Journal of the Operations Research
Society of Japan
Vol. 24, No. 4, December 1981

A UNIFIED APPROACH TO ALGORITHMS WITH A SUBOPTIMALITY TEST IN DISCOUNTED SEMI-MARKOV DECISION PROCESSES

Katsuhisa Ohno
Kyoto University

(Received May 15, 1980; Revised April 25, 1981)

Abstract This paper deals with computational algorithms for obtaining the optimal stationary policy and the minimum cost of a discounted semi-Markov decision process. Van Nunen [23] has proposed a modified policy iteration algorithm with a suboptimality test of MacQueen type, where the modified policy iteration algorithm is policy iteration method with the policy evaluation routine by a finite number of iterations of successive approximations and includes the method of successive approximations and policy iteration method as special cases. This paper devises a modified policy iteration algorithm with the suboptimality test of Hastings and Mello type and proves that it constructs a finite sequence of policies whose last element is either a unique optimal policy or an ϵ-optimal policy. Moreover, a new notion of equivalent decision processes is introduced, and many iterative methods for solving a system of linear equations such as the Jacobi method, simultaneous overrelaxation method, Gauss-Seidel method, successive overrelaxation method, stationary Richardson's method and so on are shown to convert the original semi-Markov decision process to equivalent decision processes. Various transformed algorithms are derived from the modified policy iteration algorithm with the suboptimality test applied to those equivalent decision processes. Numerical comparisons are made for Howard's automobile replacement problem. They show that the modified policy iteration algorithm with the suboptimality test is much more efficient than van Nunen's algorithm and is superior to the policy iteration method, linear programming and some transformed algorithms.

1. Introduction

Markov decision processes are Markov chains controlled by actions. In particular, finite state discounted Markov decision processes have been studied extensively and thoroughly, and the following three methods for solving them have been developed: method of successive approximations, policy iteration method and linear programming [6]. The most approved method of these is the policy iteration method, which consists of the policy evaluation routine

and the policy improvement routine [10]. The former routine amounts to solving a system of linear equations in N unknowns, N being the number of states. Thus, when N is large, say in the range of 10^3 to 10^6, it is quite difficult to use the policy iteration method. In fact, many important problems such as inventory control systems, controlled queueing systems, stochastic optimal control problems and so on can be precisely or approximately formulated by Markov or semi-Markov decision processes with N in the above range.

MacQueen [13, 14] has devised a method of successive approximations with a suboptimality test for solving discounted Markov decision processes with a large N. His suboptimality test, however, is not computationally efficient, and Hastings and Mello [7] have derived an improved suboptimality test for discounted semi-Markov decision processes. A comprehensive survey of the method of successive approximations with a suboptimality test for semi-Markov decision processes has been made by White [27]. Van Nunen [23, 24] has recently proposed a method of value oriented successive approximations with a suboptimality test for solving discounted Markov decision processes and proved its convergence under a certain condition. It is to be noted, however, that his suboptimality test is of MacQueen type. He has obtained his method by adding an extra routine computing the approximate expected total cost of an improved policy to the method of successive approximations. On the other hand, Puterman and Shin [19] have derived the same method without any suboptimality test by replacing the policy evaluation routine by a finite number of iterations of successive approximations and called it a modified policy iteration algorithm. Since their derivation is more natural than van Nunen's, their terminology is used in this paper. Moreover these derivations show that a modified policy iteration algorithm includes the method of successive approximations and the policy iteration method as special cases.

This paper deals with discounted semi-Markov decision processes. Its main purposes are as follows: to devise a modified policy iteration algorithm with the suboptimality test of Hastings and Mello type, to prove that this constructs a finite sequence of policies whose last element is a unique or an ε-optimal policy, and to derive various transformed algorithms with the suboptimality test from a new notion of equivalent decision processes. The organization of this paper is as follows. Section 2 contains statements of the discounted semi-Markov decision process and the method of successive approximations. In Section 3, new lower and upper bounds for the minimum cost are derived for the sequence generated by a modified policy iteration algorithm. These bounds yield a suboptimality test of Hastings and Mello type which can eliminate suboptimal actions in the policy improvement routine. A modified

policy iteration algorithm with the suboptimality test is proposed and its convergence is proved. In Section 4, equivalent decision processes are defined and many iterative methods for solving a system of linear equations such as the Jacobi method, simultaneous overrelaxation method, Gauss-Seidel method, successive overrelaxation method, stationary Richardson's method and so on [28] are shown to convert the original semi-Markov decision process to equivalent decision processes. Section 5 discusses many transformed algorithms which are derived from the modified policy iteration algorithm with the suboptimality test applied to the equivalent decision processes. Section 6 shows numerical comparisons between the modified policy iteration algorithm with the suboptimality test, van Nunen's algorithm [23], policy iteration method, linear programming and transformed algorithms for Howard's automobile replacement problem [10].

2. Discounted Semi-Markov Decision Processes

Consider a semi-Markov process with a finite number of states $i\in I=\{1,\ldots, N\}$. Whenever state i is reached, an action k is chosen from a finite set K_i. This choice of action determines the expected discounted cost r_{ik} generated in the current state, and the probability $p_{ij}(k)$ that the process next moves to state j, where costs are assumed to be continuously discounted at interest rate $\alpha>0$. Let $G_{ij}(t;k)$ be the conditional probability distribution function of the transition time from state i to j given that the process moves from i to j under action k. Then the discounted transition probability $q_{ij}(k)$ is defined as

$$(2.1)\quad q_{ij}(k)=p_{ij}(k)\int_0^\infty e^{-\alpha t}dG_{ij}(t;k) \qquad (i,j\in I,\ k\in K_i).$$

It is assumed [22, p.157] that there exist positive constants ε_1 and ε_2 such that for all $i\in I$ and $k\in K_i$,

$$\sum_{j\in I} p_{ij}(k)G_{ij}(\varepsilon_1;k) \le 1-\varepsilon_2.$$

This assumption, which is satisfied for all practical semi-Markov processes, implies that for all $i\in I$ and $k\in K_i$,

$$(2.2)\quad \alpha_i(k) \equiv \sum_{j\in I} q_{ij}(k) < 1.$$

It is to be noted that for discrete time Markov decision processes with discount factor β,

$$(2.3)\quad q_{ij}(k) = \beta p_{ij}(k) \quad \text{and} \quad \alpha_i(k)=\beta \quad \text{for all } i,j\in I \text{ and } k\in K_i.$$

The Cartesian product of all $K_i(i\in I)$ is called the policy space denoted by F.

For $f=(f_1,\ldots,f_N)\in F$, let $r(f)$ be the N dimensional column vector whose ith component is r_{if_i}, and $Q(f)$ the N×N matrix whose (i,j) component is $q_{ij}(f_i)$. A policy is a sequence $\pi=(f^1,f^2,\ldots)$ of elements f^n of F, and the expected total cost vector adopting policy π is given by

$$v(\pi) = r(f^1) + \sum_{n=1}^{\infty} Q(f^1)\cdots Q(f^n)r(f^{n+1}).$$

Then the discounted semi-Markov decision process is defined as the quadruplet (I,F,Q,r) associated with the problem of determining an optimal policy π^* that minimizes $v(\pi)$ over all policies.

A policy of the form $f=(f,f,\ldots)$ is called a stationary policy. It is well-known [4, 5, 25] that there exists an optimal stationary policy $f^*=(f^*, f^*,\ldots)$ that minimizes $v(\pi)$ over all policies. Denote the minimum cost $v(f^*)$ by $v^*\in E^N$, where E^N is N dimensional Euclidean space. There are three approaches to obtaining the optimal policy $f^*\in F$ and the minimum cost $v^*\in E^N$: the method of successive approximations, policy iteration method and linear programming [6]. The method of successive approximations is as follows: for any starting vector $v^0=(v_i^0)\in E^N$ and $n=1,2,\ldots$, compute $v^n=(v_i^n)$ by

$$v_i^n = \min_{k\in K_i} \{r_{ik} + \sum_{j\in I} q_{ij}(k)v_j^{n-1}\} \quad (i\in I). \tag{2.4}$$

Let f_i^n be an action k that attains the minimum value of the right hand side. Denote by $\|v\|$ and $\|Q(f)\|$ the ℓ_∞-norm and the corresponding matrix norm, respectively. That is,

$$\|v\| = \max_{i\in I}|v_i| \quad \text{and} \quad \|Q(f)\| = \max_{i\in I}\sum_{j\in I}|q_{ij}(f_i)|.$$

Since all $q_{ij}(k)$ are nonnegative, if $v\le w$, then

$$Q(f)v\le Q(f)w, \tag{2.5}$$

where $v\le w$ means $v_i\le w_i$ for all $i\in I$. Moreover, for column vector $e=(1,\ldots,1)\in E^N$,

$$\|Q(f)\| = \|Q(f)e\| = \max_{i\in I}\alpha_i(f_i). \tag{2.6}$$

Define β and γ by

$$\beta = \max_{i\in I,k\in K_i} \alpha_i(k) \quad \text{and} \quad \gamma = \min_{i\in I,k\in K_i} \alpha_i(k). \tag{2.7}$$

Then (2.2) and (2.6) imply that for all $f\in F$,

$$\|Q(f)\| \le \beta < 1. \tag{2.8}$$

Define the operators $T(f)$ and A mapping E^N into E^N by

$$T(f)v = r(f) + Q(f)v \tag{2.9}$$

and

$$Av=\min_{f\in F} T(f)v, \tag{2.10}$$

where minimization is taken componentwise. Then (2.4) can be rewritten as

(2.11) $v^n = Av^{n-1} = T(f^n)v^{n-1}$.

Definition 1. $T(f)$ and A are said to be *monotone*, if for all $v,w\in E^N$ such that $v\leq w$,

(2.12) $T(f)v \leq T(f)w$ and $Av \leq Aw$;

They are called *contractive*, if there exists $\tilde{\beta}<1$ such that for all $v,w\in E^N$,

(2.13) $\|T(f)v-T(f)w\| \leq \tilde{\beta}\ \|v-w\|$ and $\|Av-Aw\| \leq \tilde{\beta}\|v-w\|$.

The following lemma is a direct consequence of the definition.

Lemma 1. $T(f)$ is monotone and contractive if and only if $Q(f)$ is non-negative and satisfies (2.8). If $T(f)$ is monotone and contractive for all $f\in F$, then A is also monotone and contractive.

Lemma 1 shows that $T(f)$ and A are monotone and contractive. Consequently, from the fixed-point theorem [5] it follows that $T(f)$ and A have unique fixed-points $v(f)$ and v^*, respectively, *i.e.*

(2.14) $v(f) = (I-Q(f))^{-1}r(f)$

and

(2.15) $v^* = \min_{f\in F} v(f)$.

Note that the spectral radius $\sigma(Q(f))$ of matrix $Q(f)$ satisfies

(2.16) $\sigma(Q(f)) \leq \|Q(f)\| \leq \beta < 1$,

and $(I-Q(f))$ is nonsingular [28]. The following lemma is due to Denardo [5].

Lemma 2.

(i) If $Av\leq v$, then $v^*\leq v$.

(ii) For any $v^0\in E^N$ and $f\in F$, $T(f)^n v^0$ and $A^n v^0$ converge to $v(f)$ and v^*, respectively, where $T(f)^n v^0$ is determined recursively by

(2.17) $T(f)^0 v^0 = v^0$

and for $n=1,2,\ldots$,

(2.18) $T(f)^n v^0 = T(f)(T(f)^{n-1}v^0)$,

and $A^n v^0$ is determined in the same way.

3. Modified Policy Iteration Algorithm with a Suboptimality Test

Another approach to finding f^* and v^* is the policy iteration method [1, 10]. It consists of the following two routines. For any starting policy $f^0\in F$ and $n=1,2,\ldots$,

policy evaluation routine: solve $T(f^{n-1})v=v$ to obtain $v(f^{n-1})$;

policy improvement routine: find $f^n\in F$ such that $Av(f^{n-1})=T(f^n)v(f^{n-1})$.

In the policy evaluation routine, the system of N linear equations $T(f^{n-1})v=v$

must be solved. The Gauss elimination method which is usually used in finding $v(f^{n-1})$ requires $(N^3+3N^2-N)/3$ multiplications and divisions. Therefore, when N is large, for example, over several thousands, it is difficult to obtain the exact value of $v(f^{n-1})$ by the Gauss elimination method. Another method of finding $v(f^{n-1})$ is the method of successive approximations given by (2.18). Its convergence to $v(f^{n-1})$ is shown in Lemma 2. Since the computation of $T(f^{n-1})v$ requires N^2 multiplications, this method is superior to the Gauss elimination method if convergence occurs within N/3 iterations. This implies that the method of successive approximations ought to be used in the policy evaluation routine when N is large. Note that an approximate value of $v(f^{n-1})$ works well in the policy improvement routine, if it gives the same policy as the exact value of $v(f^{n-1})$ does. This suggests that the policy evaluation routine can be replaced by a finite number of iterations of successive approximations. Let m be a given number of iterations. Moreover, starting from v^0 instead of f^0 yields a modified policy iteration algorithm: For any starting vector v^0 and $n=1,2,\ldots$, compute f^n, w^n and v^n by

(3.1) $\quad w^n = Av^{n-1} = T(f^n)v^{n-1}$

and

(3.2) $\quad v^n = T(f^n)^m w^n,$

where m is the given nonnegative integer. It should be noted that this algorithm with $m=0$ is identical to the method of successive approximations (2.11) and that as m increases, the algorithm approaches the original policy iteration method. Puterman and Shin [19] have discussed the modified policy iteration algorithm and van Nunen [23] has devised the algorithm with suboptimality test of MacQueen type. They have proved the convergence of their algorithms under a certain condition. The purposes of this section are to devise the modified policy iteration algorithm with the suboptimality test of Hastings and Mello type [7] which is more efficient than that of MacQueen type (see Remark 2) and to prove its convergence without assuming any condition. In Sections 4 and 5, many algorithms with the suboptimality test will be derived from the present algorithm.

The following lemma is due to Porteus [16], who has proved it using only the monotonicity and the contractiveness of $T(f)$ and A.

Lemma 3. If for scalars a and b, $be \le w-v \le ae$, then

(i) $\quad b\gamma(b)^m e \le T(f)^m w - T(f)^m v \le a\beta(a)^m e$ for all $f \in F$,

(ii) $\quad b\gamma(b)^m e \le A^m w - A^m v \le a\beta(a)^m e,$

where $\beta(a)$ and $\gamma(b)$ are given by

(3.3) $\quad \beta(a)=\beta$ if $a \ge 0$, $=\gamma$, otherwise,

and

(3.4) $\gamma(b) = \gamma$ if $b \geq 0$, $= \beta$, otherwise.

Let scalars Δ_n, ∇_n, a_n, b_n, c_n and d_n be defined as follows:

(3.5) $\Delta_n = \max_{i \in I}\{v_i^n - v_i^{n-1}\}$, $\nabla_n = \min_{i \in I}\{v_i^n - v_i^{n-1}\}$,

(3.6) $a_n = \max_{i \in I}\{w_i^n - v_i^n\}$, $b_n = \min_{i \in I}\{w_i^n - v_i^n\}$,

(3.7) $c_n = \max_{i \in I}\{w_i^n - v_i^{n-1}\}$, $d_n = \min_{i \in I}\{w_i^n - v_i^{n-1}\}$.

Then an upper bound u^n and a lower bound ℓ^n for v^* are given by the following theorem.

Theorem 1.

(3.8) $\ell^n = v^n + \eta_n e \leq v^* \leq u^n = v^n + \xi_n e$,

where ξ_n and η_n are given by

(3.9) $\xi_n = \min\{-b_n\beta(-b_n)^m/(1-\beta(-b_n)^m),\ \max\{(\Delta_n\beta(\Delta_n) + a_n)/(1-\beta),$
$(\Delta_n\beta(\Delta_n) + a_n)/(1-\gamma)\}\}$

and

(3.10) $\eta_n = \min\{(\nabla_n\gamma(\nabla_n) + b_n)/(1-\beta),\ (\nabla_n\gamma(\nabla_n) + b_n)/(1-\gamma)\}$.

In addition, the following bounds for v^* also hold:

(3.11) $w^n + \min\{\beta d_n/(1-\beta),\ \gamma d_n/(1-\gamma)\}e \leq v^* \leq w^n + \max\{\beta c_n/(1-\beta),\ \gamma c_n/(1-\gamma)\}e$.

Proof: Since by lemma 2, $v(f^n) = \lim_{\ell \to \infty} T(f^n)^\ell w^n$ and by (3.2), $v^n = T(f^n)^m w^n$,

$$v(f^n) - v^n = \sum_{\ell=1}^{\infty} \{T(f^n)^{(\ell+1)m} w^n - T(f^n)^{\ell m} w^n\}$$
$$= \sum_{\ell=1}^{\infty} \{T(f^n)^{\ell m} v^n - T(f^n)^{\ell m} w^n\}.$$

Lemma 3 implies that

$$T(f^n)^{\ell m} v^n - T(f^n)^{\ell m} w^n \leq -b_n\beta(-b_n)^{\ell m} e,$$

because $\max_{i \in I}\{v_i^n - w_i^n\} = -b_n$. Therefore

$$v(f^n) \leq v^n - b_n \sum_{\ell=1}^{\infty} \beta(-b_n)^{\ell m} e = v^n - \{b_n\beta(-b_n)^m/(1-\beta(-b_n)^m)\}e.$$

Since $v^* \leq v(f^n)$, it holds that

(3.12) $v^* \leq v^n - \{b_n\beta(-b_n)^m/(1-\beta(-b_n)^m)\}e$.

From (2.15) and (3.1) it follows that

(3.13) $v^* - v^n = v^* - w^n + w^n - v^n = Av^* - Av^{n-1} + w^n - v^n$
$= Av^* - Av^n + Av^n - Av^{n-1} + w^n - v^n$.

Lemma 3 leads to

$$\nabla_n\gamma(\nabla_n)e \leq Av^n - Av^{n-1} \leq \Delta_n\beta(\Delta_n)e$$

and $\phi_n\gamma(\phi_n)e \leq Av^* - Av^n \leq \psi_n\beta(\psi_n)e$,

where $\psi_n = \max_{i\in I}\{v_i^*-v_i^n\}$ and $\phi_n = \min_{i\in I}\{v_i^*-v_i^n\}$. Consequently (3.13) yields

(3.14) $\psi_n \le \psi_n\beta(\psi_n) + \Delta_n\beta(\Delta_n) + a_n$

and

(3.15) $\phi_n \ge \phi_n\gamma(\phi_n) + \nabla_n\gamma(\nabla_n) + b_n$.

Thus,

$$\psi_n \le (\Delta_n\beta(\Delta_n) + a_n)/(1-\beta(\psi_n))$$
$$\le \max\{(\Delta_n\beta(\Delta_n) + a_n)/(1-\beta),\ (\Delta_n\beta(\Delta_n) + a_n)/(1-\gamma)\}$$

and

$$\phi_n \ge (\nabla_n\gamma(\nabla_n) + b_n)/(1-\gamma(\phi_n))$$
$$\ge \min\{(\nabla_n\gamma(\nabla_n) + b_n)/(1-\beta),\ (\nabla_n\gamma(\nabla_n) + b_n)/(1-\gamma)\}.$$

Combination of these results and (3.12) proves (3.8) through (3.10). From (3.1) it follows that

$$v^*-w^n = Av^*-Aw^n + Aw^n-Av^{n-1}.$$

In much the same way as in the proof of (3.14) and (3.15), Lemma 3 leads to (3.11).

Remark 1. When $m=0$, the modified policy iteration algorithm is identical to the method of successive approximations. Then $a_n=b_n=0$ and the upper and lower bounds (3.8) in Theorem 1 are reduced to

(3.16) $u^n = v^n + \max\{\beta\Delta_n/(1-\beta),\ \gamma\Delta_n/(1-\gamma)\}e$

and

(3.17) $\ell^n = v^n + \min\{\beta\nabla_n/(1-\beta),\ \gamma\nabla_n/(1-\gamma)\}e$

respectively. These bounds agree with ones shown in [2,7,16] for the method of successive approximations (2.11).

Corollary 1. Suppose that $v^{n-1}\ge w^n\ge v^n\ge v^*$. Then the upper and the lower bounds (3.8) for v^* hold with ξ_n and η_n given by

(3.18) $\xi_n = \min\{\gamma^m b_n/(1-\gamma^m),\ (\gamma\Delta_n + a_n)/(1-\gamma)\}$ and

(3.19) $\eta_n = (\beta\nabla_n + b_n)/(1-\beta)$.

Moreover,

(3.20) $w^n + \{\beta d_n/(1-\beta)\}e \le v^* \le w^n + \{\gamma c_n/(1-\gamma)\}e$.

Proof: The assumption implies that $a_n\ge b_n\ge 0\ge\Delta_n\ge\nabla_n$ and $\phi_n\le\psi_n\le 0$. Consequently, (3.3), (3.4), (3.12), (3.14) and (3.15) yield the bounds (3.8) with ξ_n and η_n given by (3.18) and (3.19). Since $d_n\le c_n\le 0$, (3.11) is reduced to (3.20).

A suboptimality test of Hastings and Mello type is derived by using the upper and lower bounds (3.8) for v^*.

Theorem 2. Action k in state i is *suboptimal* if

(3.21) $r_{ik} + \sum_{i\in I} q_{ij}(k)v_j^n > u_i^n - \alpha_i(k)\eta_n$,

where $\alpha_i(k)$ is defined by (2.2) and u^n and η_n are given by (3.8) through (3.10), or they are given by (3.8), (3.18) and (3.19) when the assumption of corollary 1 holds.

Proof: (3.8) implies that

$$r_{ik} + \sum_{j\in I} q_{ij}(k)v_j^* \geq r_{ik} + \sum_{j\in I} q_{ij}(k)\ell_j^n$$
$$= r_{ik} + \sum_{j\in I} q_{ij}(k)v_j^n + \alpha_i(k)\eta_n.$$

Consequently, when (3.21) is satisfied, it holds that

$$r_{ik} + \sum_{j\in I} q_{ij}(k)v_j^* > u_i^n \geq v_i^*.$$

Thus action k in state i can not be optimal.

Theorem 1 or Corollary 1 and Theorem 2 suggest the modified policy iteration algorithm with suboptimality test of Hastings and Mello type, which will sometimes be called the basic algorithm.

Basic Algorithm:

Step 1: Choose an appropriate vector $v^0 \in E^N$, a nonnegative integer m and a positive constant ε. Set $K_i^0 = K_i$ for all $i\in I$, $u^0 = Me$, $\eta_0 = 0$ and $n=1$, where M is a sufficiently large number.

Step 2: For each $i\in I$, compute

(3.22) $w_i^n = \min_{k\in K_i^{n-1}} \{r_{ik} + \sum_{j\in I} q_{ij}(k)v_j^{n-1}\}$

and determine at the same time

(3.23) $K_i^n = \{k\in K_i^{n-1};\ r_{ik} + \sum_{j\in I} q_{ij}(k)v_j^{n-1} \leq u_i^{n-1} - \alpha_i(k)\eta_{n-1}\}$.

If f_i^{n-1} attains the minimum value w_i^n, then set f_i^n as f_i^{n-1}; else, set f_i^n as any action which attains w_i^n. If K_i^n consists of a single action f_i^n for all $i\in I$, go to Step 5.

Step 3: For $\ell = 0, \ldots, m-1$, compute $y^{\ell+1}$ by

(3.24) $y^{\ell+1} = r(f^n) + Q(f^n)y^\ell$,

where $y^0 = w^n$. Set $v^n = y^m$.

Step 4: Calculate Δ_n, ∇_n, a_n, b_n by (3.5), (3.6) and ξ_n, η_n by (3.9), (3.10) or (3.18), (3.19). If

(3.25) $\xi_n - \eta_n \geq 2\varepsilon$,

then compute u^n by (3.8), set $n=n+1$ and go back to Step 2. Otherwise,

(3.26) $v = v^n + \{(\xi_n + \eta_n)/2\}e$

is an ε-approximate value of v^*. Calculate ε_n and δ_n by

(3.27) $\delta_n = \min_{i\in I} \{v_i - r_{if_i^n} - \sum_{j\in I} q_{ij}(f_i^n)v_j\}$

and

$$(3.28)\quad \varepsilon_n = \varepsilon - [\delta_n/(1-\min_{i\in I}\alpha_i(f_i^n))], \text{ if } \delta_n \geq 0;$$
$$= \varepsilon - [\delta_n/(1-\max_{i\in I}\alpha_i(f_i^n))], \text{ otherwise.}$$

The policy f^n is ε_n-optimal.

Step 5: f^n is the unique optimal policy. Compute c_n and d_n by (3.7). The minimum value v^* is estimated by (3.11) or (3.20) as follows:

$$(3.29)\quad v = w^n + [\{\max\{\beta c_n/(1-\beta),\ \gamma c_n/(1-\gamma)\} + \min\{\beta d_n/(1-\beta), \gamma d_n/(1-\gamma)\}\}/2]e,$$

or

$$= w^n + [\{\gamma c_n/(1-\gamma) + \beta d_n/(1-\beta)\}/2]e.$$

In the basic algorithm, the set K_i^n given by (3.23) is justified by Theorem 2 and the approximate values v of v^* given by (3.26) and (3.29) are justified by Theorem 1 or Corollary 1. However, the ε_n-optimality of f^n given by (3.28) has not been proved to be valid. Its validity is shown in the following theorem.

Theorem 3. Suppose that $\xi_n - \eta_n < 2\varepsilon$. Then the policy f^n is ε_n-optimal, where ε_n is given by (3.28).

Proof: From (2.14) and (3.27) it follows that

$$v - r(f^n) - Q(f^n)v = (I - Q(f^n))(v - v(f^n)) \geq \delta_n e.$$

Since $(I - Q(f^n))^{-1} = \sum_{\ell=0}^{\infty} Q(f^n)^\ell \geq 0$, it holds that $v - v(f^n) \geq \delta_n \sum_{\ell=0}^{\infty} Q(f^n)^\ell e$.

Therefore, (3.8), (3.26) and the assumption imply that

$$v^* - v(f^n) \geq \ell^n - v + \delta_n \sum_{\ell=0}^{\infty} Q(f^n)^\ell e$$
$$\geq -\varepsilon_n e,$$

because

$$\sum_{\ell=0}^{\infty} (\min_{i\in I} \alpha_i(f_i^n))^\ell e \leq \sum_{\ell=0}^{\infty} Q(f^n)^\ell e \leq \sum_{\ell=0}^{\infty} (\max_{i\in I} \alpha_i(f_i^n))^\ell e.$$

Thus it has been proved that

$$v^* - v(f^n) + \varepsilon_n e > 0,$$

which implies the ε_n-optimality of f^n. Finally, combination of the above inequality and $v(f^n) - v^* \geq 0$ leads to $\varepsilon_n > 0$.

As mentioned in Remark 1, the basic algorithm with $m=0$ is identical to the method of successive approximations with the Hastings and Mello's suboptimality test, although the ε_n-optimality of f_n in Step 4 is new. We add a few remarks on the basic algorithm, before proving its convergence.

Remark 2: The bounds for v^* derived by van Nunen [23] under the assumption of Corollary 1 are

$$v^{n-1} + \{d_n/(1-\beta)\}e \leq v^* \leq v^{n-1} + \{c_n/(1-\gamma)\}e.$$

These bounds utilize the value of w^n so that they can not eliminate suboptimal actions during the computation of (3.22) as in Step 2. Thus the suboptimality test using these bounds is necessarily MacQueen test, which must be done in an additional step requiring about the same computational effort as in (3.22). Therefore the basic algorithm is more efficient than the algorithm proposed by van Neunen. Moreover, the bounds (3.20) are tighter than his bounds, because

$$\min_{i\in I}\{w_i^n - v_i^{n-1}\}e \le w^n - v^{n-1} \le \max_{i\in I}\{w_i^n - v^{n-1}\}e.$$

Remark 3. In the basic algorithm, the sets K_i^n ($i\in I$) decrease monotonically as n increases. Thus β and γ can change with n:

$$\beta_n = \max_{i\in I, k\in K_i^n}\{\alpha_i(k)\} \quad \text{and} \quad \gamma_n = \min_{i\in I, k\in K_i^n}\{\alpha_i(k)\}. \tag{3.30}$$

Clearly, $0<\gamma=\gamma_1\le\ldots\le\gamma_n\le\gamma_{n+1}\le\ldots\le\beta_{n+1}\le\beta_n\le\ldots\le\beta_1=\beta<1$. It may be useful to revise the values of β or γ by (3.30) when action k that attains $\beta=\alpha_i(k)$ or $\gamma=\alpha_i(k)$ is excluded from K_i^n, or simply in every ℓth iteration. However, since for Markov decision processes, $\alpha_i(k)=\beta=\gamma$ for all $i\in I$ and $k\in K_i$, $\beta=\gamma$ remains constant during the whole iterative procedure.

Remark 4: In the basic algorithm, the integer m is assumed to be constant during the whole iterative procedure. However, it may vary with n, and one simple way may be to iterate (3.22) until $\|y^{\ell+1}-y^{\ell}\| < \varepsilon'$ is satisfied for given constant ε'.

The *convergence of the basic algorithm* is shown in the following theorem.

Theorem 4. For any nonnegative integer m and any $\varepsilon > 0$, the basic algorithm starting from an arbitrary vector v^0 constructs a finite sequence $\{f^n\}$ whose last element is either a unique optimal policy or an ε_n-optimal policy. In particular, if the semi-Markov decision process has the unique optimal policy and ε is sufficiently small, then the basic algorithm finds the unique optimal policy with a finite number of iterations.

Proof: Note that f_i^n determined in Step 2 is always included in K_i^n. This shows that f^n in Step 2 is equal to f^n determined from (3.1). Therefore, values of w^n and v^n in the algorithm are equal to those of w^n and v^n generated by (3.1) and (3.2) (see[27]). Since $w^1 - v^0 \le c_1 e$, where c_1 is given by (3.7), Lemma 3 implies that

$$T(f^1)^{m+1}w^1 - T(f^1)^{m+1}v^0 \le c_1\beta(c_1)^{m+1}e.$$

Consequently, from (3.1) and (3.2) it follows that

$$w^2 - v^1 = Av^1 - v^1 \le T(f^1)v^1 - v^1 \le c_1\beta(c_1)^{m+1}e.$$

By induction, the following inequality holds: for $n=0,1,2,\ldots$,

$$w^{n+1} - v^n \le \bar{A}_n e, \tag{3.31}$$

where $A_n = c_1\beta(c_1)^{n(m+1)}$. Therefore, Lemma 3 leads to

$$T(f^{n+1})w^{n+1} - w^{n+1} = T(f^{n+1})w^{n+1} - T(f^{n+1})v^n \le A_n\beta(c_1)e.$$

Adding this inequality to (3.31) yields

$$T(f^{n+1})w^{n+1} - v^n \le A_n(1 + \beta(c_1))e.$$

Induction leads to

$$T(f^{n+1})^{m-1}w^{n+1} - v^n \le A_n(1 + \beta(c_1) + \cdots + \beta(c_1)^{m-1})e.$$

Since $v^{n+1} - w^{n+1} = T(f^{n+1})^m w^{n+1} - T(f^{n+1})v^n$, the following inequality holds: for $n=0,1,2,\ldots$,

(3.32) $$v^{n+1} - w^{n+1} \le B_n e,$$

where $B_n = \{A_n\beta(c_1)(1 - \beta(c_1)^m)\}/(1 - \beta(c_1))$. Again, (3.31) and Lemma 3 imply that for $\ell=0,1,2,\ldots$,

$$A^{\ell+1}w^{n+1} - A^{\ell}w^{n+1} = A^{\ell+1}w^{n+1} - A^{\ell+1}v^n \le A_n\beta(c_1)^{\ell+1}e.$$

Since $v^* - w^{n+1} = \sum_{\ell=0}^{\infty} (A^{\ell+1}w^{n+1} - A^{\ell}w^{n+1})$, it holds that

(3.33) $$v^* - w^{n+1} \le \{A_n\beta(c_1)/(1 - \beta(c_1))\}e.$$

Adding this inequality to (3.31) leads to

(3.34) $$v^* - v^n \le \{A_n / (1 - \beta(c_1))\}e.$$

Now from (3.32) and Lemma 3 it follows that for $\ell=0,1,2,\ldots$,

$$v^{n+\ell+1} - T(f^*)v^{n+\ell} \le v^{n+\ell+1} - Av^{n+\ell} = v^{n+\ell+1} - w^{n+\ell+1} \le B_{n+\ell}\, e.$$

Since $B_{n+\ell} = \beta(c_1)^{\ell(m+1)}B_n$, the following inequality holds: for $\ell=0,1,2,\ldots$,

(3.35) $$v^{n+\ell+1} - T(f^*)v^{n+\ell} \le B_n\beta(c_1)^{\ell(m+1)}e.$$

When $\ell=0$, this inequality is reduced to

$$v^{n+1} - T(f^*)v^n \le B_n e.$$

Thus, by Lemma 3,

$$T(f^*)v^{n+1} - T(f^*)^2v^n \le B_n\beta(c_1)e.$$

Adding this inequality to (3.35) with $\ell=1$ yields

$$v^{n+2} - T(f^*)^2v^n \le B_n\beta(c_1)\{1 + \beta(c_1)^m\}e.$$

Again, by Lemma 3,

$$T(f^*)v^{n+2} - T(f^*)^3v^n \le B_n\beta(c_1)^2\{1 + \beta(c_1)^m\}e.$$

Adding this inequality to (3.35) with $\ell=2$ yields

$$v^{n+3} - T(f^*)^3v^n \le B_n\beta(c_1)^2\{1 + \beta(c_1)^m + \beta(c_1)^{2m}\}e.$$

Repeating this routine leads to for $\ell=1,2,\ldots$,

(3.36) $$v^{n+\ell} - T(f^*)^\ell v^n \le B_n\beta(c_1)^{\ell-1}[(1 - \beta(c_1)^{\ell m})/(1 - \beta(c_1)^m)]e$$
$$\le [A_n\beta(c_1)^\ell / (1 - \beta(c_1))]e.$$

Hence,

$$v^{n+\ell} - v^* = v^{n+\ell} - T(f^*)^{\ell}v^n + T(f^*)^{\ell}v^n - T(f^*)^{\ell}v^*$$
$$\leq [\{A_n\beta(c_1)^{\ell} / (1 - \beta(c_1))\} + \beta^{\ell}\| v^n - v^*\|]e.$$

Setting $n = 0$ and $\ell = n$ in this inequality produces for $n = 1,2,\ldots,$

$$(3.37) \quad v^n - v^* \leq [\{c_1\beta(c_1)^n / (1 - \beta(c_1))\} + \beta^n\| v^0 - v^*\|] e.$$

Since by (3.31),

$$w^{n+\ell+1} - v^{n+\ell} \leq A_{n+\ell}e = A_n\beta(c_1)^{\ell(m+1)}e,$$

adding this inequality to (3.36) produces

$$w^{n+\ell+1} - T(f^*)^{\ell}v^n \leq A_n\beta(c_1)^{\ell}[\beta(c_1)^{\ell m} + \{1 / (1 - \beta(c_1))\}]e.$$

In the same way as in the proof of (3.37), the following inequality can be obtained: for $n = 1,2,\ldots,$

$$(3.38) \quad w^{n+1} - v^* \leq [c_1\beta(c_1)^n\{\beta(c_1)^{nm} + (1 / (1 - \beta(c_1)))\} + \beta^n\|v^0 - v^*\|]e.$$

In the above, inequalities (3.33), (3.34), (3.37) and (3.38) have been proved. These inequalities show that $\|v^n - v^*\|$ and $\|w^n - v^*\|$ decrease geometrically to zero as n increases. Consequently, for an arbitrary number $\delta > 0$ there exists an integer M_1 such that for all $n \geq M_1$, $\|v^n - v^*\| < \delta$ and $\|w^n - v^*\| < \delta$. Since for $n > M_1$,

$$\| v^n - v^{n-1}\| \leq \| v^n - v^*\| + \| v^* - v^{n-1}\| < 2\delta$$

and $$\| w^n - v^n \| \leq \| w^n - v^*\| + \| v^* - v^n \| < 2\delta,$$

it holds that for $n > M_1$,

$$(3.39) \quad -2\delta < \nabla_n < 2\delta \quad \text{and} \quad -2\delta < b_n < 2\delta,$$

where ∇_n and b_n are given by (3.5) and (3.6), respectively. The basic algorithm terminates at Step 4 or Step 5. Therefore, in order to prove the first part of the theorem, it suffices to show that for $m \geq 1$, there exists a positive δ such that $\xi_n - \eta_n < 2\varepsilon$. From (3.9), (3.10) and (3.39),

$$\xi_n - \eta_n \leq -b_n\beta(-b_n)^m/(1 - \beta(-b_n)^m) - \min\{(\nabla_n\gamma(\nabla_n) + b_n)/(1 - \beta),$$
$$(\nabla_n\gamma(\nabla_n) + b_n) / (1 - \gamma)\}$$
$$< \{2\delta\beta^m / (1 - \beta^m)\} + \{(2\delta\beta + 2\delta) / (1 - \beta)\}.$$

Thus for $n > M_1$

$$(3.40) \quad \xi_n - \eta_n < 2\delta(1 + \beta) / \{(1 - \beta)(1 - \beta^m)\}.$$

Consequently, taking δ as $\delta = \varepsilon(1 - \beta)(1 - \beta^m) / (1 + \beta)$ guarantees that $\xi_n - \eta_n < 2\varepsilon$ holds. Suppose that the semi-Markov decision process has the unique optimal policy f^*. Then there exists a positive number ε_1 such that

$$(3.41) \quad r_{ik} + \sum_{j \in I} q_{ij}(k)v_j^* > v_i^* + \varepsilon_1 \quad (i \in I)$$

holds for all k except f_i^*. To prove the second part of the theorem, it suffices to show that there exists a positive δ such that K_i^n given by (3.23) consists of a single action for all i. Since $v^* - \delta < v_i^{n-1} < v^* + \delta$ for $n > M_1$, by (3.40) and (3.41)

$$\begin{aligned} & r_{ik} + \sum_{j\in I} q_{ij}(k)v_j^{n-1} - v_i^{n-1} - \xi_{n-1} + \alpha_i(k)\eta_{n-1} \\ > & r_{ik} + \sum_{j\in I} q_{ij}(k)v_j^* - v^* - (1+\beta)\delta - 2\delta(1+\beta)/\{(1-\beta)(1-\beta^m)\} \\ & -2\delta(1-\gamma)(1+\beta)/(1-\beta) > \varepsilon_1 - \delta(1+\beta)\{1 + 4/(1-\beta)(1-\beta^m)\}. \end{aligned}$$

Hence, taking δ as $\delta = \varepsilon_1(1-\beta)(1-\beta^m)/\{(1+\beta)(4+(1-\beta)(1-\beta^m))\}$ and ε as zero, guarantees that the basic algorithm terminates at Step 5 within $(M_1 + 2)$ iterations. The proof of the theorem is concluded.

Define a set $V \subset E^N$ as

$$V = \{v \in E^N;\ Av \le v\}. \tag{3.42}$$

It is clear that $v(f) \in V$ for all $f \in F$. Van Nunen [23] has proved the convergence of his algorithm under the condition that $v^0 \in V$. The convergence of a modified policy iteration algorithm has been proved in [19] under the same condition and in [24] without any conditions.

Corollary 2. Suppose that $v^0 \in V$. Then for $n = 1,2,\ldots$,

$$v^0 \ge \cdots \ge v^{n-1} \ge w^n \ge v^n \ge w^{n+1} \ge \cdots \ge v^*.$$

Proof: The assumption implies that $w^1 \le v^0$. Therefore, c_1 in the proof of Theorem 4 can be taken as zero. Then (3.31) and (3.32) are reduced to $v^n \ge w^{n+1}$ and $w^n \ge v^n$, respectively. Moreover, (3.33) and (3.34) are reduced to $w^{n+1} \ge v^*$ and $v^n \ge v^*$, respectively. The proof is concluded.

This corollary shows that v^n generated by the basic algorithm decreases monotonically and the assumption of Corollary 1 is satisfied for all n. Therefore, a starting vector $v^0 \in V$ should be used, if such a vector is easily found. Let us assume that $v^0 = ce$ for some scalar c. Then $v^0 \in V$ means that for all $i \in I$,

$$\min_{k\in K_i} \{r_{ik} + c\alpha_i(k)\} \le c.$$

Since $c\alpha_i(k) \le c\beta(c)$ for all $i \in I$ and $k \in K_i$, if

$$\max_{i\in I} \min_{k\in K_i} \{r_{ik}\} + c\beta(c) \le c,$$

then $v^0 = ce \in V$. Consequently, the starting vector $v^0 \in V$ can be taken as ce, where

$$c = \max_{i\in I} \min_{k\in K_i} \{r_{ik}\}/(1-\beta), \quad \text{if } \max_{i\in I} \min_{k\in K_i} \{r_{ik}\} \ge 0, \tag{3.43}$$

$$= \max_{i\in I} \min_{k\in K_i} \{r_{ik}\}/(1-\gamma), \text{ otherwise.}$$

In the basic algorithm starting from v^0 given by (3.43), ξ_n and η_n in Step 4 should be computed by (3.18) and (3.19) and the second equation of (3.29) should be adopted. Moreover, it is to be noted that, as shown in Theorem 4, the basic algorithm attains its full power in solving semi-Markov decision processes with unique optimal policies, because the termination at Step 5 can occur only for those processes.

4. Equivalent Decision Processes

Consider a decision process (I, F, $\tilde{Q}$, $\tilde{r}$) which has the same state and policy spaces as the semi-Markov decision process (I, F, Q, r) discussed in the preceding sections. However $\tilde{Q}$ may not be a discounted transition probability matrix. Define as in (2.9) and (2.10),

$$\tilde{T}(f)v = \tilde{r}(f) + \tilde{Q}(f)v \tag{4.1}$$

and

$$\tilde{A}v = \min_{f\in F} \tilde{T}(f)v \ . \tag{4.2}$$

Definition 2: A decision process (I, F, $\tilde{Q}$, $\tilde{r}$) is called *equivalent* to the semi-Markov decision process (I, F, Q, r), if

(i) $\tilde{T}(f)$ is monotone and contractive for all $f\in F$,

and

(ii) the fixed point $\tilde{v}(f)$ of $\tilde{T}(f)$ is identical with the fixed point $v(f)$ of $T(f)$ for each $f\in F$.

Now consider the class MC of all decision processes satisfying the above condition (i). Then the above definition induces an equivalence relation R in MC which is defined as (I, F, Q, r)R(I, F, $\tilde{Q}$, $\tilde{r}$) if Condition (ii) is satisfied. In fact it is easy to show that R satisfies the reflexive, symmetric and transitive laws. Thus Definition 2 defines an equivalence relation in the class MC. Denote by $\tilde{f}^*$ and $\tilde{v}^*$ an optimal policy and the minimum cost of the decision process (I, F, $\tilde{Q}$, $\tilde{r}$) and define $\tilde{V}$ as

$$\tilde{V} = \{v\in E^n;\ \tilde{A}v\le v\}.$$

Theorem 5: Let $\tilde{w}^n$, $\tilde{v}^n$ and $\tilde{f}^n$ (n=1,2,...) be the sequences generated by the basic algorithm applied to the decision process (I, F, $\tilde{Q}$, $\tilde{r}$). If this process is equivalent to (I, F, Q, r), then these sequences are finite and the last element of $\{\tilde{f}^n\}$ is either a unique optimal policy or an $\tilde{\varepsilon}_n$-optimal policy of (I, F, Q, r), where $\tilde{\varepsilon}_n$ is given by (3.28) with Q and r replaced by $\tilde{Q}$

and $\tilde{r}$. In particular, if (I, F, O, r) has the unique optimal policy f^* and ε is sufficiently small, then the basic algorithm applied to (I, F, $\tilde{Q}$, $\tilde{r}$) finds f^* with a finite number of iterations. Moreover, if $\tilde{v}^0 \in \tilde{V}$,

(4.3) $\tilde{v}^0 \geq \ldots \geq \tilde{v}^{n-1} \geq \tilde{w}^n \geq \tilde{v}^n \geq \tilde{w}^{n+1} \geq \ldots \geq \tilde{v}^*$

and $\tilde{v}_n$ converges monotonically to $\tilde{v}^*=v^*$, the minimum cost of (I, F, Q, r).

Proof: Lemma 1 implies that $\tilde{q}_{ij}(k) \geq 0$ and $1 \geq \tilde{\beta} \geq \sum_{j \in I} \tilde{q}_{ij}(k)$ for all i, $j \in I$ and $k \in K_i$. Thus $\tilde{\beta}$ and $\tilde{\gamma}$ can be defined as in (2.7). Note that Lemmas 2 and 3 hold for $\tilde{T}(f)$, $\tilde{A}$, $\tilde{\beta}$ and $\tilde{\gamma}$. In addition, Theorem 1 holds for $\tilde{v}^*$, $\tilde{w}^n$, $\tilde{v}^{n-1}$ and $\tilde{v}^n$, because its proof utilizes only the monotonicity and the contractiveness of T and A. Similarly, Theorems 2 through 4 and Corollary 2 hold for (I, F, $\tilde{Q}$, $\tilde{r}$). From (ii) of Definition 2 it follows that

$$\tilde{v}^* = \min_{f \in F} \tilde{v}(f) = \min_{f \in F} v(f) = v^* \quad \text{and} \quad \tilde{f}^* = f^*.$$

The main part of the theorem follows from Theorem 4 and (4.3) from Corollary 2. The proof is concluded.

This theorem shows that an optimal policy f^* and the minimum cost v^* of (I, F, Q, r) can be obtained by the basic algorithm which is applied to any decision process (I, F, $\tilde{Q}$, $\tilde{r}$) equivalent to (I, F, Q, r).

Consider a system of linear equations

(4.4) $(I-Q(f))v = r(f)$.

Let $L(f)$, $D(f)$ and $U(f)$ denote the strictly lower triangular, the diagonal and the strictly upper triangular matrices of $Q(f)$. Define strictly lower and strictly upper triangular matrices $\hat{L}$ and $\hat{U}$ by

(4.5) $\hat{L}(f) = (I-D(f))^{-1}L(f)$ and $\hat{U}(f) = (I-D(f))^{-1}U(f)$.

These matrices are the strictly lower and the strictly upper triangular matrices of $\hat{Q}(f)$ whose elements are determined through $k=f_i$ from

(4.6) $\hat{q}_{ij}(k) = q_{ij}(k)/(1-q_{ii}(k))$ for $i \neq j \in I$, $k \in K_i$

$= 0$ for $i=j \in I$, $k \in K_i$.

Similarly, $\hat{r}(f) = (I-D(f))^{-1}r(f)$ is determined from

(4.7) $\hat{r}_{ik} = r_{ik}/(1-q_{ii}(k))$ for $i \in I$, $k \in K_i$.

There are numerous iterative methods for solving (4.4) [28]. Here we confine our consideration to linear stationary iterative methods which have the form

(4.8) $\tilde{v}^n = \tilde{T}(f)\tilde{v}^{n-1} = \tilde{r}(f) + \tilde{Q}(f)\tilde{v}^{n-1}$ $(n=1,2,\ldots)$.

If $\tilde{T}(f)$ is contractive, then Lemma 2 implies that $\tilde{v}^n$ converges to $\tilde{v}(f)=(I-\tilde{Q}(f))^{-1}\tilde{r}(f)$. Note that iterative methods satisfying $\tilde{v}(f)=v(f)$ are said to be completely consistent with (4.4) [28, p.64]. Thus decision processes (I, F, $\tilde{Q}$, $\tilde{r}$) derived from consistent iterative methods $\tilde{T}$ are equivalent to (I, F, Q, r), if $\tilde{T}$ are monotone and contractive. Since all iterative methods used popu-

larly are consistent with (4.4), Lemma 1 implies that if $\tilde{Q}(f)$ in (4.8) is a nonnegative matrix and satisfies

(4.9) $\| \tilde{Q}(f)e \| < 1$ for all $f \in F$,

then (I, F, $\tilde{Q}$, $\tilde{r}$) with $\tilde{Q}$ and $\tilde{r}$ given by (4.8) is equivalent to (I, F, Q, r). Let ω and Ω be a relaxation parameter (factor) and a nonsingular diagonal matrix, respectively. Basic linear stationary iterative methods for solving (4.4) are [28]:

(i) *Jacobi method (J method):*

(4.10) $\tilde{Q}(f) = \hat{L}(f) + \hat{U}(f)$, $\tilde{r}(f) = \hat{r}(f)$;

(ii) *Simultaneous overrelaxation method (JOR method):*

(4.11) $\tilde{Q}(f) = \omega(\hat{L}(f) + \hat{U}(f)) + (1-\omega)I$, $\tilde{r}(f) = \omega\hat{r}(f)$;

(iii) *Gauss-Seidel method (GS method):*

(4.12) $\tilde{Q}(f) = (I-\hat{L}(f))^{-1}\hat{U}(f)$, $\tilde{r}(f) = (I-\hat{L}(f))^{-1}\hat{r}(f)$;

(iv) *Successive overrelaxation method (SOR method):*

(4.13) $$\tilde{Q}(f) = (I-\omega\hat{L}(f))^{-1}(\omega\hat{U}(f) + (1-\omega)I),$$
$$\tilde{r}(f) = \omega(I-\omega\hat{L}(f))^{-1}\hat{r}(f);$$

(v) *Stationary generalized Richardson's method (GRF method):*

(4.14) $\tilde{Q}(f) = I - \Omega + \Omega Q(f)$, $\tilde{r}(f) = \Omega r(f)$;

(vi) *Stationary Richardson's method (RF method):*

(4.15) $\tilde{Q}(f) = (1-\omega)I + \omega Q(f)$, $\tilde{r}(f) = \omega r(f)$.

Theorem 6. Decision processes (I, F, $\tilde{Q}$, $\tilde{r}$) derived from the J method, the JOR method with ω such that $0<\omega\leq 1$, the GS method, the SOR method with ω such that $0<\omega\leq 1$, the GRF method with Ω whose ith diagonal element ω_i satisfies $0<\omega_i\leq\bar{\omega}_i$ ($i\in I$) and the RF method with ω such that $0<\omega\leq\bar{\omega}$ are equivalent to (I,F, Q, r), where for $\bar{q}_i=\min_{k\in K_i}\{q_{ii}(k)\}$ and $\bar{q}=\min_{i\in I}\{\bar{q}_i\}$, $\bar{\omega}_i$ and $\bar{\omega}$ are given by

(4.16) $\bar{\omega}_i = 1/(1-\bar{q}_i)$ and $\bar{\omega} = 1/(1-\bar{q})$.

Proof: As noted in the above, it suffices to prove that each $\tilde{Q}(f)$ is nonnegative and satisfies (4.9). By (4.6) it holds for the JOR method (4.11) that for $\omega\leq 1$, $\tilde{q}_{ii}(k) = 1-\omega\geq 0$ and nondiagonal elements $\tilde{q}_{ij}(k)\geq 0$, and for $\omega>0$,

(4.17) $$\| \tilde{Q}(f) \| = 1 - \omega + \omega \max_{i\in I, k\in K_i} \{(\alpha_i(k)-q_{ii}(k))/(1-q_{ii}(k))\}$$
$$= 1 - \omega \min_{i,k}\{(1-\alpha_i(k))/(1-q_{ii}(k))\} < 1.$$

Since the J method is identical to the JOR method with $\omega=1$, the J method and the JOR method with $0<\omega\leq 1$ give equivalent decision processes. For the SOR

method, the nonnegativity of $\tilde{q}_{11}(k)=1-\omega$ requires $\omega\leq 1$. From (4.13) it follows that for $\omega>0$,

$$(4.18)\quad \tilde{\alpha}_1(k) = \sum_{j\in I} \tilde{q}_{1j}(k) = 1 - \omega + \omega(\alpha_1(k)-q_{11}(k))/(1-q_{11}(k)) = 1-\omega(1-\alpha_1(k))/(1-q_{11}(k))<1.$$

Suppose that $\tilde{\alpha}_j(k)<1$ for all $j<i$. Then by (4.13),

$$(4.19)\quad \tilde{\alpha}_i(k) \leq 1-\omega + \omega(\alpha_i(k)-q_{ii}(k)) /(1-q_{ii}(k)) < 1$$

Since the SOR method with $\omega=1$ is reduced to the GS method, the theorem holds for the GS method and the SOR method with $0<\omega\leq 1$. For the GRF method (4.14),

$$\tilde{q}_{ii}(k) = 1-\omega_i + \omega_i q_{ii}(k) \text{ and } \| \tilde{Q}(f) \| = \max_{i,k}\{1-\omega_i + \omega_i\alpha_i(k)\},$$

which implies that for $0<\omega_i\leq\bar{\omega}_i$, $\tilde{q}_{ii}(k)\geq 0$, nondiagonal $\tilde{q}_{ij}(k)\geq 0$ and $\|\tilde{Q}(f)\| \leq 1$. The RF method is the GRF method with $\Omega=\omega I$, and the theorem is proved for these methods, too.

Porteus [17] has proposed several transformations including the J method and the GS method that can be used to convert the semi-Markov decision process into an equivalent one that may be easier to solve by the method of successive approximations with a suboptimality test. His definition of equivalence, however, imposes only Condition (i) of Definition 2. The J method has been dealt with by van Nunen [24], too. The SOR method has been discussed by Porteus and Totten [18] as a computational method of $v(f)$. In order to distinguish the original operator $T(f)$ associated with (I, F, Q, r) with the J method, it will be called the *pre-Jacobi method (PJ method)* in the sequel. Kushner and Kleinman [11] have proposed the method of successive approximations which uses, instead of the PJ method, the GRF method and the following methods:

(vii) *Pre-Gauss Seidel method (PGS method):*

$$(4.20)\quad \tilde{Q}(f) = (I-L(f))^{-1}(D(f) + U(f)),\ \tilde{r}(f) = (I-L(f))^{-1}r(f);$$

(viii) *Pre-successive overrelaxation method (PSOR method):*

$$(4.21)\quad \tilde{Q}(f) = (I-\omega L(f))^{-1}(\omega D(f) + \omega U(f) + (1-\omega)I),\quad \tilde{r}(f) = (I-\omega L(f))^{-1}r(f).$$

The method of successive approximations using the PSOR method has been discussed by Reetz [20]. He also has devised the method of successive approximations with suboptimality test which uses the PGS method instead of the PJ method [21]. The PGS method has been dealt with in [3, 24]. In addition to the above eight methods, a stationary ℓ-th degree method is stated in [28]:

(ix) *Stationary ℓ-the degree method:*

$$(4.22)\quad \tilde{T}(f) = T(f)^{\ell}.$$

Hinderer [9] has derived upper and lower bounds for v^* for the method of successive approximations using the stationary ℓ-th degree PJ method, and Hastings and van Nunen [8] have investigated a suboptimality test utilizing analogous bounds.

Theorem 7. Decision processes (I, F, $\tilde{Q}$, $\tilde{r}$) derived from the PGS method, the PSOR method with ω satisfying $0<\omega\leq\bar{\omega}$ and the stationary ℓ-th degree method are equivalent to (I, F, Q, r), where $\bar{\omega}$ is given by (4.16).

Proof: First let us prove the theorem for the PSOR method. Since $\bar{q}_{11}(k)=1-\omega+\omega q_{11}(k)$ and $\tilde{\alpha}_1(k)=1-\omega+\omega\alpha_1(k)$, the nonnegativity of $\tilde{q}_{11}(k)$ and (4.9) require $0<\omega\leq 1/(1-\bar{q}_1)$. Similarly, since $\tilde{q}_{ii}(k)\geq 1-\omega+\omega q_{ii}(k)$ for $\omega>0$ and $\tilde{\alpha}_i(k)=1$ for $\omega=0$, $\tilde{Q}(f)$ of the PSOR method satisfies the nonnegativity and (4.9) for $0<\omega\leq\bar{\omega}$. The PSOR method with $\omega=1$ is identical with the PGS method so that the theorem holds for the PGS method. It is clear that $\tilde{T}(f)=T(f)^{\ell}$ is monotone and $\|\tilde{T}(f)\|=\|T(f)\|^{\ell}=\beta^{\ell}<1$. Moreover, $\tilde{T}(f)v(f)=T(f)^{\ell}v(f)=v(f)$. Thus the proof is concluded.

It should be noted that from the proof, the stationary ℓ-th degree method based on any method in (i) through (viii) yields an equivalent decision process (I, F, $\tilde{Q}$, $\tilde{r}$).

5. Transformed Algorithms

In the preceding section, it has been shown that all methods given by (i) through (ix) convert the semi-Markov decision process (I, F, Q, r) to equivalent processes (I, F, $\tilde{Q}$, $\tilde{r}$) that can be solved by the basic algorithm in Section 3. Consequently, we have obtained ten modified policy iteration algorithms with the suboptimality test including the basic algorithm which uses the PJ method. These algorithms are all new.

Theorem 1 and the proof of Theorem 4 imply that the bounds for v^* become sharper and the convergence of $\{\tilde{v}^n\}$ becomes more rapid as the constants $\tilde{\beta}$ and $\tilde{\gamma}$ given by (2.7) for (I, F, $\tilde{Q}$, $\tilde{r}$) decrease. Thus the algorithms can be compared in this respect. First, consider (I, F, $\tilde{Q}$, $\tilde{r}$) derived from the SOR method. Since the operator (4.8) with $\tilde{Q}$ and $\tilde{r}$ given by (4.13) is equivalent to

$$\tilde{v}_i^n=(1-\omega)\tilde{v}_i^{n-1}+\omega\{\sum_{j<i}\hat{q}_{ij}(f_i)\tilde{v}_j^n+\sum_{j>i}\hat{q}_{ij}(f_i)\tilde{v}_j^{n-1}+\hat{r}_{if_i}\}\qquad(i\in I),\tag{5.1}$$

the constants $\tilde{\beta}(\omega)$ and $\tilde{\gamma}(\omega)$ for the SOR method can be computed by

$$\tilde{\beta}_i(\omega)=1-\omega+\omega\max_{k\in K_i}\{\sum_{j<i}\hat{q}_{ij}(k)\tilde{\beta}_j(\omega)+\sum_{j>i}\hat{q}_{ij}(k)\}\tag{5.2}$$

$$(5.3)\quad \tilde{\gamma}_i(\omega) = 1-\omega+\omega \min_{k\in K_i}\{\sum_{j<i}\hat{a}_{ij}(k)\tilde{\gamma}_j(\omega) + \sum_{j>i}\hat{q}_{ij}(k)\}$$

and

$$(5.4)\quad \tilde{\beta}(\omega) = \max_{i\in I}\tilde{\beta}_i(\omega) \quad \text{and} \quad \tilde{\gamma}(\omega) = \min_{i\in I}\tilde{\gamma}_i(\omega)\ .$$

By (4.6)

$$\tilde{\beta}_1(\omega) = 1-\omega \min_{k\in K_1}\{(1-\alpha_1(k))/(1-q_{11}(k))\}$$

and

$$\tilde{\gamma}_1(\omega) = 1-\omega \max_{k\in K_1}\{(1-\alpha_1(k))/(1-q_{11}(k))\}.$$

Therefore $\tilde{\beta}_1(\omega)$ and $\tilde{\gamma}_1(\omega)$ are strictly decreasing in ω. Suppose that $\tilde{\beta}_j(\omega)$ and $\tilde{\gamma}_j(\omega)$ are strictly decreasing in ω for all $j<i$. Then (5.2) implies that for $\delta>0$ and $\omega>0$,

$$\tilde{\beta}_i(\omega+\delta) - \tilde{\beta}_i(\omega) = -\delta+(\omega+\delta)\max_{k\in K_i}\{\sum_{j<i}\hat{q}_{ij}(k)\tilde{\beta}_j(\omega+\delta) + \sum_{j>i}\hat{q}_{ij}(k)\}$$

$$-\omega \max_{k\in K_i}\{\sum_{j<i}\hat{q}_{ij}(k)\tilde{\beta}_j(\omega) + \sum_{j>i}\hat{q}_{ij}(k)\}$$

$$\le -\delta\{1-[\sum_{j<i}\hat{q}_{ij}(k')\tilde{\beta}_j(\omega+\delta)+\sum_{j>i}\hat{q}_{ij}(k')]\}+\omega\{\sum_{j<i}\hat{q}_{ij}(k')(\tilde{\beta}_j(\omega+\delta)-\tilde{\beta}_j(\omega))\}$$

$$\le 0,$$

where $\tilde{\beta}_i(\omega+\delta)$ is attained at $k'\in K_i$. Similarly, (5.3) implies that

$$\tilde{\gamma}_i(\omega+\delta)-\tilde{\gamma}_i(\omega) \le -\delta\{1-[\sum_{j<i}\hat{q}_{ij}(k'')\tilde{\gamma}_j(\omega+\delta) + \sum_{j>i}\hat{q}_{ij}(k'')]\}$$

$$+\omega\{\sum_{j<i}\hat{q}_{ij}(k'')(\tilde{\gamma}_j(\omega+\delta)-\tilde{\gamma}_i(\omega))\} \le 0,$$

where $k''\in K_i$ attains $\tilde{\gamma}_i(\omega)$. Consequently, $\tilde{\beta}_i(\omega)$ and $\tilde{\gamma}_i(\omega)$ are strictly decreasing in ω for all $i\in I$, which implies that $\tilde{\beta}(\omega)$ and $\tilde{\gamma}(\omega)$ are also strictly decreasing in ω. In a way similar to the above, it can be proved that $\tilde{\beta}(\omega)$ and $\tilde{\gamma}(\omega)$ for the JOR, GRF, RF and PSOR methods are all strictly decreasing in ω. Summarizing the above results, we have the following theorem:

Theorem 8. The constants $\tilde{\beta}$ and $\tilde{\gamma}$ for decision processes $(I, F, \tilde{Q}, \tilde{r})$ derived from the JOR, SOR, GRF, RF and PSOR methods are strictly decreasing in $\omega>0$.

From Theorems 6 and 8 it follows that the relaxation factor ω of the SOR method should be one. In this case, the SOR method is identical to the GS method. Thus the SOR method can be ignored in the following discussion. Similarly, the JOR method can be also ignored. If $\bar{q}_i = 0$ for all $i\in I$, then $\bar{\omega}_i=1$ for all i, and if $\bar{q}=0$, then $\bar{\omega}=1$. Hence, if $\bar{q}_i=0$ for all $i\in I$, then the GRF and RF methods are reduced to the PJ method, and if $\bar{q}=0$, then the RF

method is reduced to the PJ method. Moreover, Theorem 7 implies that the PGS method can be ignored in the following discussion, because $\bar{\omega}\geq 1$. However, only when $\bar{q}=0$, the PSOR method is reduced to the PGS method

As an example of transformed algorithms derived from equivalent decision processes $(I, F, \tilde{Q}, \tilde{r})$, consider the algorithm based on the GS method.

Gauss-Seidel Algorithm:

Step 1: Compute $\hat{p}_{ij}(k)$ and $\hat{r}_{ik}$ by (4.6) and (4.7) for all i, $j\in I$ and $k\in K_i$. Calculate c by (3.43) with r_{ik} replaced by $\hat{r}_{ik}$ and set $\tilde{v}^0=ce$. Choose a non-negative integer m and a positive constant ε, and set $K_i^0=K_i$ $(i\in I)$, $u^0=Me$, $\eta_0=0$ and $n=0$. Compute $\tilde{\beta}_i$ and $\tilde{\gamma}_i$ for $i=1,\ldots,N$ by

$$(5.5)\quad \tilde{\beta}_i = \max_{k\in K_i^n} \{ \sum_{j<i} \hat{q}_{ij}(k)\tilde{\beta}_j + \sum_{j>i} \hat{q}_{ij}(k) \}$$

and

$$(5.6)\quad \tilde{\gamma}_i = \min_{k\in K_i^n} \{ \sum_{j<i} \hat{q}_{ij}(k)\tilde{\gamma}_j + \sum_{j>i} \hat{q}_{ij}(k) \}$$

and set

$$(5.7)\quad \tilde{\beta} = \max_{i\in I} \tilde{\beta}_i \quad \text{and} \quad \tilde{\gamma} = \min_{i\in I} \tilde{\gamma}_i .$$

Store $i(\beta)$, $i(\gamma)$, $k(\beta)$ and $k(\gamma)$ such that $\tilde{\beta}=\tilde{\beta}_{i(\beta)}$, $\tilde{\gamma}=\tilde{\gamma}_{i(\gamma)}$ and $\tilde{\beta}_i$ and $\tilde{\gamma}_i$ are attained at $k=k(\beta)$ and $k=k(\gamma)$, respectively. Set $n=1$.

Step 2: For each $i\in I$, compute

$$(5.8)\quad \tilde{w}_i^n = \min_{k\in K_i^{n-1}} \{\hat{r}_{ik} + \sum_{j<i} \hat{q}_{ij}(k)\tilde{w}_j^n + \sum_{j>i} \hat{q}_{ij}(k)\tilde{v}_j^{n-1}\}$$

and determine at the same time

$$(5.9)\quad K_i^n = \{k\in K_i^{n-1};\ \hat{r}_{ik} + \sum_{j<i} \hat{q}_{ij}(k)\tilde{w}_j^n + \sum_{j>i} \hat{q}_{ij}(k)\tilde{v}_j^{n-1} \leq u_i^{n-1} - \min\{\tilde{\beta}_i\eta_{n-1},\ \tilde{\gamma}_i\eta_{n-1}\}\}.$$

If $\tilde{f}_i^{n-1}$ attains the minimum value $\tilde{w}_i^n$, then set $\tilde{f}_i^n$ as $\tilde{f}_i^{n-1}$; else, set $\tilde{f}_i^n$ as any action which attains $\tilde{w}_i^n$. If $k(\tilde{\beta})\notin K_{i(\tilde{\beta})}^n$ or $k(\tilde{\gamma})\notin K_{i(\tilde{\gamma})}^n$, then revise $\tilde{\beta}_i$, $\tilde{\gamma}_i$, $\tilde{\beta}$ and $\tilde{\gamma}$ by (5.5) through (5.7). If K_i^n consists of a single action $\tilde{f}_i^n$ for all $i\in I$, go to Step 5.

Step 3: For $\ell=0,\ldots,m-1$, compute $y_i^{\ell+1}$ $(i\in I)$ by

$$(5.10)\quad y_i^{\ell+1} = \hat{r}_{if_i^n} + \sum_{j<i} \hat{q}_{ij}(f_i^n)y_i^{\ell+1} + \sum_{j>i} \hat{q}_{ij}(f_i^n)y_j^{\ell},$$

where $y^0=\tilde{w}^n$. Set $\tilde{v}^n=y^m$.

Step 4: Calculate Δ_n, ∇_n, a_n and b_n by (3.5) and (3.6) with v^n and w^n replaced by $\tilde{v}^n$ and $\tilde{w}^n$, and ξ_n and η_n by (3.18) and (3.19) with β and γ replaced by $\tilde{\beta}$ and $\tilde{\gamma}$. If (3.25) holds, then compute u^n by (3.8), set $n=n+1$ and go back to Step 2. Otherwise, v given by (3.26) with v^n replaced by $\tilde{v}^n$ is an ε-

approximate value of v^*. Calculate δ_n and $\tilde{\varepsilon}_n$ by

$$(5.11)\quad \delta_n = \min_{i\in I}\{v_i - \hat{r}_{i\tilde{f}_i^n} - \sum_{j\in I}\hat{q}_{ij}(\tilde{f}_i^n)v_j\}$$

and

$$(5.12)\quad \tilde{\varepsilon}_n = \varepsilon - [\delta_n/(1-\tilde{\gamma})], \quad \text{if} \quad \delta_n \geq 0;$$
$$= \varepsilon - [\delta_n/(1-\tilde{\beta})], \quad \text{otherwise.}$$

The policy $\tilde{f}^n$ is $\tilde{\varepsilon}_n$-optimal.

Step 5: $\tilde{f}^n$ is the unique optimal solution. Compute c_n and d_n by (3.7) with v^{n-1} and w^n replaced by $\tilde{v}^{n-1}$ and $\tilde{w}^n$. The minimum value v^* is estimated by

$$(5.13)\quad v = \tilde{w}^n + [\{\tilde{\gamma}c_n/(1-\tilde{\gamma}) + \tilde{\beta}d_n/(1-\tilde{\beta})\}/2]e.$$

Transformed algorithms based on the J, **GRF**, **RF** and **PSOR** methods can be described in the same way as the GS algorithm. In order to derive the transformed algorithm based on the stationary ℓ-th degree method, it is necessary to compute $Q(f)^\ell$ for each $f\in F$, which requires much computational effort. That is, the transformed algorithm based on the stationary ℓ-th degree method is not efficient. In the case of $m=0$ which corresponds to the method of successive approximations, however, it is possible to devise the transformed algorithm based on the stationary ℓ-th degree method which is faster than the method of successive approximations with the Hastings and Mello test [15].

6. A Numerical Example

The most popular test problem in discounted Markov decision processes is Howard's automobile replacement problem [10, p.89]. The state space I and the action sets $K_i (i\in I)$ for the problem are $I=\{0,1,\ldots,40\}$ and $K_i=K=\{1,2,\ldots,41\}$; the discounted transition probabilities $q_{ij}(k)$ are $\beta P_{ij}(k)$, where $\beta = 0.97$ and for $i, j\in I$,

$$(6.1)\quad P_{ij}(1) = p_i (j=i+1),\ =1-p_i\ (j=40),\ =0\ (j\neq i+1 \text{ or } 40)$$

and for $k\geq 2$,

$$(6.2)\quad P_{ij}(k) = p_{k-2}\ (j=k-1),\ =1-p_{k-2}\ (j=40),\ =0\ (j\neq k-1 \text{ or } 40).$$

The values of p_i and r_{ik} $(i\in I,\ k\in K)$ are given in [10, p.56].

First, the problem was solved by the basic algorithm (PJ algorithm) and van Nunen's algorithm [23] with m varying from one to twenty. The programs were written in FORTRAN and designed for general situations. Hence they did not exploit the sparsity of the discounted transition matrix. The value of ε was 0.1 and all iterations terminated at Step 5 giving the unique optimal policy. The minimum value v^* accurate to at least five decimal places was obtained by the basic algorithm. The problem was also solved by the policy ite-

ration method and linear programming in order to compare these conventional methods with the basic algorithm. The computational results are given in Table 1. All computations were carried out with double precision (except linear programming) on a FACOM M-200 computer of Data Processing Center, Kyoto University. Table 1 verifies that the basic algorithm is faster than van Nunen's one, as noted in Remark 2. Moreover, it indicates that the basic algorithm with properly chosen values of m is somewhat more efficient than the policy iteration method even for this small problem. This suggests that the basic algorithm, which was originally devised for solving semi-Markov decision processes with one thousand or more states, will be superior to the policy iteration method even when the number of states is on the order of hundreds.

Table 1

Comparison between the Basic Algorithm and the Known Algorithms

	basic algorithm (PJ algorithm)		Van Nunen's algorithm [23]	
	computation time (milliseconds)	number of iterations	computation time (milliseconds)	number of iterations
m = 1	1316	40	3054	38
2	1039	30	2461	27
3	1038	26	2438	24
4	911	21	2206	20
5	942	19	2140	17
6	930	18	1997	16
7	853	15	1864	15
8	851	15	1849	14
9	957	16	2133	15
10	913	14	2041	13
11	906	13	2005	12
12	864	13	1889	12
13	909	12	1967	11
14	822	11	1776	10
15	845	11	1793	10
16	817	10	1830	10
17	912	11	2036	11
18	845	10	1831	10
19	867	10	1847	10
20	854	10	1889	10
	policy iteration method		linear programming	
	912	6	8171	——

Next the problem was solved by transformed algorithms devised in Sections 4 and 5. From (6.1) and (6.2) it follows that $q_{ii}(i+1)=p_{i-1}$ for $i\geq 1$ and $q_{ii}(k)=0$ for other i and k, which implies that $\bar{q}_i=\bar{q}=0$ and $\bar{\omega}_i=\omega=1$ for all i. Therefore, as noted in the preceding section, the GRF and RF algorithms are reduced to the basic algorithm (PJ algorithm). Thus the transformed algorihms applicable to the problem are the PSOR, J and GS algorithms. The program was

designed for comparing the basic algorithm with the transformed algorithms for m varying from zero to fifty. The value of ϵ was 0.1 and the other details were also the same as noted above. Table 2 summarizes computational results by

Table 2

Comparison of Algorithms

(Computation Times in Milliseconds/Number of Iterations)

	PJ	PSOR	J	GS
m = 0	2513/75	9624/234	10047/255	9853/235
1	1649/40	5345/119	6031/128	5702/120
2	1305/30	3883/81	4500/87	4380/82
3	1347/26	3116/62	3722/66	3643/63
4	1208/21	2680/50	3197/53	3190/51
5	1194/19	2338/43	2868/45	2812/43
6	1197/18	2152/37	2657/39	2615/38
7	1090/15	1988/33	2398/34	2413/34
8	1086/15	1882/30	2232/31	2279/31
9	1242/16	1837/28	2171/28	2165/28
10	1183/14	1729/26	1994/26	2101/26
11	1184/13	1670/24	1942/24	1907/24
12	1117/13	1647/22	1858/22	1977/23
13	1151/12	1647/21	1810/21	1900/21
14	1058/11	1635/20	1760/20	1869/20
15	1081/11	1621/19	1725/19	1853/19
16	1072/10	1588/18	1820/18	1823/18
17	1177/11	1558/17	1758/17	1828/17
18	1097/10	1578/17	1718/16	1828/17
19	1115/10	1574/16	1742/16	1818/16
20	1134/10	1509/15	1686/15	1864/16
25	1175/10	1514/13	1681/13	1788/13
30	1268/10	1460/12	1633/11	1804/12
35	1310/10	1523/11	1627/10	1812/11
40	1389/10	1543/11	1618/9	1792/10
45	1467/10	1592/10	1690/9	1814/9
50	1520/10	1680/10	1787/9	1836/9

the PJ, PSOR, J and GS algorithms. The algorithms for m=0 correspond to the method of successive approximations with the suboptimality test. However, the program comprised several subroutines and many IF sentences so that the computation times were considerably longer than by programs coded separately for PJ, PSOR, J and GS algorithms. Moreover, the revision of $\tilde{\beta}$ and $\tilde{\gamma}$ in Step 2 of the transformed algorithms consumed much time. Thus the program was modified to omit the routine of the revision. Table 3 shows the computational results by the basic algorithm and the transformed algorithms without the revision of $\tilde{\beta}$ and $\tilde{\gamma}$. Tables 2 and 3 indicate that the basic algorithm (PJ algorithm) is superior to the PSOR, J and GS algorithms. The reason may be that the bounds for v^* in the basic algorithm are tighter than those in the transformed algorithms, although $\tilde{v}^n$ in the transformed algorithms is closer to v^* than v^n in the basic algorithm. Moreover, those tables show an irregualr

behavior of the computation times of the basic and the transformed algorithms as m varies. Therefore it will not be easy to determine in advance an optimal value of m that minimizes computation time. Puterman and Shin [19], for example, have concluded that it is not reasonable to consider the modified policy iteration algorithm with m greater than $(N/3)+1$. Their conclusion, however, is not true for the basic and the transformed algorithms, because the minimum computation times for these algorithms are attained at m greater than 14, as shown in Tables 1 and 3.

Table 3

Comparison of Algorithms Omitting Revision of $\tilde{\beta}$ and $\tilde{\gamma}$
(Computation Times in Milliseconds/Number of Iterations)

	PJ	PSOR	J	GS
m = 0	2470/75	9779/252	9892/277	9403/253
1	1641/40	5250/127	5244/139	5101/128
2	1318/30	3796/86	3769/94	3678/87
3	1330/26	3040/65	2948/71	2942/66
4	1171/21	2575/53	2441/57	2564/54
5	1155/19	2303/45	2141/48	2244/45
6	1159/18	2075/39	1943/42	2066/40
7	1057/15	1938/35	1779/37	1885/35
8	1081/15	1785/31	1658/33	1749/32
9	1239/16	1713/29	1585/30	1663/29
10	1190/14	1652/26	1492/27	1598/27
11	1169/13	1550/24	1440/25	1545/25
12	1130/13	1569/23	1419/24	1516/23
13	1170/12	1565/21	1397/22	1431/22
14	1059/11	1525/20	1314/21	1436/21
15	1104/11	1500/19	1298/20	1399/19
16	1120/10	1485/18	1362/19	1447/19
17	1205/11	1502/18	1275/18	1414/18
18	1118/10	1479/17	1268/17	1397/17
19	1120/10	1438/16	1256/16	1391/16
20	1158/10	1470/16	1277/16	1424/16
25	1204/10	1481/14	1244/13	1388/13
30	1272/10	1421/12	1218/11	1400/12
35	1342/10	1448/11	1220/10	1388/11
40	1394/10	1499/11	1273/9	1338/10
45	1459/10	1579/10	1346/9	1357/9
50	1508/10	1619/10	1379/9	1438/9

As noted in the above, the computational results indicate that the basic algorithm is far superior to van Nunen's algorithm and is more efficient than the policy iteration method, linear programming, or indeed the PSOR, J and GS algorithms for the automobile replacement problem. A problem remains with the discounted Markov decision process with a unique optimal policy; the GRF and RF algorithms are reduced to the basic algorithm, and the PGS algorithm is identical to the PSOR algorithm. Consequently, many other numerical comparisons for real discounted semi-Markov decision processes must be made to

conclude which algorithm is best among the basic and the transformed algorithms.

7. Concluding Remarks

This paper proposes the modified policy iteration algorithm with the suboptimality test of Hastings and Mello type, which is called the basic algorithm, and proves without assuming any conditions that it constructs a finite sequence of policies whose last element is a unique optimal or an ε_n-optimal one. Equivalent decision processes are defined in Definition 2, and many iterative methods for solving a system of linear equations are shown to convert the original semi-Markov decision process to equivalent decision processes. Various transformed algorithms such as the PGS, PSOR, J,GS,GRF and RF algorithms are derived from the basic algorithm applied to equivalent decision processes. It is shown that all these transformed algorithms have the same convergence property as the basic algorithm. Condition (i) of Definition 2 is essential for this to hold. Since the column reduction proposed in [17] and discussed in [26] satisfies Condition (i),its combination with the basic algorithm will provide a new transformed algorithm with the convergence property.

The numerical comparisons in Section 6 show that the basic algorithm is the most efficient among the policy iteration method, linear programming, van Nunen's algorithm and the PJ, PSOR, J and GS algorithms for Howard's automobile replacement problem. Since the programs did not exploit the sparsity of the discounted transition matrix of the problem, the basic algorithm exploiting it will solve the problem with much shorter computation times than those shown in Table 1. The problem, however, has the discounted transition matrix with the special structure and the unique optimal policy. Some of the transformed algorithms may be superior to the basic algorithm for problems with several optimal policies. Therefore, many other numerical comparisons should be made to conclude which algorithm is most efficient among the basic and the transformed algorithms. In such comparisons, the ordering of the states for the transformed algorithms [12, p.354] may be taken into consideration. It is hoped that the proposed algorithms will be applied to real semi-Markov decision processes such as controlled queueing systems, inventory control processes and stochastic control processes.

The author would like to express his appreciation to Prof. H. Mine for his encouragement and to Mr. T. Shitomi for his assistance in preparing the computation results of Table 1. The author is also indebted to the anonymous

reviewers for helpful comments on the earlier drafts of this paper.

References

[1] Bellman, R: *Dynamic Programming*, Princeton Univ. Press, Princeton, 1957.

[2] Bertsekas, D.P.: On Error Bounds for Successive Approximation Methods, *IEEE Transactions on Automatic Control*, Vol.AC-21 (1976), 394-396.

[3] Bertsekas, D.P.: *Dynamic Programming and Stochastic Control*, Academic Press, New York, 1976.

[4] Blackwell, D.: Discrete Dynamic Programming, *Annals of Mathematical Statistics*, Vol.33 (1962), 719-726.

[5] Denardo, E.: Contraction Mappings in the Theory Underlying Dynamic Programming, *SIAM Review*, Vol.9 (1967), 165-177.

[6] Derman, C.: *Finite State Markovian Decision Processes*, Academic Press, New York, 1970.

[7] Hastings, N.A.J., and Mello, J.M.C.: Tests for Suboptimal Actions in Discounted Markov Programming, *Management Sciences*, Vol.19 (1973), 1019-1022.

[8] Hastings, N.A.J., and van Neunen, J.A.E.E.: The Action Elimination Algorighm for Markov Decision Processes. *Markov Decision Theory* (ed. H.C. Tijms and J. Wessels). Mathematical Center Tracts 93, Amsterdam, 1979, 161-170.

[9] Hinderer, K.: Estimates for Finite-Stage Dynamic Programs, *Journal of Mathematical Analysis and Applications*, vol.55 (1976), 207-238.

[10] Howard, R.: *Dynamic Programming and Markov Processes*, The M.I.T. Press, *Cambridge*, 1960.

[11] Kushner, H.J., and Kleinman, A.J.: Accelerated Procedures for the Solution of Discrete Markov Control Problems, *IEEE Transactions on Automatic Control*, Vol.AC-16 (1971), 147-152.

[12] Kushner, H.: *Introduction to Stochastic Control*, Holt, Rinehart and Winston, Inc., New York, 1971.

[13] MacQueen, J.: A Modified Dynamic Programming Method for Markovian Decision Problems, *Journal of Mathematical Analysis and Applications*, Vol.14 (1966), 38-43.

[14] MacQueen, J.: A Test for Suboptimal Actions in Markovian Decision Problems, *Operations Research*, Vol.15 (1967), 559-561.

[15] Ohno, K.: Computational Algorithms in Markov Decision Processes, *Preprints of 11th JAACE Symposium on Stochastic Systems*, pp.9-12, 1979 (in Japanese).

[16] Porteus, E.L.: Some Bounds for Discounted Sequential Decision Processes, *Management Sciences*, Vol.18 (1971), 7-11.

[17] Porteus, E.L.: Bounds and Transformations for Discounted Finite Markov Decision Chains, *Operations Research*, Vol.23 (1975), 761-784.

[18] Porteus, E.L., and Totten, J.C.: Accelerated Computation of the Expected Discounted Return in a Markov Chain, *Operations Research*, Vol.26 (1978), 350-358.

[19] Puterman, M.L., and Shin, M.C.: Modified Policy Iteration Algorithms for Discounted Markov Decision Problems, *Management Sciences*, Vol.24 (1978), 1127-1137.

[20] Reets, D.: Solution of a Markovian Decision Problem by Successive Overrelaxation, *Zeitschrift für Operations Research*, Vol.17 (1973), 29-32.

[21] Reets, D.: A Decision Exclusion Algorithm for a Class of Markovian Decision Processes, *Zeitschrift für Operations Research*, Vol.20 (1976), 125-131.

[22] Ross, S.M.: *Applied Probability Models with Optimization Applications*, Holden-Day, San Francisco, 1970.

[23] Van Nunen, L.A.E.E.: A Set of Successive Approximation Methods for Discounted Markovian Decision Problems, *Zeitschrift für Operations Research*, Vol.20 (1976), 203-208.

[24] Van Nunen, J.A.E.E.: *Contracting Markov Decision Processes*, Mathematical Center Tracts 71, Amsterdam, 1976.

[25] Veinott, Jr., A.F.: Discrete Dynamic Programming with Sensitivity Discount Optimality Criteria, *Annals of Mathematical Statistics*, Vol.40 (1969), 1635-1660.

[26] Vickson, R.G.: Generalized Value Bounds and Column Reduction in Finite Markov Decision Problems, *Operations Research*, Vol.28 (1980), 387-394.

[27] White, D.J.: Elimination of Non-optimal Actions in Markov Decision Processes. *Dynamic Programming and its Applications* (ed. M.L. Puterman). Academic Press, New York, 1978, 131-160.

[28] Young, D.M.: *Iterative Solution of Large Linear Systems*, Academic Press, New York, 1971.

Katsuhisa OHNO: Department of Applied Mathematics and Physics, Kyoto University, Yoshidahonmachi, Sakyoku, Kyoto, 606, Japan.