Progressive State Space Disaggregation for Dynamic Programming

O. Forghieri[1], H. Castel-Taleb[4], E. Hyon[2,3], E. Le Pennec[1]

[1] Ecole Polytechnique, Institut Polytechnique de Paris
orso.forghieri@polytechnique.edu erwan.le-pennec@polytechnique.edu
[2] LIP6, Université Paris Sorbonne, CNRS, Paris, France
[3] Université de Paris Nanterre, Nanterre, France
emmanuel.hyon@parisnanterre.fr
[4] Telecom Sud Paris, Evry, France
hind.castel@telecom-sudparis.eu

Mots-clés : *Markov Decision Process, Hierarchical, Aggregated Markov Decision Process*

1 Introduction

Markov Decision Process is a framework to solve stochastic dynamic control problem in which the environments evolve under the actions of an agent and in which actions could optimize an expected gain [10]. The goal is therefore to find the optimal sequence of actions (named policy) that gives the maximal return. A various categories of problems can be modeled in this way as [3] achieves a recent overview. One can quote inventory control, energy control, networks of queues and stochastic shortest past in robot exploration. There is a great interest in solving these problems exactly as it is sometimes necessary to ensure optimal decision-making in uncertain environments. The solution to this problem can be either an optimal policy or a value function that assigns each state a visit priority [13].

The widely acknowledged curse of dimensionality in the MDP framework hinders its scalability in practical problems due to an excessively large state space, leading to inefficient computational outcomes. Various strategies aim to address this challenge by decomposing an MDP into a more manageable form. Notably, Factored MDPs [7] represent the state as a feature vector and Reduced-Rank MDPs [11] uses scalar product transition function. Additionally, Hierarchical solutions [8] provides options as actions that lasts in time or aggregation of state space into regions [14, 9]. This study focuses on state aggregation, aiming to determine optimal state groupings and evaluate the aggregation quality. The selection of merging criteria is pivotal, with prior works proposing diverse approaches. Bisimulation [5], soft aggregation with probabilistic state inclusion [12], limited number of regions [6] or various aggregation critera [1] are implemented which induces approximation bound on the optimal policy [15, 1]. The aggregation process can enhance solving algorithms like Policy Iteration [2, 4].

2 Our Contribution

We propose a new class of algorithms based on dynamic programming methods. Our method has two main advantages : it saves computation time by updating the value of several states at once, these states being grouped together under a certain similarity criterion.

More precisely, we first gather all the states in the same region. We then divide this region into several pieces, then each of these regions into several pieces, and so on. To divide a region into several pieces, we separate the states that evolve differently under the action of the Bellman operator, which updates the value function. Between two divisions, we update the value of each

region with a projected Bellman operator, which is less costly to calculate. It is therefore a progressive disaggregation of states, starting from a single region. At the end of the process, the states that have remained together evolve in the same way when the Bellman operator is updated. These states have the same value and therefore the same role in the MDP. The final value function obtained in this process can approximate the optimal value function to an arbitrary ε precision, thus obtaining an optimal policy, guarantee based on the following result we proved :

$$\|\tilde{V} - V^*\|_\infty \leq \frac{1}{1-\gamma}\left(\|\tilde{V} - \Pi\mathcal{T}^*\tilde{V}\|_\infty + \mathrm{Span}_{S_k}\,\mathcal{T}^*\tilde{V}\right)$$

where $\tilde{V}$ is the value on region, V^* is the optimal value function, γ is the discount factor, $\mathcal{T}^*$ is the optimal Bellman operator, $\Pi\mathcal{T}^*$ is the projected optimal Bellman operator and $\mathrm{Span}_S\, V$ is defined as $\max_{s\in S} V(s) - \min_{s\in S} V(s)$.

In this study, we streamline the problem by transforming it into an abstract Markov Decision Process (MDP) and precisely address this refined scenario. We present an algorithm designed to yield an optimal policy for decision-making problems. By consolidating states, we effectively diminish the intricacies associated with planning. Consequently, our numerical experiments underscore substantial time efficiencies in resolving extensive problems.

Références

[1] David Abel, David Hershkowitz, and Michael Littman. Near optimal behavior via approximate state abstraction. In *International Conference on Machine Learning*, 2016.

[2] D. P. Bertsekas and D. A. Castanon. Adaptive aggregation methods for infinite horizon dynamic programming. *IEEE Transactions on Automatic Control*, 34(6) :589–598, 1989.

[3] R. Boucherie and N.M. van Dijk. *Markov Decision Processes in Practice*. Springer, 2017.

[4] G. Chen, J. D. Gaebler, M. Peng, C. Sun, and Y. Ye. An adaptive state aggregation algorithm for Markov Decision Processes. In *AAAI 2022 Workshop on Reinforcement Learning in Games*, 2022.

[5] Thomas Dean and Robert Givan. Model minimization in Markov Decision Processes. In *AAAI/IAAI*, pages 106–111, 1997.

[6] J.G. Ferrer-Mestres, T. Dietterich, O. Buffet, and I. Chadès. Solving K-MDPs. In *ICAPS*, volume 30, pages 110–118, 2020.

[7] C. Guestrin, D. Koller, R. Parr, and S. Venkataraman. Efficient solution algorithms for factored mdps. *Journal of Artificial Intelligence Research*, 19 :399–468, 2003.

[8] Bernhard Hengst. Hierarchical approaches. In *Reinforcement learning*, pages 293–323. Springer, 2012.

[9] Lihong Li, Thomas J Walsh, and Michael L Littman. Towards a unified theory of state abstraction for mdps. In *AI&M*, 2006.

[10] Martin L Puterman. *Markov decision processes : discrete stochastic dynamic programming*. John Wiley & Sons, 1994.

[11] S. Siddiqi, B. Boots, and G. Gordon. Reduced-rank hidden markov models. In *Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 741–748, 2010.

[12] Satinder Singh, Tommi Jaakkola, and Michael Jordan. Reinforcement learning with soft state aggregation. *Advances in neural information processing systems*, 7, 1994.

[13] Sutton and Barto. *Reinforcement learning : An introduction*. MIT press, 2018.

[14] R.S. Sutton, D. Precup, and S. Singh. Between MDPs and semi-MDPs : A framework for temporal abstraction in reinforcement learning. *Artificial intelligence*, 112(1-2) :181–211, 1999.

[15] John N Tsitsiklis and Benjamin Van Roy. Feature-based methods for large scale dynamic programming. *Machine Learning*, 22(1-3) :59–94, 1996.