Robust partially observable Markov decision process

Takayuki Osogami OSOGAMI@JP.IBM.COM
IBM Research - Tokyo, Tokyo, Japan

Abstract

We seek to find the robust policy that maximizes the expected cumulative reward for the worst case when a partially observable Markov decision process (POMDP) has uncertain parameters whose values are only known to be in a given region. We prove that the robust value function, which represents the expected cumulative reward that can be obtained with the robust policy, is convex with respect to the belief state. Based on the convexity, we design a value-iteration algorithm for finding the robust policy. We prove that our value iteration converges for an infinite horizon. We also design point-based value iteration for fining the robust policy more efficiency possibly with approximation. Numerical experiments show that our point-based value iteration can adequately find robust policies.

1. Introduction

The partially observable Markov decision process (POMDP) has been studied extensively as a model of sequential decision making under stochastic and partially observable environment (Shani et al., 2013; Young et al., 2013). The POMDP has a wide range of applications including the systems of spoken dialog (Young et al., 2009), fault recovery (Shani & Meek, 2009), human-robot interaction (Pineau et al., 2003b), and the assistance of persons (Hoey et al., 2010). In these applications, parameters of a POMDP are often set manually by experts (Hoey et al., 2010; Pineau et al., 2003b; Young et al., 2009). There also exist systematic methods for learning parameters of a POMDP (Makino & Takeuchi, 2012; Shani et al., 2005; Shani & Meek, 2009), but precise estimation is generally difficult. The robustness against the uncertainties in the parameters of a POMDP is of paramount importance to open up new but critical applications such as finance.

Proceedings of the 32^{nd} International Conference on Machine Learning, Lille, France, 2015. JMLR: W&CP volume 37.

For a (fully observable) Markov decision process (MDP), the prior work has investigated ways to obtain the (optimally) *robust policy* that maximizes the expected cumulative reward for the worst case when the values of the parameters of the MDP have uncertainties and are only known to be in a given set called an uncertainty set (Nilim & El Ghaoui, 2005; Osogami, 2012). Although a POMDP can be considered as an MDP with the continuous state space consisting of belief states (Smallwood & Sondik, 1973), existing computational procedures for finding the robust policy for an MDP rely on the finiteness of the state space (Nilim & El Ghaoui, 2005).

A key property of the POMDP is that the value function is piecewise linear and convex (PWLC) with respect to the belief state when decisions are made over a finite horizon (Smallwood & Sondik, 1973). Computing the value function is essentially equivalent to finding the optimal policy, and the prior work has exploited the PWLC property for finding the optimal policy either exactly (Monahan, 1982; Smallwood & Sondik, 1973) or approximately (Kurniawati et al., 2009; Pineau et al., 2003a; Smith & Simmons, 2004).

When the parameters of the POMDP are only known to lie in an uncertainty set, we study the *robust value function* that gives the expected cumulative reward for the worst case. Computing the robust value function directly leads to finding the robust policy.

Specifically, we prove that the robust value function is convex when the uncertainty set is convex, which is the first contribution of this paper. A lower bound on the robust value function can then be given by a PWLC function. Our proof is by induction and immediately suggests *robust value iteration* for computing the robust value function or its lower bound with dynamic programming.

We then prove that the robust value iteration converges for an infinite horizon when future reward is discounted, which is our second contribution. This means that the robust value function for an infinite horizon can be approximated arbitrarily closely with a PWLC function, which can be found via robust value iteration. The exact procedure of robust value iteration is important primarily because it gives the foundation of approximate but efficient procedures.

We design *robust point-based value iteration* (robust PBVI) for computing the robust value function approximately, which is our third contribution. The robust PBVI is directly motivated by the standard PBVI for POMDPs (Pineau et al., 2003a). The application of this point-based approach, however, can only be justified and motivated by the convexity of the robust value function, which we prove in this paper. Also, details of the robust PBVI need to be specified for computational purposes.

In particular, we show that an optimization problem appearing in the robust PBVI is convex and becomes linear program when the uncertainty set is represented by a set of linear inequalities. This includes a particularly interesting case, studied for example in Osogami (2012), where each parameter has a nominal value, and the true value can deviate from the nominal value at most by a constant factor. These specifications of the robust PBVI constitute our fourth contribution.

To choose actions based on a robust policy, the belief state needs to be updated every time observation is made. Because the parameters have uncertainty, the belief state cannot be updated with the Bayes rule, as in the standard case of no uncertainty. We design *robust belief update* for updating the belief state to be used with a robust policy, which is our fifth contribution. The robust belief update solves the optimization problem that appears in the robust PBVI. Hence, the particular structure of the uncertainty set is also desirable for efficiency of the robust belief update.

We then conduct numerical experiments of finding a robust policy by the use of the robust PBVI and executing the robust policy with robust belief update. These numerical experiments constitute our final contribution. Here, we use the robust PBVI in the framework of heuristic search value iteration (HSVI) by Smith & Simmons (2004) (i.e., *robust HSVI*). The robust HSVI is applied to *Heaven and Hell*, a standard instance of a POMDP that has been used for example in Braziunas & Boutilier (2004); Poupart (2005); Shani et al. (2007). For our purposes, we assume that the probability of erroneous observation can deviate its nominal value by up to a constant factor. We show that the robust policy is found in 24 seconds and indeed robust against uncertainty.

The rest of the paper is organized as follows. In Section 2, we show the convexity of the robust value function and design robust value iteration, whose convergence is established for an infinite horizon. In Section 3, we propose robust PBVI and discuss how we can exploit the particular structure of the uncertainty set in the robust PBVI as well as in robust belief update. Section 4 shows the results of numerical experiments. In Section 5, we discuss the related work and conclude the paper.

2. Robust value iteration

We start by considering the POMDP with a finite horizon, $[0, N]$. Let $\mathcal{S}$ be the finite set of states, let $\mathcal{A}$ be the finite set of actions, and let $\mathcal{Z}$ be the finite set of observations. Let $r(s, a)$ be the immediate reward that can be obtained by taking $a \in \mathcal{A}$ from $s \in \mathcal{S}$. Let $p_n^a(t, z|s)$ be the probability of transitioning to $t \in \mathcal{S}$ and observing $z \in \mathcal{Z}$ given that $a \in \mathcal{A}$ is taken from $s \in \mathcal{S}$ at time $n \in [0, N)$. Let $0 < \gamma \leq 1$ be the discount rate. The standard objective is to find the policy, which determines the action to take for each belief state at each time, that maximizes the expected cumulative reward, where the reward at time n is discounted by the factor of γ^n. A belief state gives the probability of being in each $s \in \mathcal{S}$. This is the probability that the decision maker believes. Let $\mathcal{B}$ be the space of the belief states, i.e. all of the probability vectors having dimension $|\mathcal{S}|$.

Here, we study the robust POMDP, where $p_n^a(\cdot, \cdot|s)$ is only known to be in an uncertainty set, $\mathcal{P}_s^a$, for each $s \in \mathcal{S}, a \in \mathcal{A}, n \in [0, N)$. Throughout, we assume that the uncertainty set, $\mathcal{P}_s^a$, is convex. Our objective is to find the robust policy that maximizes the expected cumulative reward for the worst case with this uncertainty. Let $p_n^a \in \mathcal{P}^a$ denote $p_n^a(\cdot, \cdot|s) \in \mathcal{P}_s^a, \forall s \in \mathcal{S}$. For simplicity, we assume that $\mathcal{P}_s^a$ is independent of n, but, when $N < \infty$, it is straightforward to relax this assumption in the following. Although we assume full knowledge about $r(\cdot, \cdot)$, it is straightforward to consider an uncertainty set, $\mathcal{R}_s^a$, of $r(s, a)$ for $(s, a) \in \mathcal{S} \times \mathcal{A}$. The worst case is simply given by the one in $\mathcal{R}_s^a$ that has the minimum expected value, independently for each $(s, a) \in \mathcal{S} \times \mathcal{A}$ (Osogami, 2012).

The robust value function, $V_n(\mathbf{b})$, gives the maximum expected cumulative reward that can be obtained, in the worst case, from time $n \in [0, N]$ when the belief state at time n is $\mathbf{b} \in \mathcal{B}$. For $n \in [0, N)$, the robust Bellman equation gives fundamental relation between V_n and V_{n+1}:

$$V_n(\mathbf{b}) = \max_{a \in \mathcal{A}} \min_{p_n^a \in \mathcal{P}^a} \sum_{s \in \mathcal{S}} \mathbf{b}(s) \left(r(s, a) + \gamma \sum_{t, z \in \mathcal{S} \times \mathcal{Z}} p_n^a(t, z|s)\, V_{n+1}(\mathbf{b}'_{\mathbf{b}, a, z}) \right), \quad (1)$$

where $\mathbf{b}'_{\mathbf{b}, a, z}$ is the belief state after taking $a \in \mathcal{A}$ from $\mathbf{b} \in \mathcal{B}$ and observing $z \in \mathcal{Z}$:

$$\mathbf{b}'_{\mathbf{b}, a, z}(t) = \frac{\sum_{s \in \mathcal{S}} p_n^a(t, z|s)\, \mathbf{b}(s)}{\sum_{s', t' \in \mathcal{S}^2} p_n^a(t', z|s')\, \mathbf{b}(s')}, \forall t \in \mathcal{S}. \quad (2)$$

The robust Bellman equation (1) follows directly from the Bellman equations for the POMDP (Smallwood & Sondik, 1973) and for the robust MDP (Nilim & El Ghaoui, 2005).

Algorithm 1 Robust value iteration

1: $\Lambda_N \leftarrow \{\mathbf{0}\}$
2: **for** $n \leftarrow N-1$ **to** 0 **do**
3: $\quad \Lambda_n \leftarrow$ `RobustDPbackup`(Λ_{n+1})
4: **end for**
5: **Return:** $\Lambda_n, n = 0, \dots, N-1$

We can now establish the convexity:

Theorem 1. *When $N < \infty$, the robust value function, $V_n(\mathbf{b})$, is convex with respect to $\mathbf{b}$ for each $n \in [0, N]$ as long as the uncertainty set, $\mathcal{P}_s^a$ for $s, a \in \mathcal{S} \times \mathcal{A}$, is convex.*

The proof of Theorem 1 is provided in Appendix A. We prove Theorem 1 by induction that is motivated by value iteration with incremental pruning (Monahan, 1982) for POMDPs. Robust value iteration, which we will present in the following, thus follows from the steps of the proof. Our proof applies a minimax theorem, where the convexity of the uncertainty set plays an essential role.

Robust value iteration (Algorithm 1) is equivalent to the standard value iteration for the POMDP (Monahan, 1982; Smallwood & Sondik, 1973) except Step 3. Specifically, from $n = N$ to $n = 0$, we recursively find a set of vectors, Λ_n, that represents the robust value function with

$$V_n(\mathbf{b}) = \max_{\alpha \in \Lambda_n} \left[\sum_{s \in \mathcal{S}} \alpha(s)\, \mathbf{b}(s) \right]. \tag{3}$$

Here, Step 1 initializes $\Lambda_N = \{\mathbf{0}\}$, where $\mathbf{0}$ is the vector of zeros, because $V_N(\mathbf{b}) = 0, \forall \mathbf{b} \in \mathcal{B}$. Unlike the standard POMDP, however, the set, Λ_n, is generally infinite, because the robust value function is not necessarily piece-wise linear. In the following, we will show a way to approximate the exact robust value iteration by considering only a finite subset of Λ_n. This approximate robust value iteration gives a lower bound on $V_n(\mathbf{b})$, and this lower bound is PWLC with respect to $\mathbf{b}$.

Step 3 of Algorithm 1 uses new *robust DP backup* described in Algorithm 2 to construct Λ_n from Λ_{n+1}. Analogously to the standard DP backup (Monahan, 1982; Smallwood & Sondik, 1973), Steps 4-10 of Algorithm 2 find the set of vectors, Λ_n^a, that represents the convex function, $Q^a(\mathbf{b})$, that satisfies

$$V_n(\mathbf{b}) = \max_{a \in \mathcal{A}} Q_n^a(\mathbf{b}). \tag{4}$$

Here, $Q_n^a(\mathbf{b})$ is the *robust Q-function* that represents the maximum expected cumulative reward that can be obtained from time n when $a \in \mathcal{A}$ is taken from $\mathbf{b} \in \mathcal{B}$ at time n. Hence, we can construct Λ_n via $\Lambda_n = \cup_{a \in \mathcal{A}} \Lambda_n^a$. Steps 11-12 eliminate the redundant vectors that never become the maximizers in (3), which is analogous to incremental pruning (Monahan, 1982) for POMDPs.

Algorithm 2 Robust DP backup

1: **Input:** Λ_{n+1}
2: $\Lambda_n^\star \leftarrow \emptyset$
3: **for all** $a \in \mathcal{A}$ **do**
4: $\quad \Lambda_n^a \leftarrow \emptyset$
5: $\quad$ **for all** $(\alpha_z)_{z \in \mathcal{Z}} \in (\bar{\Lambda}_{n+1})^{|\mathcal{Z}|}$ **do**
6: $\quad\quad$ **for all** $s \in \mathcal{S}$ **do**
7: $\quad\quad\quad p_n^{a,\star}(\cdot, \cdot|s) \leftarrow \underset{p_n^a(\cdot,\cdot|s) \in \mathcal{P}_s^a}{\operatorname{argmin}} \sum_{t \in \mathcal{S}} \sum_{z \in \mathcal{Z}} p_n^a(t, z|s)\, \alpha_z(t)$
8: $\quad\quad$ **end for**
9: $\quad\quad \Lambda_n^a \leftarrow \Lambda_n^a \cup \left\{ \left(r(s,a) + \gamma \sum_{t \in \mathcal{S}} \sum_{z \in \mathcal{Z}} p_n^{a,\star}(t, z|s) \alpha_z(t) \right)_{s \in \mathcal{U}} \right\}$
10: $\quad$ **end for**
11: $\quad \Lambda_n^a = \text{prune}\,(\Lambda_n^a)$
12: $\quad \Lambda_n^\star \leftarrow \text{prune}\,(\Lambda_n \cup \Lambda_n^a)$
13: **end for**
14: **Return:** $\Lambda_n^\star$

What distinguishes robust DP backup from standard DP backup (Monahan, 1982; Smallwood & Sondik, 1973) is Step 7, which finds the probability mass function, $p_n^{a,\star}(\cdot, \cdot|s)$, that gives the worst case under the given uncertainty. A simple but important remark, which follows from the use of a minimax theorem in our proof of Theorem 1, is that considering the worst case for each $s \in \mathcal{S}$ results in the worst case for any $\mathbf{b} \in \mathcal{B}$. Steps 6-8 thus find $p_n^{a,\star}(\cdot, \cdot|s)$ independently for each $s \in \mathcal{S}$. The $p_n^{a,\star}$ depends on the *robust value* (the value of the robust value function) of the next belief state, which in turn can depend on the observation, $z \in \mathcal{Z}$, that we make after taking the a. Namely, the α-vector that gives the robust value of the next belief state can depend on the z. Another consequence of the use of a minimax theorem is that, now, α-vector can be any vector in the convex hull, $\bar{\Lambda}_{n+1}$, of Λ_{n+1}.

The loop starting with Step 5 thus considers all of the ordered subsets of $|\mathcal{Z}|$ vectors in $\bar{\Lambda}_{n+1}$, where each α-vector corresponds to a $z \in \mathcal{Z}$. For each $(\alpha_z)_{z \in \mathcal{Z}}$, Step 9 constructs an α-vector, using the $p_n^{a,\star}$ that gives the worst case for that $(\alpha_z)_{z \in \mathcal{Z}}$, and includes that α-vector in Λ_n^a. The loop starting with Step 5 is generally intractable, because $\bar{\Lambda}_{n+1}$ is continuous. One can approximate this loop, for example, by replacing $\bar{\Lambda}_{n+1}$ with a finite subset, Λ_{n+1}. This approximate robust value iteration gives a lower bound on the robust value function.

Most of those $(\alpha_z)_{z \in \mathcal{Z}}$'s do not correspond to the true α-vectors that give the robust value at the next belief state. However, the α-vector constructed in Step 9 with such an $(\alpha_z)_{z \in \mathcal{Z}}$ is dominated by another and pruned in Step 11 or in Step 12.

Algorithm 3 Robust point-based DP backup

1: **Input:** $\Lambda_{n+1}, \mathcal{B}_0$
2: $\tilde{\Lambda}_n \leftarrow \emptyset$
3: **for all** $\mathbf{b} \in \mathcal{B}_0$ **do**
4: $\quad \tilde{\Lambda}_{n,\mathbf{b}} \leftarrow \emptyset$
5: $\quad$ **for all** $a \in \mathcal{A}$ **do**
6: $\quad\quad$ Solve (5) for $\mathbf{b}, a$
7: $\quad\quad$ **for all** $z \in \mathcal{Z}$ **do**
8: $\quad\quad\quad \alpha_z^\star \leftarrow$ maximizer α_z in the optimal solution of (5) that achieves (6)
9: $\quad\quad$ **end for**
10: $\quad\quad$ **for all** $s \in \mathcal{S}$ **do**
11: $\quad\quad\quad p_n^{a,\star}(\cdot,\cdot|s) \leftarrow$ minimizer $p^a(\cdot,\cdot|s)$ in the optimal solution of (5)
12: $\quad\quad\quad \alpha^\star(s) \leftarrow r(s,a) + \gamma \sum_{t\in\mathcal{S}} \sum_{z\in\mathcal{Z}} p^{a,\star}(t,z|s)\, \alpha_z^\star(t)$
13: $\quad\quad$ **end for**
14: $\quad\quad \tilde{\Lambda}_{n,\mathbf{b}} \leftarrow \tilde{\Lambda}_{n,\mathbf{b}} \cup \{\alpha^\star\}$
15: $\quad$ **end for**
16: $\quad \tilde{\Lambda}_n \leftarrow \tilde{\Lambda}_n \cup \left\{ \operatorname{argmax}_{\alpha\in\tilde{\Lambda}_{n,\mathbf{b}}} \sum_{s\in\mathcal{S}} \mathbf{b}(s)\, \alpha(s) \right\}$
17: **end for**
18: **Return:** $\tilde{\Lambda}_n$

Robust value iteration is designed for a finite horizon but can be shown to converge as $N \to \infty$ in the following sense, which we prove in Appendix B:

Theorem 2. *The robust value function, V_0, satisfying (1) converge uniformly as $N \to \infty$ if $\gamma < 1$.*

For each N, let $\Lambda^{(m)} \equiv \Lambda_{N-m}$ be the set of vectors found in the m-th iteration of the robust value iteration (Algorithm 1), where $\Lambda^{(m)}$ is independent of N. Let $V_n^{(N)}$ be the robust value function for time n when the horizon is $[0, N)$. Then $\Lambda^{(m)}$ represents $V_{N-m}^{(N)}$ for any N. Theorem 2 suggests that $V_0 \equiv V_{N-N}^{(N)}$, which is represented by $\Lambda^{(N)}$, converges as $N \to \infty$.

Theorem 2 implies that the dependency of the worst case on n disappears as $n \to \infty$. For a finite horizon, the $p_n^a \in \mathcal{P}^a$ that gives the worst case can depend on n, as is evident from the robust Bellman equation (1). Because $V_{N-m}^{(N)} \equiv V_{M-m}^{(M)}$ for any M and N, Theorem 2 implies the following: for any $N_0 > 0$, V_n for $n \in [0, N_0]$ converges to a common function as $N \to \infty$. Then the dependency of (1) on n disappears, so does the dependency of the $p_n^{a,\star}$ on n.

Robust value iteration is computationally expensive, because exact value iteration for POMDPs is computationally expensive even without the uncertainty in the parameters. The significance of the results in this section is primarily in giving the theoretical foundations for the approaches to be presented in the following and others.

3. Point-based approach

We now use the idea of the point-based approach, which has been developed for POMDPs without uncertainty (Pineau et al., 2003a), to design robust point-based value iteration (robust PBVI). We will also present the robust belief update for executing a robust policy.

3.1. Robust point-based value iteration

The point-based approach can limit the number of vectors that represent the robust value function. Specifically, we keep only those vectors in Λ_n that achieve the maximum in (3) for some $\mathbf{b}$ in a given finite set, $\mathcal{B}_0 \subset \mathcal{B}$. We can use this idea of the point-based approach, because we have proved that the robust value function is convex and can be represented with a set of vectors.

Robust PBVI replaces the robust DP backup in Algorithm 1 with the *robust point-based DP backup* shown in Algorithm 3. Here, Steps 4-16 find a single vector, α, for a given $\mathbf{b} \in \mathcal{B}_0$ and keep that vector in $\tilde{\Lambda}_n$. When the given Λ_{n+1} contains all of the vectors that represents $V_{n+1}(\cdot)$, the vector that is kept in $\tilde{\Lambda}_n$ is the $\alpha \in \Lambda_n$ that maximizes $\sum_{s\in\mathcal{S}} \mathbf{b}(s)\, \alpha(s)$. Because robust point-based DP backup is used iteratively, the given Λ_{n+1} is generally approximate.

For a given $\mathbf{b} \in \mathcal{B}_0$ and a given $a \in \mathcal{A}$, Step 6 solves the following optimization problem:

$$\begin{aligned} \text{min.} \quad & \sum_{z\in\mathcal{Z}} U(z) \\ \text{s.t.} \quad & U(z) \geq \sum_{s\in\mathcal{S}} \mathbf{b}(s) \sum_{t\in\mathcal{S}} p_n^a(t,z|s)\, \alpha_z(t), \\ & \qquad\qquad \forall \alpha_z \in \Lambda_{n+1}, \forall z \in \mathcal{Z} \\ & p_n^a(\cdot,\cdot|s) \in \mathcal{P}_s^a, \quad \forall s \in \mathcal{S}. \end{aligned} \tag{5}$$

The optimal solution of (5) is used to determine the maximizers and the minimizers in the robust Bellman equation (1). For $z \in \mathcal{Z}$, the maximizer $\alpha_z^\star \in \Lambda_{n+1}$ is the one that achieves

$$U(z) = \sum_{s\in\mathcal{S}} \mathbf{b}(s) \sum_{t\in\mathcal{S}} p_n^a(t,z|s)\, \alpha_z^\star(t) \tag{6}$$

for the optimal solution of (5). For each $z \in \mathcal{Z}$, there must exist an $\alpha_z^\star \in \Lambda_{n+1}$ that achieves (6); otherwise, the value of $U(z)$ can be decreased (so can the objective value), while satisfying the constraints in (5), which contradicts the optimality of the solution. For $s \in \mathcal{S}$, the minimizer $p_n^{a,\star}(\cdot,\cdot|s)$ is the $p_n^a(\cdot,\cdot|s)$ in the optimal solution of (5).

The $p_n^{a,\star}$ obtained from the optimal solution of (5) can be shown to be the minimizer in the robust Bellman equation (1) for the given $\mathbf{b} \in \mathcal{B}_0$ and the given $a \in \mathcal{A}$. Specifically,

(16)-(17) in Appendix A imply that

$$p_n^{a,\star} \equiv \operatorname*{argmin}_{p_n^a \in \mathcal{P}^a} \sum_{s\in\mathcal{S}} \mathbf{b}(s) \left(r(s,a) + \gamma \sum_{t,z\in\mathcal{S}\times\mathcal{Z}} p_n^a(t,z|s)\, V_{n+1}(\mathbf{b}'_{\mathbf{b},a,z}) \right) \tag{7}$$

$$= \operatorname*{argmin}_{p_n^a \in \mathcal{P}^a} \sum_{z\in\mathcal{Z}} \max_{\alpha_z \in \Lambda_{n+1}} \sum_{s\in\mathcal{S}} \mathbf{b}(s) \sum_{t\in\mathcal{S}} p_n^a(t,z|s)\, \alpha_z(t). \tag{8}$$

The last expression (8) can be represented as the optimization problem (5).

The maximizers and minimizers are extracted in Step 8 and Step 11 and used to construct a vector, $\alpha^\star$, in Step 12. This $\alpha^\star$ vector constitutes the $Q_n^a(\mathbf{b})$ in (4) such that $Q_n^a(\mathbf{b}) = \alpha^\star \cdot \mathbf{b}$, when the given Λ_{n+1} is exact. Here, for a given $\mathbf{b}$ and an a, the value of $Q_n^a(\mathbf{b})$ is given by $\sum_{s\in\mathcal{S}} \mathbf{b}(s)\, r(s,a)$ plus the optimal objective value of (5) multiplied by γ.

Because $\mathcal{P}_s^a$ is convex, (5) is a convex optimization problem. For example, consider the case studied in Osogami (2012), where $p_n^a(t,z|s)$ can range between 0 and $\hat{p}^a(t,z|s)/\kappa$ for $\kappa > 1$ (here, $\hat{p}^a(t,z|s)$ can be considered as a nominal value of $p_n^a(t,z|s)$). Then $\mathcal{P}_s^a$ is convex and can be represented with a system of linear inequalities, so that (5) reduces to the following linear program:

$$\begin{aligned}
\text{min.} \quad & \sum_{z\in\mathcal{Z}} U(z) \\
\text{s.t.} \quad & U(z) \geq \sum_{s\in\mathcal{S}} \mathbf{b}(s) \sum_{t\in\mathcal{S}} p_n^a(t,z|s)\, \alpha_z(t), \\
& \qquad \forall \alpha_z \in \Lambda_{n+1}, \forall z \in \mathcal{Z} \\
& 0 \leq p_n^a(t,z|s) \leq \frac{1}{\kappa}\, \hat{p}^a(t,z|s), \\
& \qquad \forall s,t \in \mathcal{S}^2, z \in \mathcal{Z} \\
& \sum_{t,z\in\mathcal{S}\times\mathcal{Z}} p_n^a(t,z|s) = 1, \\
& \qquad \forall s \in \mathcal{S}.
\end{aligned} \tag{9}$$

Note that (9) is linear program, because $\mathbf{b} \in \mathcal{B}_0$ is given in the robust point-based DP backup.

3.2. Robust belief update

To choose actions based on the policy that is found by the robust PBVI (or robust value iteration), we need to update the belief state. The standard belief update (2) requires the knowledge of p_n^a, which has uncertainties in our settings.

Because the robust policy is optimized for the $p_n^{a,\star}$ in (8) that gives the worst case, the belief state needs to be updated based on that $p_n^{a,\star}$. Suppose that we take $a_n \in \mathcal{A}$ at time n and observe $z_n \in \mathcal{Z}$. The robust belief update solves (5) with $\mathbf{b} = \mathbf{b}_n$ and $a = a_n$ to obtain the minimizers, $p_n^{a_n,\star}$. Then the belief state, $\mathbf{b}'_{\mathbf{b}_n,a_n,z_n}$ at time $n+1$ is obtained as

$$\mathbf{b}'_{\mathbf{b}_n,a_n,z_n}(t) = \frac{\sum_{s\in\mathcal{S}} p_n^{a_n,\star}(t,z_n|s)\, \mathbf{b}_n(s)}{\sum_{s',t'\in\mathcal{S}^2} p_n^{a_n,\star}(t',z_n|s')\, \mathbf{b}_n(s')}, \forall t \in \mathcal{S}. \tag{10}$$

Analogously to Step 6 of Algorithm 3, particular forms (e.g., (9)) of $\mathcal{P}_s^{a_n}$ can be exploited for computational efficiency.

3.3. Robust heuristic search value iteration

The robust PBVI in Section 3.1 gives a foundation of more sophisticated approaches. The state-of-the-art approaches to POMDPs (without uncertainties) iteratively update lower bounds and upper bounds on the value functions (Smith & Simmons, 2004; Kurniawati et al., 2009; Zhang et al., 2014; Brechtel et al., 2013). The lower bounds are updated with point-based value iteration, and the upper bounds are used to guide the search for the belief states to be included in $\mathcal{B}_0$. The robust PBVI can be used in the framework of these approaches.

In our numerical experiments in Section 4, we will use the robust PBVI in the framework of heuristic search value iteration (HSVI) by Smith & Simmons (2004). Namely, the lower bounds are updated with the robust PBVI in our robust HSVI. We also need to specify how the lower bounds and the upper bounds are initialized in the robust HSVI, because existing approaches do not allow uncertainties in the parameters of a POMDP. In the following, we assume infinite horizon.

The initial upper bounds can be set by first choosing an arbitrary $\dot{p}^a \in \mathcal{P}^a$ for each $a \in \mathcal{A}$ and then finding upper bounds with the assumption of $p^a = \dot{p}^a, \forall a \in \mathcal{A}$. For example, we can compute the Q-functions of a corresponding MDP (QMDP) or the *fast informed bound* (Hauskrecht, 2000) to initialize the upper bounds. These upper bounds with arbitrary p^a's are valid, because the worst case $p_n^{a,\star}$ can only lower the values.

On the other hand, lower bounds cannot be obtained by arbitrarily fixing $p^a \in \mathcal{P}^a$ for $a \in \mathcal{A}$. A popular approach to initializing lower bounds for a POMDP without uncertainties is fixed action policy (FAP) (Smith & Simmons, 2004; Kurniawati et al., 2009). We apply the idea of FAP, but additional ideas are needed due to uncertainties. Now, for a fixed $a_0 \in \mathcal{A}$, the $\underline{\mathrm{V}}$ that satisfies the following equation

gives the lower bound on the robust value function:

$$\underline{\mathrm{V}}(\mathbf{b}) = \min_{p^{a_0} \in \mathcal{P}^{a_0}} \sum_{s \in \mathcal{S}} \mathbf{b}(s) \left(r(s, a_0) + \gamma \sum_{t,z \in \mathcal{S} \times \mathcal{Z}} p^{a_0}(t, z|s)\, \underline{\mathrm{V}}(\mathbf{b}'_{\mathbf{b}, a_0, z}) \right), \quad (11)$$

because optimizing actions can only increase the values. Now, (11) can be seen as a Bellman equation for a POMDP (without uncertainties) with the objective of *minimizing* the expected cumulative reward, where each p^{a_0} is associated with an action in a possibly continuous space, $\mathcal{P}^{a_0}$. That is, $p^{a_0} \in \mathcal{P}^{a_0}$ can be seen both as an action and as the probability mass function for transition and observation given that action. The value function of a POMDP with the objective of minimization is concave, so that $\underline{\mathrm{V}}(\mathbf{b})$ is concave with respect to $\mathbf{b}$.

We want to initialize the lower bounds with a PWLC function, so that they can be used in the robust HSVI. This motivates us to compute the lower bounds on extreme belief states with FAP and linearly interpolate the extreme points. Namely, for $s \in \mathcal{S}$, let $\mathbf{e}_s$ be the extreme belief state such that $\mathbf{e}_s(s) = 1$ and $\mathbf{e}_s(s') = 0$ for $s' \neq s$. We find the lower bound on $V(\mathbf{e}_s)$ by the following Bellman equation for $s \in \mathcal{S}$:

$$\underline{\mathrm{V}}(\mathbf{e}_s) = r(s, a_0) + \gamma \min_{p^{a_0} \in \mathcal{P}^{a_0}} \sum_{t,z} p^{a_0}(t, z|s)\, \underline{\mathrm{V}}(\mathbf{e}_t). \quad (12)$$

Then we obtain the lower bound on an arbitrary $\mathbf{b} \in \mathcal{B}$ by

$$\underline{\mathrm{V}}(\mathbf{b}) = \sum_{s \in \mathcal{S}} \mathbf{b}(s)\, \underline{\mathrm{V}}(\mathbf{e}_s), \quad (13)$$

which gives a proper lower bound, because the robust value function is convex (Theorem 1). We refer to the lower bound of (13) as the *robust initial lower bound*.

4. Numerical experiments

We now apply the robust HSVI to *Heaven and Hell* (Braziunas & Boutilier, 2004; Poupart, 2005; Shani et al., 2007), a standard instance of a POMDP. We introduce uncertainties in the parameters of the POMDP and study the robustness of the policy found by the robust HSVI. We also study how the bounds are updated by the robust HSVI. Due to space limitation, we provide the results of the numerical experiments in the associated supplementary material.

In our Heaven and Hell, an agent travels the area shown in Figure 1 of the supplementary material. The agent moves one step at a time with the reward of -1 (unit cost). The agent obtains the reward of 1 upon reaching "heaven" or the reward of -10 upon reaching "hell," and terminates the travel. One of the two "?"s in Figure 1 is "heaven," and the other is "hell." The agent can observe which "?" is "heaven" at "!", but this observation is erroneous and has uncertainties. Specifically, the probability of observation error, p_e, has the nominal value of $\hat{p}_e = 0.1$ but can deviate it by a constant factor of $1/\kappa$ (i.e., $0 \leq p_e \leq \hat{p}_e/\kappa$). At the locations with numerical labels, the agent can observe the exact location without error. The state of the POMDP is the pair, (m, n), where m denotes the location of the agent, and n denotes the location of "heaven" (the *right* "?" or the *left* "?"). The initial belief state, $\mathbf{b}_0$, is that $(5, \mathrm{right})$ with probability 0.5 and $(5, \mathrm{left})$ with probability 0.5. The agent seeks to maximize the expected cumulative reward with the discount rate of $\gamma = 0.9$. When p_e is large, the agent should directly go to an arbitrary "?", because the cost of going to "!" for an observation pays off only when the observation is informative. A difficulty here is that p_e is uncertain.

5. Related work and conclusion

Our work extends the prior study on robust MDPs (Nilim & El Ghaoui, 2005; Osogami, 2012) to POMDPs, where the policy is optimized for the worst case when the parameters have uncertainties. Although this *robust POMDP* is studied for the first time, there exits the prior work that addresses the uncertainty in the parameters of POMDPs.

In particular, Bayesian reinforcement learning (RL) for a POMDP (Doshi-velez, 2009; Ross et al., 2008; 2011) assumes prior distributions over the parameters and seeks to find the policy that maximizes the expected cumulative reward based on these prior distributions. We expect that the approach of the robust POMDP is complementary to Bayesian RL, as has been the case for (fully observable) MDPs where the uncertainties are addressed both from robust MDP (Nilim & El Ghaoui, 2005; Osogami, 2012) and from Bayesian RL (Vlassis et al., 2012).

There also exists the prior work that considers the uncertainty in parameters of a POMDP but does not considers the problem of finding the robust policy, which is optimal for the worst case. For example, Itoh & Nakamura (2007) find a policy that is optimal for an arbitrary point in the uncertainty set. Ni & Liu (2013) do not explicitly state exactly what policy the proposed modified value iteration finds, but their approach is similar in spirit to Itoh & Nakamura (2007), and an arbitrary alpha-vector is selected in each step of value iteration. Ni & Liu (2012) find an optimistic policy for the best case via policy iteration.

Our work establishes the fundamental results, including the robust Bellman equation, the convexity of the robust value function, and the robust belief update, that initiate the study of robust POMDPs. The robust PBVI, the robust HSVI, and the robust initial lower bound give basic frameworks

for planning with robust POMDPs. As is discussed at the end of Section 4, there are interesting directions of future work towards efficient planning under partially observable and uncertain environment. Maximizing the expected cumulative reward for the worst case studied in Section 4 can be shown to be equivalent to maximizing the value of a certain risk measure when the parameters have the given nominal values (Osogami, 2012). This equivalence motivates the study of other risk-sensitive objectives with POMDPs, which has never been investigated in the literature.

A. Proof of Theorem 1

We prove the theorem by induction. For $n = N$, we have $V_n(\mathbf{b}) = 0$ for $\mathbf{b} \in \mathcal{B}$, which is convex with $|\Lambda_n| = 1$.

For $0 \leq n < N$, suppose that $V_{n+1}(\mathbf{b})$ is convex with respect to $\mathbf{b}$. Then there exists a possibly infinite set of vectors, Λ_{n+1}, such that

$$V_{n+1}(\mathbf{b}) = \max_{\alpha \in \Lambda_{n+1}} \left[\sum_{s \in \mathcal{S}} \alpha(s)\, \mathbf{b}(s) \right]. \tag{14}$$

Plugging (14) into (1), we obtain

$$\begin{aligned} V_n(\mathbf{b}) = & \max_{a \in \mathcal{A}} \min_{p_n^a \in \mathcal{P}^a} \sum_{s \in \mathcal{S}} \mathbf{b}(s) \Bigg(r(s,a) \\ & + \gamma \sum_{t,z \in \mathcal{S} \times \mathcal{Z}} p_n^a(t,z|s) \max_{\alpha_z \in \Lambda_{n+1}^{(a,z)}} \sum_{x \in \mathcal{S}} \mathbf{b}'_{\mathbf{b},a,z}(x) \alpha_z(x) \Bigg). \end{aligned} \tag{15}$$

Here, the value function at time $n+1$ depends on $a \in \mathcal{A}$ and $z \in \mathcal{Z}$ at time n, and this dependency is represented in the notation, $\Lambda_{n+1}^{(a,z)}$. Plugging (2) into (15), we obtain

$$\begin{aligned} V_n(\mathbf{b}) = & \max_{a \in \mathcal{A}} \min_{p_n^a \in \mathcal{P}^a} \Bigg(\sum_{s \in \mathcal{S}} \mathbf{b}(s)\, r(s,a) \\ & + \gamma \sum_{z \in \mathcal{Z}} \max_{\alpha_z \in \Lambda_{n+1}^{(a,z)}} \sum_{x,s' \in \mathcal{S}^2} p_n^a(x,z|s')\, \mathbf{b}(s')\, \alpha_z(x) \Bigg) \end{aligned} \tag{16}$$

$$\begin{aligned} = & \max_{a \in \mathcal{A}} \min_{p_n^a \in \mathcal{P}^a} \sum_{z \in \mathcal{Z}} \max_{\alpha_z \in \Lambda_{n+1}^{(a,z)}} \sum_{s \in \mathcal{S}} \mathbf{b}(s) \Bigg(\frac{r(s,a)}{|\mathcal{Z}|} \\ & + \gamma \sum_{t \in \mathcal{S}} p_n^a(t,z|s)\, \alpha_z(t) \Bigg), \end{aligned} \tag{17}$$

where we change the variables, s' and x, to s and t, respectively, in the last equality.

For a given $p_n^a(\cdot, z|s)$ and a given α_z, we can see that

$$\sum_{s \in \mathcal{S}} \mathbf{b}(s) \left(\frac{r(s,a)}{|\mathcal{Z}|} + \gamma \sum_{t \in \mathcal{S}} p_n^a(t,z|s)\, \alpha_z(t) \right) \tag{18}$$

is a linear function of $\mathbf{b}$. The maximum of linear functions is convex, and so is the sum of convex functions. Thus,

$$\sum_{z \in \mathcal{Z}} \max_{\alpha_z \in \Lambda_{n+1}^{(a,z)}} \sum_{s \in \mathcal{S}} \mathbf{b}(s) \left(\frac{r(s,a)}{|\mathcal{Z}|} + \gamma \sum_{t \in \mathcal{S}} p_n^a(t,z|s)\, \alpha_z(t) \right) \tag{19}$$

is a convex function of $\mathbf{b}$.

The minimum of convex functions is not necessarily convex, but we will see that the following is convex:

$$\begin{aligned} Q_n^a(\mathbf{b}) \equiv \min_{p_n^a \in \mathcal{P}^a} \sum_{z \in \mathcal{Z}} \max_{\alpha_z \in \Lambda_{n+1}^{(a,z)}} \sum_{s \in \mathcal{S}} \mathbf{b}(s) \Bigg(& \frac{r(s,a)}{|\mathcal{Z}|} \\ & + \gamma \sum_{t \in \mathcal{S}} p_n^a(t,z|s)\, \alpha_z(t) \Bigg). \end{aligned} \tag{20}$$

Once this convexity is established,

$$V_n(\mathbf{b}) = \max_{a \in \mathcal{A}} Q_n^a(\mathbf{b}) \tag{21}$$

is convex, because the maximum of convex functions is convex.

To see the convexity of $Q_n^a(\mathbf{b})$, exchange the summation over z and the maximum over α_z in (20) to obtain

$$\begin{aligned} Q_n^a(\mathbf{b}) \equiv \min_{p_n^a \in \mathcal{P}^a} \max_{\alpha_z \in \Lambda_{n+1}^{(a,z)}, z \in \mathcal{Z}} \sum_{z \in \mathcal{Z}} \sum_{s \in \mathcal{S}} \mathbf{b}(s) \Bigg(& \frac{r(s,a)}{|\mathcal{Z}|} \\ & + \gamma \sum_{t \in \mathcal{S}} p_n^a(t,z|s)\, \alpha_z(t) \Bigg). \end{aligned} \tag{22}$$

Now, consider a two-person zero-sum game, where the first player (maximizing $Q_n^a(\mathbf{b})$ by optimally choosing $\alpha_z, z \in \mathcal{Z}$) receives $Q_n^a(\mathbf{b})$, and the second player (minimizing $Q_n^a(\mathbf{b})$ by optimally choosing p_n^a) receives $-Q_n^a(\mathbf{b})$. By letting $\bar{\Lambda}_{n+1}^{(a,z)}$ be the convex hull of $\Lambda_{n+1}^{(a,z)}$ and letting

$$M \equiv \sum_{z \in \mathcal{Z}} \sum_{s \in \mathcal{S}} \mathbf{b}(s) \left(\frac{r(s,a)}{|\mathcal{Z}|} + \gamma \sum_{t \in \mathcal{S}} p_n^a(t,z|s)\, \alpha_z(t) \right), \tag{23}$$

Loomis' Minimax Theorem (Theorem 2.3 from Motwani & Raghavan (1995)) implies that

$$\min_{p_n^a \in \mathcal{P}^a} \max_{\alpha_z \in \Lambda_{n+1}^{(a,z)}, z \in \mathcal{Z}} M = \max_{\alpha_z \in \bar{\Lambda}_{n+1}^{(a,z)}, z \in \mathcal{Z}} \min_{p_n^a \in \mathcal{P}^a} M \tag{24}$$

if p_n^a on the left-hand side is chosen from mixed strategies, and $\alpha_z, z \in \mathcal{Z}$ on the right-hand side is chosen from mixed strategies. Our key assumption is the convexity of $\mathcal{P}_s^a$. The second player chooses p_n^a from the convex set, which can be represented with a mixed strategy. Note that an arbitrary probabilistic mixture of the transition probabilities corresponding to the extreme points in $\mathcal{P}_s^a$ is the transition probability corresponding to a point in $\mathcal{P}_s^a$. Thus, the p_n^a on

the left-hand side is chosen from mixed strategies. On the left-hand side of (24), $\alpha_z, z \in \mathcal{Z}$ is chosen from pure (deterministic) strategies. On the right-hand side of (24), we consider the convex hull $\bar{\Lambda}_{n+1}^{(a,z)}$, and this can be interpreted as choice from mixed strategies. In the expression of (23), the coefficient of α_z is a probability, so that the expression of (23) is an expectation. Thus, replacing $\alpha_z, z \in \mathcal{Z}$ with a convex combination of alpha vectors is equivalent to randomly choosing an alpha vector, following the distribution determined by the coefficients of the convex combination, and calculating the expected value.

By (24), we obtain from (20) that

$$Q_n^a(\mathbf{b}) = \max_{\alpha_z \in \bar{\Lambda}_{n+1}^{(a,z)}, z \in \mathcal{Z}} \min_{p_n^a \in \mathcal{P}^a} \sum_{z \in \mathcal{Z}} \sum_{s \in \mathcal{S}} \mathbf{b}(s) \Bigg(\frac{r(s,a)}{|\mathcal{Z}|} + \gamma \sum_{t \in \mathcal{S}} p_n^a(t,z|s)\, \alpha_z(t) \Bigg). \tag{25}$$

Because $p_n^a \in \mathcal{P}^a$ means $p_n^a(\cdot,\cdot|s) \in \mathcal{P}_s^a, \forall s \in \mathcal{S}$, we have from (25) that

$$Q_n^a(\mathbf{b}) = \max_{\alpha_z \in \bar{\Lambda}_{n+1}^{(a,z)}, z \in \mathcal{Z}} \sum_{s \in \mathcal{S}} \mathbf{b}(s) \min_{p_n^a(\cdot,\cdot|s) \in \mathcal{P}_s^a} \Bigg(r(s,a) + \gamma \sum_{t,z \in \mathcal{S} \times \mathcal{Z}} p_n^a(t,z|s)\, \alpha_z(t) \Bigg). \tag{26}$$

The last expression shows that $Q_n^a(\mathbf{b})$ can be represented with the maximum of a possibly infinite number of functions that are linear with respect to $\mathbf{b}$. Therefore, $Q_n^a(\mathbf{b})$ is convex with respect to $\mathbf{b}$, which completes the proof of the theorem.

B. Proof of Theorem 2

By the Banach fixed-point theorem (Theorem 6.2.3 from Puterman (2005)), it suffices to show that the robust Bellman operator, $\mathcal{L}$, is a contraction mapping. Here, $\mathcal{L}$ is the operator that maps $V_{n+1}(\cdot)$ to $V_n(\cdot)$ in (1). Consider two functions, V and U. Each of these functions maps a belief state to a real number.

For a fixed belief state, $\mathbf{b}$, let

$$a^\star \equiv \operatorname*{argmax}_{a \in \mathcal{A}} \min_{p^a \in \mathcal{P}^a} \sum_{s \in \mathcal{S}} \mathbf{b}(s) \Bigg(r(s,a) + \gamma \sum_{t,z \in \mathcal{S} \times \mathcal{Z}} p^a(t,z|s) V(\mathbf{b}'_{\mathbf{b},a,z}) \Bigg) \tag{27}$$

$$p^{a,\star} \equiv \operatorname*{argmin}_{p^a \in \mathcal{P}^a} \sum_{s \in \mathcal{S}} \mathbf{b}(s) \Bigg(r(s,a) + \gamma \sum_{t,z \in \mathcal{S} \times \mathcal{Z}} p^a(t,z|s)\, U(\mathbf{b}'_{\mathbf{b},a,z}) \Bigg), \forall a \in \mathcal{A}, \tag{28}$$

where $\mathbf{b}'_{\mathbf{b},a,z}$ is defined with (2). Note that $a^\star$ is defined with V, while $p^{a,\star}$ is defined with U.

Suppose that $\mathcal{L}U(\mathbf{b}) \leq \mathcal{L}V(\mathbf{b})$. Then we have

$$\begin{aligned} 0 \leq & \mathcal{L}V(\mathbf{b}) - \mathcal{L}U(\mathbf{b}) \qquad (29) \\ \leq & \sum_{s \in \mathcal{S}} \mathbf{b}(s) \Big(r(s,a^\star) + \gamma \sum_{t,z \in \mathcal{S} \times \mathcal{Z}} p^{a^\star,\star}(t,z|s) V(\mathbf{b}'_{\mathbf{b},a^\star,z}) \Big) \\ & - \sum_{s \in \mathcal{S}} \mathbf{b}(s) \Big(r(s,a^\star) + \gamma \sum_{t,z \in \mathcal{S} \times \mathcal{Z}} p^{a^\star,\star}(t,z|s) U(\mathbf{b}'_{\mathbf{b},a^\star,z}) \Big). \qquad (30) \end{aligned}$$

Here, the second inequality follows from (27) and (28). Specifically, $a^\star$ is the maximizer for V, and $p^{a,\star}$ is the minimizer for U. Hence, the first term of (30) is no smaller than $\mathcal{L}V(\mathbf{b})$, and the second term is no greater than $\mathcal{L}U(\mathbf{b})$.

Simplifying (30), we obtain

$$\begin{aligned} 0 \leq & \gamma \sum_{s \in \mathcal{S}} \mathbf{b}(s) \sum_{t,z \in \mathcal{S} \times \mathcal{Z}} p^{a^\star,\star}(t,z|s) \big(V(\mathbf{b}'_{\mathbf{b},a^\star,z}) - U(\mathbf{b}'_{\mathbf{b},a^\star,z}) \big) \\ \leq & \gamma \sum_{s \in \mathcal{S}} \mathbf{b}(s) \sum_{t,z \in \mathcal{S} \times \mathcal{Z}} p^{a^\star,\star}(t,z|s) \big| V(\mathbf{b}'_{\mathbf{b},a^\star,z}) - U(\mathbf{b}'_{\mathbf{b},a^\star,z}) \big| \\ \leq & \gamma \sum_{s \in \mathcal{S}} \mathbf{b}(s) \sum_{t,z \in \mathcal{S} \times \mathcal{Z}} p^{a^\star,\star}(t,z|s)\, ||V - U||_0 \\ = & \gamma\, ||V - U||_0, \qquad (31) \end{aligned}$$

where

$$||V - U||_0 = \sup_{\mathbf{b} \in \mathcal{B}} |V(\mathbf{b}) - U(\mathbf{b})|. \tag{32}$$

Analogously, we can establish the following when we assume $\mathcal{L}V(\mathbf{b}) \leq \mathcal{L}U(\mathbf{b})$:

$$0 \leq \mathcal{L}U(\mathbf{b}) - \mathcal{L}V(\mathbf{b}) \leq \gamma\, ||V - U||_0. \tag{33}$$

Hence, for any $\mathbf{b}$, we have

$$|\mathcal{L}V(\mathbf{b}) - \mathcal{L}U(\mathbf{b})| \leq \gamma\, ||V - U||_0. \tag{34}$$

Then

$$||\mathcal{L}V - \mathcal{L}U||_0 = \sup_{\mathbf{b} \in \mathcal{B}} |\mathcal{L}V(\mathbf{b}) - \mathcal{L}U(\mathbf{b})| \leq \gamma\, ||V - U||_0. \tag{35}$$

For $0 < \gamma < 1$, this establishes that $\mathcal{L}$ is a contraction mapping.

Acknowledgments

This research was supported by CREST, JST.

References

Braziunas, D. and Boutilier, C. Stochastic local search for POMDP controllers. In *Proceedings of the 19th National Conference on Artifical Intelligence (AAAI-04)*, pp. 690–696, July 2004.

Brechtel, S., Gindele, T., and Dillmann, R. Solving continuous POMDPs: Value iteration with incremental learning of an efficient space representation. In *Proceedings of the 30th International Conference on Machine Learning (ICML 2013)*, pp. 370–378, June 2013.

Doshi-velez, F. The infinite partially observable Markov decision process. In Bengio, Y., Schuurmans, D., Lafferty, J.D., Williams, C.K.I., and Culotta, A. (eds.), *Advances in Neural Information Processing Systems 22*, pp. 477–485. Curran Associates, Inc., 2009.

Hauskrecht, M. Value-function approximations for partially observable Markov decision processes. *Journal of Artificial Intelligence Research*, 13(1):33–94, 2000.

Hoey, J., Poupart, P., von Bertoldi, A., Craig, T., Boutilier, C., and Mihailidis, A. Automated handwashing assistance for persons with dementia using video and a partially observable Markov decision process. *Computer Vision and Image Understanding*, 114(5):503–519, 2010.

Itoh, H. and Nakamura, K. Partially observable Markov decision processes with imprecise parameters. *Artificial Intelligence*, 171(8-9):453–490, 2007.

Kurniawati, H., Hsu, D., and Lee, W. S. *SARSOP: Efficient Point-Based POMDP Planning by Approximating Optimally Reachable Belief Spaces*, pp. 65–72. MIT Press, 2009.

Makino, T. and Takeuchi, J. Apprenticeship learning for model parameters of partially observable environments. In *Proceedings of the 29th International Conference on Machine Learning*, 2012.

Monahan, G. E. A survey of partially observable Markov decision processes: Theory, models and algorithms. *Management Science*, 28:1–16, 1982.

Motwani, R. and Raghavan, P. *Randomized Algorithms*. Cambridge University Press, 1995.

Ni, Y. and Liu, Z.-Q. Policy iteration for bounded-parameter POMDPs. *Soft Computing*, 17(4):537–548, 2012.

Ni, Y. and Liu, Z.-Q. Bounded-parameter partially observable Markov decision processes: Framework and algorithm. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 21(6):821–864, 2013.

Nilim, A. and El Ghaoui, L. Robust control of Markov decision processes with uncertain transition matrices. *Operations Research*, 53(5):780–798, 2005.

Osogami, T. Robustness and risk-sensitivity in Markov decision processes. In *Advances in Neural Information Processing Systems 25*, pp. 233–241. Curran Associates, Inc., December 2012.

Pineau, J., Gordon, G., and Thrun, S. Point-based value iteration: An anytime algorithm for POMDPs. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI-03)*, pp. 1025 – 1032, August 2003a.

Pineau, J., Montemerlo, M., Pollack, M., Roy, N., and Thrun, S. Towards robotic assistants in nursing homes: Challenges and results. *Robotics and Autonomous Systems*, 42:271–281, 2003b.

Poupart, P. *Exploiting Structure to Efficiently Solve Large Scale Partially Observable Markov Decision Processes*. PhD thesis, Graduate Department of Computer Science, University of Toronto, 2005.

Puterman, M. L. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley-Interscience, 2005.

Ross, S., Chaib-draa, B., and Pineau, J. Bayes-adaptive POMDPs. In Platt, J. C., Koller, D., Singer, Y., and Roweis, S. T. (eds.), *Advances in Neural Information Processing Systems 20*, pp. 1225–1232. Curran Associates, Inc., 2008.

Ross, S., Pineau, J., Chaib-draa, B., and Kreitmann, P. A Bayesian approach for learning and planning in partially observable Markov decision processes. *Journal of Machine Learning Research*, 12:1729–1770, 2011.

Shani, G. and Meek, C. Improving existing fault recovery policies. In Bengio, Y., Schuurmans, D., Lafferty, J. D., Williams, C. K. I., and Culotta, A. (eds.), *Advances in Neural Information Processing Systems 22*, pp. 1642–1650. Curran Associates, Inc., 2009.

Shani, G., Brafman, R. I., and Shimony, S. E. Model-based online learning of POMDPs. In *Proceedings of the 16th European Conference on Machine Learning*, pp. 353–364, October 2005.

Shani, G., Brafman, R. I., and Shimony, S. E. Forward search value iteration for POMDPs. In *Proceedings of the IJCAI-07*, pp. 2619–2624, January 2007.

Shani, G., Pineau, J., and Kaplow, R. A survey of point-based POMDP solvers. *Autonomous Agents and Multi-Agent Systems*, 27(1):1–51, 2013.

Smallwood, R. D. and Sondik, E. J. The optimal control of partially observable Markov processes over a finite horizon. *Operations Research*, 21(5):1071–1088, 1973.

Smith, T. and Simmons, R. Heuristic search value iteration for pomdps. In *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*, pp. 520–527, 2004.

Vlassis, N., Ghavamzadeh, M., Mannor, S., and Poupart, P. Bayesian reinforcement learning. In Wiering, M. and van Otterlo, M. (eds.), *Reinforcement Learning: State of the Art*, chapter 11. Springer Verlag, 2012.

Young, S., Gašić, M., Keizer, S., Mairesse, F., Schatzmann, J., Thomson, B., and Yu, K. The hidden information state model: A practical framework for POMDP based spoken dialogue management. *Computer Speech and Language*, 24:150–174, 2009.

Young, S., Gasic, M., Thomson, B., and Williams, J. D. POMDP-based statistical spoken dialog systems: A review. *Proceedings of the IEEE*, 101(5):1160–1179, May 2013.

Zhang, Z., Hsu, D., and Lee, W. S. Covering number for efficient heuristic-based POMDP planning. In *Proceedings of the 31th International Conference on Machine Learning (ICML 2014)*, pp. 28–36, June 2014.