Operations Research

Letter to the Editor—An Approach to Reducing the Computing Time for Dynamic Programming

Peter J. Wong,

tion for z. Substituting (4) into (5) and employing the known result that $E[z]=1/P_k$ give the desired result

$$E(T)=\tau_a+\tau_i-\tau_h+(\tau_h+\tau_f)/P_k+[(\tau_m+\tau_f)/p][(1-u)/P_k+u-P_1] \qquad (6)$$

The reciprocal of (6) is the Lanchester attrition rate as defined by Barfoot for the case of a single P_k.

REFERENCES

1. SETH BONDER, "The Lanchester Attrition-Rate Coefficient," *Opns. Res.* **15**, 221–32 (1967).
2. C. B. BARFOOT, "The Lanchester Attrition-Rate Coefficient: Some Comments on Seth Bonder's Paper and a Suggested Alternative Method," *Opns. Res.* **17**, 888–894 (1969).

AN APPROACH TO REDUCING THE COMPUTING TIME FOR DYNAMIC PROGRAMMING

Peter J. Wong

Stanford Research Institute, Menlo Park, California

(Received August 1968)

This note presents a procedure for reducing the computing time for dynamic programming when the optimal control function can be assumed to be piecewise smooth.

IN THE standard dynamic programming procedure, one must search the entire admissible control set $U[\boldsymbol{x}, k]$ to find the optimal control $\boldsymbol{u}^0(\boldsymbol{x}, k)$ at state $\boldsymbol{x}$ and stage k. Ideally, one would like a method that would restrict the search within the control set $U[\boldsymbol{x}, k]$ to some smaller subset based on the local nature of the optimal control function $\boldsymbol{u}^0(\ , k)$. DREYFUS[1] has suggested that the search for $\boldsymbol{u}^0(\boldsymbol{x}, k)$ be restricted about previously computed values $\boldsymbol{u}^0(\boldsymbol{x}+\Delta, k)$ or $\boldsymbol{u}^0(\boldsymbol{x}, k+1)$. It can be seen that this procedure will not adequately handle rapid changes in the optimal control function $\boldsymbol{u}^0(\ , k)$, and it allows errors in calculating $\boldsymbol{u}^0(\boldsymbol{x}, k)$ to propagate both in the state variable $\boldsymbol{x}$ and the stage variable k.

The procedure presented in this paper is similar to the procedure just discussed, except the computations proceed in two stages. The first stage uses a coarse lattice structure in an attempt to determine the magnitude of local variations in the optimal control function $\boldsymbol{u}^0(\ , k)$. The second stage calculates the optimal control at points between the coarse lattice points by using information from the first stage to limit the region of search within the control set $U[\boldsymbol{x}, k]$. Although errors in calculating $\boldsymbol{u}^0(\boldsymbol{x}, k)$ propagate in the stage variable k, the procedure does not allow the errors to propagate in the state variable $\boldsymbol{x}$.

THE PROCEDURE

THE PROCEDURE will be described for the case $x \epsilon E^2$ and $u(x, k) \epsilon E^1$, the extension to the general case $x \epsilon E^n$ and $u(x, k) \epsilon E^p$ being straightforward. It will be assumed that the state space has been adequately quantized into a uniform rectangular lattice structure. Define the optimal control $u^0(x, k)$ as *fully calculated* if the entire admissible control set $U[x, k]$ is searched to compute $u^0(x, k)$, and *partially calculated* if $U[x, k]$ is partially searched.

Let S be the set of points in the lattice structure. Define S^* to be the subset of points in S that are in the uniform rectangular lattice structure with $2\Delta x_i$ as the

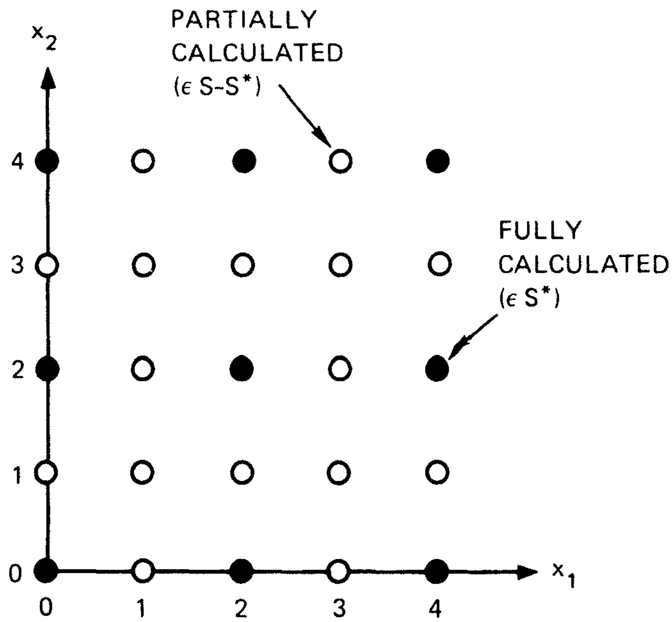

Fig. 1. Lattice structures for S and S^*.

quantization increment along the ith coordinate (*see* Fig. 1). Calculate $u^0(x^*, k)$ for each $x^* \epsilon S^*$ by searching over the entire admissible control set $U[x^*, k]$. Consequently, for each $x^* \epsilon S^*$, $u^0(x^*, k)$ is fully calculated. It remains to calculate $u^0(x, k)$ for each $x \epsilon S\text{-}S^*$. However, rather than search over the entire admissible control set $U[x, k]$ for each $x \epsilon S\text{-}S^*$, we will search $U[x, k]$ partially, with the region of search determined by values of $u^0(x^*, k)$ at neighboring lattice points, $x^* \epsilon S^*$.

Referring to Fig. 1 for any lattice point $x \epsilon S\text{-}S^*$, there is at least one pair of points α^*, $\beta^* \epsilon S^*$ such that (i) x is at the midpoint of the line joining α^* and β^*, and (ii) α^* and β^* lie on the rectangle of side $2\Delta x_i$ centered about x. A pair of points α^*, β^* satisfying these properties will be called a *bounding pair* for x. From Fig. 1, we see that the point (0, 1) has the bounding pair (0, 0), (0, 2), and the point (1, 1) has the bounding pairs (0, 0), (2, 2) and (0, 2), (2, 0). Now to calculate $\boldsymbol{u}^0(\boldsymbol{x}, k)$ for $\boldsymbol{x}\epsilon S\text{-}S^*$, we can use the previously calculated values of optimal control at a bounding pair $\boldsymbol{\alpha}^*$, $\boldsymbol{\beta}^*$ to restrict the region of search within the admissible control set $U[\boldsymbol{x}, k]$ A possible rule proved successful in practice is to search for $\boldsymbol{u}^0(\boldsymbol{x}, k)$ in the intersection of $U[\boldsymbol{x}, k]$ and the region

$$\tfrac{1}{2}[\boldsymbol{u}^0(\boldsymbol{\alpha}^*, k)+\boldsymbol{u}^0(\boldsymbol{\beta}^*, k)]-|\boldsymbol{u}^0(\boldsymbol{\alpha}^*, k)-\boldsymbol{u}^0(\boldsymbol{\beta}^*, k)|\leqq \boldsymbol{u}(\boldsymbol{x}, k)$$
$$\leqq \tfrac{1}{2}[\boldsymbol{u}^0(\boldsymbol{\alpha}^*, k)+\boldsymbol{u}^0(\boldsymbol{\beta}^*, k)]+|\boldsymbol{u}^0(\boldsymbol{\alpha}^*, k)-\boldsymbol{u}^0(\boldsymbol{\beta}^*, k)| \quad (1)$$

(*See* Fig 2) If rapid changes exist in $\boldsymbol{u}^0(\cdot, k)$ in the neighborhood of $\boldsymbol{x}$, then the absolute difference $|\boldsymbol{u}^0(\boldsymbol{\alpha}^*, k)-\boldsymbol{u}^0(\boldsymbol{\beta}^*, k)|$ should be large, thus requiring a larger

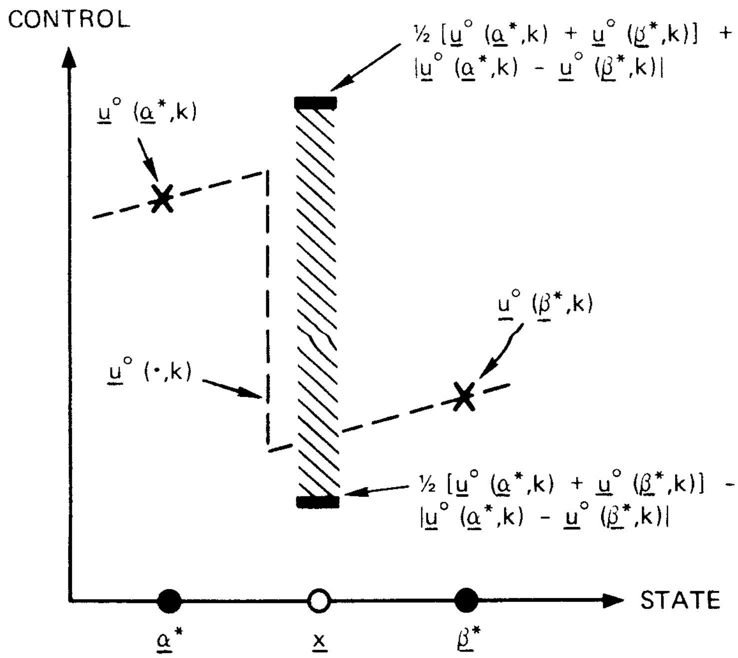

Fig 2 Region of search for $\boldsymbol{U}^0(\boldsymbol{x}, k)$

portion of the admissible control set $U[\boldsymbol{x}, k]$ to be searched Conversely, if the rate of change of $\boldsymbol{u}^0(\cdot, k)$ is small near $\boldsymbol{x}$, then a smaller portion of the control set is searched If the best control is found to be on the boundary of the region described by (1), then the region of search can be further expanded and the search for $\boldsymbol{u}^0(\boldsymbol{x}, k)$ continued until either the best control is in the interior of the region of search or on the boundary of $U[\boldsymbol{x}, k]$ Consequently, the region of search within the admissible control set is continuously adjusted to take into account the local nature of $\boldsymbol{u}^0(\cdot, k)$ It is to be noted that errors in calculating $\boldsymbol{u}^0(\boldsymbol{x}, k)$, $\boldsymbol{x}\epsilon S\text{-}S^*$ due to a partial search of $U[\boldsymbol{x}, k]$ do not propagate through the lattice structure for a given stage variable k, since the search for each $\boldsymbol{u}^0(\boldsymbol{x}, k)$, $\boldsymbol{x}\epsilon S\text{-}S^*$ is based on fully calculated values of optimal control and not on partially calculated values of optimal control

In concept, rapid changes in $\boldsymbol{u}^0(\cdot, k)$ should imply a large value for $|\boldsymbol{u}^0(\boldsymbol{\alpha}^*, k)-$

$u^0(\beta^*, k)$| However, cases exist where rapid changes in $u^0(\ , k)$ coupled with too gross a lattice structure will give inappropriate regions of search dictated by (1) Figure 3 illustrates such an example, where (1) uniquely (and incorrectly) determines the value of control without search A more complex rule to overcome the problems in Fig 3, would be to extrapolate linearly from two fully calculated values of control to the 'left' and from two fully calculated values to the 'right' of the partially calculated point x and then to search in the region bounded by these estimates (The author would like to thank a reviewer for the example and the associated rule)

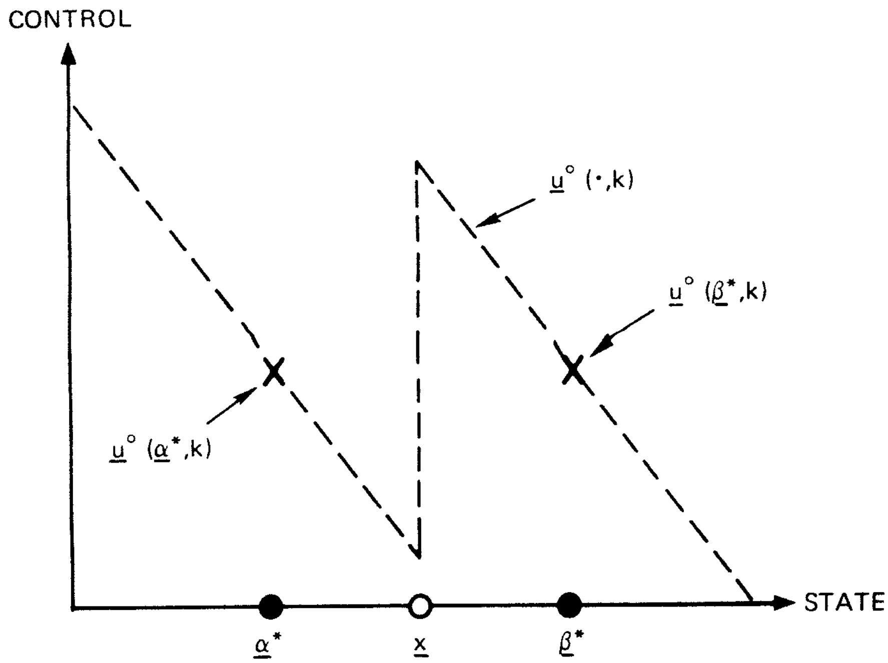

Fig 3 An example where the procedure fails

If the optimal control function $u^0(\ , k)$ is smooth over large regions of the state space, then the admissible control set is fully searched only for $x^*\epsilon S^*$ and for perhaps $x\epsilon S\text{-}S^*$, which are near regions of rapid change For a state space of dimension n, it can be shown that

$$(\text{number of points } S^*)/(\text{number of points } S) = 1/2^n$$

Consequently, depending on the smoothness of $u^0(\ , k)$, the procedure can realize a significant reduction in computing time

CONCLUSION

IF ONE knew beforehand that the optimal control function $u^0(\ , k)$ were smooth, then the computational time could be reduced by using a coarse lattice structure accompanied by the appropriate polynomial fit between lattice points However, in many cases one does not know beforehand whether the optimal control function will be sufficiently smooth to justify this approach In this case, one has no alternative but to use a fine lattice structure in calculating the optimal control function For problems in this latter category, the procedure in this paper is similar to an adaptive learning process In particular, a coarse lattice structure is first used to learn about the nature of $u^0(\, , k)$, and then this knowledge is applied to the finer lattice structure

Experience on some simple two-dimensional examples in which the optimal control function was known to be piecewise smooth gave reductions on the order of 50 percent with little or no loss in accuracy The extra programming effort to implement the procedure over the standard dynamic programming algorithm is small, and essentially consists of adding an extra iterative loop in the program

ACKNOWLEDGMENT

THIS RESEARCH was supported by the National Science Foundation

REFERENCE

1 S E DREYFUS, "Computational Aspects of Dynamic Programming," *Opns Res* 5, 408–415 (1957)

AN ALTERNATIVE PROOF OF A CONSERVATION LAW FOR THE QUEUE $G/G/1$

Linus Schrage

The University of Chicago, Chicago, Illinois

(Received January 9 1969)

KLEINROCK first showed that, for a multiclass M/G 1 queue, the expected waiting times for all classes satisfy a simple linear equality constraint that is independent of queue discipline for a large class of disciplines We generalize here the conditions under which this result holds and give a simpler proof

WE STATE and prove a theorem, first given by KLEINROCK,[2] whose original statement was for the queue with Poisson input and general independent service Here we make no similar assumptions about the arrival and service processes other than that the first two moments exist for the service process and the first moment exists for the arrival process Our proof is perhaps slightly simpler than the proof given in reference 2

Consider the queue G/G/1, that is, an arbitrary arrival stream and service process No assumptions about independence are made we require only that

Published in Operations Research on February 01, 1970 as DOI: 10.1287/opre.18.1.181.
This article has not been copyedited or formatted. The final version may differ from this version.