MATHEMATICS OF OPERATIONS RESEARCH
Vol. 12, No. 2, May 1987
Printed in U.S.A.

THE MULTI-ARMED BANDIT PROBLEM: DECOMPOSITION AND COMPUTATION*†

MICHAEL N. KATEHAKIS‡ AND ARTHUR F. VEINOTT, JR.§

This paper is dedicated to our friend and mentor,
Cyrus Derman, on the occasion of his 60th birthday.

The multi-armed bandit problem arises in sequentially allocating effort to one of N projects and sequentially assigning patients to one of N treatments in clinical trials. Gittins and Jones (1974) have shown that one optimal policy for the N-project problem, an N-dimensional discounted Markov decision chain, is determined by the following *largest-index rule*. There is an index for each state of each given project that depends only on the data of that project. In each period one allocates effort to a project with largest current index. The purpose of this paper is to give a short proof of this result and a new characterization of the index of a project in state i, viz., as the maximum expected present value in state i for the restart-in-i problem in which, in each state and period, one either continues allocating effort to the project or immediately restarts the project in state i. Moreover, it is shown that an approximate largest-index rule yields an approximately optimal policy. These results lead to more efficient methods of computing the indices on-line and/or for sparse transition matrices in large state spaces than have been suggested heretofore. By using a suitable implementation of successive approximations, a policy whose expected present value is within $100\epsilon\%$ of the maximum possible range of values of the indices can be found on-line with at most $(N + T - 1)TM$ operations where M is the number of operations required to calculate one approximation, T is the least integer majorizing the ratio $\ln \epsilon / \ln a$ and $0 < a < 1$ is the discount factor.

1. Introduction and summary. The multi-armed bandit problem can be described in terms of sequentially allocating effort to one of N independent projects or sequentially assigning patients to one of N possible treatments in clinical trials. In the sequel, we discuss the problem in terms of project scheduling. In each period, one observes the states of the N projects and activates one project in the period. The active project earns a reward that depends only on the project and its state, and then moves to a state in the next period according to a Markov transition law that also depends only on the project and its state. The inactive projects in a period earn no rewards and their states remain frozen in the period. The goal is to find a maximum- (expected-present-) value policy for choosing the active project in each period.

Gittins and Jones (1974), alternately, Whittle (1980, 1982), and more simply, Varaiya et al. (1985), have shown that the maximum-value N-armed bandit problem, an N-dimensional Markov decision problem, can be reduced to a sequence of one-dimensional stopping problems. In each of the latter problems, one finds for each state i of a project, its index $m_i \equiv \max_{\tau \geq 1} ER_\tau / (1 - Ea^\tau)$, where $\tau + 1$ is a stopping time for the

*Received August 12, 1985; revised February 7, 1986.
AMS 1980 subject classification. Primary 90C47.
IAOR 1973 subject classification. Main: Programming, Dynamic.
OR/MS Index 1978 subject classification. Primary: 117 Dynamic programming/Markov.
Key words. Multi-armed bandit problem, largest-index rule, restart problem, stopping problem, sequential design of experiments, sequential project scheduling, stopping times, Markov decision chains, policy improvement, successive approximations.
†Partially supported by USAF Contract AFOSR 840136 and NSF Grant ECS83-12356.
‡Technical University of Crete and State University of New York at Stony Brook.
§Stanford University.

0364-765X/87/1202/0262$01.25

project, $0 < a < 1$ is the discount factor, and R_τ is the present value of rewards earned in periods $1, \dots, \tau$ when the project is active in those periods. (Incidentally, the above and all other expectations are conditional on the initial state, which is suppressed throughout.) In each period, one selects as the active project one with largest index in the current state. Gittins and Jones (1974), Gittins (1979) and Whittle (1980, 1982) show that m_i is also an indifference value, i.e., a stopping reward m for which one is indifferent between stopping and continuing in state i. They suggest that m_i be computed by solving this last stopping problem parametrically for several values of m. Beale (1979), Varaiya et al. (1985), Chen and Katehakis (1986) and Kallenberg (1986) have respectively proposed policy-improvement, largest-remaining-index, linear-programming and parametric-linear-programming methods for finding the indices in the general finite-state case.

All these papers appear to have overlooked the fact that m_i is the maximum value in state i in the *restart-in-i problem* in which two actions are available in each period, viz., to continue or to restart instantaneously in state i. This observation reduces the problem of finding m_i to that of solving a single maximum-value discounted Markov decision chain problem and so permits standard methods available for solving the latter problem to be used to solve the former.

In §2 we use stopping times to give a short proof that an approximate largest-index rule is approximately optimal. This provides a short proof that the Gittins-Jones largest-index rule is optimal. In §3 we characterize the indices as values of restarting problems. We use this fact in §4 to give short proofs characterizing the optimal continuation and restarting sets for the restarting problems.

In §5 we use the above results to develop alternate methods of computing that are often more efficient than earlier methods for large state spaces. There are two general strategies for implementing the Gittins-Jones largest-index rule, viz., computing the needed indices *in advance* or *on-line*. The first strategy entails computing in advance the indices for every conceivable state of every project. The second strategy involves computing on-line the indices only for those states that projects actually do enter. The latter approach requires far less computation and also obviates the necessity for providing the user with a large table of indices for all states of all projects.

When each project can be in one of at most a finite number S of states, the most efficient of the algorithms mentioned above seems to be the largest-remaining-index method of Varaiya et al. That method runs in $O(NS^3)$ time whether or not one computes in advance, on-line or with transition matrices that are sparse, i.e., have $O(S)$ nonzero elements. By contrast, we show that in the worst case, application of successive approximations (or its Gauss-Seidel improvement) to solve the restarting problems runs in $O(NS^3)$ time (but not as fast as the largest-remaining-index method) when computing in advance and in $O(NS^2)$ time when computing on-line. Moreover, for sparse matrices, these running times fall to $O(NS^2)$ and $O(NS)$ respectively. In short, successive approximations is apparently significantly more efficient than the best previously suggested method for computation on-line and/or with sparse matrices.

2. Approximate optimality of approximate indices.

We now formulate the project-scheduling problem more precisely. There are N independent projects, labeled $1, \dots, N$, only one of which may be active in each period. The *state* i_t^n of project n in the tth period it is active is a (for simplicity) countable-state Markov chain. Put $i^n \equiv (i_1^n, i_2^n, \dots)$ and assume that i^k and i^n are independent for all $k \neq n$. The states of the $N - 1$ inactive projects in a period are frozen. If project n is active in a period when it is in state i, the project earns a (for simplicity) bounded reward r_i^n. Inactive projects earn no rewards. There is a discount factor $0 < a < 1$. A policy is a (possibly randomized) rule for activating projects that is *nonanticipative*, i.e., the project activated in a period depends only on the states of the projects observed in that and prior periods. The goal is to find a policy with maximum (expected present) value.

We now require a few definitions. Put $R^n_t \equiv \sum_1^t a^{j-1} r^n_{i^n_j}$ and $A_t \equiv 1 - a^t$. A *random time* is a $+\infty$ or nonnegative integer-valued random variable. A *stopping time for project n* is a random time σ that is nonanticipative for project n, i.e., $P(\sigma \leqslant t | i^n)$ is independent of $i^n_{t+1}, i^n_{t+2}, \ldots$ for each nonnegative integer t. Call ER^n_τ / EA_τ the *index of the stopping time $\tau + 1 > 1$ of project n in state i* and call $m^n_i \equiv \max_{\tau \geqslant 1} ER^n_\tau / EA_\tau$ the *index of project n in state i* where the maximum is over stopping times $\tau + 1$. A policy may be described inductively by *selecting*, for each $t = 1, 2, \ldots,$ a period τ_t and a project that will be activated during periods $\tau_{t-1} + 1, \ldots, \tau_t$ where $\tau_0 \equiv 0$ and $\tau_{t-1} < \tau_t$ whenever $\tau_{t-1} < \infty$. Since policies are nonanticipative and since i^k and i^n are independent for $k \neq n$, $\tau_t + 1 - \tau_{t-1}$ is the *stopping time for the project selected* in period $\tau_{t-1} + 1$ relative to the state of the project in that period.

We now show that a policy is *ϵ-optimal*, i.e., has value within ϵ of the maximum value, if the index of the stopping time of the project selected in any period is within ϵ of the largest of the indices of the projects in the period. When $\epsilon = 0$, this specializes to Gittins and Jones' result. Our proof has much in common with, but is simpler than, that of Varaiya et al. (1985).

The key fact underlying our development is that for any sequence $1 \geqslant s_1 \geqslant s_2 \geqslant \cdots \geqslant 0$ of Borel functions of $\mathbf{i} \equiv (i^1, \ldots, i^N)$, there is a random time σ for which $P(\sigma \geqslant t | \mathbf{i}) = s_t$ for $t = 1, 2, \ldots,$ so

$$ER^n_\sigma = E \sum_{t=1}^{\infty} s_t a^{t-1} r^n_{i^n_t}. \tag{1}$$

One use of this formula is to express the value of the income that a policy earns from project n as ER^n_σ for a suitable random time σ. This is done by letting $s_t = a^{l_t - t}$ where l_t is the tth period that project n is active when using the given policy.

THEOREM 1. (Approximate Optimality of Approximate Largest-Index Rule). *A sufficient condition for a policy to be ϵ-optimal is that the index of the stopping time of the project selected in any period be within ϵ of the largest of the indices of the projects in their states in that period.*

PROOF. Let π^* be a policy satisfying the hypotheses of the theorem and $\tau_{t-1} + 1$ be the tth period in which π^* selects a project. Assume project one say, is selected in period one. Put $\tau \equiv \tau_1$ and $m^1 \equiv ER^1_\tau / EA_\tau$. By hypothesis, $m^1 + \epsilon \geqslant m^n_{i^n_1}$, so for each stopping time $\sigma + 1$ for project n,

$$ER^n_\sigma \leqslant (m^1 + \epsilon) EA_\sigma. \tag{2}$$

Let V_π be the value of any policy π starting from the given initial state. Let π' be an arbitrary policy, $\theta > 0$ be any fixed number and π_0 be a policy that activates each project infinitely often and for which $V_{\pi_0} \geqslant V_{\pi'} - \theta$. Let π_1 be the policy that permutes the order in which π_0 activates projects by shifting the first τ times π_0 activates project one to the first τ periods. Let T be the period in which π_0 activates project one for the τth time if τ is finite and let $T = \infty$ otherwise.

For each fixed n and $t \geqslant 1$, let l_t (resp., $\underline{l}_t$) be the period in which π_0 (resp., π_1) activates project n for the tth time. For $1 \leqslant l_t \leqslant T$, set $s_t \equiv a^{l_t - t}$ if $n = 1$ and $s_t \equiv a^{l_t - t} - a^{\underline{l}_t - t}$ if $n > 1$. For $l_t > T$, $l_t = \underline{l}_t$ and we put $s_t \equiv 0$. Observe that $0 \leqslant s_t \leqslant 1$ is nonincreasing in $t \geqslant 1$ because $l_t - t \geqslant 0$ is nondecreasing and, if $n > 1$, $l_t - \underline{l}_t \leqslant 0$ is nondecreasing in $t \geqslant 1$. Thus, from (1), there is a random time $\sigma^n \leqslant T$ with $P(\sigma^n \geqslant t | \mathbf{i}) = s_t$ for $t \geqslant 1$. Also by (1), the difference between the values that π_0

and π_1 earn from project n is $E\sum_1^\infty(a^{l_t-1} - a^{l_t-1})r^n_{i^n_t}$, and so is $ER^1_{\sigma^1} - ER^1_\tau$ if $n = 1$ and $ER^n_{\sigma^n}$ if $n > 1$. Hence,

$$V_{\pi_1} - V_{\pi_0} = ER^1_\tau - \sum_{n=1}^N ER^n_{\sigma^n}. \tag{3}$$

If the rewards are $1 - a$ in each state, the left-hand side of (3) vanishes, so because $\sum_1^t a^{j-1}(1 - a) = 1 - a^t$, (3) reduces to

$$0 = EA_\tau - \sum_{n=1}^N EA_{\sigma^n}. \tag{4}$$

Now it is easy to show for each n that $\sigma^n + 1 \geqslant 1$ is a stopping time for project n since $P(\sigma^n \geqslant t|\mathbf{i})$ is independent of $i^n_{t+1}, i^n_{t+2}, \ldots$ and i^n is independent of i^k for $k \neq n$. Hence, from (2)–(4) and the definition of m^1,

$$V_{\pi_1} - V_{\pi_0} \geqslant m^1 EA_\tau - (m^1 + \epsilon)\sum_{n=1}^N EA_{\sigma^n} = -\epsilon EA_\tau.$$

Using the above construction, it follows by induction on $t \geqslant 1$ that there exist policies π_t that agree with π^* through period $\tau_t \geqslant t$ and have the property that $V_{\pi_t} - V_{\pi_{t-1}} \geqslant -\epsilon E(A_{\tau_t} - A_{\tau_{t-1}})$. Thus $V_{\pi_t} - V_{\pi_0} \geqslant -\epsilon EA_{\tau_t} \geqslant -\epsilon$ and $V_{\pi^*} = \lim_{t\to\infty} V_{\pi_t} \geqslant V_{\pi_0} - \epsilon \geqslant V_{\pi'} - \epsilon - \theta$, so because θ is arbitrary, $V_{\pi^*} \geqslant V_{\pi'} - \epsilon$. ■

3. Characterization of indices as values of restarting problems. In this section we characterize the index of a project in a state as the value of the project in a restarting problem. For notational simplicity, we drop the superscript designating the project in the sequel without further mention. The sequence of states an active project enters is a Markov chain on the countable state space I with transition probabilities p_{ij}. Put $r \equiv (r_i)$ and $P \equiv (ap_{ij})$. For each state i, let P_i be the ith row of P.

It follows from well-known results for discounted Markov decision chains that the maximum-value vector for the restart-in-i problem is the unique bounded solution $v = v^i = (v^i_j)$ of

$$v_j = \max(r_j + P_j v, r_i + P_i v), \qquad j \in I. \tag{5}$$

If we look at the process only at times when the process is in state i, then v_i can also be interpreted as the maximum value in state i for the corresponding embedded single-state semi-Markov decision chain. Hence, v_i satisfies $v_i = \max_{\tau \geqslant 1} E(R_\tau + a^\tau v_i)$, or equivalently, $v_i = m_i$, where $\tau + 1$ is a stopping time for the project, viz., the first period after period one in which one chooses to restart in state i in the restart-in-i problem. Thus, the maximum is attained. Moreover, $m = m_i = v_i$ satisfies

$$m = r_i + P_i v. \tag{6}$$

On substituting (6) into (5), we obtain that (cf., Gittins and Jones 1974, Gittins 1979, Whittle 1982) $(v, m) = (v(m_i), m_i)$ is the unique bounded solution of (6) and

$$v_j = \max(r_j + P_j v, m), \qquad j \in I, \tag{7}$$

where $v(m)$ is the unique bounded solution of (7). Thus $v(m)$ is the maximum-value vector for the discounted stopping problem with stopping reward m. These facts imply most of the next result.

PROPOSITION 2. *For each state i, $m_i = v_i^i$. Also m_i is the unique solution of $m_i = r_i + P_i v(m_i)$ and $v^i = v(m_i)$ is nondecreasing in m_i.*

4. Characterization of continuation and restarting sets. The *optimal continuation set* $\mathbf{C}_i$ (resp., *optimal restarting set* $\mathbf{R}_i$) for the restart-in-i problem is the set of states j for which continuation (resp., restarting in i) is optimal, i.e., for which $r_j + P_j v^i - m_i$ is nonnegative (resp., nonpositive). The optimal stopping time in the restart-in-i problem is the number of periods required for the state of the project to leave $\mathbf{C}_i$. Of course one is indifferent between continuing and restarting for each state in $\mathbf{C}_i \cap \mathbf{R}_i$. In particular, that is so of state i.

PROPOSITION 3. (Continuation and Restarting Sets: Gittins). *For each state i, $\mathbf{C}_i$ (resp., $\mathbf{R}_i$) is the set of states j for which $m_j \geqslant m_i$ (resp., $m_j \leqslant m_i$).*

PROOF. By (7), $\|v(m_j) - v(m_i)\|_\infty \leqslant |m_j - m_i|$, so $|P_j v(m_j) - P_j v(m_i)| \leqslant a|m_j - m_i|$. Hence by (6), $r_j + P_j v(m_i) - m_i \geqslant 0$ (resp., $\leqslant 0$) if and only if $m_j - m_i \geqslant 0$ (resp., $\leqslant 0$). ■

Recall from Theorem 1 that a sufficient condition for a policy to be optimal is that the largest-index rule is used in each period in which a project is selected. Proposition 3 assures that such a policy in fact uses the largest-index rule in *every* period.

Proposition 3 characterizes $\mathbf{C}_i$ and $\mathbf{R}_i$ in terms of the value of the index m_i. But what can be said about those sets if m_i is not known? To answer this question, suppose that I is partially ordered by the relation $\leqslant$. A subset J of I is called *increasing* (resp., *decreasing*) if $i \in J$ and $i \leqslant j$ (resp., $i \geqslant j$) in I implies that $j \in J$. Call P *stochastically monotone* if Pv is nondecreasing on I whenever v is nondecreasing and bounded on I. Define $\mathbf{R}_i$ by $(\mathbf{R}_i v)_j \equiv \max(r_j + P_j v, r_i + P_i v)$ for each $j \in I$ and bounded v. Observe that $\mathbf{R}_i$ is a contraction with modulus a, and that v^i is the unique fixed point thereof. Also $\mathbf{R}_i v$ is monotone in v. The next result extends related work of Gittins (1979) and Ross (1983).

PROPOSITION 4. (Monotone Continuation and Restarting Sets). *If I is partially ordered, r is nondecreasing and P is stochastically monotone, then v_j^i and m_i are respectively nondecreasing in i, j and in i. Moreover, $\mathbf{C}_i \supseteq \mathbf{C}_j$ and $\mathbf{R}_i \subseteq \mathbf{R}_j$ for each $i \leqslant j$ in I, and $\mathbf{C}_i$ (resp., $\mathbf{R}_i$) is increasing (resp., decreasing) for each i in I.*

PROOF. The hypotheses imply that $\mathbf{R}_i v$ is nondecreasing on I if v is. Thus since 0 is nondecreasing on I, so is $\mathbf{R}_i^t 0$ and $\lim_{t\to\infty} \mathbf{R}_i^t 0 = v^i$. Now by hypothesis again, $v^i = \mathbf{R}_i v^i \leqslant \mathbf{R}_k v^i$ for $i \leqslant k$ in I, so $v^i \leqslant \lim_{t\to\infty} \mathbf{R}_k^t v^i = v^k$, whence v^i is nondecreasing in i. The last two assertions follow from Proposition 3. ■

5. Comparison of computational methods. *Computation on-line.* It is interesting to ask what indices must be computed and when this must be done in order to implement the Gittins-Jones largest-index rule. In the first period, it is necessary to compute the N initial indices of each project. Subsequently, it suffices to compute at most *one* index in each period. In particular, one computes the index of a project in a period when its state first leaves the optimal continuation set of the project in the state and period of its most-recent prior selection. Thus, by Proposition 3, if the indices are computed on-line only as needed, the indices computed for each project will decrease strictly over time.

Computations in finite state spaces. If there are $S < +\infty$ states, then equation (5) for the restart-in-i problem can be solved by a variety of standard methods, e.g., successive approximations, policy improvement, or linear programming. In implementing these methods, it is convenient to observe that $v_i^i = r_i + P_i v^i$, so that one can replace $r_i + P_i v$ in (5) by v_i for many purposes. However, one must exercise some care in that event because v^i is merely the *least* solution of the revised version of (5).

Policy improvement. Beale (1979) suggested a method for finding the indices that can be shown to be equivalent to use of the policy improvement method to solve the restarting problems. However, the standard implementation of policy improvement requires only about one-half the computations required by Beale's implementation.

Largest-remaining-index method. Varaiya et al. (1985) use Proposition 3 to develop a method that first finds the index of a state with largest index, then the index of a state with next largest index, etc. This *largest-remaining-index* method requires about $S^3/3$ multiplications and additions (and far fewer comparisons) if P is positive and at least one-half that many if the number of positive elements of P is $O(S)$. In most cases the largest-remaining-index method appears to be superior to both the standard version of policy improvement and linear programming. Actually, the largest-remaining-index method finds the largest index less than the index m_i of a given state i say, provided that $\mathbf{C}_i$ is known. This fact and the remark at the beginning of this section suggest that the method will be especially useful if, when a project changes state, the index usually does not drop very far.

Successive approximations. However, it appears to us that successive approximations (or its Gauss-Seidel improvement) is a more efficient way to solve the restarting problems for large state spaces than is the policy improvement or largest-remaining-index method. To see why, we show how to use successive approximations to find a policy whose value is within $100\epsilon\%$ $(0 < \epsilon < 1)$ of $\delta \equiv \beta - \alpha$ of the maximum value where $\alpha \leqslant r_j/(1 - a) \leqslant \beta$. We allow α, β and r_j, but not δ, to depend on the project, but we suppress this dependence for notational simplicity. In view of Theorem 1, it suffices to show how to find a stopping time for each project whose index is within $100\epsilon\%$ of δ of the index m_i of the project in any state i. To that end, suppose that one has available an estimate u of v^i with the property that $\mathbf{R}_i u \leqslant u$ and $\alpha \leqslant u_j \leqslant \beta$ for all j (e.g., $u_j = \beta$ for all j will do). Then $\mathbf{R}_i^k u \downarrow v^i$ and $v_j^i \leqslant \underline{v}_j^i \leqslant v_j^i + \epsilon\delta$ for all j where $\underline{v}^i = (\underline{v}_j^i) \equiv \mathbf{R}_i^T u$ and T is the least integer majorizing $\ln \epsilon / \ln a$ because, since $\mathbf{R}_i$ is a contraction with modulus a,

$$\|\underline{v}^i - v^i\|_\infty = \|\mathbf{R}_i^T u - \mathbf{R}_i^T v^i\|_\infty \leqslant a^T \|u - v^i\|_\infty \leqslant a^T \delta \leqslant \epsilon\delta.$$

Now let $\underline{\mathbf{C}}_i$ be the continuation set of $\underline{v}^i$, i.e., the set of states j such that $r_j + P_j \underline{u}^i \geqslant \underline{v}_i^i$ where $\underline{u}^i \equiv \mathbf{R}_i^{T-1} u$. Then, as above, the value of $\underline{\mathbf{C}}_i$ (i.e., the value of the associated policy) in state i is at least $\underline{v}_i^i - \epsilon\delta \geqslant v_i^i - \epsilon\delta = m_i - \epsilon\delta$. The desired stopping time for the project in state i is, of course, the number of periods required to exit $\underline{\mathbf{C}}_i$ starting from state i.

If the computations are done on-line, then we must find a new approximate stopping time for the project only when the project enters a state $j \notin \underline{\mathbf{C}}_i$. In that event $\mathbf{R}_j \underline{v}^i \leqslant \mathbf{R}_j \underline{u}^i < \mathbf{R}_i \underline{u}^i = \underline{v}^i$, so that in applying successive approximations, we can take $\underline{v}^i$ as the initial estimate of $\underline{v}^j$. Thus, $\underline{v}^i \geqslant \underline{v}^j$, so the successive approximations of the restarting values of a project diminish.

Running time. With successive approximations, if one has an estimate v of v^i at hand, one computes a new approximation $v' = \mathbf{R}_i v$ by the rules: $v_i' = r_i + P_i v$ and $v_j' = \max(r_j + P_j v, v_i')$ for $j \neq i$. Computing each approximation entails at most M *operations*, i.e., multiplications, additions and comparisons, where M is the number of nonzero elements in P. Thus the number of operations needed to estimate a stopping time and index for a project in a state is at most $TM \leqslant TS^2$.

If we estimate the stopping times and indices in advance for each state of each project, the running time of successive approximations, measured in operations, is at most NSTM. If instead we compute on-line, then it suffices to estimate an index and stopping time for each project in its state in period one and for at most one project and in one state in subsequent periods. Also any policy can be used after period T. Thus, when implemented on-line, the running time of successive approximations is reduced by about a factor of S to at most $(N + T - 1)TM$.

Since T is independent of S, it follows that when the transition matrices are sparse (so $M = O(S)$) or the computations are done on-line, successive approximations will be more efficient than the policy improvement and largest-remaining-index methods for large enough S because the last two methods still run in $O(NS^3)$ time (as they do in general). On the other hand, if S is not too large and if both $1 - a$ and ϵ are very small, then the largest-remaining-index method will be more efficient than successive approximations.

Another advantage of successive approximations is that its average running time is often considerably less than that of its worst case. The reason for this is that one can terminate before completing T approximations if two successive approximations v' and v are found for which $\|v' - v\| \leqslant \epsilon\delta(1 - a)$, because this assures that $|m_i - v_i'| \leqslant \epsilon\delta$.

These methods can also be used to approximate the indices in infinite-state problems.

Acknowledgements. We are grateful to Cyrus Derman, Joseph Chang, Dimitri Bertsekas and the referees for helpful comments on this paper.

References

Beale, E. M. L. (1979). Discussant of J. C. Gittins (1979). 171–172.

Chen, Y. R. and Katehakis, M. N. (1986). Linear Programming for Finite State Multi-Armed Bandit Problems. *Math. Oper. Res.* **11** 180–183.

Gittins, J. C. (1979). Bandit Processes and Dynamic Allocation Indices. *J. Roy. Statist. Soc. Ser. B* **41** 148–164.

______ and Jones, D. M. (1974). A Dynamic Allocation Index for the Sequential Design of Experiments. In J. Gani, K. Sarkadi and I. Vince (Eds.), *Progress in Statistics*. European Meeting of Statisticians, 1972, 1, North Holland, Amsterdam, 241–266.

Kallenberg, L. C. M. (1986). A Note on M. N. Katehakis, and Y.-R. Chen's Computation of the Gittins Index. *Math. Oper. Res.* **11** 184–186.

Ross, S. (1983). *Introduction to Stochastic Dynamic Programming*. Academic Press, New York, 148–149.

Whittle, P. (1980). Multi-Armed Bandits and the Gittins Index. *J. Roy. Statist. Soc. Ser. B.* **42** 143–144.

______. (1982). *Optimization over Time*. 1. John Wiley, New York, 210–220.

Varaiya, P., Walrand, J. and Buyukkoc, C. (1985). Extensions of the Multiarmed Bandit Problem: The Discounted Case. *IEEE Trans. Auto. Control* **AC-30** 426–439.

KATEHAKIS: DEPARTMENT OF INDUSTRIAL ENGINEERING, TECHNICAL UNIVERSITY OF CRETE, HANIA, CRETE, GREECE 73100

VEINOTT: DEPARTMENT OF OPERATIONS RESEARCH, STANFORD UNIVERSITY, STANFORD, CALIFORNIA 94305