# Linear Program Approximations for Factored Continuous-State Markov Decision Processes

Milos Hauskrecht, Branislav Kveton
Department of Computer Science &Intelligent Systems Program
University of Pittsburgh
{*milos,bkveton*}*@cs.pitt.edu*

In many realistic planning problems formulated as finite-state Markov decision processes (MDPs), the state space $\mathbf{x}$ is factorized and represented using a set of state variables $(x_0, \ldots, x_{n-1})$. In factored models, the number of states is exponential in the number of state variables, and standard dynamic programming solutions scale-up exponentially in the number of state variables. Much of the recent work in the AI community has focused on factored structured representations of finite-state MDPs and their efficient solutions. Particularly popular are approximations based on linear representations of value functions, where the value function $V(\mathbf{x})$ is expressed as a linear combination of basis functions $f_i$ over subsets of state variables $\mathbf{x}_i$

$$V(\mathbf{x}) = \sum_i w_i f_i(\mathbf{x}_i)$$

The advantage of the representation is that for a given set of basis functions, the weights $w_i$ in the linear approximation can be calculated through linear programming techniques. In fact, the whole problem can be rewritten as a linear program

$$\begin{aligned}
&\text{minimize}_{\mathbf{w}}: && \sum_i w_i 2^{n-|\mathbf{x}_i|} \sum_{\mathbf{x}_i} f_i(\mathbf{x}_i) \\
&\text{subject to:} && \sum_i w_i \left[ f_i(\mathbf{x}_i) - \gamma \sum_{\mathbf{x}'_i} P(\mathbf{x}'_i | \mathbf{x}_{i,a}, a) f_i(\mathbf{x}'_i) \right] - R(\mathbf{x}, a) \geq 0, \forall \mathbf{x}, a
\end{aligned}$$

This formulation allows us to tackle factored MDPs with large state spaces, and control policies can be obtained in reasonable time.

Factored continuous-state Markov decision process (CMDP) is a variant of factored MDPs, where the state space $\mathbf{x}$ is not discrete, but continuous. Our main contribution to this field is an extension of the ALP framework to continuous-state MDPs, which can be formulated as

$$\text{minimize}_{\mathbf{w}}: \quad \int_{\mathbf{x}} V(x) d\mathbf{x}$$

$$\text{subject to:} \qquad V(\mathbf{x}) - \gamma \int_{\mathbf{x}'} P(\mathbf{x}'|\mathbf{x}, a) V(\mathbf{x}') d\mathbf{x}' - R(\mathbf{x}, a) \geq 0, \forall \mathbf{x}, a \quad ,$$

where $V(x)$ is assumed to be decomposable as a linear combination of basis functions. Unfortunately, this formulation raises several issues. First, the integrals may not decompose along the subsets of state variables as in the case of finite-state MDPs. Second, the integrals may be improper and even not computable. Finally, an infinite number of constraints (for all values of $\mathbf{x}$ and $a$) needs to be satisfied.

An elegant decomposition of both integrals can be achieved when the state space $\mathbf{x}$ is restricted to $[0, 1]^n$. It can be shown that the linear program simplifies to

$$\begin{aligned} \text{minimize}_{\mathbf{w}}: \qquad & \sum_i w_i \int_{\mathbf{x}_i} f_i(\mathbf{x}_i) d\mathbf{x}_i \\ \text{subject to:} \qquad & \sum_i w_i F_i(\mathbf{x}, a) - R(\mathbf{x}, a) \geq 0, \forall \mathbf{x}, a \quad , \end{aligned}$$

where

$$F_i(\mathbf{x}, a) = f_i(\mathbf{x}_i) - \gamma \int_{\mathbf{x}'_i} \left( \prod_{x'_j \in \mathbf{x}'_i} P(x'_j | \mathbf{x}_{j,a}, a) \right) f_i(\mathbf{x}'_i) d\mathbf{x}'_i$$

In addition, an appropriate choice of basis functions, as is the product of factors

$$f_i(\mathbf{x}_i) = \prod_{x_j \in \mathbf{x}_i} x_j^{m_{j,i}} \quad ,$$

together with a transition model defined by Beta densities

$$P(x_j | \mathbf{x}_{j,a}, a) = Beta(x_j | g^1_{j,a}(\mathbf{x}_{j,a}), g^2_{j,a}(\mathbf{x}_{j,a})) \quad ,$$

leads to a closed form solution of both integrals. Since $\mathbf{x}$ is continuous there may exist an infinite number of constraints restricting $V(\mathbf{x})$, and none of the optimization methods that were developed for finite-state MDPs applies. The solution to this problem is to select a finite subset of constraints and use them to define the ALP. A variety of methods can be used to select the constraints, e.g. one can choose the constraints randomly or using some heuristic.

To test the performance of the ALP on CMDPs, we compare the method to two alternative approximation methods often used to solve CMDPs: the grid-based MDP (GMDP), and least-squares fit (LS) methods. In the first case the CMDP is converted into a finite state MDP with states corresponding to points on the state grid. The approximation is found by solving the new MDP. In the least-squares fit (LS), we fit the weights $\mathbf{w}$ of basis functions using a finite number of state space samples. The experimental comparison on highly factorized network architectures shows that the ALP approach easily beats GMDP, which does not take into account any factorization of the state space, and the results are only slightly worse in the terms of computation time and policy quality than those obtained for LS. These results will be reported at NIPS-03 during the main conference.

Our recent research in this field focuses on the improvement of the ALP framework for CMDPs with heuristics aimed at both computational speed-ups and improvements in the quality of solutions. One set of heuristics we have built takes advantage of local effect of actions, which is a reasonable assumption when modeling highly distributed environments. Other set of heuristics, relies on the incremental ALP (iALP) solver, which solves the linear program in several steps for different sets of constraints, while taking into account the feedback from the previous step solutions. This allowed us to achieve a several-fold speed-ups. Finally, to generate better constraints we used Monte Carlo simulation approaches that reflect the dynamics of the system and prefer constraints that cover better the regions of the state space that are visited more often.