ELSEVIER

Artificial Intelligence 121 (2000) 49–107

Artificial Intelligence

www.elsevier.com/locate/artint

Stochastic dynamic programming with factored representations [☆]

Craig Boutilier [a,*], Richard Dearden [b], Moisés Goldszmidt [c]

[a] *Department of Computer Science, University of Toronto, Toronto, ON, M5S 3H5, Canada*
[b] *Department of Computer Science, University of British Columbia, Vancouver, BC, V6T 1Z4, Canada*
[c] *Computer Science Department, Stanford University, Stanford, CA 94305-9010, USA*

Received 16 July 1999

Abstract

Markov decision processes (MDPs) have proven to be popular models for decision-theoretic planning, but standard dynamic programming algorithms for solving MDPs rely on explicit, state-based specifications and computations. To alleviate the combinatorial problems associated with such methods, we propose new representational and computational techniques for MDPs that exploit certain types of problem structure. We use dynamic Bayesian networks (with decision trees representing the local families of conditional probability distributions) to represent stochastic actions in an MDP, together with a decision-tree representation of rewards. Based on this representation, we develop versions of standard dynamic programming algorithms that directly manipulate decision-tree representations of policies and value functions. This generally obviates the need for state-by-state computation, aggregating states at the leaves of these trees and requiring computations only for each *aggregate* state. The key to these algorithms is a decision-theoretic generalization of classic regression analysis, in which we determine the features relevant to predicting expected value. We demonstrate the method empirically on several planning problems, showing significant savings for certain types of domains. We also identify certain classes of problems for which this technique fails to perform well and suggest extensions and related ideas that may prove useful in such circumstances. We also briefly describe an approximation scheme based on this approach.

[☆] Some parts of this report appeared in preliminary form in “Exploiting Structure in Policy Construction”, in: *Proc. Fourteenth International Joint Conference on Artificial Intelligence (IJCAI-95)*, Montreal, Quebec, pp. 1550–1556 (1995); and in “Correlated Action Effects in Decision-Theoretic Regression”, in: *Proc. Thirteenth Conference on Uncertainty in Artificial Intelligence (UAI-97)*, Providence, RI, pp. 30–37 (1997).

[*] Corresponding author.

E-mail addresses: cebly@cs.toronto.edu (C. Boutilier), dearden@cs.ubc.ca (R. Dearden), moises@robotics.stanford.edu (M. Goldszmidt).

0004-3702/00/$ – see front matter
PII: S0004-3702(00)00033-3

Keywords: Decision-theoretic planning; Markov decision processes; Bayesian networks; Regression; Decision trees; Abstraction

1. Introduction

Decision-theoretic planning (DTP) has attracted a considerable amount of attention recently as AI researchers seek to generalize the types of planning problems that can be tackled in computationally effective ways. DTP is primarily concerned with problems of sequential decision making under conditions of uncertainty and where there exist multiple, often conflicting, objectives whose desirability can be quantified. *Markov decision processes* (MDPs) have been adopted as the model of choice for DTP problems in much recent work [12,26,28,30,61,78], and have also provided the underlying foundations for most work in reinforcement learning [48,76,77,84]. MDPs allow the introduction of uncertainty into the effects of actions, the modeling of uncertain exogenous events, the presence of multiple, prioritized objectives, and the solution of nonterminating process-oriented problems. [1]

The foundations and the basic computational techniques for MDPs [3,5,44,62] are well-understood and in certain cases can be used directly in DTP. These methods exploit the dynamic programming principle and allow MDPs to be solved in time polynomial in the size of the state and action spaces that make up the planning problem. Unfortunately, these classical dynamic programming methods are formulated so as to require explicit state space enumeration. As such, AI planning systems that solve MDPs are faced with Bellman's so-called *curse of dimensionality*: the number of states grows exponentially with the number of variables that characterize the planning domain. This has an impact on the feasibility of both the specification and solution of large MDPs.

The curse of dimensionality plagues not only DTP, but also classical planning techniques. However, methods have been developed that, in many instances, circumvent this problem. In classical planning one typically does not specify actions and goals explicitly using the underlying state space, but rather "intensionally" using *propositional* or *variable-based* representations. For instance, a STRIPS representation of an action describes very concisely the transitions induced by that action over a large number of states. Similarly, classical planning techniques such as regression planning [83] or nonlinear planning [22,54,58,66] exploit these representations to great effect, never requiring that one search (or implement "shortest-path" dynamic programming techniques) explicitly through state space. Intuitively, such methods aggregate states that behave identically under a given action sequence with respect to a given goal.

In this paper, we develop similar techniques for solving certain classes of large MDPs. We first describe a representation for actions with stochastic effects that uses Bayesian networks (and decision trees to represent the required families of conditional probability

[1] One form of uncertainty cannot be handled in the framework we adopt, specifically, *partial observability*, or uncertain knowledge about the state of the system being controlled. *Partially observable MDPs* (or POMDPs) [52,53,73,75] can be used in such cases. We will make further remarks on POMDPs at the end of this article.