Numerical analysis of Markov decision processes

EINDHOVEN UNIVERSITY OF TECHNOLOGY

Department of Mathematics and Computing Science

STATISTICS AND OPERATIONS RESEARCH GROUP

Memorandum COSOR 81 - 18

Numerical Analysis of Markov Decision Processes

by

L.M.M. Veugen, Delft
J. van der Wal, Eindhoven
J. Wessels, Eindhoven

Eindhoven, the Netherlands

December 1981

NUMERICAL ANALYSIS OF MARKOV DECISION PROCESSES

L.M.M. Veugen, Delft
J. van der Wal, Eindhoven
J. Wessels, Eindhoven

Kurzfassung: In diese Arbeit werden einige Aspekte der numerische Bewerkung von Markoffschen Entscheidungsprozessen mit Diskontierung diskutiert. Ins besondere wird verzucht die Problemstruktur aus zu nützen um effiziente Algorithmen zu bekommen. Als Beispiele von Spezialstrukturen die ausgenützt werden können, werden Periodizität und umfangreiche Aktionenräume hervorgehoben. Für die letzte Spezialstruktur wird untersucht wie Aggregation und später Disaggregation von Nützen sein können.

Abstract. For the numerical analysis of Markov decision processes quite a lot of algorithms have been presented in the literature. Nevertheless, really large problems cannot be solved efficiently by standard algorithms. It remains necessary to exploit the particular structure of the problem and to use these exploitation possibilities as a selection criterion for the type of algorithm. In this paper we proceed with the exploration of this area by investigating the possibilities of exploiting periodicity of demands and the structure of actions in some inventory-management models.

1. Introduction

Many different types of algorithms have been proposed for the numerical analysis of Markov decision processes. The development of new algorithms has led to an enormous increase of computational efficiency and hence to the possibility to analyze larger problems. However, really large problems are very hard to solve if one uses the new algorithms as standard algorithms. Only by exploiting the specific properties of the model, it is possible to handle large problems efficiently. For discounted Markov decision processes this has been demonstrated by Hendrikx/van Nunen/Wessels in [2]. In this paper, we will proceed with the investigation of this aspect.

The striking result in [2] is that one has to choose the algorithm primarily on the basis of the possibilities it gives to exploit the structure of the model for reducing the amount of work per iteration. For instance, for a large 3-point inventory model with 1000 states, it is shown in [2] that the relatively primitive successive approximation method is by far the most efficient. All other methods (with the exception of one version of bisection) require at least 10 times as much process time. Even action elimination is not recommendable, since the maximization step can be executed so efficiently, that the extra work for action elimination is not compensated. This efficiency of the maximization step can only be reached by using the specific structure of the problem.

The main structural property that is utilized in [2], is typical for many decision processes, particularly in the area of inventory management and replacement. It is the property that all available actions have the form of a transition to a new, sometimes intermediate, state: if the inventory level is the state, then the action is the level up to which we order.

In the usual notation for Markov decision processes, this implies that if a labels the action as well as the intermediate state, then the transition probability p_{ij}^{a} does not depend on i and hence the maximization step may be rewritten as

$$v_n(i) = \max_a \{r(i,a) + d_{n-1}(a)\}$$

where

$$d_{n-1}(a) = \beta \sum_j p_{\cdot j}^{a} v_{n-1}(j)$$

and r(i,a) is the one stage reward in state i if action a is chosen.

Often, also r(i,a) can be split up and allows further simplification of the computation. By the way, this also shows that the model choice influences the computational efficiency, viz. new inventory is a better choice for the action than order size.

The simplification given above may cause a huge diminishment of computational work, as has been shown in [2], but it will also be clear, that it cannot always be applied if one replaces the standard version of the maximization step by the Gauss-Seidel version. Here we already see that the simplification possibilities determine the choice of the algorithm.

In this paper we will present a brief discussion of two other structural properties which might be exploited to reduce the process times. Both properties will be discussed for some inventory-management models. The first property is periodicity in the demand (section 1) and the second property the action structure (section 2). The latter property can be exploited in a simple decomposition algorithm. More elaborate discussions of these topics will appear in [6] and [7].

For lack of space, we will not start with a description of the model and an overview of the numerical methods. For these we refer to [2] and the review papers [3] and [4]. The model is the standard finite state and action Markov decision process with the criterion of total expected discounted rewards.

2. The exploitation of cyclic behaviour of the demands

Cyclic behaviour in the demand distribution frequently occurs. In the model it can be incorporated by extending the state with an extra parameter which indicates the phase in the cycle, cf. Riis [5]. When using standard successive approximations as solution method, the weak point is that all transition matrices involved are periodic and hence have more than one eigenvalue on the unit circle. As a consequence, the

convergence of this method is only linear of order β. The incorporation of the cycle phase in the state does not give extra work per iteration, since, because of the structure of the matrices, one iteration in this cyclic problem corresponds computationally to one iteration in its non-cyclic analogon. The problem, however, is the slow convergence.

In order to speed up convergence, it is necessary to replace the process by a non-cyclic process which is equivalent with respect to costs and decisions. A natural candidate is the embedded process with the cycle length as time period. This is not very attractive numerically, since actions are now c-stage strategies which require the pre-computation of all c-stage transition probabilities, if c is the cycle length. However, this candidate may be approximated by a well-chosen Gauss-Seidel step for the original process. The remaining weakness is the stop criterion, since though Gauss-Seidel usually converges faster, the extrapolations are weaker. In [2] this is solved by intermitting some pre-Jacobi steps (or standard successive approximation steps) in order to obtain good upper and lower bounds for v^*. Regrettably, the periodicity deteriorates the quality of the extrapolations for pre-Jacobi procedures. A remedy is found in the construction of extrapolations for the parts of the rewards vector for each cycle phase separately. This again stems from the idea of working with c as time period. In fact this is equivalent to the construction of extrapolations based on the difference between the expected income over n and n+1 cycles in the original problem with time dependent demands. For details see [6].

As an illustration we give processing times in seconds for 2 variants of the cash-regulation problem treated in [7]: the first has 30 stock levels and the second 80. The cycle length is one week, which corresponds to c = 10, since the time unit is half a weekday. β = .999. By a star we indicate processing times of runs aborted, because of passing iteration no. 300. The methods are as follows

J-MQ ≡ pre-Jacobi with standard MacQueen extrapolations.

GS-MQ ≡ Gauss-Seidel with standard MacQueen extrapolations based on an extra inserted pre-Jacobi step.

GS-GS ≡ Gauss-Seidel with the aforementioned specially tailored extrapolations.

Methods	problem 1	problem 2
J - MQ	27.2*	104.6*
GS - MQ	24.3*	99.0*
GS - GS	.6	1.1

If one combines these methods with bisection in situations where a bisection step is possible, cf. [2] or Bartmann [1], then the second method improves considerably as is shown by the results on the next page.

Method	problem 1	problem 2
J - MQ	29.1*	111.0*
GS - MQ	1.9	3.6
GS - GS	.6	1.1

3. Aggregation and disaggregation of actions

In problems of inventory or replacement type, one may often apply the simplified successive approximation procedure as mentioned in the introduction. When using this simplified procedure the maximization step is very fast even for many actions. So, for inventory type problems, aggregation in the action space cannot be expected to be very helpful. However, if old decisions have influence on new decisions, because of some time-lag, then the old actions have to be incorporated in the state space. The effect will ususally be a huge state space. In such cases aggregation of the actions might be helpful for obtaining a first approximate solution, which can be followed by a disaggregation step. So the solution method consists of two phases. In phase 1 the action space is thinned, by only maintaining some actions as representatives of an interval of actions (order sizes). Naturally one selects these representatives as midpoints of their respective intervals. The size Q of these intervals indicates the degree of aggregation. Aggregation of this type is very simple and natural, since it does not require any approximation of transition probabilities or rewards (for more general aggregation cf. Whitt [8]). In this first phase the problem with the thinned action space is solved. In the next phase the action spaces depend on the state of the system, namely, for each state we introduce the interval of actions of which the representative was optimal for that state in the first phase. For a more detailed analysis of this and more refined procedures compare [7].

Here we will confine ourselves to the results for one typical example. This example is again a cash-regulation problem of a bank. Now, the mornings and afternoons are again supposed to have their own demand distribution (demand can be negative), but no longer vary with the day of the week, so the periodicity is only 2 and hence less important than in the previous example. At the end of an afternoon a partial decision has to be taken, namely, whether an armed car has to appear at the end of the next morning. It also has to be decided how much money this car should have available. If it is decided that the car comes, then it is possible to decide on the exact size of deposit or intake by the bank at the last moment. Of course, the intake is constrained by the available amount in the car.

As a result of this decision set-up the states in the morning consist of the stock level at the end of the morning together with the decision of the previous afternoon. For a situation with 80 allowed stock levels this state space becomes huge and can be

made much slimmer by aggregation in the action space. The effect of different levels Q of aggregation on the process time in seconds is shown below. Of course, Q = 1 means direct computation without aggregation. The method used in each phase is successive approximation with the GS- GS method of the previous section.

Q	phase 1	phase 2	total
1	32.9	-	32.9
2	17.4	5.1	22.5
4	9.9	7.5	17.4
5	8.4	7.5	15.9
10	4.9	7.4	12.3
16	3.4	12.5	15.9
20	3.4	15.0	18.4

References

[1] D. Bartmann, A method of bisection for discounted Markov decision problems. Zeitschrift für Oper.Res. 23 (1979) 275-287.

[2] M. Hendrikx, J van Nunen, J. Wessels, Some notes on iterative optimization of structured Markov decision processes with discounted rewards. Memorandum COSOR-80-20, Eindhoven University of Technology, Department of Mathematics and Computer Science (November 1980).

[3] J. van Nunen, J. Wessels, On theory and algorithms for Markov decision problems with the total reward criterion, OR-Spectrum 1 (1979), 57-67.

[4] J. van Nunen, J. Wessels, Successive approximations for Markov decision processes and Markov games with unbounded rewards, Math. Operationsforsch. Statist. Ser. Optimization, 10 (1979), 431-455.

[5] J.O. Riis,Discounted Markov programming in a periodic process, Oper. Res. 13 (1965), 920-929.

[6] L.M.M. Veugen, J. van der Wal, J. Wessels, The numerical exploitation of periodicity in Markov decision processes, (to appear).

[7] L.M.M. Veugen, J. van der Wal, J. Wessels, Decomposition and aggregation in Markov programming models for inventory control, (to appear).

[8] W. Whitt, Approximations of dynamic programs, I, Math. Oper. Res. 3 (1978) 231-243.