Symbolic Dynamic Programming for Risk-sensitive Markov Decision Process with limited budget

Daniel A. M. Moreira
University of São Paulo
Sao Paulo, Brazil
danm@ime.usp.br

Karina Valdivia Delgado
University of São Paulo
Sao Paulo, Brazil
kvd@usp.br

Leliane Nunes de Barros
University of São Paulo
Sao Paulo, Brazil
leliane@ime.usp.br

CCS CONCEPTS

• **Computing methodologies → Planning under uncertainty**;

KEYWORDS

Agent theories and models; single agent planning; Markov Decision Process; risk.

ACM Reference Format:

Daniel A. M. Moreira, Karina Valdivia Delgado, and Leliane Nunes de Barros. 2018. Symbolic Dynamic Programming for Risk-sensitive Markov Decision Process with limited budget. In *Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), Stockholm, Sweden, July 10–15, 2018,* IFAAMAS, 3 pages.

1 INTRODUCTION

Markov Decision Process (MDP) [6] is the standard model for decision planning under uncertainty and its goal is to find a policy that minimizes the expected cumulative cost. Although this optimization criterion fits well for many problems, they do not guarantee a low cost variance. Thus, in situations where the optimal policy is expected to be executed only few times, it is unacceptable to have a result with too high cost. Risk-Sensitive MDPs (RS-MDP) [4, 7] can be used to deal with such situations.

A Risk-Sensitive MDP (RS-MDP) [7] is a tuple $\langle S, s_I, A, T, C, S_g, \theta_u \rangle$ where: (i) S is a finite state set; (ii) $s_I \in S$ is the Initial state; (iii) A is a finite action set; (iv) $T\colon S \times A \times S \longrightarrow [0, 1]$ is a state transition function; (v) $C\colon S \times A \longrightarrow \mathbb{R}^+$ is a cost function; (vi) $S_g \subseteq S$ is a set of absorbing goal states; and (vii) $\theta_u \in \mathbf{\Theta}$ is the user-defined cost threshold, where $\mathbf{\Theta}$ is the set of remaining budgets obtained during the process (i.e., each state reached in the process has associated a budget $\theta \in \mathbf{\Theta}$).

The objective of an RS-MDP is to find a policy that maximizes the probability of histories $h(\pi)$ starting in s with budget θ and ending in $s \in S_g$, whose cumulative cost does not exceed θ, that is:

$$P^*(s, \theta) = \max \left[\sum_{h(\pi): c(h(\pi), s) \leq \theta} Pr(h(\pi), s, \theta) \right], \tag{1}$$

where $c(h(\pi), s)$ is the cumulative cost following the history $h(\pi)$ from state s and $Pr(h(\pi), s, \theta)$ is the probability of the history $h(\pi)$ from state s and budget θ to happen. The probability $P^*(s, \theta)$ is called *cost-threshold probability*.

Previous work has proven that optimal policies of RS-MDPs, $\pi^*\colon S \times \mathbf{\Theta} \longrightarrow A$, are stationary and deterministic [4]. The space given by the pairs (s, θ) is called augmented state space. Thus, the optimal cost-threshold probability can be recursively defined as:

$$P^*(s, \theta) = \\ \max_{a \in A} \sum_{s' \in S} \begin{cases} 0 & \text{if } C(s, a) > \theta \\ T(s'|s, a) * P^*(s', \theta - C(s, a)) & \text{if } C(s, a) \leq \theta \end{cases} \tag{2}$$

Hou et.al (2014) proposed an algorithm for RS-MDPs based on the Topological Value Iteration (TVI) [2] (which uses the Tarjan's algorithm to find Strongly Connected Components of S), called TVI-DP, that computes the optimal cost-threshold probability for θ varying from 0 to θ_u, with an increment of 1. The main limitation of TVI-DP is to compute $P^*(s, \theta)$ for the whole augmented state space.

An extension of TVI-DP, called *Improved* TVI-DP (ITVI-DP) [5], proposed two major improvements: (i) an early termination based on the *convergence behavior* theorem of TVI-DP with growing threshold budgets; and (ii) to prune augmented states generated during the forward search of the Tarjan's algorithm by only generating states belonging to trajectories that end in a goal state. The *convergence behavior* theorem states that the optimal cost-threshold probability function converges when $P(s, \theta) = P(s, \theta - c_{max}), \forall s \in S$, where c_{max} is the RS-MDP largest cost [5].

In this work, we address the computational scalability problem of existing RS-MDP algorithms by proposing the first Symbolic Dynamic Programming (SDP) algorithm for risk-sensitive MDPs. We first define a factored RS-MDP that allows real action cost values and propose a new, sound and complete SDP algorithm, called RS-SPUDD.

2 FACTORED RS-MDP

We define a factored RS-MDP, where the set of states S is a vector of n state variables $\vec{X} = (X_1, ..., X_n)$ and $s \in S$ is represented by the state vector $\vec{x} = (x_1, ..., x_n)$, where $x_i \in \{0, 1\}$ is the value of variable X_i. We have the initial state $\vec{x}_I \in S$ and S_g is a finite set of state vectors $\vec{g} \in \{0, 1\}^n$. Each vector $\vec{g}$ represents an absorbing goal state.

The definition of actions $a \in A$ and the user-defined cost threshold θ_u are the same for RS-MDPs. The cost function is $C(\vec{x}, a)$ and the transition probabilities are encoded using *Dynamic Bayesian Networks (DBNs)* [3].

In factored RS-MDPs, augmented states are represented by $(\vec{x}, \theta)$, where $\vec{x} \in \{0, 1\}^n$ and $\theta \in \mathbb{R}$. The optimal cost-threshold probability (Equation 2) for a factored RS-MDP is:

$$P^*(\vec{x}, \theta) = \\ \max_{a \in A} \sum_{\vec{x}' \in S} \begin{cases} 0 & \text{if } C(\vec{x}, a) > \theta \\ T(\vec{x}'|\vec{x}, a) * P^*(\vec{x}', \theta - C(\vec{x}, a)) & \text{if } C(\vec{x}, a) \leq \theta. \end{cases} \tag{3}$$

Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), M. Dastani, G. Sukthankar, E. André, S. Koenig (eds.), July 10–15, 2018, Stockholm, Sweden.

Equation 3 can be used to iteratively approximate the optimal solution, i.e. $P^*(\vec{x}, \theta)$, and can be efficiently computed using Algebraic Decision Diagrams (ADDs) [1]. An ADD compactly represents functions parameterized by boolean variables and the main operations for ADDs are multiplication ($\otimes$), sum ($\oplus$), subtraction ($\ominus$), minimization ($\min(\cdot, \cdot)$), maximization ($\max(\cdot, \cdot)$) and sum-out ($\sum(\cdot)$).

2.1 RS-SPUDD

RS-SPUDD uses ADDs to represent: (i) the cost function for each action a, denoted by $C_{DD}(\cdot, a)$; (ii) the cost-threshold probability at iteration i for each θ, denoted by $P^i_{DD}(\cdot, \theta)$; and (iii) the transition function for a pair (X_i, a), denoted by T_{DD}. RS-SPUDD updates all states iteratively by applying the following set of equations:

$$P^{i+1}_{DD}(\vec{x}, \theta) = \max_{a \in A} Q^i_{DD}(\vec{x}, a, \theta), \text{where} \tag{4}$$

$$Q^i_{DD}(\vec{x}, a, \theta) = \sum_{\vec{x}'} \bigotimes_{j=1}^{n} T_{DD}(x'_j | pa_a(X'_j), a) \otimes W^i_{DD}(\vec{x}, a, \theta, \vec{x}') \tag{5}$$

and

$$W^i_{DD} = \begin{cases} 0 & \text{if } C_{DD}(\vec{x}, a) > \theta \\ P^*_{DD}(\vec{x}', \theta - C_{DD}(\vec{x}, a)) & \text{if } \vec{x}' \notin S_g, 0 < C_{DD}(\vec{x}, a) \leq \theta \\ P^i_{DD}(\vec{x}, \theta) & \text{if } \vec{x}' \notin S_g, C_{DD}(\vec{x}, a) = 0 \\ 1 & \text{if } \vec{x}' \in S_g, C_{DD}(\vec{x}, a) \leq \theta. \end{cases} \tag{6}$$

The main difficulty to compute Equation 6 with ADD operations comes from the 2nd case: the cost function $C_{DD}(\vec{x}, a)$ of the current state and action is a parameter of the optimal cost-threshold probability of the next state $\vec{x}'$. Thus, the computation of $P^*_{DD}(\vec{x}', \theta - C_{DD}(\vec{x}, a))$ depends on multiples $P^*_{DD}(\cdot, \theta')$, one for each previously computed (valid) value of $\theta' = \theta - C_{DD}(\vec{x}, a)$.

To access the previously computed values of $P^*_{DD}(\cdot, \theta')$ for all successors states, RS-SPUDD merges them into a single ADD, called $W^i_{DD}(\vec{x}, a, \theta, \vec{x}')$ (Equation 6), which can be efficiently constructed using the following indicator functions: (i) $Goal_{DD}$ that takes the value 1 for goal states and 0 otherwise; and (ii) a set of indicator functions $A^{c,a}_{DD}$ for each different possible cost c not greater than the current budget θ, which take the value 1 for states with cost c and 0 otherwise. Giving those indicator function we compute $W^i_{DD}(\vec{x}, a, \theta, \vec{x}')$ by performing the following operations:

$$W^i_{DD}(\vec{x}, a, \theta, \vec{x}') = \sum_{A^c_{DD}: c \leq \theta} P^*_{DD}(\vec{x}, \theta - c)' \otimes A^{c,a}_{DD}(\vec{x}),$$

where $P^*_{DD}(\cdot, \theta)'$ is the ADD $P^*_{DD}(\cdot, \theta)$ with all the variables primed to represent the optimal cost-threshold probability of the next state.

Given $W^i_{DD}(\cdot, a, \theta, \cdot)$, now we can efficiently compute the value $Q^i_{DD}(\cdot, a, \theta)$ (Equation 5) by eliminating variable by variable (applying the sum-out operation in ADDs). Once we have computed $Q^i_{DD}(\cdot, a, \theta)$ for each action, we can compute the probability $P^{i+1}_{DD}(\cdot, \theta)$ by applying the maximization operator of ADDs over the functions $Q^i_{DD}(\cdot, a, \theta)$ (Equation 4).

Furthermore, we can find the set Θ_r of all valid budget values from 0 up to θ_u of an RS-MDP, by solving the following constraint satisfaction problem (CSP):

$$d_1 * c_1 + d_2 * c_2 + \ldots + d_m * c_m \leq \theta_u, \tag{7}$$

where each c_i, $1 \leq i \leq m$, is a possible value in the cost function of the RS-MDP; and $d_i \in \mathbb{N}$, represents the number of times that we can apply an action with cost c_i. So, this expression represents all possible combinations of remaining budget, given an RS-MDP. Our proposed algorithm, RS-SPUDD, also includes the early termination condition proposed by ITVI-DP and considers values of θ belonging to Θ_r (in crescent order).

3 EMPIRICAL RESULTS

Algorithms TVI-DP, ITVI-DP (considering budgets from Eq. 7) and RS-SPUDD, were applied in two well-known planning domains: SysAdmin, proposed by Guestrin et al. (2003) and Navigation, from the International Planning Competition, where problems were modified by adding a user-defined threshold θ_u. We set $\theta_u = 500$ and the residual error as $\epsilon = 0.01$ (convergence error). For all the experiments, we used a virtual machine running with 4 processors at 3.50 GHz and 8 GB of memory.

We tested the algorithms with grid size up to 512x512 in the Navigation domain and up to 12 computers connected in a ring configuration of the SysAdmin domain. The results show that the original TVI-DP fails to give solutions quite quickly, solving problems up to 16x16 in the Navigation domain and up to 7 computers in the SysAdmin domain. The ITVI-DP shows an improvement in convergence time in both domains and an improvement in terms of scalability only for the Navigation domain, being able to solve instances with grid size up to 128x128. However, our proposed algorithm (RS-SPUDD) shows a great improvement for both domains when comparing convergence time and scalability (solving instances with grid size up to 512x512 and up to 12 computers). Also, RS-SPUDD was up to 26.2 times faster and was able to solve instances up to 10^3 times larger when compared with the original TVI-DP.

4 CONCLUSIONS

In this work, we tackle the scalability problem of existing algorithms for RS-MDPs by proposing the first Symbolic Dynamic Programming algorithm for risk-sensitive MDPs to explore the conditional independence of the transition structure over the augmented state space. Different from the original SDP algorithm, called SPUDD, our proposed algorithm RS-SPUDD: includes ADD operations to deal with continuous value budgets; optimizes cost-threshold probabilities over the augmented state space; and adds a pruning technique that solves an SCP to only consider valid budgets. Empirical results show that RS-SPUDD can outperform the previous approaches and solve problems up to 10^3 times larger.

ACKNOWLEDGMENTS

The authors would like to thank the São Paulo Research Foundation (FAPESP) for the financial support (grant # 2015/01587-0) and CAPES.

REFERENCES

[1] Ruth Iris Bahar, Erica A. Frohm, Charles M. Gaona, Gary D. Hachtel, Enrico Macii, Abelardo Pardo, and Fabio Somenzi. 1993. Algebraic decision diagrams and their applications. In *Proceedings of the International Conference on Computer-Aided Design (ICCAD)*. IEEE Computer Society Press, Los Alamitos, CA, USA, 188–191.

[2] Peng Dai and Judy Goldsmith. 2007. Topological Value Iteration Algorithm for Markov Decision Processes.. In *Proceedings of International Joint Conferences on Artificial Intelligence*, Manuela M. Veloso (Ed.). 1860–1865.

[3] Thomas Dean and Keiji Kanazawa. 1990. A model for reasoning about persistence and causation. *Computational Intelligence* 5, 3 (1990), 142–150.

[4] Ping Hou, William Yeoh, and Pradeep Varakantham. 2014. Revisiting Risk-Sensitive MDPs: New Algorithms and Results. In *Proceedings of the Twenty-Fourth International Conference on Automated Planning and Scheduling, ICAPS 2014, Portsmouth, New Hampshire, USA, June 21-26, 2014.*

[5] D. A. M. Moreira, K. V. Delgado, and L. N. de Barros. 2017. Risk-Sensitive Markov Decision Process with Limited Budget. In *Brazilian Conference on Intelligent System (BRACIS)*. 109–114.

[6] Martin L. Puterman. 1994. *Markov Decision Processes.* John Wiley and Sons, New York.

[7] Stella X Yu, Yuanlie Lin, and Pingfan Yan. 1998. Optimization Models for the First Arrival Target Distribution Function in Discrete Time. *J. Math. Anal. Appl.* 225, 1 (1998), 193 – 223.