Analytic Decision Analysis via Symbolic Dynamic Programming for Parameterized Hybrid MDPs

Shamin Kinathil
ANU and Data61, CSIRO
Canberra, ACT, Australia
shamin.kinathil@anu.edu.au

Harold Soh
University of Toronto
Toronto, ON, Canada
harold.soh@utoronto.ca

Scott Sanner
University of Toronto
Toronto, ON, Canada
ssanner@mie.utoronto.ca

Abstract

Decision analysis w.r.t. unknown parameters is a critical task in decision-making under uncertainty. For example, we may need to (i) perform inverse learning of the cost parameters of a multi-objective reward based on observed agent behavior; (ii) perform sensitivity analyses of policies to various parameter settings; or (iii) analyze and optimize policy performance as a function of policy parameters. When such problems have mixed discrete and continuous state and/or action spaces, this leads to parameterized hybrid MDPs (PHMDPs) that are often approximately solved via discretization, sampling, and/or local gradient methods (when optimization is involved). In this paper we combine two recent advances that allow for the first exact solution and optimization of PHMDPs. We first show how each of the aforementioned use cases can be formalized as PHMDPs, which can then be solved via an extension of symbolic dynamic programming (SDP) even when the solution is piecewise nonlinear. Secondly, we can leverage recent advances in non-convex solvers that require symbolic forms of the objective function for non-convex global optimization in (i), (ii), and (iii) using SDP to derive symbolic solutions for each PHMDP formalization. We demonstrate the efficacy and scalability of our optimal analytical framework on nonlinear examples of each of the aforementioned use cases.

1 Introduction

Markov Decision Processes (MDPs) are the de facto standard framework for decision theoretic planning in fully observable environments (Boutilier, Dean, and Hanks 1999). Traditional MDP solution techniques often assume that the parameters of the model are known. However, in practice, model parameters are usually estimated from limited data or elicited from humans and are naturally uncertain. Hence decision analysis w.r.t. unknown parameters is a critical task in decision-making under uncertainty with applications to: (i) inverse learning of parameters of multi-objective rewards; (ii) sensitivity analyses of policies to various parameter settings; and (iii) analyzing and optimizing policy performance as a function of policy parameters. Formalizing models to address each of the aforementioned use cases is often fraught, due to the specification leading to hybrid (mixed discrete and continuous state and/or action) MDPs with nonlinear and/or piecewise structure that have been traditionally very difficult to solve.

In this paper we make the following key contributions:

- We present *Parameterized Hybrid MDPs* (PHMDPs) as a unified model of the aforementioned use cases and provide an algorithm that solves PHMDPs exactly and in closed-form by defining a parameterized variant of Symbolic Dynamic Programming (SDP) (Boutilier, Reiter, and Price 2001) extended to hybrid MDPs (Sanner, Delgado, and Nunes de Barros 2011).
- We provide the *first* completely symbolic encodings of the aforementioned use cases, which in turn enables the use of recent advances in symbolic non-convex optimization techniques with *guarantees* (Gao, Kong, and Clarke 2013).
- We present the *first exact* symbolic analysis of vaccination policies in an SIR epidemiological model (Kermack and McKendrick 1927), as well exact solutions to the inverse learning of parameters in a multi-objective reward domain and sensitivity analyses of portfolio execution strategies.

2 Related Work

In this section we briefly survey prior art in the areas of multi-objective reasoning, exact sensitivity analysis and nonlinear parameterized policy optimization and conclude with a discussion of alternate uses of the term *parameterized* in the MDP literature that contrasts with our work.

The techniques used to solve Multi-objective MDPs (MOMDPs) with unknown preferences depend on the nature of the scalarization function used to weight each reward component (Roijers et al. 2013). Methods such as the Convex Hull Value Iteration algorithm (Barrett and Narayanan 2008) can be used for discrete *enumerated state* MOMDPs with any linear preference function. Nonlinear scalarization functions require the calculation of the Pareto front, which can be prohibitively large. As a result, Pareto front approximation techniques such as those of (Chatterjee, Majumdar, and Henzinger 2006) and (Pirotta, Parisi, and Restelli 2015) or Lorenz optimal refinements such as (Perny et al. 2013) are often used. In this work we present the first exact *factored hybrid* MOMDP solutions as a symbolic function of multiobjective weights via PHMDPs and SDP.

To date, most research into sensitivity analysis of MDP parameters has focused on uncertainty within the specification of the transition function (Kalyanasundaram, Chong, and Shroff 2004), reward function (Tan and Hartman 2011), or a combination of both (Givan, Leach, and Dean 2000), in discrete MDPs. The framework that we introduce in this paper enables *exact* sensitivity analysis for PHMDPs that allows it to be applied in continuous state settings and permits the derivation and analysis of the *optimal* policy as a symbolic function of these parameters.

Policy gradient methods rely upon optimizing parameterized policies with respect to the expected return by gradient descent. Two of the most prominent approaches have been the finite-difference methods, such as those of (Ng and Jordan 2000), and Monte Carlo methods, such as (Sutton et al. 1999; Baxter and Bartlett 2000), both of which only converge to local optima. Our use of PHMDPs and SDP allows us to solve for a globally optimal policy as a parameterized function of policy parameters.

Finally, as a point of differentiation from other uses of the term *parameterized* in the MDP literature, we remark that other works (Doshi-Velez and Konidaris 2016; Duff 2002; Dearden, Friedman, and Andre 1999; Gopalan and Mannor 2015) have used Parameterized MDP to refer to MDPs with latent parameters whose beliefs can be updated by observing reward and transition samples. In contrast, in this work we assume strict uncertainty of continuous MDP parameters in models that are otherwise fully specified; in this way we can treat parameters simply as free variables that can be parametrically analyzed via recent advances in symbolic solution methods and non-convex optimizers (Gao, Kong, and Clarke 2013).

3 Parameterized Hybrid MDPs

In this section we introduce Parameterized Hybrid Markov Decision Processes (PHMDPs).

3.1 Definition

A parameterized hybrid Markov Decision Process (PHMDP) is defined by the tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \mathcal{H}, \gamma, \theta \rangle$. $\mathcal{S}$ specifies a vector of states given by $(\vec{d}, \vec{x}) = (d_1, \ldots, d_m, x_1, \ldots, x_n)$, where each $d_i \in \{0, 1\}$ $(1 \leq i \leq m)$ is discrete and each $x_j \in \mathbb{R}$ $(1 \leq j \leq n)$ is continuous. $\mathcal{A}_s^h$ specifies a finite set of state and horizon dependent actions. $\vec{\theta}$ are free parameters from the parameter space Θ. PHMDPs are naturally factored (Boutilier, Dean, and Hanks 1999) in terms of the state variables $\vec{d}$ and $\vec{x}$. Hence, the joint transition model can be written as:

$$\mathcal{T} : \mathbb{P}\left(\vec{d}', \vec{x}' \middle| \vec{d}, \vec{x}, a, \vec{\theta}\right) = \prod_{i=1}^{m} \mathbb{P}\left(d_i' \middle| \vec{d}, \vec{x}, a, \vec{\theta}\right) \prod_{j=1}^{n} \mathbb{P}\left(x_j' \middle| \vec{d}, \vec{d}', \vec{x}, a, \vec{\theta}\right), \quad (1)$$

where $a \in \mathcal{A}_s^h$. The transition model permits discrete noise in the sense that $\mathbb{P}\left(x_j' | \vec{d}, \vec{d}', \vec{x}, a, \vec{\theta}\right)$ may condition on $\vec{d}'$, which are stochastically sampled according to their conditional probability functions. We note that this framework can be extended to Dynamic Bayesian Networks with arbitrary intermediate variable layers that allow one to emulate synchronic arc dependencies and relax the discrete and continuous stratifications.

$\mathcal{R} : \mathcal{S} \times \mathcal{A} \times \vec{\theta} \to \mathbb{R}$ is the reward function which encodes the preferences of the agent. $\mathcal{H}$ represents the number of decision steps until termination and the discount factor $\gamma \in [0, 1)$ is used to geometrically discount future rewards. A policy $\pi : \mathcal{S} \times \mathcal{H} \times \vec{\theta} \to \mathcal{A}$, specifies the action to take in every state and horizon. The value function of the optimal policy π^* satisfies:

$$V^{\pi^*}\left(\vec{d}, \vec{x}; \vec{\theta}\right) = \max_{a \in \mathcal{A}} \left\{ Q^{\pi}\left(\vec{d}, \vec{x}, a; \vec{\theta}\right) \right\}. \quad (2)$$

$Q^{\pi}\left(\vec{d}, \vec{x}, a; \vec{\theta}\right)$ gives the expected return starting from state $(\vec{d}, \vec{x}) \in \mathcal{S}$, taking action $a \in \mathcal{A}_s^h$, and then following policy π. In general, an agent's objective is to find an optimal policy π^* which maximises the expected sum of discounted rewards over horizon $\mathcal{H}$[1]. We again remark that in our formulation of PHMDPs $\vec{\theta}$ are free parameters and not learned from reward and transition samples.

4 Parameterized Symbolic Dynamic Programming

Symbolic Dynamic Programming (SDP) (Boutilier, Reiter, and Price 2001) is the process of performing dynamic programming via symbolic manipulation. In the following sections we present a brief overview of SDP operations and how it can be adapted to solve PHMDPs.

4.1 Symbolic Case Calculus

SDP assumes that all functions can be represented in case statement form (Boutilier, Reiter, and Price 2001):

$$f = \begin{cases} \phi_1 : & f_1 \\ \vdots & \vdots \\ \phi_k : & f_k \end{cases}$$

Here, f_i are nonlinear expressions over $(\vec{d}, \vec{x}, \vec{\theta})$ and ϕ_i are logical formulae defined over $(\vec{d}, \vec{x}, \vec{\theta})$ that can consist of arbitrary logical combinations of tests on $\vec{d}$ and inequalities $(\geq, >, <, \leq)$ over nonlinear expressions of $(\vec{x}, \vec{\theta})$. We assume that the set of conditions $\{\phi_1, \ldots, \phi_k\}$ disjointly and exhaustively partition $(\vec{d}, \vec{x}, \vec{\theta})$ such that f is well-defined for all $(\vec{d}, \vec{x}, \vec{\theta})$. Henceforth, we refer to functions with linear ϕ_i and piecewise linear f_i as linear piecewise linear (LPWL) and functions with nonlinear ϕ_i and piecewise nonlinear f_i as nonlinear piecewise nonlinear (NPWN) functions.

Operations on case statements may be either unary or binary. Unary operations on a single case statement f, such as scalar multiplication $c \cdot f$ where $c \in \mathbb{R}$, are applied to each f_i $(1 \leq i \leq k)$. Binary operations such as addition, subtraction and multiplication are executed in two stages. Firstly, the cross-product of the logical partitions of each case statement is taken, producing paired partitions. Finally, the binary operation is applied to the resulting paired partitions. The "cross-sum" $\oplus$ operation can be performed on two cases:

[1] All of the code can be found at https://github.com/skindev/xadd-inference-1/src/cmomdp.

$$\begin{cases}\phi_1 : f_1\\ \phi_2 : f_2\end{cases} \oplus \begin{cases}\psi_1 : g_1\\ \psi_2 : g_2\end{cases} = \begin{cases}\phi_1 \wedge \psi_1 : & f_1 + g_1\\ \phi_1 \wedge \psi_2 : & f_1 + g_2\\ \phi_2 \wedge \psi_1 : & f_2 + g_1\\ \phi_2 \wedge \psi_2 : & f_2 + g_2\end{cases}$$

"cross-subtraction" $\ominus$ and "cross-multiplication" $\otimes$ are defined in a similar manner but with the addition operator replaced by the subtraction and multiplication operators, respectively. Some partitions resulting from case operators may be inconsistent and are thus removed. All of the operations presented thus far are closed-form for NPWN functions (Sanner, Delgado, and Nunes de Barros 2011).

A case statement can be maximized with respect to a continuous parameter y as $f_1(\vec{x}) = \max_y f_2(\vec{x}, y)$. Continuous maximization is used for continuous $\mathcal{A}$ PHDMPs and is closed-form for LPWL functions; maximization of discrete $\mathcal{A}$ remains closed-form for all NPWN functions. We refer the reader to (Sanner, Delgado, and Nunes de Barros 2011; Zamani, Sanner, and Fang 2012) for more details.

In principle, case statements can be used to represent all PHMDP components. In practice, case statements are implemented using a more compact representation known as Extended Algebraic Decision Diagrams (XADDs) (Sanner, Delgado, and Nunes de Barros 2011), which also support efficient versions of all of the aforementioned operations.

4.2 SDP for Parameterized Hybrid MDPs

Value iteration (VI) (Bellman 1957) can be modified to solve PHMDPs in terms of the following case operations:

$$Q^h\left(\vec{d}, \vec{x}, a; \vec{\theta}\right) = \mathcal{R}\left(\vec{d}, \vec{x}, a; \vec{\theta}\right) \oplus \gamma$$

$$\bigoplus_{\vec{d'}} \int_{\vec{x'}} \mathbb{P}\left(\vec{d'}, \vec{x'} \middle| \vec{d}, \vec{x}, a; \vec{\theta}\right) \otimes V^{h-1}\left(\vec{d'}, \vec{x'}; \vec{\theta}\right) \, d\vec{x'} \quad (3)$$

$$V^h\left(\vec{d}, \vec{x}; \vec{\theta}\right) = \text{casemax}_{a \in \mathcal{A}} \left\{ Q^h\left(\vec{d}, \vec{x}, a; \vec{\theta}\right) \right\} \quad (4)$$

$\mathbb{P}\left(\vec{d'}, \vec{x'} \middle| \vec{d}, \vec{x}, a; \vec{\theta}\right)$ is specified in Equation (1). The parameters θ_i are stationary free variables and hence do not change during the backup operation. Continuous state parameters $\vec{x}$ are handled in a similar fashion. Symbolic integration over continuous variables are carried out with respect to a deterministic Dirac δ function. This is a consequence of the discrete noise restriction mentioned in section 3.1 and yields a closed-form backup operation even with NPWN $\mathcal{T}$ or $\mathcal{R}$ components (Sanner, Delgado, and Nunes de Barros 2011).

A particular strength of SDP is that all operations will automatically condition the value and policy on the θ_i, without needing to know their value a priori, yielding the parameterized value function in Equation (4).

In the case of discrete $\mathcal{A}$ it can be proved that all of the SDP operations used in Equations (3) and (4) are closed-form for NPWN functions (Sanner, Delgado, and Nunes de Barros 2011). In the case of continuous $\mathcal{A}$ all of the operations are closed-form for only LPWL functions (Zamani, Sanner, and Fang 2012).

Inverse Learning for Multi-objective PHMDPs A possible formulation for the inverse learning problem for multi-objective MDPs is to constrain the Q-values corresponding to the observed behavior and maximize $\vec{\theta}$, which can be interpreted as multi-objective weights that best explain the observed behavior:

$$\max_{\vec{\theta}} \max_{a_k, a_k \neq \pi} Q^h\left(\vec{d}, x, a_k; \vec{\theta}\right) \ominus Q^h\left(\vec{d}, x, a_{-k}; \vec{\theta}\right), \quad (5)$$

where x can either be fixed or a region specified in the constraints, a_k refers to the action taken under the policy π in a particular state and a_{-k} refers to all other available actions in that state. We note that Equation (5) is one of many possible formulations to inverse reinforcement learning and refer the reader to (Ng and Russell 2000) for alternate approaches.

Sensitivity Analysis for PHMDPs Sensitivity analysis for PHMDPs can be analysed exactly and in closed-form via SDP by first calculating Equation (4) and then taking symbolic derivatives, up to any order, with respect to $\vec{\theta}$.

Nonlinear Parameterized Policy Optimization Methods for PHMDPs Parameterized policies $\pi(\vec{\theta})$ for PHMDPs, where $\vec{\theta}$ may be nonlinear, can be analyzed exactly and in closed-form via SDP by substituting $\pi(\vec{\theta})$ in for a in Equation (3). This precludes the need for action maximization in Equation (4) and makes SDP efficient in both computation time and space. The parametric nature of this function allows us to directly apply non-convex optimization tools that require symbolic forms of the objective function. This yields a global optimization of the function in contrast to policy gradient methods which only guarantee local optimization.

5 Results

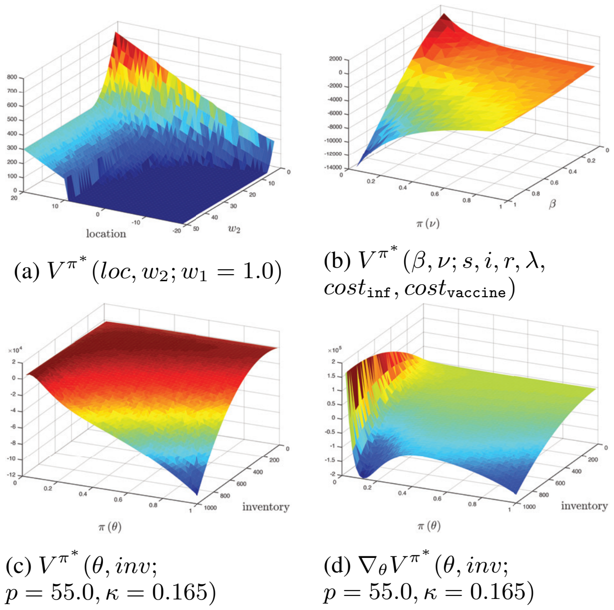
In this section we demonstrate the efficacy and tractability of PHMDPs by calculating the first known optimal solutions to three difficult nonlinear sequential decision problems. We note that while dOp (Gao, Kong, and Clarke 2013) offers strong δ-optimality guarantees, we found that nonlinear solvers such as `fmincon` (The MathWorks Inc. 2015), an interior-point algorithm, perform comparably well and are much more efficient, hence we use `fmincon`.

5.1 Inverse Learning for Navigation

The domain is specified as follows: $\mathcal{S} = \langle loc \rangle$, where loc is the location of the vehicle. $\mathcal{A} \in \{0.0, 5.0\}$ is the amount by which vehicle moves relative to its current location. $\mathcal{T}(loc' | loc, a) = \delta\left[loc' - (loc + a)\right]$, where $a \in \mathcal{A}$. $\mathcal{R}(\vec{w}, loc, loc') = w_1 \cdot \mathcal{R}_{\mathtt{region}} + w_2 \cdot \mathcal{R}_{\mathtt{move}}$ where,

$$\mathcal{R}_{\mathtt{region}}(loc') = \begin{cases}(loc' \geq 10.0) : & loc'\\ \text{otherwise} : & 0.0\end{cases} \qquad \mathcal{R}_{\mathtt{move}}(loc, loc') = -|loc' - loc|$$

Figure 1a shows the optimal value function at $\mathcal{H} = 15$ and reveals the trade-off between reaching the goal region and incurring a movement cost $w_2 \cdot \mathcal{R}_{\mathtt{move}}$, when $w_2 \in [0.0, 50.0]$. The vehicle will incur the movement cost as long as it is mitigated by the $\mathcal{R}_{\mathtt{region}}$ reward. Furthermore, the range of acceptable non-zero movement costs decreases the further the vehicle is from the goal region. In Figure 2a we utilise Equation (5) to learn the parameters (weights) of the multi-objective reward under the following sub-optimal policy: $\tilde{\pi}(0 < loc < 10) = 5.0, \tilde{\pi}(loc < 0 \, \text{or} \, loc > 10) = 0.0$. We



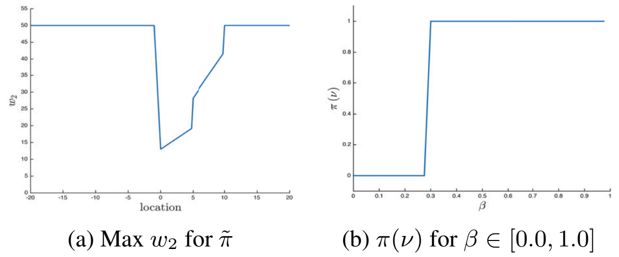
(a) $V^{\pi^*}(loc, w_2; w_1 = 1.0)$

(b) $V^{\pi^*}(\beta, \nu; s, i, r, \lambda, cost_{\texttt{inf}}, cost_{\texttt{vaccine}})$

(c) $V^{\pi^*}(\theta, inv; p = 55.0, \kappa = 0.165)$

(d) $\nabla_\theta V^{\pi^*}(\theta, inv; p = 55.0, \kappa = 0.165)$

Figure 1: Optimal Value functions for each domain.



(a) Max w_2 for $\tilde{\pi}$

(b) $\pi(\nu)$ for $\beta \in [0.0, 1.0]$

Figure 2: Nonlinear optimization for Navigation and SIR.

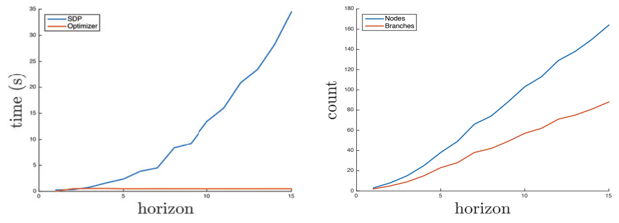


Figure 3: Time and Space versus $\mathcal{H}$ for Navigation.

observe that when $a = 0.0$, $w_2 = 50.0$, its maximum value. When $a = 5.0$, there are two different gradients for w_2, one when $(0 < loc < 5)$ and another when $(5 < loc < 10)$. The steeper gradient in the latter region indicates that being one step closer to the goal region allows the vehicle to accumulate an additional $\mathcal{R}_{\texttt{region}}$ reward over $\mathcal{H}$.

5.2 SIR Epidemic

The well studied SIR epidemic (Kermack and McKendrick 1927) domain is specified as follows: $\mathcal{S} = \langle s, i, r \rangle$, where s, i, and r refer to the size of the susceptible, infected and recovered sub-populations, respectively. $\mathcal{A} \in \{\pi(\nu)\}$ where $\nu \in [0.0, 1.0]$ is the proportion of s to vaccinate at each stage. The transition function $\mathcal{T}$ for each state variable in $\mathcal{S}$ is given by:

$$\begin{aligned}
\mathcal{T}(s'|s,i,r,\pi(\nu)) &= \delta\left[s' - (s - \beta \cdot s \cdot i - \pi(\nu) \cdot s)\right] \\
\mathcal{T}(i'|s,i,r,\pi(\nu)) &= \delta\left[i' - (i + \beta \cdot s \cdot i - \lambda \cdot i)\right] \\
\mathcal{T}(r'|s,i,r,\pi(\nu)) &= \delta\left[r' - (r + \lambda \cdot i + \pi(\nu) \cdot s)\right]
\end{aligned}$$

where β is the infection rate and λ is the spontaneous recovery rate. The reward is specified as $\mathcal{R}(cost_{\texttt{inf}}, cost_{\texttt{vaccine}}, s, i, r, \pi(\nu)) = (s \cdot (-cost_{\texttt{vaccine}} \cdot \pi(\nu) + (1 - \pi(\nu)))) - cost_{\texttt{inf}} \cdot i + r$. $cost_{\texttt{inf}}$ is the incident cost of infection and $cost_{\texttt{vaccine}}$ is the unit cost of vaccination. We assume that the total population is constant and that vaccinated individuals go straight from s to r without being infected. The decision maker must balance the cost of vaccination and the burden of disease on the population.

Figure 1b shows the optimal value function at $\mathcal{H} = 7$ when $s = 1000.0, i = 100.0, r = 0.0, \lambda = 0.25, cost_{\texttt{vaccine}} = 4.0$ and $cost_{\texttt{inf}} = 10.0$. The value function shows that it is not always optimal to vaccinate the entire population. In fact, Figure 2b reveals that vaccinating the entire population is only optimal when $\beta > 0.25$, that is, when the *basic reproductive ratio* $R_0 \,(= \beta/\lambda)$ (Heffernan, Smith, and Wahl 2005) exceeds 1.0. Scenarios where $R_0 > 1.0$ can lead to an epidemic.

To the best of our knowledge, this is the *first* exact symbolic analysis of vaccination policies in an SIR model. Furthermore, PHMDPs and SDP can be used to solve *any* SIR model without needing an analytical solution.

5.3 Optimal Execution

The domain is specified as follows $\mathcal{S} = \langle p, inv \rangle$, where p is the price of the asset and inv is the inventory remaining. $\mathcal{A} \in \{\pi(\theta)\}$, where $\theta \in (0.0, 1.0)$ is the proportion of inventory to be sold. The transition function $\mathcal{T}$ for each state variable in $\mathcal{S}$ is given by:

$$\begin{aligned}
\mathcal{T}(p'|p, inv, \pi(\theta)) &= \delta\left[p' - (p - \kappa \cdot (inv \cdot \pi(\theta)) + \epsilon)\right] \\
\mathcal{T}(inv'|p, inv, \pi(\theta)) &= \delta\left[inv' - (inv - inv \cdot \pi(\theta))\right]
\end{aligned}$$

where $\kappa > 0$ is a market-impact parameter and ϵ is a discrete noise parameter. The reward is specified by $\mathcal{R}(p', inv, \pi(\theta)) = p' \cdot inv \cdot \pi(\theta)$. When transacting a large number of shares, investors often face a trade-off between adverse market impact and the volatility of slow execution.

Figures 1c and 1d show the optimal value function at $\mathcal{H} = 10$ and its derivative w.r.t the parameter θ, respectively. When inventory is low, the value function is high at higher θ and the corresponding derivative is relatively stable. When the inventory is high, the value function is high at lower θ and the corresponding derivative shows maximum sensitivity. This indicates that when inventory is low, high θ allows the investor to capture the current price and when inventory is high, lower θ captures a more stable set of future prices.

5.4 Time and Space Complexity

Figure 3 shows an approximate linear relationship between the horizon $\mathcal{H}$ and the computational time and space for the Navigation domain, which is a promising scalability property of the overall framework.

References

Barrett, L., and Narayanan, S. 2008. Learning All Optimal Policies with Multiple Criteria. In Cohen, W. W.; McCallum, A.; and Roweis, S. T., eds., *Proceedings of the 25th International Conference on Machine Learning*, volume 307 of *ACM International Conference Proceeding Series*, 41–47. New York, NY, USA: ACM.

Baxter, J., and Bartlett, P. L. 2000. Direct Gradient-based Reinforcement Learning. In *Circuits and Systems*, volume 3, 271–274. IEEE.

Bellman, R. E. 1957. *Dynamic Programming*. Princeton, NJ: Princeton University Press.

Boutilier, C.; Dean, T.; and Hanks, S. 1999. Decision-Theoretic Planning: Structural Assumptions and Computational Leverage. *Journal of Artificial Intelligence Research* 11:1–94.

Boutilier, C.; Reiter, R.; and Price, B. 2001. Symbolic Dynamic Programming for First-order MDPs. In Bernhard Nebel., ed., *Proceedings of the 17th International Joint Conference on Artificial intelligence*, 690–697. Morgan Kaufmann Publishers Inc.

Chatterjee, K.; Majumdar, R.; and Henzinger, T. A. 2006. *Markov Decision Processes with Multiple Objectives*. Berlin, Heidelberg: Springer Berlin Heidelberg. 325–336.

Dearden, R.; Friedman, N.; and Andre, D. 1999. Model Based Bayesian Exploration. In Kathryn B. Laskey and Henri Prade., ed., *Proceedings of the 15th Conference on Uncertainty in Artificial Intelligence*, UAI'99, 150–159. Morgan Kaufmann Publishers Inc.

Doshi-Velez, F., and Konidaris, G. 2016. Hidden Parameter Markov Decision Processes: A Semiparametric Regression Approach for Discovering Latent Task Parametrizations. In Kambhampati, S., ed., *Proceedings of the 25th International Joint Conference on Artificial Intelligence*, 1432–1440. IJCAI/AAAI Press.

Duff, M. O. 2002. *Optimal Learning: Computational Procedures for Bayes-adaptive Markov Decision Processes*. Ph.D. Dissertation, University of Massachusetts Amherst.

Gao, S.; Kong, S.; and Clarke, E. M. 2013. *dReal: An SMT Solver for Nonlinear Theories over the Reals*. 208–214.

Givan, R.; Leach, S.; and Dean, T. 2000. Bounded-parameter Markov Decision Processes. *Artificial Intelligence* 122(12):71–109.

Gopalan, A., and Mannor, S. 2015. Thompson Sampling for Learning Parameterized Markov Decision Processes. In Grünwald, P.; Hazan, E.; and Kale, S., eds., *Proceedings of The 28th Conference on Learning Theory*, volume 40 of *JMLR Workshop and Conference Proceedings*, 861–898. JMLR.org.

Heffernan, J.; Smith, R.; and Wahl, L. 2005. Perspectives on the Basic Reproductive Ratio. *Journal of The Royal Society Interface* 2(4):281–293.

Kalyanasundaram, S.; Chong, E. K. P.; and Shroff, N. B. 2004. Markov Decision Processes with Uncertain Transition Rates: Sensitivity and Max-min Control. *Asian Journal of Control* 6(2):253–269.

Kermack, W. O., and McKendrick, A. G. 1927. A Contribution to the Mathematical Theory of Epidemics. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences* 115(772):700–721.

Ng, A. Y., and Jordan, M. I. 2000. PEGASUS: A policy search method for large MPDs and POMDPs. In Didier Dubois., ed., *Proceedings of the 16th Conference in Uncertainty in Artificial Intelligence*, volume 15 of *UAI '00*, 406–415. Morgan Kaufmann.

Ng, A. Y., and Russell, S. J. 2000. Algorithms for Inverse Reinforcement Learning. In Pat Langley., ed., *Proceedings of the 17th International Conference on Machine Learning*, ICML '00, 663–670. Morgan Kaufmann.

Perny, P.; Weng, P.; Goldsmith, J.; and Hanna, J. P. 2013. Approximation of Lorenz-Optimal Solutions in Multiobjective Markov Decision Processes. In *Workshops at the 27th AAAI Conference on Artificial Intelligence*.

Pirotta, M.; Parisi, S.; and Restelli, M. 2015. Multi-Objective Reinforcement Learning with Continuous Pareto Frontier Approximation. In Bonet, B., and Koenig, S., eds., *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, 2928–2934. AAAI Press.

Roijers, D. M.; Vamplew, P.; Whiteson, S.; and Richard Dazeley. 2013. A Survey of Multi-Objective Sequential Decision-Making. *Journal of Artificial Intelligence Research* 48:67–113.

Sanner, S.; Delgado, K.; and Nunes de Barros, L. 2011. Symbolic Dynamic Programming for Discrete and Continuous State MDPs. In Cozman, F. G., and Pfeffer, A., eds., *Proceedings of the 27th Conference on Uncertainty in Artificial Intelligence*, 1–10. AUAI Press.

Sutton, R. S.; McAllester, D. A.; Singh, S. P.; and Mansour, Y. 1999. Policy Gradient Methods for Reinforcement Learning with Function Approximation. In Solla, S.; Leen, T.; and Müller, K., eds., *Proceedings of the 12th International Conference on Neural Information Processing Systems*, 1057–1063. Cambridge, MA, USA: MIT Press.

Tan, C. H., and Hartman, J. C. 2011. Sensitivity Analysis in Markov Decision Processes with Uncertain Reward Parameters. *Journal of Applied Probability* 48(4):954–967.

The MathWorks Inc. 2015. *MATLAB version 8.5.197613 (R2015a) and Optimization Toolbox 7.2*. Natick, Massachusetts, United States: The MathWorks Inc.

Zamani, Z.; Sanner, S.; and Fang, C. 2012. Symbolic Dynamic Programming for Continuous State and Action MDPs. In Jörg Hoffmann and Bart Selman., ed., *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, 1–7. AAAI Press.