\pi}(\sigma)$ gives service attention to the queues with label $1 \leqslant k < k^{\pi}(\sigma)$ in the *same* slots as if the assignment π were used from time σ onward.

Under both assignments π and $\tilde{\pi}(\sigma)$, each queue is given service attention during the exact same number of slots over the horizon $[\sigma, \sigma + \nu^{\pi}(\sigma))$. Using the convention adopted in Section 3, one observes the following facts:

For $1 \leqslant k < k^{\pi}(\sigma)$,

$$\tilde{T}_k(n) = T_k(n), \quad n = 1,2,\ldots . \tag{5.6}$$

For $k = k^\pi(\sigma)$,

$$\tilde{T}_k(n)\begin{cases} < T_k(n) & \text{on the set } [k^\pi(\sigma) > 0] \text{ if } N_k^\pi(\sigma) < n \leqslant N_k^\pi(\sigma) + \gamma^\pi(\sigma), \\ = T_k(n) & \text{otherwise.} \end{cases} \tag{5.7}$$

For $k^\pi(\sigma) < k \leqslant K$,

$$\tilde{T}_k(n)\begin{cases} > T_k(n) & \text{on the set } [k^\pi(\sigma) > 0] \text{ if } N_k^\pi(\sigma) < n \leqslant N_k^\pi(\sigma + \nu^\pi(\sigma)), \\ = T_k(n) & \text{otherwise.} \end{cases} \tag{5.8}$$

Combining (3.5) and (5.6), one readily sees that

$$\tilde{J}_\beta(\tilde{\pi}(\sigma)) - \tilde{J}_\beta(\pi) = E\left[\sum_{k=1}^{K} \delta[k, k^\pi(\sigma)] \sum_{l=k}^{K} \mu_l c_l \sum_{n=1}^{\infty} \left[\beta^{\tilde{T}_l(n)} - \beta^{T_l(n)}\right]\right]. \tag{5.9}$$

Upon inspection, it follows from (5.8) that for $1 \leqslant k < l \leqslant K$,

$$\delta[k, k^\pi(\sigma)] \sum_{n=1}^{\infty} \left[\beta^{\tilde{T}_l(n)} - \beta^{T_l(n)}\right] \leqslant 0, \tag{5.10}$$

and subsequently by making use of (3.4) one concludes that

$$\delta[k, k^\pi(\sigma)] \sum_{l=k+1}^{K} \mu_l c_l \sum_{n=1}^{\infty} \left[\beta^{\tilde{T}_l(n)} - \beta^{T_l(n)}\right] \geqslant \delta[k, k^\pi(\sigma)] \mu_k c_k \sum_{l=k+1}^{K} \sum_{n=1}^{\infty} \left[\beta^{\tilde{T}_l(n)} - \beta^{T_l(n)}\right]. \tag{5.11}$$

Substituting (5.11) into (5.9) one immediately concludes that

$$\tilde{J}_\beta(\tilde{\pi}(\sigma)) - \tilde{J}_\beta(\pi) \geqslant E\left[\sum_{k=1}^{K} \delta[k, k^\pi(\sigma)] \mu_k c_k \sum_{l=k}^{K} \sum_{n=1}^{\infty} \left[\beta^{\tilde{T}_l(n)} - \beta^{T_l(n)}\right]\right]. \tag{5.12}$$

But one verifies by the very definition of $\tilde{\pi}(\sigma)$ that for $1 \leqslant k \leqslant K$,

$$\delta[k, k^\pi(\sigma)] \sum_{l=k}^{K} \sum_{n=1}^{\infty} \left[\beta^{\tilde{T}_l(n)} - \beta^{T_l(n)}\right] = 0 \tag{5.13}$$

and the conclusion than $\tilde{J}_\beta(\pi) \leqslant \tilde{J}_\beta(\tilde{\pi}(\sigma))$ follows from (5.12).

The discussion is summarized by the following proposition, where $\tau^\pi(\sigma)$ is to be identified with $\sigma + \gamma^\pi(\sigma)$:

Proposition 5.1. *For every non-idling assignment π in $\mathcal{N}_{\text{ext}}$ and every N-valued RV σ, there always exists a non-idling assignment $\tilde{\pi}(\sigma)$ in $\mathcal{N}_{\text{ext}}$ that agrees with the assignment π on $[1, \sigma)$, behaves like the μc-rule π^* on $[\sigma, \tau^\pi(\sigma))$, where $\tau^\pi(\sigma)$ is an N-valued RV with $\sigma + 1 \leqslant \tau^\pi(\sigma)$, and improves upon the assignment π.*

Proposition 5.1 constitutes the key step for showing the optimality of the μc-rule. Indeed, start with an arbitrary *policy* π in $\mathcal{N}$ and recursively generate via Proposition 5.1 a sequence $\{\sigma_n\}_1^\infty$ of N-valued RV's and a sequence $\{\pi^{(n)}\}_1^\infty$ of assignments in $\mathcal{N}_{\text{ext}}$ as follows: Set $\sigma_1 \equiv 1$ and $\pi^{(1)} \equiv \pi$ and for all $n = 1,2,\ldots$,

$$\pi^{(n+1)} := \pi^{(n)}(\sigma_n), \qquad \sigma_{n+1} := \tau^{\pi(n)}(\sigma_n). \tag{5.14}$$

Now, $\tilde{J}_\beta(\pi^{(n)}) \leqslant \tilde{J}_\beta(\pi^{(n+1)})$ and $\pi^{(n)}$ behaves like the μc-rule π^* on the horizon $[1, \sigma_n)$ with $\lim_n \sigma_n = \infty$. The following result is now straightforward:

Theorem 5.2. *For every β in* (0, 1) *the μc-rule π^* solves the problem $(\tilde{N}_\beta)$ as it maximizes the cost* (3.4) *over the class $\mathcal{N}$ of all non-idling policies.*

It is worth observing here that although for most n, the assignment $\pi^{(n)}$ is *not* in $\mathcal{N}$, the 'limit' of the $\pi^{(n)}$'s is the μc-rule π^* which of course is in $\mathcal{N}$ and the conclusion of Theorem 5.2 follows. Finally, one can now show that idling does not pay:

Theorem 5.3. *For every β in* (0, 1), *the μc-rule π^* solves problem (P_β) as it maximizes the cost* (3.1) *over the class $\mathcal{P}$ of all admissible policies.*

Proof. The basic idea of the proof is to embed the problem of K competing queues into one for $K+1$ competing queues with the following special structure:

(i) The costs for the K first queues are still c_k, $1 \leqslant k \leqslant K$, but now $c_{K+1} = 0$.

(ii) The $(K+1)$-st queue is initially non-empty and at least one new customer arrives to it in each time slot; for sake of concreteness take $\xi_{K+1} = 1$ and $A_{K+1}(t) = 1$ for all $t = 1, \ldots$. As a consequence, the $(K+1)$-st queue never becomes empty under the action of any policy.

To complete the embedding, any admissible policy π for the original problem is put in correspondence with an admissible policy $\pi(K+1)$ for the $K+1$ competing queue problem as follows: $\pi(K+1)$ takes the same actions as π when π does not idle but gives service attention to the $(K+1)$-st queue when π idles *or* when all K first queues are empty. Since idling incurs no reward and $c_{K+1} = 0$, it is easy to see that the costs for operating under π (in the K competing queue system) and under $\pi(K+1)$ (in the $K+1$ competing queue system) are the *same*. Note that if π^* is the μc-rule for the original problem, then $\pi^*(K+1)$ is exactly the μc-rule for the $K+1$ competing queue system by virtue of (i) and (ii). Hence, invoking Theorem 5.2, which is valid for systems with an *arbitrary* number of competing queues, one concludes that the μc-rule maximizes the cost (3.2) over the class of *all* admissible policies!

Combining Proposition 3.1 and Theorems 5.2 and 5.3, one obtains the main result of this note:

Theorem 5.4. *Under assumption* (A), *the μc-rule π^* solves problems (P_β) and (N_β) for every β in* (0, 1).

The simplicity, and yet generality, of these results is quite remarkable and somewhat surprising in the light of works by of Harrison [3] and others for the continuous-time dynamic scheduling problem. This may probably be traced back to the discrete nature of time, the geometric assumption on the service times and to the fact that the service discipline is pre-emptive *resume*. The reader is also referred to the paper of Meilijson and Yechiali [4] for a discussion of the optimality of the μc-rule in the context of $G|G|1$ queueing systems with arbitrary arrival streams.

Extensions of Theorems 5.2 and 5.3 can be given for general discrete-time Jacksonian networks *without* feedback where at each node service requirements are geometric and independent from customer to customer.

References

[1] J.S. Baras, A.J. Dorsey and A.M. Makowski, Two competing queues with linear costs: the μc-rule is often optimal, *J. Appl. Probab.*, submitted (1983).

[2] C. Buyukkoc, P. Varaiya and J. Walrand, the $c\mu$ rule revisited, *J. Appl. Probab.*, submitted (1983).

[3] J.M. Harrison, Dynamic scheduling of a multiclass queue: discount optimality, *Oper. Res.* **23** (1975) 270–282.

[4] I. Meilijson and U. Yechiali, On optimal right-of-way policies at a single-server station when insertion of idle times is permitted, *Stochast. Proc. Appl.* **6**(1) (1977) 25–32.

[5] P. Varaiya, J. Walrand and C. Buyukkoc, Extensions of the multi-armed bandit problem, *IEEE Trans. Automat. Control.*, submitted (1983).

[6] P. Whittle, Multi-armed bandits and the Gittins index, *J. Roy. Statist. Soc.* **42** (1980) 143–149.