DOMINANT STRATEGIES IN STOCHASTIC ALLOCATION AND SCHEDULING PROBLEMS

P. Nash, R.R. Weber

Cambridge University

ABSTRACT

Some problems of stochastic allocation and scheduling have the property that there is a single strategy which minimizes the expected value of the costs incurred up to every finite time horizon. We present a sufficient condition for this to occur in the case where the problem can be modelled by a Markov decision process with costs depending only on the state of the process. The condition is used to establish the nature of the optimal strategies for problems of customer assignment, dynamic memory allocation, optimal gambling, maintenance and scheduling.

1. DOMINANT STRATEGIES

The aim in many stochastic allocation and scheduling problems is to attain a desired state at the least possible cost. In job scheduling problems one may want to minimize the expected value of the makespan or flowtime. In a reliability problem one may want to minimize the expected repair costs. Some problems of this type have the special property that there exists a single strategy which not only minimizes the expected cost of reaching the desired state, but also minimizes the expected value of the cost incurred at every time. A strategy with this property shall be called expectation dominant (ED). A strategy will be called stochastic dominant (SD) if for all times t it minimizes in distribution the cost incurred at time t. Weeks and Wingler [14] have argued that stochastic dominance is the property most desired in a scheduling strategy.

To give an example of a problem in which the optimal strategy

M. A. H. Dempster et al. (eds.), Deterministic and Stochastic Scheduling, 343–353.

is stochastic dominant we shall describe the problem of assigning customers to parallel servers which has been discussed by Winston [16]. Identical jobs with exponentially distributed processing requirements arrive at a service facility in a Poisson stream. As each job arrives it must be assigned to one of a number of identical servers which operate in parallel. Each server serves jobs in its queue in a first come first served order. No jockeying amongst the queues is allowed. We have shown [11] that the strategy of assigning each arrival to the shortest queue minimizes the expected average waiting time of the first k jobs to arrive. In fact the optimal strategy is SD since it minimizes for all times t the distribution of the number of jobs in the system at time t. Before proceeding with a proof of this and other results, we present a general method for investigating expectation and stochastic dominant strategies.

2. A SUFFICIENT CONDITION FOR DOMINANCE

We will conduct most of our analysis in the setting of a continuous time Markov decision process on a finite state space $(i=1,\dots,N)$. Suppose that the cost of residence in state i is c_i per unit time. Starting from state i the rate of transition to state j is given by Q_{ij}, where Q is an NxN matrix in the set of feasible matrices Ω. The set Ω is bounded, closed and convex with the property that when a collection of 1st to Nth rows from different matrices in Ω is assembled together then this is itself a matrix in Ω. An ED strategy was defined as one which minimizes for all t the expected cost incurred at time t. It corresponds to a choice $\bar{Q}\varepsilon\Omega$ minimizing every component of V(t) where,

$$V(0) = c, \quad \text{and} \quad \dot{V}_i(t) = \min_{Q\varepsilon\Omega} \sum_j Q_{ij}V_j(t) = \sum_j \bar{Q}_{ij}V_j(t).$$

If the states have been indexed from greatest to least costly $(c_1>c_2>\cdots>c_N)$, then a SD strategy must minimize for all t and k $(1<k\leqslant N)$ the probability that the process is in a state less than k at time t. Thus a SD strategy is one which is simultaneously ED for (N-1) cost vectors c(k) $(k=1,\dots,N-1)$ of the form

$$c_j(k) \begin{array}{l} = 1, \; j=1,\dots,k, \\ = 0, \; j=k+1,\dots,N. \end{array}$$

Stochastic scheduling and allocation problems may be studied in either discrete or continuous time formulations. The continuous time formulation is often simpler for the reason that only one event (such as a completion of a job) can occur at any instant of time. Any convenience which is lost in setting the problem in continuous rather than discrete time can be recovered by treating the Markov process as a jumping process in which the residence times between jumps do not depend on the current state. We choose a θ, with $\theta\geqslant \max\{-Q_{ii}: 1\leqslant i\leqslant N, Q\varepsilon\Omega\}$, and let $\Pi=\{P: P=(Q+\theta I)/\theta, Q\varepsilon\Omega\}$. Starting from state i the original Markov decision process is realized by waiting a time which is exponentially distributed with parameter θ, choosing a matrix $P \in \Pi$, and then jumping to state j with probability P_{ij}. This process does not rule out the possibility of jumps from a state to itself. The strategy whose transition matrix is $\bar{Q}$ is ED for the continuous time Markov decision process if and only if $\bar{P}$ is the transition matrix of an ED strategy for the discrete time Markov decision process with the same cost vector and set of feasible transition matrices Π. The truth of this statement may be verified by considering the equation

$$V_i(t) = (1-e^{-\theta t})c_i + \int_0^t \sum_j P_{ij}V_j(t-s)\theta e^{-\theta s}\, ds.$$

This construction, which Keilson [5] calls uniformizing, has been discussed in number of papers, including [7] and [9].

We shall now investigate conditions under which a matrix $\bar{P}$ is ED for the discrete time jumping chain. To be minimizing, $\bar{P}$ must satisfy

$$(P-\bar{P})\bar{P}^n c \geq 0, \text{ for all } n \text{ and } P \in \Pi.$$

Let R be a matrix whose rows consist of all possible rows of $(P-\bar{P})$ with $P \in \Pi$.

Lemma 1. $\bar{P}$ is ED if and only if there exist sequences of matrices $\{H_n\}$, $\{A_n\}$, $\{B_n\}$, $(n=0,1,2,\ldots)$ such that

$$\begin{aligned} H_0 Rc &\geq 0, \\ R &= A_n H_n,\ A_n \geq 0, \text{ and} \\ H_{n+1}\bar{P} &= B_n H_n,\ B_n \geq 0. \end{aligned}$$

Proof. The sufficiency of the conditions follows immediately from

$$R\bar{P}^n c = A_n H_n \bar{P}^n c = \cdots = A_n B_{n-1} \cdots B_0 H_0 c \geq 0.$$

The necessity comes from considering the cone K_n whose generators are $\{c, \bar{P}c, \cdots, \bar{P}^n c\}$. Because K_n is a convex polyhedral cone it may be written as $\{x\colon H_n x \geq 0\}$. Now if $\bar{P}$ is ED then $H_n x \geq 0$ implies $Rx \geq 0$. Thus $\{x\colon Rx \geq 0\}$ is contained in $\{x\colon H_n x \geq 0\}$. When one cone is contained in another we can write $R = A_n H_n$ for some $A_n \geq 0$. Since $H_n x \geq 0$ implies $H_{n+1}\bar{P} \geq 0$ we must also have $H_{n+1}\bar{P} = B_n H_n$ for some $B_n \geq 0$.

Although this condition is not directly helpful, it does suggest that we may easily be able to establish the optimality of an SD strategy if the sequence H_n can be replaced by a single finite-dimensional matrix H. Certainly the existence of such a matrix is a sufficient condition, and it is this that we shall use in the rest of the paper to show that strategies are ED or SD.

Lemma 2. $\bar{P}$ is ED if there exist matrices H, $A \geqslant 0$ and $B \geqslant 0$ such that

$$
\begin{aligned}
Hc &\geqslant 0, \\
R\_ &= AH, \text{ and} \\
H\bar{P} &= BH.
\end{aligned}
$$

3. PROCESSES WHICH PARTIALLY ORDER THE STATES

Let e_i be the ith row of the $N \times N$ identity matrix. When the rows of $(P-\bar{P})$ are proportional to rows of the form (e_j-e_i) the problem may be particularly simple to analyse. $\bar{P}$ defines a partial order on the states. We say that i is better than j (and denote this by $i \downarrow j$) if there is a row of $(P-\bar{P})$ proportional to (e_j-e_i). We also have $i \downarrow i$. It is natural to consider the matrix H whose rows include

e_j-e_i iff $i \downarrow j$, and there is no k with $i \downarrow k$ and $k \downarrow j$, and
e_i iff there is no $j \neq i$ such that $i \downarrow j$.

From lemma 2 we deduce that $\bar{P}$ is ED if there exists a $B \geqslant 0$ with $H\bar{P} = BH$. This happens if for $i \downarrow j$ the ith row of $\bar{P}$ may be obtained from the jth row of $\bar{P}$ by operations on row j which shift probability from one state to others which are better in the partial order. The following lemma gives a condition guaranteeing that this shifting is possible (the proof is straightforward).

Lemma 3. $\bar{P}$ is ED if for all states $i \downarrow j$ and states k,

$$c_i \leqslant c_j, \text{ and } \sum_{k \downarrow h} \bar{P}_{ih} \leqslant \sum_{k \downarrow h} \bar{P}_{jh}. \tag{1}$$

There is a dual sufficient condition in which (1) is replaced by

$$c_i \leqslant c_j, \text{ and } \sum_{h \downarrow k} \bar{P}_{ih} \geqslant \sum_{h \downarrow k} \bar{P}_{jh}. \tag{2}$$

When the states can be totally ordered with $1 \downarrow 2, 2 \downarrow 3, \ldots$, then both conditions are equivalent to the statement that for $i<j$ and all k we must have

$$c_i \leqslant c_j, \text{ and } \sum_{h \geqslant k} \bar{P}_{ih} \leqslant \sum_{h \geqslant k} \bar{P}_{jh}. \tag{3}$$

$\bar{P}$ is then called a monotone matrix. Some properties of monotone matrices are discussed by Keilson and Kester [5].

Example 1. Assignment of Customers to Parallel Servers

We consider the problem of customer assignment that was described at the start of the paper. Suppose that there are m servers, and that each has a large but finite waiting room. Let x(i) be a vector whose kth component is equal to the number of jobs in the kth longest queue of state i. We partially order the states of the system, writing $i \downarrow j$ if

$$\sum_{h=1}^{k} x_h(i) \leqslant \sum_{h=1}^{k} x_h(j), \quad \text{for all } k=1,\dots,m. \tag{4}$$

The condition of lemma 3 simply says that for all k and $i \downarrow j$ the probability of a transition to a state in the partial ordering which is as bad as k must be no more starting from i than from j. This is obvious from the nature of the shortest queue strategy and the partial order defined by (4). If we let c_i be equal to 1 or 0 as the number of jobs in the system in state i is or is not greater than k, then $i \downarrow j$ implies $c_i \leqslant c_j$, and thus assignment to the shortest queue minimizes for all k and t the probability that there are more than k jobs in the system at time t.

Example 2. Dynamic Memory Allocation

Benes [1] proves a number results for problems of stochastic memory allocation. In one simple example he describes a linear computer memory having room for exactly three units of program. Computer jobs, which require just one or two contiguous units of memory, arrive according to Poisson processes of rates λ_1 and λ_2. The execution times of the programs in memory are exponentially distributed with parameters μ_1 and μ_2. If a program arrives to find insufficient contiguous room in the memory, then the computer crashes. We let (-—) denote the state in which there is one program of each of the lengths in the memory. In an obvious fashion, we can write the states of the system as

1=(   ) 2=(-) 3=(-) 4=(—)
5=(- -) 6=(--) 7=(-—) 8=(---) 9=(crashed).

A decision must be made as to where to allocate a program of length 1 when it arrives to find that the system is in states 1 or 2. The following transitions are then possible.

$$1=(\quad) \rightarrow \begin{matrix} 2=(-\ \ ) \\ \text{or} \\ 3=(\ -\ ) \end{matrix} \qquad \text{and} \qquad 2=(-\ \ ) \rightarrow \begin{matrix} 5=(-\ -) \\ \text{or} \\ 6=(--\ ) \end{matrix}$$

We can use lemma 3 to show that in each case the first choice is the one which minimizes the probability that the system crashes by time t. The acorrect partial order is $1\downarrow 2$, $2\downarrow 3$, $2\downarrow 5$, $3\downarrow 6$, $5\downarrow 6$, $6\downarrow 8$, $4\downarrow 7$, $7\downarrow 9$, $8\downarrow 9$. It is easy to write down a transition matrix P and verify lemma 3 with $c=(0,\dots,0,1)$. Thus the probability of a blockage occurring by time t is minimized by the indicated strategy. A similar analysis can be carried out when the memory is 4 cells long. However, we have computed the optimal strategies for some examples with longer memories and found that the optimal strategy is not generally SD.

The optimality of the strategies in examples 1 and 2 may also be established using simple arguments based on realizations of the sample paths. Sample path arguments also show that the optimal strategies remain SD when the arrival processes are general rather than Poisson. This seems to be a general feature of SD strategies. If a strategy is SD optimal for a problem in which there are Poisson arrivals then it is SD optimal for arbitrary arrivals.

4. APPLICATIONS OF THE SUFFICIENT CONDITION

We consider some examples in which ED strategies exist but neither lemma 3 nor sample path arguments are sufficient to prove their optimality. In these cases we work with lemma 2 directly. We try a H consisting of all rows of R (all rows of $P-\bar{P}$) and any other rows which are in some way obvious from the nature of the problem. We then test to see whether we can write $Hc \geq 0$ and $H\bar{P}=BH$ for some nonnegative B. If not, then we construct a new H whose rows are those of the old H and $H\bar{P}$. The process is continued until we have an H satisfying $Hc \geq 0$ and $H\bar{P}=BH$. In practice, we have found that every problem with an ED optimal strategy has a finite-dimensional H satisfying the conditions of lemma 2. Although we have not been able to show that this must necessarily happen, we do have the following result.

Lemma 4. If $\bar{P}$ defines an ED optimal strategy and all the eigenvalues of $\bar{P}$ are real, then there exists a finite dimensional H satisfying the conditions of lemma 2.

Proof. The proof is not particularly illuminating and will be given in outline only. Although the proof can be adapted to include the cases in which eigenvalues are not simple or some have equal modulus, we suppose for simplicity that $|\lambda_1| > \cdots > |\lambda_N|$. We can then write

$$\bar{P}^n = \sum_i \lambda_i^n (z_i y_i'),$$

where y_i and z_i are the left and right eigenvectors of $\bar{P}$ corresponding to the eigenvalue λ_i (y_i' denotes the transpose of y_i). Consider a row of R, say r'. Since $r'\bar{P}^n c \geq 0$ for all n, we deduce that $\lambda_i > 0$ for the least i such that $(r'z_i)(y_i'c) \neq 0$. We repeat this analysis for all rows of R and let $I = \{i$: i is the least index for which $(r'z_i)(y_i'c) \neq 0$, r' some row of R$\}$. Let G be a matrix consisting of all rows of the form,

$$\sum_{i=1}^{j} \beta_i (r'z_i) y_i',$$

where r' is a row of R, $1 \leq j \leq N$, $\beta_i = 1$ for $i \in I$, and $\beta_i = \pm 1$ for $i \notin I$. We can now check that

$Gx \geqslant 0$ implies $G\bar{P}x \geqslant 0$,
$G\bar{P}^n c \geqslant 0$ for all $n \geqslant$ some n_0, and
$Gx \geqslant 0$ implies $R\bar{P}^n x \geqslant 0$ for all $n \geqslant$ some n_1.

Let $n_2 = n_0 + n_1$. We now take H to consist of all the rows appearing in the matrices $R, R\bar{P}, \ldots, R\bar{P}^{n_2}, G\bar{P}^{n_0}$ and check that this H satisfies the conditions of lemma 2.

Note that, $\bar{P}$ has real eigenvalues if it is similar to a symmetric matrix [6]. This happens if $\bar{P}$ is the transition matrix of a time reversible Markov chain or if it can be represented as the limit of transition matrices of time reversible chains.

Example 3. Optimal Gambling

Ross [8] considers a problem of choosing the optimal stake in an advantageous gamble. He supposes that a gambler who has an amount of money £i can gamble any portion of that money in a bet. He wins an amount equal to his stake with probability p ($p \geqslant 0.5$) and loses his stake with probability $q = (1-p)$. The gambler wants to chose the size of his bets (which must be integral) so as to maximize the probability of eventually increasing his capital to £(N-1). Ross shows that the gambler achieves this by always making the minimum bet of £1, and that this strategy minimizes for all t the probability that he is ruined by time t. We show how this may be proved using lemma 2.

The problem is analyzed in discrete time. Let state i be the state in which the gambler has £(i-1). States 1 and N are absorbing, and $\bar{P}$ is the transition matrix when bets of 1 are made at each turn. Let I be the NxN identity matrix and let J be a NxN matrix with 1's above the diagonal and 0's elsewhere. Let $G = (qI - pJ)$. It is simple to check that $G\bar{P} = CG$ for some $C \geqslant 0$. Letting $F = (I-J)$ we can also check that $FG\bar{P} = DFG$ for some $D \geqslant 0$. Rows of R are positive linear combinations of rows of FG, so we take H as all rows of G and FG. Then $R = AH$ some $A \geqslant 0$, $H\bar{P} = BH$ some $B \geqslant 0$, and $Hc \geqslant 0$ for $c = (1, 0, \ldots, 0)$. Thus the strategy of betting the minimum amount minimizes the probability of being ruined by time t.

Example 4. Repair of the Series System

Derman, Lieberman and Ross [3], Katehatis [4], Smith [10] and Weber [12] have considered the problem of optimally maintaining a series system of n components with a single repairman. When functioning, component i fails with constant hazard rate μ_i. When failed, component i takes a time to repair which is exponentially distributed with parameter λ_i. The repairman desires to allocate his repair effort to maximize the probability that all component are functioning at time t. He may allocate his repair effort amongst the failed components in any way he likes, and he may stop the repair of one component in order to begin repair of another. Smith showed that amongst strategies which repair the components according to a fixed precedence list order the least failure rate strategy of always repairing the component with the least value of μ_i is optimal. He conjectured that the strategy should be optimal in the class of all strategies. Derman and others proved this for the case when all the λ_i are equal, and Weber and Katehakis proved it more generally.

We can establish the result using the sufficient condition of lemma 2. Suppose $\mu_1 < \cdots < \mu_n$. Suppose the states of the system are indexed by α $(1 \leq \alpha \leq N)$. Let $L(\alpha)$ be the set of components which are functioning in the αth state. Let $E\alpha$ be a row vector of N components having 1 in the αth position and 0 in other positions. For $i,j \notin L(\alpha)$ define the row vectors

$$S(i)\alpha = \lambda_i(E\alpha - E\alpha^i), \text{ and}$$
$$T(ij)\alpha = S(i)\alpha - S(j)\alpha,$$

where α^i is the index of the state in which the functioning components are $L(\alpha)+\{i\}$. Similarly, we shall let α_h denote the state in which the functioning components are $L(\alpha)-\{h\}$. If i is the component of least μ amongst those which are failed, k is the component of next largest μ, and $i<j$ then,

$$S(i)\alpha\{\bar{Q}+(\mu_i+\lambda_i+\sum_{h\varepsilon L}\mu_h)I\} = \lambda_i S(k)\alpha^i + \sum_{h\varepsilon L}\mu_h S(i)\alpha_h.$$

$$T(ij)\alpha\{\bar{Q}+(\mu_j+\lambda_i+\sum_{h\varepsilon L}\mu_h)I\} = (\mu_j-\mu_i)S(i) + \lambda_i T(kj)\alpha^i + \sum_{h\varepsilon L}\mu_h T(ij)\alpha_h.$$

If component $k\neq i<j$ is the one of least μ amongst those that are failed then,

$$S(i)\alpha\{\bar{Q}+(\mu_i+\lambda_k+\sum_{h\varepsilon L}\mu_h)I\} = \lambda_k S(i)\alpha^k + \sum_{h\varepsilon L}\mu_h S(i)\alpha_h.$$

$$T(ij)\alpha\{\bar{Q}+(\mu_j+\lambda_k+\sum_{h\varepsilon L}\mu_h)I\} = (\mu_j-\mu_i)S(i) + \lambda_k T(ij)\alpha^k + \sum_{h\varepsilon L}\mu_h T(ij)\alpha_h.$$

The matrix H which is appropriate for lemma 2 consists of all rows of the form $S(i)\alpha$ and $T(ij)\alpha$, for all α with $(i<j)\notin L(\alpha)$. The matrix R has rows which are positive linear combinations of the rows $T(ij)\alpha$. The statements above and $(\mu_j-\mu_i)\geq 0$ are sufficient for there to exist a θ (equal to the sum of all the λ's and μ's) such that $H(\bar{Q}+\theta I)=\theta BH$ for some nonnegative matrix B. Taking c as a vector which is equal to 1 in all components except the one corresponding to the state in which all components are functioning, we deduce that the least failure rate repair strategy maximizes for all t the probability that the all components are functioning at time t.

Example 5. Job Scheduling

A number of authors have proved that the longest expected processing time order strategy (LEPT) minimizes the expected value of the makespan when n jobs with exponentially distributed processing requirements are to be processed on m identical parallel machines (see [2] and [15]). In [13] we have shown that LEPT is stochastic dominant in that it minimizes the makespan in distribution. We shall indicate how the proof can be accomplished using lemma 2. Suppose the failure rates are $\lambda_1 < \cdots < \lambda_n$. We use the same notation as in example 4 and define $S(i)\alpha$ and $T(ij)\alpha$ as before. Suppose $L(\alpha)$ is the set of jobs which are not yet completed in the αth state. Let $K(\alpha)$ be the set of m jobs of least hazard rate amongst those which are uncompleted. For $i,j \in K(\alpha)$, $k \notin K(\alpha)$ and $k \in K(\alpha^i)$ we also define

$$U(ijk)\alpha = \lambda_i T(kj)\alpha^i + \lambda_j T(ik)\alpha^j.$$

Let H be a matrix whose rows are all the rows of $S(i)\alpha$, $T(ij)\alpha$ with $i<j$, and $U(ijk)\alpha$ with $i<j\in K(\alpha)$. All rows of R and $R\bar{P}$ are positive linear combinations of the rows of H. The $S(i)\alpha$ rows state that it is better to start from a state in which a given job is already complete rather than in a state which differs by that job being uncomplete. With $(\Sigma\lambda_i)\bar{P}=\bar{Q}+(\Sigma\lambda_i)I$ we find

$$S(i)\alpha\bar{P} = \sum_{h\in K}\lambda_h S(i)\alpha^h + \sum_{h\notin K}\lambda_h S(i)\alpha + \lambda_i S(k)\alpha^i \quad \text{for } i\in K, \tag{5}$$

$$S(i)\alpha\bar{P} = \sum_{h\in K}\lambda_h S(i)\alpha^h + \sum_{h\notin K}\lambda_h S(i)\alpha + \lambda_i S(k)\alpha \quad \text{for } i\notin K, \tag{6}$$

$$\begin{aligned} T(ij)\alpha\bar{P} = {} & \sum_{h\in K}\lambda_h T(ij)\alpha^h + \sum_{h\notin K}\lambda_h T(ij)\alpha + \lambda_i T(kj)\alpha^i \\ & + \lambda_j T(ik)\alpha^j \quad \text{for } i,j\in K, \end{aligned} \tag{7}$$

$$\begin{aligned} T(ij)\alpha\bar{P} = {} & \sum_{h\in K}\lambda_h T(ij)\alpha^h + \sum_{h\notin K}\lambda_h T(ij)\alpha + \lambda_i T(kj)\alpha^i \\ & + \lambda_j T(ij)\alpha \quad i\in K, \quad \text{for } j\notin K, \end{aligned} \tag{8}$$

$$\begin{aligned} T(ij)\alpha\bar{P} = {} & \sum_{h\in K}\lambda_h T(ij)\alpha^h + \sum_{h\notin K}\lambda_h T(ij)\alpha + \lambda_i T(ij)\alpha \\ & + \lambda_j T(ij)\alpha \quad \text{for } i,j\notin K, \text{ and} \end{aligned} \tag{9}$$

$$\begin{aligned} U(ijk)\alpha\bar{P} = {} & \sum_{h\in K}\lambda_h U(ijk)\alpha^h + \sum_{h\notin K}\lambda_h U(ijk)\alpha + \lambda_i U(ijk)\alpha \\ & + (\lambda_j-\lambda_i)S(ik)\alpha^j + \lambda_k U(ij\ell)\alpha^k. \end{aligned} \tag{10}$$

In each of (5)-(10) the sums on the right hand side should be read to exclude any of the indices i,j,k which appear as arguments on the left hand side. In (10) ℓ is a job such that $\ell\notin K(\alpha^k)$ and $\ell\in L(\alpha^{ik})$. If there is no such ℓ then the last term of (10) is replaced by $\lambda_k\{\lambda_j S(i)\alpha^k-\lambda_i S(j)\alpha^k\}$. These equations establish $H\bar{P}=BH$ for some

$B \geq 0$, and the result follows by taking c as a vector which is 0 in the component corresponding to all jobs complete and 1 in all other components.

5. CONCLUDING REMARKS

The examples of the previous sections have shown that lemma 2 may be useful in establishing the optimality of a dominant strategy for a stochastic allocation or scheduling problem. In fact, for every problem which we know to have an expectation dominant strategy the optimality of that strategy can be established by this method. Winston's problem [17] of customer assignment to heterogeneous servers is another example where the method is useful.

Example 5 illustrates that it can sometimes be quite difficult to construct the appropriate matrix H. Indeed the H required to prove the results of [13] is even more complicated, though the difficulty is one of notation and not methodology. At the beginning of section 4 we described a method of building the appropriate matrix H from rows of of $R, R\bar{P}, R\bar{P}^2, \ldots$, along with rows which were somehow obvious from the nature of the problem. In fact, the obvious rows were simply ones which stated that a transition was preferred or not preferred to remaining in the current state. It would be interesting to know whether the constructive approach will always succeed, or whether there is a problem with a dominant optimal strategy whose optimality cannot be established by this method. One might also consider whether other sufficient conditions, perhaps more easily verified, would ensure that a strategy were ε-expectation dominant.

REFERENCES

[1] Benes, V.: 1981, Models and problems of dynamic memory allocation. "Proceedings of the ORSA-TIMS meeting, Applied Probability-Computer Science: the Interface", (to appear).

[2] Bruno J., Downey P., and Frederickson, G.N.: 1981, Sequencing tasks with exponential service times to minimize the expected flowtime or makespan. JACM 28, pp. 100-113.

[3] Derman, C., Lieberman, G.U. and Ross, S.M.: 1980, On the optimal assignment of servers and a repairman. J. Appl. Prob. 17, pp. 245-257.

[4] Katehakis, M.: 1980, On the repair of a series system. (private correspondence).

[5] Keilson, J. and Kester, A.: 1977, Monotone matrices and monotone Markov processes, Stoc. Proc. and Appl. 5, pp. 231-241.

[6] Keilson, J.: 1979, "Markov Chain Models-Rarity and Exponentiality", Springer-Verlag, New York.

[7] Lippman, S.: 1975, Applying a new device in the optimization of exponential queueing systems. Opns. Res. 23, pp. 687-710.

[8] Ross, S.M.: 1974, Dynamic programming and gambling models. J. Appl. Prob. 6, pp. 593-606.

[9] Serfozo, R.F.: 1979, An equivalence between continuous and discrete time Markov decision processes. Opns. Res. 27, pp. 616-620.

[10] Smith, D.R.: 1968, Optimal repair of the series system. Opns. Res. 4, pp. 653-662.

[11] Weber, R.R.: 1978, On the optimal assignment of customers to parallel servers. J. Appl. Prob. 15, pp. 406-413.

[12] Weber, R.R.: 1980, "Optimal Organization of Multi-Server Systems", Ph.D. Thesis, University of Cambridge.

[13] Weber, R.R.: 1982, Scheduling jobs with unknown processing requirements on parallel machines to minimize makespan or flowtime. J. Appl. Prob. 19, (to appear).

[14] Weeks, J.K. and Wingler, T.R.: 1979, A stochastic dominance ordering of scheduling rules. Decision Sci. 10, pp. 245-257.

[15] Weiss, G. and Pinedo, M.: 1979, Scheduling tasks with exponential service times on non-identical processors to minimize various cost functions. J. Appl. Prob. 17, pp. 187-202.

[16] Winston, W.L.: 1977, Optimality of the shortest line discipline. J. Appl. Prob. 14, pp. 181-189.

[17] Winston, W.L.: 1977, Assignment of customers to servers in a heterogeneous queueing system with switching. Opns. Res. 25, pp. 469-483.