OPTIMAL SCHEDULING IN A MACHINE WITH STOCHASTIC VARYING PROCESSING RATE

Philippe NAIN[1]* and Don TOWSLEY[2]†

[1]INRIA, B.P. 93, 06902, Sophia Antipolis Cedex, France

[2]Department of Computer Science
University of Massachusetts, Amherst, MA 01003, USA

Published in: **IEEE Trans. Automatic Control, 39 (1994) 1853-1855**

Abstract

We address the problem of allocating the capacity of a machine among jobs of different classes when the machine processing rate varies stochastically over time. We establish that the policy that always allocates the maximum available processing rate to the class having the maximum weight minimizes, pathwise, a weighted sum of the remaining service requirements of the different classes, at any point in time. This result is based on the application of elementary forward induction arguments and holds over the class of all policies (e.g., including randomized policies). As an easy corollary of this result we generalize a recent work by Hirayama and Kijima [5] on the optimality of the μc-rule in a multiclass G/M/1 queueing system in which the server processing rate varies stochastically with time. To the best of our knowledge, our proof is the first one in this context that only uses direct pathwise arguments.

*This author was supported in part by the CEC DG XIII under the ESPRIT BRA grant QMIPS.
†This author was supported in part by NSF under grants ASC-8802764 and NCR-9116183.

1 Mathematical Formulation

There are K classes of jobs to be processed on a machine whose processing rate varies stochastically over time. Each arriving job carries with it a random service requirement and leaves the machine as soon as it completes service.

At point in time a controller has to decide which fraction of the machine processing rate should be allocated to each class of job. The way the processing rate allocated to a class is split among the jobs of this class is irrelevant here because of the cost functions to be considered. Therefore, we shall assume without loss of generality that the oldest job of each class present in the system receives all of the processing rate allocated to the class it belongs to (i.e., first-in-first out service discipline; see Remark 3.2).

The objective is to find an allocation policy of the processing rate that minimizes some weighted cost function, namely, a weighted sum of the remaining service requirements and a weighted sum of the expected number of jobs in the different classes.

We now give a precise description of the mathematical model. Consider the spaces $\Omega_k^1 = \Omega_k^2 = [0,+\infty]^{\mathbb{N}}$, $k = 1,2,\ldots,K$, where $\mathbb{N}$ is the set of all nonnegative integers. Let $(A_{n,k})_{n=1}^{\infty}$ and $(S_{n,k})_{n=1}^{\infty}$ be the coordinate processes of Ω_k^1 and Ω_k^2, respectively, namely $A_{n,k}((x_n)_n) = S_{n,k}((x_n)_n) = x_n$ for all $(x_n)_n \in [0,+\infty)^{\mathbb{N}}$, $k = 1,2,\ldots,K$. $A_{n,k}$ and $S_{n,k}$ will represent the arrival time and service requirement, respectively, of the nth job of class k. We also introduce Ω^3 to be the set of all measurable mappings from $[0,+\infty)$ into $[0,1]$. Let $R := \{R_t, t \geq 0\}$ be the coordinate process of Ω^3 defined by $R_t(f) = f(t) \in [0,1]$ for all $f \in \Omega^3$, $t \geq 0$. In the following, R_t will represent the processing rate available at time t.

We assume that $\Omega^i := \times_{k=1}^K \Omega_k^i$, $i = 1,2$, is endowed with the σ-algebra $\mathcal{A}^i$, $i = 1,2$, of its Borel sets and that Ω^3 is endowed with the smallest σ-algebra $\mathcal{A}^3 := \sigma(R_t,\, t \geq 0)$ with respect to which every R_t is measurable (see [10, Ch. III, Sec. 3]).

Let P be any probability measure on the measurable space $(\Omega, \mathcal{F}) := (\times_{i=1}^3 \Omega^i, \otimes_{i=1}^3 \mathcal{A}^i)$ such that

(1) $0 \leq A_{1,k} \leq A_{2,k} \leq \cdots \leq A_{n,k} \leq A_{n+1,k} \leq \cdots$ a.s. for $k = 1,2,\ldots,K$

(2) $\lim_{n\to\infty} A_{n,k} = +\infty$ a.s. for $k = 1,2,\ldots,K$

(3) $S_{n,k} < +\infty$ a.s. for $k = 1,2,\ldots,K$, $n = 1,2,\ldots$.

Let $\mathcal{B}$ be the Borel σ-algebra of $[0,+\infty)$. A *Resource Allocation Policy* (RAP) is a $\mathcal{B}\otimes\mathcal{F}$ -measurable mapping $\pi : [0,+\infty)\times\Omega \to [0,1]^K$ with $\pi(t,\omega) := (\pi_1(t,\omega),\ldots,\pi_K(t,\omega))$, such that $\sum_{k=1}^K \pi_k(t,\cdot) \leq 1$ for every $t \geq 0$.

For every $k = 1,2,\ldots,K$, $\pi_k(t,\omega) \in [0,1]$ will represent the fraction of processing rate allocated to the oldest job of class k (if any) at time t on the path ω. So, if $\omega = (\omega_1,\omega_2,\omega_3)$, $\omega_i \in \Omega^i$, $i = 1,2,3$,

then $\int_s^t \pi_k(u,\omega)\omega_3(u)du$ will give the total processing rate allocated to jobs a class k in the interval of time (s,t) on the path ω under policy π.

We shall denote by Π the set of all RAP's.

It is worth observing from the definition of a RAP that a new allocation of the machine processing rate may be made at any time. For instance, the process π defined by $\pi_k(t,\omega) = |\sin(t\,g(\omega))|/K$ for $k = 1, 2, \ldots, K$, where g is a $\mathcal{F}$-measurable mapping from Ω into $(-\infty, +\infty)$, and where the allocation of the machine processing rate changes continuously over the time, belongs to Π. Also observe that at any point in time, a RAP can use information regarding future arrivals, service times of future jobs as well as service times of jobs presently in the system.

We shall assume without loss of generality that the machine is empty at time $t = 0^-$. Then, any RAP π defines the $\mathcal{B} \otimes \mathcal{F}$-measurable processes $Q^\pi := (Q_1^\pi, \ldots, Q_K^\pi) \in \mathbb{N}^K$ and $V^\pi := (V_1^\pi, \ldots, V_K^\pi) \in [0, +\infty]^K$, where $Q_k^\pi(t,\omega)$ and $V_k^\pi(t,\omega)$ represent the number of jobs of class $k = 1, 2, \ldots, K$ at time t and the total remaining service requirement of jobs of class k at time t, respectively, on the path ω. We shall assume that the (piecewise continuous) sample-paths of Q_k^π and V_k^π, $k = 1, 2, \ldots, K$, are right-continuous. The construction of both processes Q^π and V^π is a standard exercise that is left to the reader.

In Section 2 we consider a weighted sum of the remaining service requirements of the different classes. We establish that the RAP that always gives the maximum available processing rate to the class with the highest weight minimizes pathwise the cost function $\sum_{k=1}^K r_k\, V_k^\pi(t)$ when $r_1 \geq r_2 \geq \cdots \geq r_K \geq 0$. The proof relies on elementary forward induction arguments.

In Section 3 we derive a new proof of the optimality of the celebrated *μc-rule* (see for instance [1], [2], [3], [4], [5], [6]) for G/M/1 queueing systems (i.e., when job service requirements are exponentially distributed) with randomly varying processing rates as an easy corollary of the result in Section 2. More precisely, we show that the cost function $\sum_{k=1}^K c_k\, E[Q_k^\pi(t)]$ where $c_1, c_2, \ldots, c_K$ are nonnegative constants is minimized by the RAP that always allocates the maximum available processing rate to the class with the highest value of $\mu_k c_k$ where $1/\mu_k$ is the expected service requirement for jobs of class k. The minimization is over the set $\Gamma \subset \Pi$ of policies that do not know present and future service requirements (see Section 3). This generalizes a recent result by Hirayama and Kijima [5] (in [5, Theorem 5] only nonidling policies are considered and $R_t = 0$ or 1; note, however, that non-exponential service requirement distributions are considered in [5]).

To the best of the authors' knowledge this is the first time that the optimality of the μc-rule is established via direct pathwise arguments as opposed to all previous proofs that are based either on interchange arguments [1], [2], [5], [8], [9], [11], dynamic programming arguments [1], [4] or on polymatroid theory [12].

We conclude this paper (Section 4) by extending the above mathematical model to randomized policies. We shall observe that the policies found optimal in Sections 2 and 3 yet remain optimal over the set Π extended to randomized policies.

2 Optimizing a Weighted Sum of the Remaining Processing Requirements

In this section we consider the cost function $\sum_{k=1}^{K} r_k V_k^{\pi}(t)$ where the weights $(r_k)_{k=1}^{K}$ satisfy $r_1 \geq r_2 \geq \cdots \geq r_K \geq 0$. Denote by γ the policy that gives *at any time* the maximum available processing rate to the class with the highest weight. In other words, if n_k is the number of jobs of class k at a reallocation time then all of the available processing rate is allocated to class $j = \min\{k = 1, 2, \ldots, K,\ n_k > 0\}$. The proof that γ belongs to Π is left to the reader.

The main result of this section is the following:

Proposition 2.1

$$\sum_{k=1}^{K} r_k V_k^{\gamma}(t) \leq \sum_{k=1}^{k} r_k V_k^{\pi}(t) \quad a.s. \tag{2.1}$$

for all $t \geq 0$ *and* $\pi \in \Pi$.

Proposition 2.1 follows from the following two lemmas:

Lemma 2.1 *Let* $(a_1, \ldots, a_K)$ *and* $(b_1, \ldots, b_K)$ *be* $[0, \infty)^K$*-valued vectors such that* $\sum_{i=1}^{k} a_i \leq \sum_{i=1}^{k} b_i$ *for* $k = 1, 2, \ldots, K$. *Then,*

$$\sum_{k=1}^{K} r_k a_k \leq \sum_{i=1}^{K} r_k b_k. \tag{2.2}$$

Proof. With $r_{K+1} = 0$ one has

$$\sum_{k=1}^{K} r_k a_k = \sum_{k=1}^{K} (r_k - r_{k+1}) \sum_{i=1}^{k} a_i$$

from which the result follows.

■

Lemma 2.2

$$\sum_{i=1}^{k} V_i^{\gamma}(t) \leq \sum_{i=1}^{k} V_i^{\pi}(t) \quad a.s. \tag{2.3}$$

for $k = 1, 2, \ldots, K$, $t \geq 0$ *and for all* $\pi \in \Pi$.

Proof. Let $\pi \in \Pi$ be an arbitrary RAP. Let $(t_n)_{n=1}^{\infty}$, $0 \leq t_1 < t_2 < \cdots$ be the sequence resulting from the superposition of the K arrival processes $(A_{n,k})_{n,k}$, of the K departure processes in the

system governed by policy γ, and of the K departure processes in the system governed by policy π (simultaneous events are allowed). Observe that $\lim_{n\to\infty} t_n = +\infty$ a.s. thanks to condition (2) in Section 1.

Fix ω in Ω. The proof is by induction on the times of events.

Basis step. Because the machine is empty at time $t = 0^-$, (2.3) trivially holds for $0 \le t \le t_1$.

Induction step. Assume that (2.3) holds for $0 \le t \le t_n$ and let us show that it is still true for $t_n < t \le t_{n+1}$. There are two steps.

Step 1: $t_n < t < t_{n+1}$.

If $\sum_{i=1}^K V_i^\gamma(t_n) = 0$ then (2.3) clearly holds for $t_n < t < t_{n+1}$.

Consider the case that $\sum_{i=1}^K V_i^\gamma(t_n) > 0$ and let $l = \min\{i = 1, 2, \ldots, K : V_i^\gamma(t_n) > 0\}$. By the definition of γ and of the sequence $(t_n)_{n=1}^\infty$ we have

$$(V_1^\gamma(t), \ldots, V_K^\gamma(t)) = \left(0, \ldots, 0, V_l^\gamma(t_n) - \int_{t_n}^t R_s\, ds, V_{l+1}^\gamma(t_n), \ldots, V_K^\gamma(t_n)\right). \tag{2.4}$$

For $k = 1, 2, \ldots, l-1$, it is seen from (2.4) that

$$0 = \sum_{i=1}^k V_i^\gamma(t) \le \sum_{i=1}^k V_i^\pi(t).$$

On the other hand, we have for $k = l, l+1, \ldots, K$, cf. (2.4),

$$\sum_{i=1}^k V_i^\gamma(t) = \sum_{i=1}^k V_i^\gamma(t_n) - \int_{t_n}^t R_s\, ds \le \sum_{i=1}^k V_i^\pi(t_n) - \int_{t_n}^t R_s\, ds \le \sum_{i=1}^k V_i^\pi(t)$$

where the first inequality follows from the induction hypothesis.

Step 2: $t = t_{n+1}$.

Clearly, for $\delta = \gamma$ and $\delta = \pi$

$$V_i^\delta(t_{n+1}) = V_i^\delta(t_{n+1}^-) + \sum_{l=1}^\infty S_l(i)\, \mathbf{1}_{\{A_{l,i} = t_{n+1}\}}$$

for $i = 1, 2, \ldots, K$. Here $\mathbf{1}_A$ stands for the indicator function of any event $A \in \mathcal{F}$. Inequality (2.3) at time t_{n+1} then follows from Step 1.

■

3 Optimizing a Weighted Sum of the Expected Number of Jobs

In this section we address the minimization of the cost function $\sum_{k=1}^{K} c_k \, E\left[Q_k^{\pi}(t)\right]$ where c_k's are arbitrary nonnegative constants.

We shall restrict the analysis to policies in Π that do not know present and future service requirements. More precisely, we now consider the set of policies $\Gamma \subset \Pi$ such that for all $t \geq 0$ the mapping $(s, \omega) \to \pi(s, \omega)$ from $[0, t] \times \Omega$ into $[0, 1]^K$ is $\mathcal{B}([0, t]) \otimes \mathcal{F}(t)$-measurable, where $\mathcal{B}([0, t])$ denotes the Borel σ-algebra on $[0, t]$ and $\mathcal{F}(t) := \mathcal{A}^1 \otimes \mathcal{A}^3 \otimes \sigma\left(\left(Q_k^{\pi}(s), s \in [0, t]\right), k = 1, 2, \ldots, K\right)$.

Observe that the policy γ introduced in Section 2 belongs to Γ. We assume that the sequences of service requirements $(S_{n,k})_{n=1}^{\infty}$, $k = 1, 2, \ldots, K$, are mutually independent i.i.d. sequences of r.v.'s such that $P(S_{n,k} \leq x) = 1 - \exp(-\mu_k \, x)$ (exponential service requirements), further independent of $(A_{n,k})_{n,k}$ and $(R(t),\, t \geq 0)$.

Lemma 3.1 *For any policy $\pi \in \Gamma$*

$$E\left[Q_k^{\pi}(t)\right] = \mu_k \, E\left[V_k^{\pi}(t)\right] \tag{3.1}$$

for $k = 1, 2, \ldots, K$, $t \geq 0$.

Proof. Fix $k \in \{1, 2, \ldots, K\}$, $t \geq 0$ and $\pi \in \Gamma$. Clearly,

$$V_k^{\pi}(t) = \sum_{j=1}^{Q_k^{\pi}(t)} \sigma_{j,k}^{\pi}(t) \tag{3.2}$$

where $\sigma_{j,k}^{\pi}(t)$ is the remaining processing requirement of the jth oldest customer of class k in the system at time t. Because of the memoryless assumption on the service requirements and because π does not know present and future service requirements it is seen that $E[\sigma_{j,k}^{\pi}(t) \,|\, Q_k^{\pi}(t) = n] = 1/\mu_k$ for all $n = 1, 2, \ldots$, $j = 1, 2, \ldots, n$, which yields (3.1) from (3.2). ∎

Combining Proposition 2.1 (with $r_k = \mu_k c_k$, $k = 1, 2, \ldots, K$) and Lemma 3.1 yields the following

Proposition 3.1 *Assume that $\mu_1 c_1 \geq \mu_2 c_2 \geq \cdots \geq \mu_K c_K \geq 0$. Then, the RAP that allocates, at any time, the maximum available processing rate to the nonempty class with the highest $\mu_k c_k$ (the so-called μc-rule) minimizes the cost function $\sum_{k=1}^{K} c_k \, E\left[Q_k^{\pi}(t)\right]$ over the policies in Γ, for all $t \geq 0$.*

Proposition 3.1 says that the μc-rule is optimal out of the policies that may know future arrival times and future processing rates but not present and future service requirements in a multiclass G/M/1 queueing system with stochastic time-varying processing rate.

Remark 3.1 *The discrete-time version of the problem (see [1] and [2]) can be addressed using the approach developed in this section.*

Remark 3.2 *Proposition 2.1 holds for any service discipline within classes. The same is true for Lemma 3.1 provided that the rule that determines which customer(s) should get served within the class selected by the RAP does not depend on present and future service requirements.*

4 Extension of the Mathematical Setting to Randomized Policies

Although the set of policies Π that has been considered so far is fairly large, it however does not allow one to make randomized allocations (hereafter referred to as randomized decisions) of the machine processing rate. The aim of this section is to extend the mathematical setting introduced in Section 1 so that randomized decisions may be generated.

We shall first introduce the extended setting. Then, we shall explain in what sense randomized decisions may be made within this setting.

Let Ω^4 be the set of all measurable mappings from $[0,+\infty) \to [0,1]^K$ and let $\xi := (\xi_t := (\xi_t^1, \ldots, \xi_t^K),\ t \geq 0)$ be the coordinate process of Ω^4 defined by $\xi_t^k(f) = f_k(t)$ for $k = 1, 2, \ldots, K$, where $f(t) := (f_1(t), f_2(t), \ldots, f_K(t)) \in [0,1]^K$. Let $\mathcal{A}^4 := \sigma(\xi_t,\ t \geq 0)$ be the smallest σ-algebra with respect to which every ξ_t is measurable. Define $\tilde{\mathcal{F}}(t) := \mathcal{F} \otimes \sigma(\xi_s,\ 0 \leq s \leq t)$ where $\mathcal{F}$ was introduced in Section 1.

Let $\tilde{P}$ be any probability measure on the measurable space $(\tilde{\Omega}, \tilde{\mathcal{F}}) := (\Omega \times \Omega^4, \mathcal{F} \otimes \mathcal{A}^4)$ such that conditions (1)-(3) in Section 1 along with the following conditions:

(4) for any $0 \leq t_1 < t_2 < \cdots < t_n$, $n = 1, 2, \ldots$, $(\xi_{t_i}^k, i = 1, 2, \ldots, n)_{k=1}^K$ is a collection of independent r.v.'s uniformly distrituted on $(0,1)$

(5) ξ is independent of $(A_{n,k})_{n,k}$, $(S_{n,k})_{n,k}$ and R.

We define a Randomized RAP (R-RAP) as a mapping $\pi : [0,+\infty) \times \tilde{\Omega} \to [0,1]^K$ such that for all $t \geq 0$ the mapping $(s,\omega) \to \pi(s,\omega)$ from $[0,t] \times \tilde{\Omega}$ into $[0,1]^K$ is $\mathcal{B}([0,t]) \otimes \tilde{\mathcal{F}}(t)$-measurable, and such that $\sum_{k=1}^K \pi_k(t,\cdot) \leq 1$ for all $t \geq 0$.

Let us now comment on what we mean by randomized decisions. Let $F_A(t, x_1, \ldots, x_K)$ be the conditional probability distribution function (c.p.d.f.) that the processing rate allocated to classes $1, \ldots, K$ at time t is less than or equal to $x_1, \ldots, x_K$, respectively, given the event (history) $A \in \tilde{\mathcal{F}}(t^-) := \mathcal{F} \otimes \sigma(\xi_s,\ 0 \leq s < t)$. We shall assume that $F_A(t, x_1, \ldots, x_K) = 1$ when $\sum_{k=1}^K x_k \geq 1$ so as to reflect the constraint that $\sum_{k=1}^K \pi_k(t,\cdot) \leq 1$.

From $F_A(t,\cdot)$ we may determine the c.p.d.f. $G_{A,x_1,\ldots,x_{k-1}}(t,x)$ that the processing rate allocated to class k at time t is less than or equal to x given that the processing rates allocated to classes $1,\ldots,k-1$ at time t are less than or equal to $x_1,\ldots,x_{k-1}$, respectively, and given the event $A \in \tilde{\mathcal{F}}(t^-)$. Then, thanks to conditions (4) and (5) above it is easily seen that the r.v.'s $(\pi_k(t))_{k=1}^K$ recursively defined by $\pi_k(t) = G^{-1}_{A,\pi_1(t),\ldots,\pi_{k-1}(t)}(t,\xi_t^k))$ with $G^{-1}_{A,x_1,\ldots,x_{k-1}}(t,y) = \inf\left\{x \geq 0 : G_{A,x_1,\ldots,x_{k-1}}(t,x) > y\right\}$ have p.d.f. $F_A(t,\cdot)$ on the event $A \in \tilde{\mathcal{F}}(t^-)$. (This construction is known as the *inversion transform method*; cf. [13].)

In other words, we have shown that it is always possible to choose the process π such that decisions may be generated according to fixed p.d.f.'s.

If we now define $\tilde{\Pi}$ to be the set of all R-RAP's it is seen that the results in Sections 2 and 3 still hold with Π replaced by $\tilde{\Pi}$. In particular (see Section 3), this shows that the μc-rule minimizes the cost function $E\left[\sum_{k=1}^K c_k\, Q_k^\pi(t)\right]$ over the policies in $\tilde{\Pi}$ that do not know present and future processing requirements.

References

[1] J. S. Baras, D.-J. Ma and A. M. Makowski, "K competing queues with geometric requirements and linear costs: the μc-rule is always optimal," *Systems Control Lett.* **6**, 173-180, 1985

[2] C. Buyukkoc, P. Varaiya and J. Walrand, "The μc-rule revisited," *Adv. Appl. Prob.* **17**, 237-238, 1985.

[3] E. Gelenbe and I. Mitrani, *Analysis and Synthesis of Computer Systems*, Academic Press, 1980.

[4] J. M. Harrison, "Dynamic scheduling of a multiclass queue: Discount optimality," *Operations Res.* **23**, 270-282, 1975.

[5] T. Hirayama and M. Kijima "Single machine scheduling problem when the machine capacity varies stochastically," *Opns. Res.*, Vol. 40, No. 2, pp. 376-383, 1992.

[6] T. Hirayama, M. Kijima and S. Nishimura, "Further results for dynamic scheduling of multiclass G/G/1 queues," *J. Appl. Prob.* **26**, 595-603, 1989.

[7] D. Krass, *Contribution to the Theory and Applications of Markov Decision Processes*, Ph. D. Thesis, Johns Hopkins University, Baltimore, 1989.

[8] I. Meilijson and U. Yechiali, "On optimal right-of-way policies at a single-server station when insertion of idles times is permitted," *Stoch. Proc. and their Appl.* **6**, 25-32, 1977.

[9] P. Nain, "Interchange arguments for classical scheduling problems in queues," *Systems Control Lett.* **12**, 177-184, 1989.

[10] J. Neveu, *Mathematical Foundations of the Calculus of Probability*, Holden-Day Inc., San Francisco, 1965.

[11] R. Righter and J. G. Shanthikumar, "Scheduling multiclass single server queueing systems to stochastically maximize the number of successful departures," *Prob. Eng. and Inf. Sci.* **3**, 323-33, 1989.

[12] J. G. Shanthikumar and D. D. Yao, "Multiclass queueing systems: polymatroidal structure and optimal scheduling control," *Opns. Res.*, Vol 40, Supp. No. 2, S293-S299, May-June 1992.

[13] R. Y. Rubinstein, *Monte Carlo Optimization, Simulation and Sensitivity of Queueing Networks*, John Wiley & Sons, New York, 1986.