

# Computational Complexity in Markov Decision Theory

John N. Tsitsiklis  
Massachusetts Institute of Technology  
Cambridge, Mass., U.S.A.

## Abstract

Markov Decision Problems (MDPs) are the standard formulation of the problem of sequential decision making in a stochastic environment. As such, they arise in a multitude of contexts, from economics to engineering. We provide an overview of the current theoretical understanding of the computational issues that they present, with an emphasis on discrete (finite-state, finite-action, discrete-time) formulations. We discuss perfectly and imperfectly observed MDPs, provide a few comments on continuous counterparts, and conclude with a discussion of some nonstandard problems, such as nonclassical information structures, mean-variance optimization, and robust formulations.

## I. INTRODUCTION

The problem of sequential decision making in a stochastic environment is pervasive in many branches of science and engineering, e.g., in feedback control of engineering systems, portfolio management, supply chain management, etc. The standard mathematical formulation of this problem involves MDPs, whereby the state of the system is modeled as a Markov chain whose transition probabilities depend on the action chosen, and a state-action dependent cost is incurred at each stage.

Markov Decision Problems can be addressed, in principle, using the methods of dynamic programming. However, in many cases (e.g., if the state space dimension is large, or if the state of the system is imperfectly observed) the required computational effort is prohibitively large, a phenomenon that Bellman has termed the ‘curse of dimensionality’. This raises the question whether this is a limitation of the dynamic programming methodology or whether the problem is intrinsically difficult. In this paper, we overview the current understanding of the intrinsic difficulty of

Markov decision problems, using the tools and concepts of computational complexity theory [12].

The paper is organized as follows. In Section II, we define Markov decision problems, for the case where the state and action spaces are finite, and introduce the relevant performance criteria. We review the classical dynamic programming solution, and indicate some major open problems. In Section III, we discuss the case where the state space is continuous. In Section IV, we consider the case where the state is imperfectly observed and discuss the intrinsic complexity of the problem. Finally, in Section V, we discuss nonstandard versions of the problem, specifically, nonclassical information structures, mean-variance optimization, and robust formulations.

The literature on this subject is large, and we will make no attempt to be fair or comprehensive. A fairly complete account of the literature until 1999 is given in [3].

## II. PERFECTLY OBSERVED MDPs

A (finite and time-homogeneous) Markov Decision Problem is specified by a finite state space  $X$  (of cardinality  $n$ ), a finite action space  $U$  (of cardinality  $m$ ), a (possibly infinite) time horizon  $T$ , a discount factor  $\alpha \in [0, 1]$ , transition probabilities  $p_{xx'}(u)$ , and one-stage costs  $g(x, u)$ . At a typical time, the state is equal to some  $x \in X$ , an action  $u \in U$  is applied, and a cost  $g(x, u)$  is incurred. The next state is chosen at random and is equal to  $x'$  with probability  $p_{xx'}(u)$ . Let  $x_t$  and  $u_t$  be the state and the action at time  $t$ , respectively,

A (stationary and Markovian) *policy*  $\pi$  is specified by a mapping  $\mu : X \rightarrow U$ . When a particular policy  $\pi$  is used, the action at time  $t$  is determined according to  $u_t = \mu(x_t)$ . Once a policy is fixed, the state evolves as a homogeneous Markov chain, with one-step transition probabilities  $\mathbf{P}(x_{t+1} = x' \mid x_t = x) = p_{xx'}(\mu(x_t))$ . Let  $\alpha < 1$  be a given discount factor. We define the (infinite horizon, discounted) expected cost associated with policy  $\pi$ , starting from state  $x$ , as

$$J^\pi(x) = \mathbf{E} \left[ \sum_{k=0}^{\infty} \alpha^k g(x_k, u_k) \mid x_0 = x \right].$$

Here,  $x_k$  evolves according to the homogeneous Markov chain resulting from policy  $\pi$ , and  $u_k = \mu(x_k)$ .

The optimal cost-to-go is defined as

$$J^*(x) = \min_{\pi} J^\pi(x).$$

We are interested in calculating  $J^*(x)$  for every initial state  $x \in X$ , and an optimal policy, that is, a policy  $\pi$  such that  $J^\pi(x) = J^*(x)$ , for all  $x \in X$ . Strictly speaking, the usual notions of complexity theory (NP-completeness, decidability, etc.) refer to problems that admit a binary (yes/no) answer. Accordingly, whenever in the sequel we say that a problem is, for example, NP-complete, we will be referring to the problem of deciding whether the optimal value of the objective function (for a given initial state) is less than a given number.

There are several variants of the above problem. In one variant, the time horizon  $T$  is finite and we are interested in the expected cost accumulated during this time horizon. In this case, one usually allows the transition probabilities  $p_{xx'}(u)$  and the one-stage costs  $g(x, u)$  to change with time; without loss of generality, the discount factor can then be set to one. Accordingly, one also allows policies to be time-dependent, in the sense that  $u_t = \mu_t(x_t)$ .

Other variations include infinite-horizon average cost problems, and so-called stochastic shortest path problems, in which the time horizon  $T$  is a random variable (e.g., the process terminates as soon as a special termination state is reached) [2].

Finite horizon problems are fairly straightforward. They can be solved by the classical dynamic programming algorithm in time  $O(n^2mT)$ . We will therefore focus on infinite horizon (discounted) problems. In this case, the optimal cost-to-go function satisfies the *Bellman equation*

$$J^*(x) = \min_u \left[ g(x, u) + \alpha \sum_{x'} p_{xx'}(u) J^*(x') \right], \quad x \in X. \quad (1)$$

This is a nonlinear system of  $n$  equations, in  $n$  unknowns. It is known to have a unique solution, which can be found by iterative methods (value iteration), finitely terminating methods (policy iteration), or by a reformulation as an equivalent linear programming problem [16], [2]. Since linear programming can be solved in polynomial time, the same is therefore true for MDPs; this is essentially the only known polynomial time algorithm for MDPs. Interestingly, this linear programming approach does not seem to exploit much of the problem's structure, and one wonders whether computational efficiency gains are possible. Ideally, one would like to have an algorithm that is strongly polynomial, in the sense that it uses a number of arithmetic operations bounded by a polynomial in  $m$  and  $n$ . Whether this is possible for linear programming is an important open problem. However MDPs are a very structured special case, and one would hope that a strongly polynomial algorithm (if one exists) would be easier to

develop. Still, this issue remains unresolved and is a major open problem in dynamic programming.

In practice, one of the fastest computational methods for MDPs is policy iteration and its variants. Every iteration of this algorithm requires at most  $O(n^3 + n^2m)$  arithmetic operations, and the algorithm terminates with an optimal policy after a finite number of iterations. However, very little is known on the required number of iterations, although it appears to be rather small in practice. This raises various questions, all of which are open: is the number of iterations bounded above by (a) a polynomial in the instance size? (b) a polynomial in  $n$  and  $m$ , but possibly depending on  $\alpha$ ? (c) a polynomial in  $n$  and  $m$ , independent of  $\alpha$ ?

Even though MDPs can be solved in time that increases polynomially in the number of states, many problems of practical interest involve a very large number of states, while the problem data (e.g., the transition probabilities) are succinctly specified by a small number of parameters. Prominent examples here are the multi-armed bandit problem, and many problems in the control of queueing networks. For example, in a typical queueing network model, the number of parameters (arrival rates, service rates, routing probabilities) is usually a polynomial function of the number of queues and servers, whereas the size of the state space increases exponentially with the number of queues.

An important question is whether such problems can be solved in time polynomial in the size of the problem description (e.g., the number of parameters). It turns out that the multi-armed bandit problem is a rare case of a polynomial-time solvable problem, whereas standard queueing network problems have provably exponential complexity [15].

### III. PROBLEMS WITH CONTINUOUS STATE SPACES

When the state space is continuous (as opposed to finite), any discussion of computational complexity needs to be prefaced by the specification of an appropriate model of computation. In one model, one assumes that the problem data are given by an oracle that can provide, at request, information on the value of various input functions, at specified points. A lower bound on the problem's complexity (number of steps required to provide an answer within some desired accuracy) can then be obtained by lower bounding the number of necessary queries. For several classes of continuous MDPs, such lower bounds turn out to be tight, and indicate an exponential complexity increase with the dimension of the state space [4], [5].

#### IV. IMPERFECTLY OBSERVED PROBLEMS

The discussion so far has focused on policies of the form  $u_t = \mu(x_t)$ , so that there is an implicit assumption that the current state is known by the decision maker. In contrast, in so-called *partially observable Markov decision problems* (POMDPs), the decision maker does not know  $x_t$  but at each time  $t$  has access to an observation  $y_t$ . We assume that  $y_t$  takes values in a finite set, and is distributed according to a particular probability mass function that depends on the current state  $x_t$ . Without loss of generality, one can actually assume that  $y_t$  is a deterministic function,  $h(x_t)$ .

For the case of POMDPs, there are two different classes of policies of potential interest. In the more general class, an action can be chosen on the basis of the entire history of past observations and actions, according to  $u_t = \mu_t(y_0, u_0, y_1, u_1, \dots, u_{t-1}, y_t)$ . In a more restricted class, the action at time  $t$  is only based on the current observation, according to  $u_t = \mu(y_t)$ . In general, the optimal performance is strictly better when general history-dependent policies are allowed.

A POMDP can be reformulated as an equivalent MDP, but with an augmented state space; the augmented state can be taken to be either the past history of the process or the posterior distribution of  $x_t$  conditioned on  $(y_0, u_0, \dots, u_{t-1}, y_t)$  [2]. In either case, the state space is infinite. While the dynamic programming algorithm still applies in principle, it cannot be implemented on a digital computer. In fact, the problem turns out to be undecidable [6]. If we let the time horizon be finite, then the augmented state space is exponentially large but finite, and a finite algorithm is possible. A straightforward application of dynamic programming to this reformulated problem requires exponential time, and one may wonder whether a more efficient algorithm is possible. However, the problem is PSPACE-complete, so this is unlikely [14]. Even in the very special case of open-loop control of MDPs (the case of no observations) the problem is NP-hard [14].

#### V. NONSTANDARD FORMULATIONS

##### A. Nonclassical Information Structures

Nonclassical information structures arise primarily within the context of decentralized control, or when the structure of the controller is restricted to a specific form, such as “static output feedback.” Typically, the presence of nonclassical information structures tends to increase the difficulty of the problem.

Let  $I_t$  be the information used in choosing the action at time  $t$ . That is, we consider policies of the form  $u_t = \mu_t(I_t)$ . For the imperfectly observed

problems considered in Section IV, we have  $I_t = (y_0, u_0, \dots, u_{t-1}, y_t)$ . In particular, any information contained in  $I_t$  is also contained in  $I_{t+1}$ , and the information structure is *nested*. In nonclassical information structures,  $I_t$  is defined differently, and the nestedness property is usually absent. This is the case, for example, with static output feedback, where  $I_t = y_t$ . For another example, suppose that for odd  $t$ , we have  $I_t = (y_k : k \text{ odd}, k \leq t)$ , and for even  $t$  we have  $I_t = (y_k : k \text{ even}, k \leq t)$ . In this case, we have essentially two decision makers, who take turns in making decisions, but the information each decision maker accumulates is not shared with the other.

The case of a general information structure subsumes the case of imperfectly observed problems, and therefore the PSPACE-hardness result from Section IV applies. However, the problem remains intractable even for some very special cases. For instance, suppose there are only two decision epochs, and that the decisions applied are of the form  $u_1 = \mu_1(y_1)$  and  $u_2 = \mu_2(y_2)$ . Then, the problem of optimizing (over  $\mu_1$  and  $\mu_2$ ) a cost function of the form  $\mathbf{E}[g(x_3)]$  is NP-complete [13], [18]. This case can be viewed alternatively as one involving decentralized decision making (the decision  $u_t$  is applied by agent  $t$ , based on private information), or static output feedback (there is only one controller, but it does not keep past observations in memory). Thus, all of these problems are intractable even when the state and action spaces are finite and the time horizon is very short. This is to be contrasted with the case of nested information structures; a POMDP with time horizon 2 (or any other *fixed* time horizon) can be solved in polynomial time.

### B. Mean-Variance Optimization

In many applications, especially in finance, one is not just interested in the expected cost, but also in the variance of the cost. Let us consider a finite-state, finite-action MDP, and let  $J^\pi(x)$ ,  $V^\pi(x)$  be the expectation and the variance, respectively, of the infinite horizon discounted cost  $\sum_{k=0}^{\infty} \alpha^k g(x_k, u_k)$ , when the initial state is  $x$ . Let  $c$  be a given constant. We are interested in the problem of minimizing  $J^\pi(x)$ , over all policies in some set  $\Pi$ , subject to the constraint  $V^\pi(x) \leq c$ .

The complexity of this problem depends on the particular class of policies being considered. Let  $z_t$  be the cost that has been accumulated until time  $t$ , that is,  $z_t = \sum_{k=0}^t \alpha^k g(x_k, u_k)$ . In the most general class of policies,  $u_t$  is allowed to be a function of the entire history  $h_t = (x_0, u_0, z_0, \dots, x_{t-1}, u_{t-1}, z_{t-1}, x_t)$ , together with an independent randomization variable  $w_t$ . It turns out that the problem can be addressed in

principle [10] using the methodology of dynamic programming with side constraints [1], applied to the augmented state  $(x_t, z_t)$ . In particular, one can restrict to policies of the form  $u_t = \mu_t(x_t, z_t, w_t)$ . Consistent with what is known for general constrained MDPs, randomization is essential: the optimal performance deteriorates if we exclude randomized policies.

The difficulty with the above outlined approach is that  $z_t$  now takes values in an infinite set, which makes an exact solution difficult, if not impossible. In particular, it can be shown that the problem is NP-hard [10]. By discretizing the set of all possible values of  $z_t$ , an approximate solution becomes possible. A solution, within some desired accuracy  $\epsilon$  can be obtained in time which is polynomial in the cardinalities of the state and action spaces, in  $1/\epsilon$ , and in  $1/(1 - \alpha)$ .

Sometimes, there may be a preference for “simpler” policies, for example, randomized policies of the form  $u_t = \mu_t(x_t, w_t)$ , or deterministic policies of the form  $u_t = \mu_t(x_t)$ . Once again, there are simple examples that show that performance can be strictly worse if randomization is prohibited. Furthermore, the optimal performance obtained with such policies can be strictly worse than the optimal performance obtained with general history-dependent policies. Finally, optimizing within this restricted class of policies (with or without randomization) is a strongly NP-complete problem [10]. We conclude that such restricted policies are inferior in terms of both performance and complexity.

### C. Robust Formulations

One of the reasons why MDPs are sometimes impractical is the assumption that the model parameters (e.g., the transition probabilities) are known exactly. However, in many situations, these parameters are only estimated from data, and modeling errors are to be expected. Furthermore, the performance of a policy which is optimal for a nominal problem, may be very sensitive to parameter variations. This motivates so-called robust approaches, in which the model is only assumed to lie in a given set of possible models, and performance is optimized for the worst case over possible models. Unfortunately, as we will now discuss, such robust formulations usually lead to hard problems.

Suppose that we have a collection of MDPs, all with the same finite state and action spaces, but with possibly different transition probabilities or one-stage costs. Let  $M$  be the set of such MDPs, assumed finite, and let  $J_m^\pi(x)$  be the expected (infinite horizon, discounted) cost if policy  $\pi$  is applied to MDP  $m$ . The robust formulation aims at minimizing the worst-case cost  $\max_{m \in M} J_m^\pi(x)$ .

If we have a prior distribution on the set of possible models, we can consider the problem of minimizing  $\mathbf{E}[J_m^\pi(x)]$ , where the expectation is taken over the random model  $m$ . This problem can be addressed by using the augmented state  $\bar{x}_t = (m, x_t)$ , but what we obtain is a partially observed MDP (because  $m$  is not observed). It turns out that the proof of PSPACE-hardness for finite horizon POMDPs applies to the present context as well. In particular, the problems of minimizing either  $\max_{m \in M} J_m^\pi(x)$  or  $\mathbf{E}[J_m^\pi(x)]$ , over the set of all policies, is PSPACE-hard [9].

If we restrict ourselves to policies of the form  $u_t = \mu(x_t)$ , performance will in general deteriorate. While the problem is apparently a little easier, it is still intractable: under either of the criteria considered above, the problem is NP-complete [9], even if the set  $M$  contains only two alternative models. Furthermore, NP-completeness results can be obtained even for the special case of deterministic models where the only uncertain parameters are the values of the costs  $g(x, u)$ .

The main exception to the above negative results on robust MDPs is the case where the objective is the worst-case cost  $\max_{m \in M} J_m^\pi(x)$ , and the set  $M$  of models has a particular Cartesian product structure, so that effectively we are dealing with an adversary who can choose independently the model parameters applicable to each state. In this case, we are dealing with a so-called zero-sum Markov game, for which a suitable Bellman-like equation (the Shapley equation) applies [17]. Even though it is not known whether the infinite-horizon problem can be solved in polynomial time, at least the finite-horizon version of the problem can be solved in time polynomial in the instance size and the time horizon [7], [11].

## VI. CONCLUSIONS

Markov decision theory is a rich problem domain as far as computational complexity is concerned. In particular, there are natural problems exemplifying several natural complexity classes. More interestingly, complexity considerations provide useful insights on many important problems, of practical significance.

## ACKNOWLEDGMENT

The author would like to acknowledge collaborations with Vincent Blondel, Yann Le Tallec, Shie Mannor, and Christos Papadimitriou in developing several of the results reported here. Parts of this paper are developed along the lines of the survey paper [3] coauthored with Vincent Blondel. The results in Sections V-B and V-C represent joint work with Shie Mannor and Yann Le Tallec, respectively.

## REFERENCES

- [1] E. Altman and A. Shwartz, "Markov decision problems and state-action frequencies," *SIAM J. Control and Optimization*, Vol. 29, No. 4, pp. 786-809, 1991.
- [2] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, Athena Scientific, Belmont, MA, 2005
- [3] V. D. Blondel and J. N. Tsitsiklis, "A survey of computational complexity results in systems and control," *Automatica*, Vol. 36, No. 9, pp. 1249-1274, 2000.
- [4] C.-S. Chow and J. N. Tsitsiklis, "The complexity of dynamic programming," *Journal of Complexity*, Vol. 5, pp. 466-488, 1989.
- [5] C.-S. Chow and J. N. Tsitsiklis, "An optimal one-way multigrid algorithm for discrete-time stochastic control," *IEEE Transactions on Automatic Control*, Vol. 36, pp. 898-914, 1991.
- [6] O. Madani, S. Hanks, and A. Condon, "On the undecidability of probabilistic planning and related stochastic optimization problems", *Artificial Intelligence*, Vol. 147, No. 1-2, pp. 5-34, 2003.
- [7] G. Iyengar, "Robust dynamic programming," *Mathematics of Operations Research*, Vol. 30, 2005.
- [8] H. J. Kushner and P. G. Dupuis, *Numerical Methods for Stochastic Control Problems in Continuous Time*, 2nd Ed. Springer-Verlag, New York, 2001.
- [9] Y. Le Tallec, "Robust, Risk-Sensitive, and Data-driven Control of Markov Decision Processes," doctoral dissertation, Operations Research Center, MIT, Cambridge, MA, 2007.
- [10] S. Mannor and J. N. Tsitsiklis, "Mean-variance criteria in Markov decision problems," in preparation, 2007.
- [11] A. Nilim and L. El Ghaoui, "Robust solutions to Markov decision problems with uncertain transition matrices," *Operations Research*, Vol. 53, No. 5, 2005.
- [12] C. H. Papadimitriou, *Computational Complexity*, Addison Wesley, 1993.
- [13] C. H. Papadimitriou and J. N. Tsitsiklis, "On the complexity of designing distributed protocols," *Information and Control*, Vol. 53, pp. 211-218, 1982.
- [14] C. H. Papadimitriou and J. N. Tsitsiklis, "The complexity of Markov decision processes," *Mathematics of Operations Research*, Vol. 12, No. 3, pp. 441-450, 1987.
- [15] C. H. Papadimitriou and J. N. Tsitsiklis, "The complexity of optimal queueing network control," *Mathematics of Operations Research*, Vol. 24, No. 2, pp. 293-305, 1999.
- [16] M. Puterman, *Markov Decision Processes*, Wiley-Interscience, New York, 1994.
- [17] L. S. Shapley, "Stochastic games," in *Proceedings of the National Academy of Sciences*, Vol. 39, pp. 1095-1100, 1953.
- [18] J. N. Tsitsiklis and M. Athans, "On the complexity of decentralized decision making and detection problems," *IEEE Transactions on Automatic Control*, Vol. 30, pp. 440-446, 1985.