Available online at www.sciencedirect.com
ScienceDirect
Operations Research Letters 35 (2007) 319–323
Operations Research Letters
www.elsevier.com/locate/orl

NP-Hardness of checking the unichain condition in average cost MDPs☆

John N. Tsitsiklis*

Massachusetts Institute of Technology, Room 32-D662, M.I.T., 77 Massachusetts Avenue, Cambridge, MA 02421, USA

Received 15 May 2006; accepted 25 June 2006
Available online 21 August 2006

Abstract

The unichain condition requires that every policy in an MDP result in a single ergodic class, and guarantees that the optimal average cost is independent of the initial state. We show that checking whether the unichain condition fails to hold is an NP-complete problem. We conclude with a brief discussion of the merits of the more general weak accessibility condition.
© 2006 Elsevier B.V. All rights reserved.

Keywords: Markov decision processes; Average cost; Unichain condition

1. Introduction

We consider a finite-state, finite-action average cost Markov decision process (MDP), specified in terms of a state space $S = \{1, \ldots, N\}$, an action space $U = \{1, \ldots, M\}$, transition probabilities $p_{ij}(u)$ for every $i, j \in S$ and $u \in U$, and a cost $c_i(u)$ for every $i \in S$ and $u \in U$. A function μ: $S \mapsto U$ specifies the (stationary and Markovian) policy that selects action $\mu(i)$ whenever in state i. Let Π be the set of all such policies.

A policy $\mu \in \Pi$ defines a homogeneous Markov chain $\{X_t^\mu\}$ with transition probabilities

$$\mathbf{P}(X_{t+1}^\mu = j \mid X_t^\mu = i) = p_{ij}(\mu(i)).$$

The infinite horizon average cost associated with such a policy is defined as

$$\lambda^\mu(i) = \limsup_{T\to\infty} \frac{1}{T}\mathbf{E}\left[\sum_{t=1}^{T} c_{X_t^\mu}(U_t^\mu) \mid X_0^\mu = i\right],$$

where $U_t^\mu = \mu(X_t^\mu)$. The optimal average cost is defined as

$$\lambda^*(i) = \min_{\mu\in\Pi} \lambda^\mu(i).$$

As is well known, the problem of finding an optimal policy, with respect to the average cost criterion, is

☆ Research supported in part by the NSF under contract ECS-0312921.
* Tel.: +1 617 253 6175.
E-mail address: jnt@mit.edu.

0167-6377/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.orl.2006.06.005

best behaved when the optimal average cost $\lambda^*(i)$ is the same for all initial states i. A simple criterion for this to happen is provided by the following condition [2,7].

Unichain condition (*UC*): For every policy $\mu \in \Pi$, the resulting Markov chain $\{X_t^\mu\}$ has a single ergodic class.

Although the unichain condition is very intuitive, it is not clear how to check it without enumerating all policies (of which there are exponentially many). The question of whether this can be done in polynomial time is raised in [3], and is listed as an open problem in [4]. While the problem is known to be polynomial for deterministic MDPs [6], our main contribution is to show that the general case is NP-hard.

2. NP-Completeness result

Formally, we introduce the following decision problem.

Problem MULTICHAIN. *Input*: Positive integers N and M, and nonnegative rational numbers $p_{ij}(u)$, for $i, j \in \{1, \ldots, N\}$ and $u \in \{1, \ldots, M\}$, such that $\sum_{j=1}^{N} p_{ij}(u) = 1$ for all i and u.

Question: Does there exist a policy under which the resulting Markov chain has multiple ergodic classes?

Note that our description of an instance (the "input") does not include the cost coefficients $c_i(u)$. This is because the costs are irrelevant as far as the unichain condition is concerned.

In the sequel, we will be using the following terminology. We say that a state j is *accessible* from a state i if there exists a sequence of states $i = i_1, i_2, \ldots, i_k = j$ and a set of actions $u_1, \ldots, u_{k-1}$ such that $p_{i_t, i_{t+1}}(u_t) > 0$ for $t = 1, \ldots, k-1$.

Theorem 1. *The Problem* MULTICHAIN *is NP-complete.*

Proof. Suppose that there exists a policy that results in multiple ergodic classes. Such a policy serves as a certificate that the answer is "yes," and therefore the problem is in NP. (This is because the policy can be concisely described, and the multichain property can be checked in polynomial time in a straightforward manner, using a graph connectivity algorithm.)

To prove that the problem is NP-complete we use a reduction from the 3-satisfiability problem (3SAT). An instance of 3SAT consists of n Boolean variables $x_1, \ldots, x_n$, and m clauses $C_1, \ldots, C_m$, with three literals per clause. Each clause is the disjunction of three literals, where a literal is either a variable or its negation. (For example, $x_2 \vee \overline{x}_4 \vee x_5$ is such a clause, where a bar stands for negation.) The question is whether there exists an assignment of truth values ("true" or "false") to the variables such that all clauses are satisfied.

Suppose that we are given an instance of 3SAT, with n variables, and m clauses $C_1, \ldots, C_m$. We construct an instance of the MULTICHAIN problem, with the following states:

(a) two special states a and b;
(b) for $i = 1, \ldots, n$, states s_i, s_i', t_i, f_i;
(c) for $j = 1, \ldots, m$, states c_j.

There are three actions available at each state, and the transitions are as follows:

(a) Out of state a, there is equal probability $1/(n+m)$, of transitioning to each state s_i and c_j, independent of the action.
(b) At state c_j, the action determines the next state deterministically. In particular, if the kth literal in clause C_j is of the form x_i, action k moves the state to t_i; if the kth literal in clause C_j is of the form $\overline{x}_i$, action k moves the state to f_i. (For example, if the clause is of the form $x_2 \vee \overline{x}_4 \vee x_5$, the action chooses whether the next state will be t_2, f_4, or t_5.)
(c) At any state of the form s_i or s_i', the action determines whether the next state will be t_i or f_i, deterministically. (For example, under action 1, the next state is t_i, and under action 2 or 3, the next state is f_i.)
(d) At any state of the form t_i or f_i, the action determines whether the next state will be a or b, deterministically.
(e) Out of state b, there is equal probability $1/n$, of transitioning to each state s_i', independent of the action.

The transition diagram is illustrated in Fig. 1.

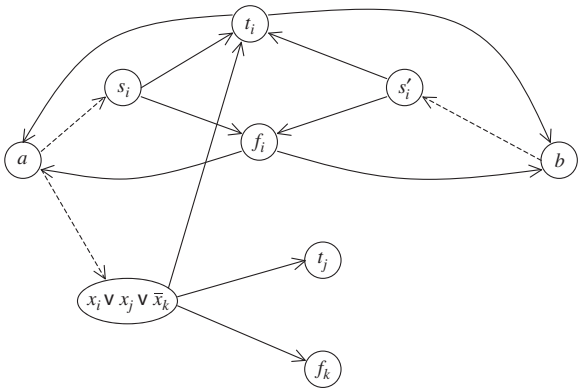

Fig. 1. Illustration of the reduction. The figure shows a representative fragment of the transition diagram. We only show one state of the form c_ℓ, associated with a clause $x_i \vee x_j \vee \overline{x}_k$. Dashed lines indicate transitions that happen with positive probability. Solid lines indicate possible (deterministic) transitions, depending on the action chosen. Let us focus on the top six states $(a, s_i, t_i, f_i, s'_i, b)$, and suppose for a moment that these are the only states. We see that there exists a policy that results in two ergodic classes, namely, $\{a, s_i, t_i\}$ and $\{b, s'_i, f_i\}$ (think of this policy as setting x_i to "true"), as well as another policy that results in two ergodic classes, namely, $\{a, s_i, f_i\}$ and $\{b, s'_i, t_i\}$ (think of this policy as setting x_i to "false").

We claim that we have a "yes" instance of MULTICHAIN if and only if we have a "yes" instance of 3SAT. Indeed, suppose that we have a "yes" instance of 3SAT. Consider an assignment of truth values ("true" or "false") to the variables such that all clauses are satisfied. We then define the following policy:

(a) At every state of the form c_j, consider the associated clause and a literal in the clause which is "true." If that literal is unnegated (say, x_k), pick the action that moves the state to t_k; if that literal is negated (say $\overline{x}_k$), pick the action that moves the state to f_k.
(b) At every state of the form s_i, let the next state be t_i if x_i is "true," and f_i if x_i is "false."
(c) At every state of the form s'_i, let the next state be f_i if x_i is "true," and t_i if x_i is "false."
(d) At every state of the form t_i, let the next state be a if x_i is "true," and b if x_i is "false."
(e) At every state of the form f_i, let the next state be b if x_i is "true," and a if x_i is "false."

Suppose that the Markov chain is initialized at a. If the next state is s_i, the subsequent states will be t_i and then a (if x_i is "true"), or they will be f_i and then a (if x_i is "false"). If the next state is c_j and the unnegated literal x_k makes this clause true, the subsequent states will be t_k and then a; if the negated literal $\overline{x}_k$ makes this clause true, the subsequent states will be f_k and then a. We conclude that a is a recurrent state and that starting from a, state b is never visited.

Suppose now that the Markov chain is initialized at b, and that the next state is s'_i. If x_i is "true," the subsequent states will be f_i and then b. If x_i is "false," the subsequent states will be t_i and then b. We conclude that b is also a recurrent state, and that the two states a and b belong to different ergodic classes. Therefore, we have a "yes" instance of MULTICHAIN.

For the converse, suppose that we have a "yes" instance of MULTICHAIN, and fix a policy that results in multiple ergodic classes. Given the structure of the possible transitions, the state belongs to the set $\{a, b\}$ once every three transitions. Since we have multiple ergodic classes, it follows that a and b are both recurrent but do not belong to the same ergodic class; in particular, b is not accessible from a, and vice versa. Consider the following truth assignment: if the transition out of state s_i leads to state t_i (respectively, f_i), set x_i to "true" (respectively, "false"). We need to show that this truth assignment satisfies all clauses.

Suppose that the transition out of s_i leads to t_i. Since b is not accessible from a, it follows that b is not accessible from t_i, and therefore the action out of t_i leads back to a. Furthermore, since a is not accessible from b, the transition out of s'_i leads to f_i and then back to b.

Similarly, suppose that the transition out of s_i leads to f_i. Since b is not accessible from a, it follows that b is not accessible from f_i, and therefore the action out of f_i leads back to a. Furthermore, since a is not accessible from b, the transition out of s'_i leads to t_i and then back to b.

Consider now a clause C_j. Suppose that the transition out of c_j leads to t_i (note that this implies that x_i appears in C_j unnegated). In particular, t_i is accessible from a. Since b is not accessible from a, it follows that t_i leads back to a. Using the remarks in the two preceding paragraphs, it follows that the transition out of s_i leads to t_i, and therefore x_i is set to "true," and the clause is satisfied.

Suppose now that the transition out of c_j leads to f_i (note that this implies that x_i appears in C_j negated).

In particular, f_i is accessible from a. Since b is not accessible from a, it follows that f_i leads back to a. Using the earlier remarks, it follows that the transition out of s_i also leads to f_i, and therefore x_i is set to "false," and the clause is satisfied.

We conclude that with the proposed truth assignment, all clauses are satisfied, and we have a "yes" instance of 3SAT, which completes the proof. □

3. Discussion

It is well known that if every state is accessible from every other state, then the optimal average cost is the same for every initial state [2,7]. In fact, this requirement need not be imposed on states that are transient under all policies, since such states will only be visited a finite number of times anyway. In particular, the optimal average cost is the same for every initial state under the following condition [7].

Weak accessibility (*WA*): The state space can be partitioned into two subsets, S_1 and S_2, such that:

(a) Every state in S_1 is transient, under every policy.
(b) For every two states $i, j \in S_2$, j is accessible from i.

We argue that Condition WA is the natural one for the question of whether the optimal average cost is the same for all initial states. Our argument rests on three observations:

(a) Unlike the unichain condition, condition WA is easy to check (in polynomial time, using standard graph connectivity algorithms).
(b) It is more general than the unichain condition [7].
(c) It is the most general possible condition that does not involve an explicit calculation of the optimal average cost (Proposition 1).

Consider an MDP that involves two sets of states that do not communicate with each other, no matter which policy is used. Then, the optimal average cost problem decouples into two independent subproblems. The optimal average cost may still turn out to be constant over the state space, if for some accidental reason the optimal average costs in the two independent subproblems happen to be equal. Determining whether this will be the case or not is impossible without actually solving the two subproblems. This indicates that one cannot hope for a necessary and sufficient condition for a constant optimal average cost that is any simpler than a complete solution of the problem. However, a necessary and sufficient condition is possible if one is willing to disregard "numerical accidents."

Definition 1. Suppose we are given positive integers N, M, and nonnegative numbers $p_{ij}(u)$, for $i, j \in \{1, \ldots, N\}$ and $u \in \{1, \ldots, M\}$, such that $\sum_{j=1}^{N} p_{ij}(u) = 1$ for all i and u. We say that *the optimal average cost is generically constant* if the set of vectors $(c_i(u);\ i = 1, \ldots, N;\ u = 1, \ldots, M)$ that result in nonconstant average cost has Lebesgue measure zero.

Proposition 2. *The optimal average cost is generically constant if and only if the WA condition holds.*

Proof. If WA holds, then the optimal average cost is always constant. Suppose that WA fails to hold. Let i and j be states that can be made recurrent (under suitable policies) and such that j is not accessible from i. Suppose that $c_j(u) < 0$ and $c_k(u) \in [0, \varepsilon]$, for all u and $k \neq j$, where ε is a small positive number. Starting from state j, an optimal policy makes j recurrent and, as long as ε is sufficiently small, results in a negative optimal average cost. Starting from state i, the process will never reach j, and the optimal average cost will be nonnegative. Thus, the optimal average cost is not constant. Furthermore, the set of cost vectors that satisfy the conditions we just introduced has positive Lebesgue measure. □

Despite the above arguments, the unichain condition is not completely without interest. In particular, consider a constrained average cost MDP, as in [1]. If the unichain condition holds, then there exists an optimal (randomized) policy which is Markovian and stationary [1]. On the other hand, if the unichain condition fails to hold, there exists a choice of cost and constraint coefficients under which an optimal Markovian policy exists, but any such optimal policy must be nonstationary; see, e.g., Example 3.1 in [5]. Thus, the unichain condition is the natural one as far as the structure of optimal policies in constrained MDPs is concerned.

Acknowledgments

The author is grateful to Eugene Feinberg for bringing the problem to his attention and providing references, to Huizhen Yu for her comments on the manuscript and for bringing up the connection with constrained MDPs, and to Dimitri Bertsekas for useful discussions and comments.

References

[1] E. Altman, Constrained Markov Decision Processes, Chapman & Hall, London, 1999.
[2] D.P. Bertsekas, Dynamic Programming and Optimal Control, vol. II, second ed., Athena Scientific, Belmont, MA, 2001.
[3] L.C.M. Kallenberg, Classification problems in MDPs, in: Z. How, J.A. Filar, A. Chen (Eds.), Markov Processes and Controlled Markov Chains, Kluwer, Boston, 2002, pp. 151–165.
[4] L.C.M. Kallenberg, Finite state and action MDPs, in: E.A. Feinberg, A. Shwartz (Eds.), Handbook of Markov Decision Processes, Kluwer, Boston, 2002.
[5] S. Mannor, J.N. Tsitsiklis, On the empirical state-action frequencies in Markov decision processes under general policies, Math. Oper. Res. 30 (3) (2005) 545–561.
[6] W. McCuaig, Intercyclic digraphs, in: N. Robertson, P. Seymour (Eds.), Graph Structure Theory, Contemporary Mathematics, vol. 147, Amererican Mathematical Society, Providence, RI, 1993, pp. 203–245.
[7] M. Puterman, Markov Decision Processes: Discrete Stochastic Dynamic Programming, Wiley, Hoboken, NJ, 1994.