Operations Research

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Technical Note—Elimination of Suboptimal Actions in Markov Decision Problems

Richard C. Grinold,

Elimination of Suboptimal Actions in Markov Decision Problems

Richard C. Grinold

University of California, Berkeley, California

(Received July 5, 1972)

This note points out that upper and lower bounds on the optimal value function of a finite discounted Markov decision problem can be computed easily when the problem is solved by linear programming or policy iteration. These bounds can be used to identify suboptimal actions.

THIS NOTE follows MacQueen[3, 4] in showing how suboptimal decisions can be eliminated in finite-state, finite-action discounted Markov decision problems. MacQueen[4] demonstrated that suboptimal actions can be identified if lower and upper bounds on the optimal value function are available. MacQueen's value-iteration scheme[3] develops upper and lower bounds, and therefore can be used to reduce the problem's size as the computation proceeds. Other bounds and algorithms that exploit them can be found in Porteus[5] and Totten.[7]

This note points out that upper and lower bounds are readily available when a Markov decision problem is being solved by policy iteration or linear programming. Moreover, very little extra calculation is needed to identify the suboptimal decisions. In what follows we state the result, discuss its implementation, present computational evidence, and finally prove the result. The proof is a straightforward exercise using the theory developed by MacQueen.[3, 4]

The problem is to maximize expected discounted reward and the notation is that of Howard.[2] In particular, p_{ij}^k is the probability that the system moves from state i to state j if action k is employed. We assume $\sum_j p_{ij}^k = 1$.

Let A be our current policy and suppose v^A is the value (Howard,[2] pp. 84–86) of policy A. For each state i and action k possible in state i, we define γ_i^k by

$$\gamma_i^k = q_i^k + \beta \sum_j p_{ij}^k v_j^A - v_i^A, \tag{1}$$

and let $\gamma^* = \max_{k,i} \gamma_i^k$.

Result. *Action k in state i is suboptimal if* $(\beta-1)^{-1}(\beta)\gamma^* > \gamma_i^k$.

DISCUSSION

In general, $\gamma^* \geqq 0$. If $\gamma^* = 0$, the current policy is optimal. Since $\beta < 1$, we have $0 > (\beta-1)^{-1}\beta\gamma^*$. As $\beta \to 1$, the bound becomes worse.

If the problem is being solved by linear programming, then the numbers γ_i^k are the reduced profit coefficients. If, in addition, the linear programming code selects as the incoming column the one with the maximum reduced profit coefficient, then γ^* will be calculated also. Therefore, in a problem with M states and a total

TABLE I

PROPORTION OF ACTIONS ELIMINATED AS A FUNCTION OF THE ITERATION COUNT AND THE DISCOUNT FACTOR

Number of iterations	$\beta = 0.8333$	$\beta = 0.86956$	$\beta = 0.909$	$\beta = 0.9532$
1	0	0	0	0
2	0.06	0	0	0
3	0.26	0.10	0	0
4	opt	0.45	0.08	0
5		opt	0.16	0
6			0.37	opt
7			opt	

of N possible actions, the result can be used with only the N comparisons $(\beta-1)^{-1}\beta\gamma^*>\gamma_i^k$.

If the problem is being solved by policy iteration, then the numbers $q_i^k+\beta\sum_j p_{ij}^k v_j^A$ and $\max_k [q_i^k+\beta \sum_j p_{ij}^k v_j^A]$ are already calculated. The γ_i^k are obtained with N extra additions. We can then find

$$\gamma^*=\max_i \{-v_i^A+\max_k [q_i^k+\beta \textstyle\sum_j p_{ij}^k v_j^A]\}$$

with M more additions and comparisons. In total, $(N+M)$ additions and comparisons are needed to employ the result.

As a computational check, we solved Howard's auto-replacement problem [reference 2, p. 90]. The problem has $M=40$ states and $N=1640$ possible actions. Table I shows the proportion of actions eliminated as a function of the iteration count and the discount factor. The same starting solution was used in each case. Notice the bound is more effective for low discount factors and it is more effective near the optimal solution.

Frequently, one wishes to solve problems for a variety of discount factors, say $(\beta_1, \beta_2, \beta_3, \beta_4)$, where $\beta_1<\beta_{i+1}$, to check the sensitivity of the optimal policy to the discount factor. If the bound is being employed, it is best to solve first for β_1, then use the optimal policy in that problem as a start in the β_2 problem. When this technique is used in Howard's problem the results are the ones shown in Table II.

PROOF OF THE RESULT

LET v_i^* be the optimal-value function

$$v_i^*=\max_k [q_i^k+\beta \textstyle\sum_j p_{ij}^k v_j^*].$$

TABLE II

RESULTS FOR HOWARD'S PROBLEM

Number of Iterations	$\beta = 0.8333$	$\beta = 0.86956$	$\beta = 0.909$	$\beta = 0.9532$
1	0	0.31	0.16	0
2	0.06	0.89	0.37	0.86
3	0.26	opt	opt	opt
4	opt			

For any policy A we have $v_i^A \leqq v_i^*$ for all i. BLACKWELL [reference 1, Theorem 6, p. 232, and MacQueen [reference 3, Theorem 1, p. 40] have shown that any v satisfying

$$v_i \geqq \max_k [q_i^k + \beta \sum p_{ij}^k v_j], \text{ for all } i, \tag{2}$$

is an upper bound on v^*.

With the aid of (1), the reader can verify that $v_i = v_i^A + (1-\beta)^{-1}\gamma^*$ satisfies (2); therefore, $v_i^A \leqq v_i^* \leqq v_i^A + (1-\beta)^{-1}\gamma^*$ for all i.

Again, following MacQueen [reference 4, page 559], action k in state i is suboptimal if $q_i^k + \beta \sum_j p_{ij}^k v_j^* < v_i^*$. The upper bound on v_i^* and (1) imply

$$q_i^k + \beta \sum_j p_{ij}^k v_j^* \leqq q_i^k + \beta \sum p_{ij}^k [v_i^A + (1-\beta)^{-1}\gamma^*] = v_i^A + \gamma_i^k + (1-\beta)^{-1}\beta\gamma^*.$$

The lower bound on v_i^* and the condition $\gamma_i^k + (1-\beta)^{-1}\beta^{\gamma^*} < 0$ then imply

$$q_i^k + \beta \sum p_{ij}^k v_j^* \leqq v_i^A + \gamma_i^k + (1-\beta)^{-1}\beta\gamma^* < v_i^*.$$

Thus if, $(\beta-1)^{-1}\beta\gamma^* > \gamma_i^k$, action k in state i is suboptimal.

The result can also be established in several other ways. In the original version of this note, it was based on a linear-programming formulation of the Markov decision problem. It also follows from Theorem 2 of Porteus,[5] when the correct values of a and b are identified.

In addition, Porteus[6] has pointed out that

$$\gamma_i^k < \gamma_i + (\gamma_* - \gamma^*)\beta(1-\beta)^{-1} \tag{3}$$

implies that action k is nonoptimal in state i. Here $\gamma_i = \max_k \gamma_i^k$, and $\gamma_* = \min_i \gamma_i$. This test can be proved in the manner followed above. Using formula (7.14) on p. 87 of reference 2, we can show that $v_i^A \geqq v_i^A + \gamma_i + \beta\gamma_* \geqq v_i^*$. Unfortunately, this stronger test was not any more effective when used on the automobile-replacement problem. The proportion of actions eliminated using (3) was never more than 1 per cent greater than the proportion eliminated by using the simpler rule.

ACKNOWLEDGMENT

ERIC DENARDO, JAMES MACQUEEN, EVAN PORTEUS, and two referees were helpful on several points.

REFERENCES

1. D. BLACKWELL, "Discounted Dynamic Programming," *Ann. Math. Stat.* **36,** 226–235 (1965).
2. R. A. HOWARD, *Dynamic Programming and Markov Processes,* The Massachusetts Institute of Technology Press, Cambridge, Mass., 1960.
3. J. B. MACQUEEN, "A Modified Dynamic Programming Method for Markovian Decision Problems," *JMAA* **14,** 38–43(1966).
4. ———, "A Test for Suboptimal Actions in Markovian Decision Problems," *Opns. Res.* **15,** 559–561(1967).

5. E. L. PORTEUS, "Some Bounds for Discounted Sequential Decision Processes," *Management Sci.* **18**, 7–11 (1971).
6. ———, private communication.
7. J. C. TOTTEN, "Computational Methods for Finite State Finite Valued Markovian Decision Problems," Report 71–9, Operations Research Center, University of California, Berkeley, 1971.

Published in Operations Research on June 01, 1973 as DOI: 10.1287/opre.21.3.848.
This article has not been copyedited or formatted. The final version may differ from this version.