AT&T Technical Journal
Vol. 64, No. 1, January 1985
Printed in U.S.A.

Optimal Resource Allocation for Two Processes*

By C. COURCOUBETIS† and P. VARAIYA‡

(Manuscript received May 9, 1984)

Two processes compete for access to n resources. A scheduling policy allocates the resource when the processes request it simultaneously. The objective is to minimize the average value of a state-dependent cost. The optimal policy can be calculated explicitly for the case of one resource. In the general case $n > 1$, an adaptive scheduling algorithm is proposed. The algorithm measures average transition times of the system and converges to the optimal policy.

I. INTRODUCTION

Two processes share n resources. A process operates in one of these states: thinking, requesting, or holding a given resource. The thinking and resource holding times are geometric, with means depending on the process and the resource.

Time for the system is discrete. A decision has to be made when the two processes simultaneously request the same resource k, in which case a scheduling policy assigns the resource to process 1 with probability $u_k \in [0, 1]$. A resource cannot stay idle when there is a process requesting it. The problem is to choose $u^* \in [0, 1]^n$, which minimizes the average value of a cost depending on the state of the system.

A simple example of such a system is the case of two processes sharing the same broadcast facility. Such a process goes through the following phases. It thinks for an arbitrary amount of time (any

* Research supported by the Office of Naval Research Contract N00014-80-C-0507.
† AT&T Bell Laboratories. ‡ University of California at Berkeley.

Copyright © 1985 AT&T. Photo reproduction for noncommercial use is permitted without payment of royalty provided that each reproduction is done without alteration and that the Journal reference and copyright notice are included on the first page. The title and abstract, but no other portions, of this paper may be copied or distributed royalty free by computer-based and other information-service systems without further permission. Permission to reproduce or republish any other portion of this paper must be obtained from the Editor.

activity other than broadcasting), requests the resource (broadcast channel), uses the resource for an arbitrary time (broadcasts), and then resumes thinking. If the resource was busy upon request, the process has to wait until the resource is released. If two processes request the resource simultaneously (during the same slot), the scheduler of the resource will decide, according to the scheduling policy of the system, which process will proceed first.

In Section III the problem is solved explicitly for the case $n = 1$ and when the cost is taken to be resource idle time. It is shown that for large ratios of average thinking times, u^* does not depend on the holding times and contrasts with the familiar "$c\mu$" results, where the average waiting cost is minimized when priority is given to processes with large waiting cost and small holding times (e.g., Refs. 1 through 6). It turns out that in that case the optimal policy consists in serving the process with the least thinking time first.

This is a new result for nonpreemptive systems. Similar results appeared in the literature concerning n processes sharing a single resource in a preemptive-resume way. For such systems, the "least thinking time first" scheduling rule has been proved optimal (see Refs. 7 and 8). For example, in Ref. 8 the author considers a "mirror image" problem, where the goal is to minimize the utilization (repair time) of the scarce resource (repairman). In this case the policy of serving the process with longest thinking time first is optimal. Because preemptive policies were considered and a single resource was shared, results are not applicable to the problem considered here.

In Section IV we consider the case of n resources and an arbitrary state-dependent cost. Since Section III suggests that an analytic expression for the optimal policy u^* as a function of the parameters of the processes would be extremely hard or even impossible to get, we proceed with an adaptive algorithm to compute u^*. This algorithm converges in a finite number of steps to u^* by using measurements of the average transition times of the system. The analysis is based on the results contained in Refs. 9 through 15, which are reviewed in Section 4.1. In Section 4.4 the structure of the problems for which the results apply is generalized. We permit a process to consist of an arbitrary set of thinking states, an arbitrary set of resource holding states for each resource in the system (one set of states per resource), and a number of resource request states (one state per resource). Finally, in Section V we present some open problems.

II. MODEL DESCRIPTION

In this section we give a formal description of the model for $n = 1$. The state of process i ($i = 1, 2$) at time t ($t \geq 0$) is X_t^i. If $X_t^i = 0$, then process i is thinking. If $X_t^i = 1$ [resp. 2], then process i is requesting [resp. holding] the resource. See Fig. 1.

The resource can accommodate only one process at any time. When the two processes are simultaneously requesting the resource, then the resource is assigned with probability $u \in [0, 1]$ to process 1.

Denote by p_i the probability that process $i (i = 1, 2)$ will complete its thinking time in a given time unit. Similarly define q_i for completion of service (holding). From this definition, one finds that the process $X_t = (X_t^1, X_t^2)$, $t = 0, 1, \cdots$, is a Markov chain with transition probability matrix $P = \{P(x, y) \mid x, y \in X = \{0, 1, 2\}^2\}$ defined as follows:

$$P(00, 10) = p_1(1 - p_2),\ P(00, 01) = (1 - p_1)p_2,\ P(00, 11) = p_1 p_2;$$

$$P(01, 12) = p_1,\ P(10, 21) = p_2;$$

$$P(01, 02) = (1 - p_1),\ P(10, 20) = (1 - p_2);$$

$$P(11, 21) = u,\ P(11, 12) = 1 - u;$$

$$P(02, 12) = p_1(1 - q_2),\ P(02, 00) = (1 - p_1)q_2;$$

$$P(20, 21) = (1 - q_1)p_2,\ P(20, 00) = q_1(1 - p_2);$$

$$P(12, 20) = q_2,\ P(21, 02) = q_1. \tag{1}$$

The diagonal elements of P are defined so that the rows sum to one. For any choice of u, X_t is ergodic.

Define as π_u the invariant measure of X_t, and let k: $X \to R$ be the state-dependent cost vector. Then the expected cost-per-unit time is

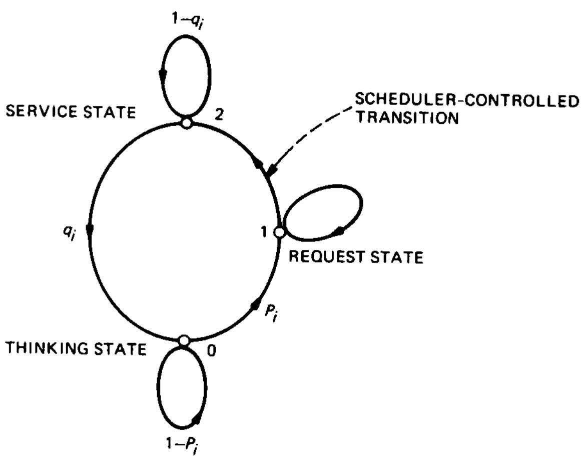

Fig. 1—Process i (i = 1, 2).

$$J(u) := \lim_{T\to\infty} \frac{1}{T+1} E \sum_{t=0}^{T} k(X_t) = \sum_{x\in X} \pi_u(x)k(x). \tag{2}$$

Our goal is to find u^* that minimizes (2).

III. MAXIMIZING RESOURCE UTILIZATION

Throughout this section we will assume $k(00) = 1$, $k(ij) = 0$, $ij \neq 00$.

Let π be the invariant probability measure associated with P. Then the expected idle time of the resource is

$$J(u) = \sum_{i,j\in\{0,1\}} \pi_u(ij). \tag{3}$$

Direct computation of π_u shows that (3) takes the following form:

$$J(u) = \frac{Au + B}{Cu + D}, \tag{4}$$

where A, B, C, D are functions of $\{p_1, p_2, q_1, q_2\}$. From (4) it follows that

$$\text{sign}\left(\frac{\partial J(u)}{\partial u}\right) = \text{sign}(AD - BC). \tag{5}$$

In what follows, we will choose $u^* \in [0, 1]$, which minimizes (4). From (5) it follows that u^* is an extreme point of $[0, 1]$. We state now the key results, which are proved in the Appendix.

Fact:

$$AD - BC = p_1p_2q_1^2q_2^2F_1F_2,$$

where $F_i = F_i(p_1, p_2, q_1, q_2)$, $i = 1, 2$.

Theorem 1 implies that the sign of $AD - BC$ is determined by the sign of F_1.

Theorem 1:

$$F_2 \geq 0 \quad \textit{for all} \quad p_1, p_2, q_1, q_2 \in [0, 1].$$

We state now the main theorem.

Theorem 2: 1. The curve $\{(p_1, p_2) \in [0, 1]^2 | F_1(p_1, p_2, q_1, q_2) = 0\}$ *lies between the lines* $p_1 = p_2$ *and* $p_2 = p_1/2$ *for* $q_1 \leq q_2$, *and between* $p_1 = p_2$ *and* $p_1 = p_2/2$ *for* $q_2 \leq q_1$.

2. $p_1 \geq 2p_2$ *implies* $F_1 \geq 0$, *and* $p_2 \geq 2p_1$ *implies* $F_1 \leq 0$, *for all* $q_1q_2 \in [0, 1]$.

The following corollary follows now from Theorem 2 and (5):

Corollary 2.1: Let u^* *be the value of the control minimizing* (4). *Then the following hold:*

1. $p_1 \geq 2p_2$ *implies* $u^* = 1$, *and* $p_2 \geq 2p_1$ *implies* $u^* = 0$;

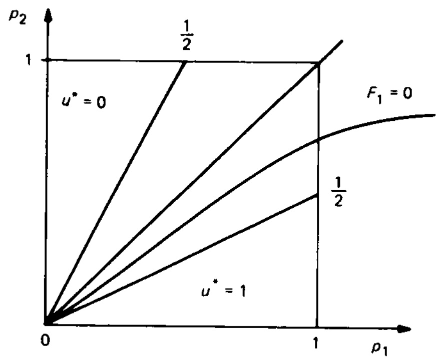

Fig. 2—Optimal policy u^* as a function of the parameter p_1, p_2.

2. If $p_1 = p_2$, then $q_2 \leq q_1$ implies $u^* = 0$, else $u^* = 1$;
3. If $q_1 = q_2$, then $p_1 \geq p_2$ implies $u^* = 1$, else $u^* = 0$.

Theorem 2 and Corollary 2.1 are illustrated in Fig. 2, where $F_1 = 0$ is considered for some fixed values of q_1, q_2, $q_2 < q_1$. The surprising result here is that there exist "safe" regions of the parameters, where u^* does not depend on the service times of the processes.

IV. ESTIMATION OF u^*

In this section we give a method for calculating u^* for an arbitrary cost vector k. The algorithm we propose requires no a priori knowledge of the parameters of the processes and is performed adaptively to the system. It starts by applying some arbitrary policy u^0, and then by monitoring the evolution of the system and estimating certain transition times, it updates the policy used until u^* is reached; this is always achieved in a finite number of steps, depending on the number of the shared resources. Since the algorithm is adaptive, it could be effectively used in systems with slowly varying parameters. This, together with the simplicity of the computation involved, makes this approach an interesting alternative to a direct calculation of u^* by using standard policy iteration algorithms (see Ref. 15). In Section 4.1 some general results are stated. We discuss in Section 4.2 the case of two processes and one resource, and in Section 4.3 the case of two processes and n resources.

4.1 Some results in Markov decision theory

Propositions 1, 2, and 3 consist of results already known and are stated without proof in order to make the present work self-contained. The key propositions used to calculate u^* are Propositions 4, 5, and 6, which consist of new results; their proofs are included in the Appendix.

Let X_t be a Markov chain on $\{1, \cdots, s\}$. If $X_t = i$, then any control $u \in U(i)$ may be used, where $U(i)$ is a compact set. A stationary strategy is any element $u = (u(1), \cdots, u(s)) \in U = U(1) \times \cdots \times U(s)$. Let $P(u)$ be the $s \times s$ transition probability matrix describing X_t, and assume that X_t consists of a single ergodic class and that $P(u)$ is continuous on U. Let $Q(u) := P(u) - I$, and let $k := (k(1), \cdots, k(s))'$ be the cost vector, not depending on u. The cost to be minimized is $J(u) = \lim_{T\to\infty} 1/(T+1)E\sum_{t=0}^{T} k(X_t)$. The following two propositions give the optimality conditions for u.

Proposition 1: (see, e.g., Ref. 14, Lemma 3.1) For $u \in U$ consider the s linear equations in the $s + 1$ variables $\gamma \in R$, $c \in R^s$,

$$\gamma\bot = Q(u)c + k, \tag{6}$$

where $\bot = (1, \cdots, 1)'$. Then:

1. If (γ, c) is a solution, then $\gamma = J(u)$.
2. If (γ, c) is a solution, then so is $(\gamma, c + \delta\bot)$ for all δ.
3. A solution always exists and is almost unique in the sense of (2).

Let $H(c, u) = Q(u)c + k$, $H = (H_1, \cdots, H_s)'$. Note that $H_i(c, u) = H_i(c, u(i))$ depends only on $u(i)$. Let $h(c) = (h_1, \cdots, h_s)$, $h_i(c) = \min \{H_i(c, v) \mid v \in U(i)\}$.

Proposition 2: (see, e.g., Theorems 3.1 and 3.2) The control u is optimal (minimizing $J(u)$) iff there exist (γ, c) such that $\gamma\bot = h(c) = H(c, u)$.

The following propositions will be used in the part dealing with the estimation of u^*.

Consider the equation

$$V_a = (I - aP(u))^{-1}k. \tag{7}$$

Then $V_a(i)$ is the expected discounted cost with the discount factor a, $a \in [0, 1]$, starting from state i, i.e., $V_a(i) = E_i\{\sum_{t=0}^{\infty} a^t k(X_t)\}$. Let $Z^a := (I - aP(u))^{-1}$.

Proposition 3: (see, e.g., Ref. 9, Chapter 3) Z^a is the fundamental matrix of the absorbing chain X_t^a defined on $\{0, 1, \cdots, s\}$ with the $(s + 1) \times (s + 1)$ transition probability matrix P_a such that

$$P_a(i, j) = aP(i, j), \qquad i, j \in \{1, \cdots, s\},$$

$$P_a(i, 0) = 1 - a \quad \text{for} \quad i \neq 0, \qquad P_a(0, 0) = 1.$$

Let $\bar{N}_i^a[j]$ be the expected number of visits to state j starting from i of X_t in a geometrically distributed interval of time with mean $(1 - a)^{-1}$. Then $z_{ij}^a = \bar{N}_i^a[j]$.

Proposition 4:

$$\lim_{\alpha\to 1}[V_\alpha(i) - V_\alpha(k)] = c(i) - c(k),$$

where c satisfies (6).

Proposition 5: Let $\bar{T}_{ij}$ be the expected time before the first visit to state j by X_t starting from i. Then

$$\lim_{\alpha\to 1}[\bar{N}_i^\alpha[j] - \bar{N}_k^\alpha[j]] = \begin{cases} \dfrac{\bar{T}_{kj} - \bar{T}_{ij}}{\bar{T}_{jj}} & \text{if} \quad k \neq j \neq i \\ -\dfrac{\bar{T}_{ij}}{\bar{T}_{jj}} & \text{if} \quad k = j \neq i. \end{cases}$$

The next proposition is key, since it relates the dual variable c to transition times of the chain X_t, which can be estimated quite readily. The proposition can be readily obtained from Propositions 3, 4, 5, and eq. (7).

Proposition 6:

$$c(i) - c(k) = \sum_{j=1}^{s} \left[\frac{\bar{T}_{kj} - \bar{T}_{ij}}{\bar{T}_{jj}}\right] k(j) + k(i) - k(k).$$

4.2 Optimal resolution of conflict for two processes and one resource

In this section we provide the following results. In Theorem 3 we characterize the value of u^*. In Theorem 4 we relate the value of u^* with the sign of a quantity that can be estimated using Proposition 6 from the behavior of the system. We finally propose Algorithm 1, which uses these results to adaptively calculate u^*.

Theorem 3: u^* can always be restricted to the set $\{0, 1\}$.

Proof: Let (γ^*, c^*) be the variables in (6) corresponding to the optimal policy u^*. Then by Proposition 2 and the fact that u enters only in the row of P corresponding to the state (11),

$$\begin{aligned} \gamma^* &= \min\{-c^*(11) - (1 - u)c^*(12) \\ &\qquad + uc^*(21) + k(11) \mid u \in U\} \\ &= \min\{[c^*(12) - c^*(11) + k(11)] \\ &\qquad + u[c^*(21) - c^*(12)] \mid u \in U\} \end{aligned} \tag{8}$$

and the minimum is achieved at u^*. Let $A = c^*(21) - c^*(12)$. Then $A > 0$ implies $u^* = 0$, $A < 0$ implies $u^* = 1$, and $A = 0$ implies that any $u \in [0, 1]$ will do. □

Theorem 4: Let (γ, c) be a solution to (6). Then $\text{sign}[c(21) - c(12)] =$ const. *for all $u \in U$.*

Proof: Suppose there exists a u_0 such that $c^{u_0}(21) - c^{u_0}(12) = 0$. Then γ^{u_0}, c^{u_0} are optimal dual variables since the optimality condition (8) is trivially satisfied. Then every $u \in U$ will satisfy the optimality conditions with $c^*(21) - c^*(21) = c^{u_0}(21) - c^{u_0}(12) = 0$; hence u is also optimal. From this and Proposition 4.1 it follows that $c^{u_1}(21) - c^{u_1}(12) = c^{u_0}(21) - c^{u_0}(12) = 0$, for all $u_1 \in U$. This, with the next Fact, proves Theorem 4.

Fact: The vector c of (6) can be chosen to be a continuous function of u.

This can be proved as follows. Let π_u be the invariant probability measure of $P(u)$. Since $\gamma = \pi_u k$, by using (6) we get

$$\pi_u k \bot - k = Q(u)c. \tag{9}$$

Since π_u is continuous in u (by the ergodicity of $P(u)$), and $Q(u)$ is of rank $s - 1$ for all $u \in [0, 1]$, then there is always a solution $\bar{c} = (\bar{c}(1), \cdots, \bar{c}(s-1), 0)$ of (9) continuous in u, and by the "almost" uniqueness of c the Fact follows. □

Corollary 4.1: Let (γ, c) *be a solution to* (6). *Then* $c(21) - c(12) > 0$ *implies* $u^* = 0$; $c(21) - c(12) < 0$ *implies* $u^* = 1$; *and if* $c(21) - c(12) = 0$, *any* u^* *will do.*

Corollary 4.1 suggests the following algorithm to estimate u^* adaptively.

Algorithm 1:

1. Start the system with an arbitrary u.
2. Use Proposition 6 to estimate the sign of $c(21) - c(12)$.
3. Use Corollary 4.1 to choose u^*.

We will conclude this section with an example. Consider the case of minimizing the probability of conflict, i.e., the probability that both processes request simultaneously. In this case, $k(i) = 0$ for $i \neq 11$. It follows that

$$c(21) - c(12) = k(11)\,\frac{\bar{T}_{12,11} - \bar{T}_{21,11}}{\bar{T}_{11,11}},$$

and

$$\text{sign}[c(21) - c(12)] = \text{sign}[\bar{T}_{12,11} - \bar{T}_{21,11}].$$

An intuitive justification for the above equation is the following. Minimizing $P(11)$ is equivalent to maximizing $\bar{T}_{11,11}$, which is equivalent to giving priority to the process so that the busy period of the system corresponding to that process being serviced first is maximized. It is easy to see by using renewal arguments that this is equivalent to choosing the largest among the $\bar{T}_{12,11}$, $\bar{T}_{21,11}$.

4.3 The case of n resources

In this section we will generalize some of the previous results. Each process has a thinking state 0 as before, and pair $(1_k, 2_k)$ of request and resource holding states for every resource k. Hence there are n conflict states, and the control u is the n-tuple $(u_1, \cdots, u_n)$, where u_k is the probability of assigning resource k to process 1 in the conflict state $(1_k, 1_k)$. See Fig. 3. Again, we are concerned with the estimation of u^*. To simplify notation, we will prove the theorems for $n = 2$; the same proofs hold for any n larger than 2. We follow the sequence of the previous section.

Theorem 5: u^* can always be selected from $\{(0, 0), (0, 1), (1, 0), (1, 1)\}$.

Proof: Assume that each process consists of the states $(0, 1_1, 2_1, 1_2, 2_2)$, where states 1_1, 2_1 are associated with resource 1, and 1_2, 2_2 with resource 2. The proof is similar to the proof of Theorem 3, since the optimality conditions are

$$\gamma^* = \min\{[c^*(1_12_1) - c^*(1_11_1) + k(1_11_1)] + u_1[c^*(2_11_1) - c^*(1_12_2)] \mid u_1 \in U_1\}, \tag{10a}$$

$$= \min\{[c^*(1_22_2) - c^*(1_21_2) + k(1_21_2)] + u_2[c^*(2_21_2) - c^*(1_22_2)] \mid u_2 \in U_2\}. \quad \square \tag{10b}$$

The following algorithm is a policy iteration algorithm that starts with an arbitrary initial choice of u and in a finite number of repetitions converges to u^*. Step 2 corresponds to the value determination operation, and Step 3 to the policy improvement operation. The novel feature of this algorithm is the simple and adaptive execution of the value determination operation by using Proposition 6. The ergodicity of the system ensures that every improvement of the policy corresponds to a strict decrease in cost (see Refs. 15 and 16). Define $A_1 :=$

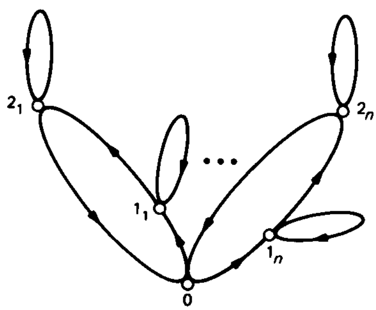

Fig. 3—Process i ($i = 1, 2$), n resources.

$c(21) - c(12)$ and $A_2 := c(2'1') - c(1'2')$. (Observe that A_1, A_2 are the coefficients of u_1, u_2 in eq. [10].)

Algorithm 2:

1. Start with some arbitrary $u^k = (u_1^k, u_2^k)$.

2. Using Proposition 6, estimate the sign of A_1^k and A_2^k, where $A_i^k = A_i(u^k)$, $i = 1, 2$.

3. Choose in an extreme way the u_i^{k+1}, $i = 1, 2$, in order to decrease each $A_i^k u_i^{k+1}$ separately (i.e., if $A_i(u^k) > 0$, choose $u_i^{k+1} = 0, \cdots$). If $u^{k+1} \neq u^k$, go to Step 2 by using $u = u^{k+1}$. If $u^{k+1} = u^k$, then $u^k = u^*$.

The convergence follows, since every choice of a different u^k corresponds to a strict decrease of the cost (by ergodicity), and there is a finite number of choices for u^k (finite extreme points in U). The essential difference between the cases $n = 1$ and $n > 1$ is that Theorem 4 cannot be generalized for $n > 1$. This prevents us from inventing a "one step" algorithm for the estimation of u^*.

Note that although not addressed in this paper, the problem of estimating the T_{ij}'s is of great importance for the algorithm. Substantial errors in the estimation procedure could lead to a wrong choice of u^* by the algorithm.

4.4 A generalization

One can notice that throughout Sections 4.2 and 4.3, the only parts of the structure of P used were the rows corresponding to conflict states. This leads to a generalization of the form of the processes. A process can consist of an arbitrary set S_0 of thinking states and a pair $(1_k, S_2^k)$ for each resource k, where S_2^k is an arbitrary set of service states. The only constraint on the transition diagram of the process is that there is a unique state in S_2^k to which a process k can transit from state 1_k.

V. CONCLUSIONS

One open problem is the relation between the distributions of service and thinking times, and the safe regions of Section III. In other words, is there a general rule suggesting that for large ratios of thinking time of the processes, the choice of u^* is independent of service times? This would be nice, since no further calculations are needed to obtain the optimal u^*.

Another open problem is the generalization to m processes sharing n resources. A realistic model for this situation would suggest a decentralized information structure for each resource scheduler. By this we mean that the scheduler of resource r should base its decision on "local" information only, i.e., the state of resource r and the identity of processes requesting resource r. If one uses Markov Decision Theory as was done here, the optimal decision of scheduler r concerning conflict between processes i, j would depend on "global" information about the system, i.e., the state of all other processes even if they are not involved in this particular conflict. This is unsatisfactory. Instead one would like to obtain a rule for making local decisions that are optimal "in the average", by "smoothing out" what happens in the rest of the system.

As a final open question, we state the generalization of Section III. One should be able to prove the existence of the safe regions of Theorem 2 without explicitly calculating the invariant probability measure.

VI. ACKNOWLEDGMENTS

The authors would like to thank Professor Jean Walrand for many helpful discussions and for his shortened version of the proof of Proposition 5.

REFERENCES

1. J. Harrison, "Dynamic Scheduling of a Multiclass Queue: Discount Optimality," Oper. Res., *23* (March–April 1975), pp. 270–82.
2. J. Harrison, "A Priority Queue With Discounted Linear Costs," Oper. Res., *23* (March–April 1975), pp. 260–9.
3. G. Klimov, "Time-Sharing Service Systems, I," Theory Probab. Appl., *19* (September 1974), pp. 532–51.
4. I. Meilijson and G. Weiss, "Multiple Feedback at a Single-Server Station," Stochastic Processes and Applications, *5* (1977), pp. 195–205.
5. L. Schrage, "A Proof of Optimality of the Shortest Remaining Processing Time Discipline," Oper. Res., *16* (1968), pp. 687–90.
6. I. Meilijson and U. Yechiali, "On Optimal Right-of-Way Policies at a Single Server Station When Insertion of Idle Times is Permitted," Stochastic Processes and Applications, *6* (November 1977), pp. 25–32.
7. P. Nash and R. Weber, "Dominant Strategies in Stochastic Allocation and Scheduling Problems," in *Deterministic and Stochastic Scheduling*, M. A. H. Dempster, ed., Dordrecht, The Netherlands: Reidel, 1982, pp. 343–53.
8. D. Smith, "Optimal Repair of a Series System," Oper. Res., *26*, No. 4 (July–August 1978), pp. 653–62.
9. J. Kemeny and J. Snell, *Finite Markov Chains*, New York: Springer Verlag, 1976.
10. N. Jaiswal, *Priority Queues*, New York: Academic Press, 1968.
11. P. Billingsley, *Statistical Inference for Markov Processes*, Chicago: The University of Chicago Press, 1961.
12. J. Kemperman, *The Passage Problem for a Stationary Markov Chain*, Chicago: The University of Chicago Press, 1961.
13. V. Borkar and P. Varaiya, "Adaptive Control of Markov Chains, I: Finite Parameter Set," IEEE Trans. Automat. Contr., *24*, No. 6 (December 1979), pp. 953–7.
14. P. Varaiya, "Optimal and Suboptimal Control for Markov Chains," IEEE Trans. Automat. Contr., *23*, No. 3 (June 1978), pp. 388–94.
15. S. Ross, *Applied Probability Models with Optimization Applications*, San Francisco: Holden Day, 1970.
16. H. Mine and S. Osaki, *Markovian Decision Processes*, New York: Elsevier, 1970.

APPENDIX

Proof of Theorem 1: After calculating, we have

$$AD - BC = p_1 p_2 q_1^2 q_2^2 F_1 F_2,$$

where

$$F_1 = p_1^3p_2q_2 - p_1^2p_2q_2 - p_1^2q_2 - p_1p_2^3q_1 - p_1p_2^2q_1 + p_2^2q_1 + p_1p_2^2 - p_2^2 - p_1^2p_2 - p_2 + p_1 + p_1^2 \quad (11)$$

$$F_2 = q_1q_2[p_1p_2^2 + p_1^2p_2 - p_1p_2 - p_1 - p_2 + 1] + (p_2 - p_1p_2)q_2 + (p_1 - p_1p_2)q_1. \quad (12)$$

Let $L(p_1, p_2) := p_1^2p_2 + p_1p_2^2 - p_1p_2 - p_1 - p_2 + 1$. Then by (12)

$$F_2 = q_1q_2L + (p_2 - p_1p_2)q_2 + (p_1 - p_1p_2)q_1.$$

Consider $G_1(q_1) := F_2(p_1, p_2, q_1, q_2)$. Then

$$G_1(q_1) = q_1[q_2L + (p_1 - p_1p_2)] + (p_2 - p_1p_2)q_2.$$

Since $G_1(q_1)$ is linear in q_1, to prove $G_1 \geqslant 0$ for all $p_1, p_2, q_1, q_2 \in [0, 1]$, it is enough to show $G_1(0) \geqslant 0$ and $G_1(1) \geqslant 0$. But $G_1(0) = (p_2 - p_1p_2)q_2 \geqslant 0$; hence we only have to prove $G_1(1) \geqslant 0$. Let $G_2(q_2) := G_1(1)$. Using a similar argument, since G_2 is linear in q_2 and $G_2(0) = p_1 - p_2p_2 \geqslant 0$, we only have to show $G_2(1) \geqslant 0$. But this holds since $G_2(1) = p_1p_2(p_1 + p_2 - 2) - p_1p_2 + 1 \geqslant 0$, as one can readily check. □

Proof of Theorem 2: Consider the function $G_1(p_2) = F_1(p_1, p_2, q_1, q_2)$ and $q_1 \leqslant q_2$. Then proving the theorem is equivalent to proving the following:

1. $G_1(p_2)$ has a unique root p_2^0 in the interval $[p_1/2, p_1]$.
2. $G_1(p_2)$ has no root in the intervals $[0, p_1/2]$, $[p_1, 1]$.
3. $p_2 \leqslant p_2^0$ implies $G_1(p_2) \geqslant 0$, and $p_2 \geqslant p_2^0$ implies $G_1(p_2) \leqslant 0$.

One can now prove 1 through 3 since

$$G_1(1) \leqslant 0,\ G_1(p_1) \leqslant 0,\ G_1(p_1/2) \geqslant 0,\ G_1(0) \geqslant 0,\ p_2^0p_2^1p_2^2 > 1,$$

where p_2^0, p_2^1, p_2^2 are the roots of $G_1(p_2) = 0$, and $F_1(a, b, c, d) = -F_1(b, a, d, c)$, as one can easily check.

Proof of Proposition 4: Define $\bar{c} := (c(1), \cdots, c(s - 1), 0)$, and $Q := (P - I)$. Then

$$\gamma\perp = Q\bar{c} + k \quad \text{has a unique solution} \quad (\gamma, \bar{c}). \quad (13)$$

Let $y_a := V_a - V_a(s)\perp$. Then by using (7) and subtracting, we get

$$(1 - a)V_a(s)\perp = (aP - I)y_a + k. \quad (14)$$

By multiplying (7) by the invariant measure π, we get $\pi[(1 - a)V_a - k] = 0$, which, together with the ergodicity assumption (i.e., all components of π are >0) and the fact that $V_a \geqslant 0$, implies that no component of V_a tends to ∞ as $a \rightarrow 1$. Let $\gamma_a := (1 - a)V_a(s)$. By the previous argument it follows that there is a converging subsequence of γ_a as $a \to 1$, and for this subsequence let $\gamma^* := \lim_{a\to\infty}\gamma_a$. By using (14) we find

$$\gamma_a \bot = (aP - I)y_a + k. \tag{15}$$

Consider (15) as $a \to 1$ along the above subsequence. Since it has a unique solution y_a for every a, it follows that there is a unique $y^* := \lim_{a\to 1} y_a$, and (13) implies that $\gamma = \gamma^*$ and $\bar{c} = y^*$. The proof now follows by observing that $V_a(i) - V_a(k) = y_a(i) - y_a(k)$, and $c(i) - c(k) = \bar{c}(i) - \bar{c}(k)$ by Proposition 1. □

Proof of Proposition 5: Let X_t be the chain under consideration, $X_t \in \sum_1 := \{1, \cdots, s\}$, and define Y_t, $t = 0, 1, \cdots$, $Y_t \in \sum_2 := \{0, 1\}$ such that $P[Y_{t+1} = 1 \mid Y_t = 0] = 1 - a$, $P[Y_{t+1} = 1 \mid Y_t = 1] = 1$, i.e., is an absorbing chain with one being the absorbing state. We also define the following:

$N_i^a[j] :=$ Number of visits by X_t to state j starting from i before absorption,

$T_1 := \text{Min}\{n \geqslant 0 \mid X_n = j\}$,

$T_2 := \text{Min}\{n \geqslant 0 \mid Y_n = 1\}$,

$z :=$ Number of visits to state j before absorption. □

Lemma: $\bar{N}_i^a[j] = E_{(j,0)}[z]P_i[T_2 > T_1]$.

Proof: Start the system (X_t, Y_t) from state $(i, 0)$ and count visits to state j before absorption. One can always start counting from time T on, since no visit to state j occurs before T. The count z will not be identically zero only when $(X_t, Y_t) = (j, 0)$, and this occurs with probability $P_i[T_2 > T_1]$. By the strong Markov property, one can always restart the system from state $(j, 0)$, and the result follows since the new expected count will be $E_{(j,0)}[z]$.

Note that $\bar{N}_j^a[j] = E_{(j,0)}[z]$ since $P_j[T_2 > T_1] = 1$.

Let $\beta_{ij}^a := P_i\{X_t^a = j \text{ for some } t > 0\}$. Then by using the previous lemma, it follows that

$$\bar{N}_i^a[j] = \bar{N}_j^a[j]\beta_{ij}^a, \qquad \text{for} \quad i \neq j,$$

and by using similar renewal arguments,

$$\bar{N}_j^a[j] = 1 + \beta_{jj}^a \bar{N}_j^a[j];$$

hence

$$\bar{N}_j^a[j] = \frac{1}{1 - \beta_{jj}^a}.$$

Since $\beta_{ij}^a = E[a^{T_{ij}}]$, by using dominated convergence we obtain

$$\frac{\partial}{\partial a}\beta_{ij}^a = E[T_{ij}a^{T_{ij}-1}]$$

and

$$\lim_{a\to 1}\frac{\partial}{\partial a}\beta_{ij}^a = \bar{T}_{ij}.$$

Therefore if $i \neq j \neq k$, then

$$\begin{aligned}\lim_{a\to 1}[\bar{N}_i^a[j] - \bar{N}_k^a[j]] &= \lim_{a\to 1}[(\beta_{ij}^a - \beta_{kj}^a)\bar{N}_j^a[j]] \\ &= \lim_{a\to 1}\frac{\beta_{ij}^a - \beta_{kj}^a}{1 - \beta_{jj}^a} \\ &= \frac{\bar{T}_{kj} - \bar{T}_{ij}}{\bar{T}_{jj}},\end{aligned}$$

by de l'Hospital's rule. The proof for $k = j$ is similar. □

AUTHOR

Costas Courcoubetis, Diploma (Electrical and Mechanical Engineering), 1977, National Technical University of Athens, Greece; M.S., and Ph.D. (Electrical Engineering and Computer Science), 1980 and 1982, respectively, University of California at Berkeley; AT&T Bell Laboratories, 1982—. He was a Research Assistant in the Electronic Research Lab at U.C. Berkeley, and an Associate Member of Technical Staff at Bell Laboratories during the summer of 1981. He is currently a Member of Technical Staff in the Mathematics and Statistics Research Center. His interests are distributed algorithms and computation, queueing theory, performance analysis of computer systems, and design and verification of computer communication protocols.