THESIS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

Efficient Bayesian Planning

DIVYA GROVER

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
CHALMERS UNIVERSITY OF TECHNOLOGY
GOTHENBURG, SWEDEN 2022

Efficient Bayesian Planning
DIVYA GROVER

ISBN 978-91-7905-758-9

Doktorsavhandlingar vid Chalmers tekniska högskola
Ny serie nr 5224
ISSN 0346-718X
Technical Report No. 228D

Division of Data Science and Artificial Intelligence
Department of Computer Science and Engineering
Chalmers University of Technology
SE-412 96 Gothenburg, Sweden
Telephone +46 (0)31-772 1000

Cover:
A tree structure representing the planning process in an unknown environment. The circular nodes represent the current estimate on the unknown environment. The square node represents the action taken and the edges represent the observation as a consequence of the interaction.

Printed by Chalmers Reproservice
Gothenburg, Sweden 2022

To my parents, Monila & Suraj Prakash

ABSTRACT

Artificial Intelligence (AI) is a long-studied and yet very active field of research. The list of things differentiating humans from AI grows thinner but the dream of an artificial general intelligence remains elusive. Sequential Decision Making is a subfield of AI that poses a seemingly benign question "How to act optimally in an unknown environment?". This requires the AI agent to *learn* about its environment as well as *plan* an action sequence given its current knowledge about it. The two common problem settings are partial observability and unknown environment dynamics. *Bayesian planning* deals with these issues by simultaneously defining a single planning problem which considers the simultaneous effects of an action on both learning and goal search. The technique involves dealing with infinite tree data structures which are hard to store but essential for computing the optimal plan. Finally, we consider the minimax setting where the Bayesian prior is chosen by an adversary and therefore a worst case policy needs to be found.

In this thesis, we present novel Bayesian planning algorithms. First, we propose DSS (*Deeper, Sparser Sampling*) for the case of unknown environment dynamics. It is a meta-algorithm derived from a simple insight about the Bayes rule, which beats the state-of-the-art across the board from discrete to continuous state settings. A theoretical analysis provides a high probability bound on its performance. Our analysis is different from previous approaches in the literature in terms of problem formulation and formal guarantees. The result also contrasts with those of previous comparable BRL algorithms, which typically provide asymptotic convergence guarantees. Suitable Bayesian models and their corresponding planners are proposed for implementing the discrete and continuous versions of DSS. We then address the issue of partial observability via our second algorithm, FMP (*Finite Memory Planner*). This uses depth-dependent partitioning of the infinite planning tree. Experimental results demonstrate comparable performance to the current state-of-the-art for both discrete and continuous settings. Finally, we propose algorithms for finding the best policy for the worst case belief in the Minimax Bayesian setting.

Keywords: Planning, Bayesian Reinforcement Learning, Partially Observable MDP

ACKNOWLEDGMENTS

I would not have been able to complete my PhD journey without the support of all the people around me. I would like to thank a few of them here for their time and generous support.

Firstly, I would like to thank my advisor Christos Dimitrakakis for his enormous support. Your guidance has been invaluable to me. You have been extremely patient and accommodating to all my fallacies, including times when I have been less than productive. Your unopinionated and balanced approach on various solution techniques has helped me fill gaps in my knowledge and your sharp examples have helped me wrap my head around very many concepts. Your encouragement to collaborate is highly appreciated, e.g., my Harvard trip as well as to the small village of Halden, Norway. You supported attending the niche but premier workshop of EWRL, which has been an amazing experience. Thank you for helping me out with the writing of my all papers, including this thesis. Writing hasn't been my forte and you have been very patient about this with me. I am honoured and very glad to have you as my mentor.

I would like to thank Debabrota Basu for his continued support and encouragement to pursue the same line of research that I once took for a dead-end. You always have a fresh perspective and never discard any potential solution technique instantly. This is a quality I tried to learn from you. I would like to thank Emilio Jorge and Hannes Eriksson. Your presence in my life is highly appreciated. You have been my co-authors as well as buddies. You were there in practically almost every part of my journey. I remember many meals we shared, discussions ranging from research to startups to random, mundane stuff. You have helped me in numerous small ways and continue to do so. In the hindsight,

I have never hesitated to ask help from you, which is a proof of the warmth you have shown me. I am grateful to Aristide Tossou, especially for his presence in the early days my PhD journey. You initiated and oriented me in the ways of a PhD student. I appreciate your bits of advice about life in Sweden.

I also feel honored and humbled to have Alessandro Lazaric as the opponent of my thesis. I am grateful to the grading committee members Alexandre Proutiere, Jana Tumova and Marc Diesenroth. I thank you all for taking your time out to review my work. I am grateful to Morteza Haghir Chehreghani who agreed to help with the defense if administrative issues to came in the way.

I am grateful to my examiner Devdatt Dubashi for our many interactions, as a guide, colleague and examiner. I also acknowledge the administrative support at Chalmers such as Agneta, Fatima, Clara, Eva, Rebecca and thank them for their help. A big shout-out to the people at the Division of Data Science and AI. I have had many memorable interactions with you all.

Finally, I am indebted to my wife, Leila Jamshidian sales. Your support has been literally enormous and I could not have completed this thesis without the mental support that you provided me. Listing all the ways you supported would be very long and therefore, I will keep it this short. My dream to have a PhD was constantly supported by my mother, father and my sister. I am uncountably grateful to you all. I also thank my family and friends back home. Finally, I show my gratitude to the many people whose name are not mentioned here but who worked behind the scenes to make this event possible in my life.

LIST OF PUBLICATIONS

The following manuscripts are included in the thesis.

- ▷ **Divya Grover**, Debabrota Basu, Christos Dimitrakakis. "Bayesian Reinforcement Learning via Deep, Sparse Sampling" in *23^{rd} International Conference on Artificial Intelligence and Statistics*. Palermo, Italy, June 3-5, 2020.
- ▷ **Divya Grover**, Debabrota Basu, Christos Dimitrakakis. "Bayesian Reinforcement Learning via Approximate Planning in Bayes Adaptive MDP" submitted to the *Journal of Artificial Intelligence Research*, 2022.
- ▷ **Divya Grover**, Christos Dimitrakakis. "Adaptive Belief Discretization for POMDP Planning" in *15^{th} European Workshop on Reinforcement Learning*. Milano, Italy, September 19-21, 2022.
- ▷ Thomas Kleine Buening, Christos Dimitrakakis, Hannes Eriksson, **Divya Grover**, Emilio Jorge. "Minimax Bayesian Reinforcement Learning" in *15^{th} European Workshop on Reinforcement Learning*. Milano, Italy, September 19-21, 2022.

The following manuscripts are not included in this work.

- ▷ Christos Dimitrakakis, Hannes Eriksson, Emilio Jorge, **Divya Grover**, Debabrota Basu. "Inferential Induction: Joint Bayesian Estimation of MDPs and Value Functions" in ICBIN'20 - NeurIPS Workshop.

Contents

List of Figures

Part I

Extended Summary

Introduction

Artificial Intelligence (AI) is an expansive research field aiming at machines that can replicate any human skill or behaviour. We can categorize AI research based on the type of tasks an AI agent is expected to perform, namely *Learning & inference* and *Planning*. The act of storing new knowledge is known as learning while inference refers to the act of extracting conclusions given this limited knowledge base. These two are tightly knit by the design of the knowledge base. Finally, the process of deciding on long-term actions or plans given one's current knowledge is called *Planning*. Early successes like the STRIPS [Fikes and Nilsson, 1971] planner and expert systems like MYCIN [Shortliffe and Buchanan, 1975] focused on planning while relying on human-encoded domain knowledge (learning didn't exist and the inference was trivial). Post 90's focus shifted to *Learning* & *Inference* tasks like pattern recognition, data compression, time series prediction etc. In many problems, one aspect is easier than the other. For example in path planning, knowledge is encoded trivially using graphs and the hard part is planning, while in face recognition, it is hard to mathematically define what constitutes a face, but the act of deciding is trivial given an appropriate learnt representation. The subfield of AI which we study brings together these two tasks by posing a seemingly benign question "How to act optimally in an unknown environment?".

Decision making. This refers to those situations where an algorithm must interact with a system to achieve some desired objective. A fundamental characteristic of such problems is the feedback effect that these interactions have on the system. In AI, this is analogous to the situation where an autonomous agent

acts in an environment. In many cases, one uses a mathematical model that encapsulates the basics of this interactive system. The process of developing long-term actions in such a setting is known as *Planning*.

Decision making problems can be divided into two types, depending on the amount of information. The first is decision making under no uncertainty, which refers to the situation where one has full knowledge of the system's behaviour (potentially stochastic). Even in this case, it is not a trivial problem to decide how to act. The second type is decision making under (epistemic) uncertainty, which is the focus of this thesis.

Decision making under uncertainty In real-world processes, along with inherent (aleatoric) uncertainty[1], there also exists uncertainty because of our lack of knowledge about how the system behaves, known as epistemic uncertainty. Reinforcement Learning (RL) is an important problem in this category. According to Duff [2002], "RL attempts to import concepts from classical decision theory and utility theory to the domain of abstract agents operating in uncertain environments. It lies at the intersection of control theory, operations research, artificial intelligence and animal learning." Its first successful application was the development of a state-of-the-art AI [Tesauro, 1994] for playing Backgammon. More recent successes include game-playing AI [Mnih et al., 2015, Silver et al., 2017].

Planning in this general setting requires taking into account future events and observations that may change our conclusions. Typically, this involves creating long-term plans covering possible future eventualities, i.e. when planning under uncertainty, we also need to take into account the possible future knowledge that could be generated while acting. Executing actions also involves trying out new things, to gather more information, but it is hard to tell how beneficial this information will be. The choice between acting in a manner that is known to produce good results, or experimenting with something new, is known as the *exploration-exploitation* dilemma. It is a central problem in RL research.

Exploration-Exploitation

[1] Like the randomness associated with a game of chance.

Consider the problem of choosing your education stream for your long-term career. Let's say you are inclined toward Engineering. However, Project Management has recently become quite popular and is financially more rewarding. It is tempting to try it out! But there is a risk involved. It may turn out to be much worse than Engineering, in which case you will regret switching streams. On the other hand, it could also be much better. What should you do? It all depends on how much information you have about either of the career choices and how many more years are you willing to spend to get a degree. If you already have a PhD, then it's probably a better idea to go with Engineering. However, if you just finished your bachelor's degree, Project Management may be a good bet since you may get a much higher salary for the remainder of your life. Otherwise, you would miss out only by a year making the potential risk quite small.

Bayesian Reinforcement Learning

One way to approach the exploration-exploitation dilemma is to take decisions that explicitly take into account the uncertainty, both in the present and the future. One may use the Bayesian approach for this; essentially any algorithm is Bayesian in nature if it maintains probabilistic beliefs on quantities of interest and updates them using evidence collected. Formulating the RL problem in a Bayesian framework is known as Bayesian RL (BRL). In this framework, the 'learning' task can be subsumed into a planning problem as shown later.

Minimax Reinforcement Learning

One can take the Bayesian formulation one step further by assuming that even the environment that the agent acts in, is selected by a stochastic adversary using some unknown probability distribution[2]. In other words, what is the best policy against an adversary who pits the agent against an environment selected randomly at the start of each episode and the agent is unaware of the probability distribution that the adversary uses to sample the environments.

[2]Over a set of possible environments.

1.1 Thesis outline

The remainder of this chapter is structured as follows. First, we formally define the quantities of interest and introduce the necessary background that will help us understand the remainder of this thesis. Next, we summarize the contributions of this thesis. Thereafter, the relevant literature is reviewed. The final section includes our concluding remarks and discusses some avenues for potential future work.

1.2 Background

In this section, we present the technical background necessary to understand the context of this thesis and its contribution. We study various sequential decision making problems where the common theme is an agent interacting with an environment. The interaction involves the agent taking an action a_t at time t and obtaining a numerical reward r_t. The agent's goal is to maximize its total reward over time, i.e. $\sum_t r_t$. The typical model used to represent this interaction between the agent and the environment is called a Markov Decision Process (MDP), which we introduce next.

1.2.1 Markov Decision Processes (MDP)

A Markov Decision Process (MDP) is a discrete time stochastic process that provides a formal framework for reinforcement learning problems.

Definition 1. An MDP (S, A, P, R) is a tuple composed of a state space S, an action space A, a reward function R and a transition function P. When we refer to a specific MDP μ, we will add a subscript to denote its transition function P_μ and reward R_μ. The transition function $P_\mu \triangleq \mathbb{P}_\mu(s_{t+1}|s_t, a_t)$ dictates the distribution over the next states s_{t+1} given the present state-action pair (s_t, a_t). The reward function $R_\mu : S \times A \to [0, 1]$ dictates the reward r_{t+1} obtained after each (s_t, a_t).

A policy $\pi \in \Pi$ in a policy space Π is an algorithm for selecting actions given previous observations. It is Markov if, at any time t, the action a_t chosen by the policy only depends on the current state s_t. We denote the action distribution of

such a policy as $\pi_t(a_t \mid s_t)$. The *value function* of a policy for a specific MDP μ is the expected sum of discounted rewards obtained from time t to T while selecting actions in the MDP μ starting from state s:

$$V_{t,T}^{\pi,\mu}(s) = \mathbb{E}_{\mu}^{\pi}(\sum_{k=0}^{T-t} \gamma^{t+k} r_{t+k} \mid s_t = s), \tag{1.1}$$

where $\gamma \in (0, 1]$ is called the discount factor and $\mathbb{E}_{\mu}^{\pi}$ denotes the expected value of any random variable dependent on the stochastic process arising from the induced Markov chain (by following policy π in the MDP μ)[3]. For $\gamma < 1$, the following limit exists $V_{\mu}^{\pi} \triangleq \lim_{T\to\infty} V_{0,T}^{\pi,\mu}$. Define $V_{\mu}^{*} \triangleq \max_{\pi} V_{\mu}^{\pi}$ as the *optimal value function*, and $\pi_{\mu}^{*} \triangleq \arg\max_{\pi} V_{\mu}^{\pi}$ as the *optimal policy*. The optimal policy and value function are computable via Backwards induction [Puterman, 1994].

$$V_t^*(s) = \max_{a\in A} \left[R_\mu(s,a) + \gamma \int_{s'\in S} P_\mu(s'|s,a) V_{t+1}^*(s') \right] \tag{1.2}$$

That is, starting with $V_T = R_\mu(s,a)$, we compute each of $V_{T-1}, V_{T-2} \ldots V_0$ in order and for each possible transition $(s_t = s, a_t = a) \to s_{t+1} = s'$. When $\gamma < 1$, the process converges to a fixed point and the algorithm is called *value iteration*. Note that in order to compute the optimal value function $V_t^*(s_t)$ and action $a_t = \pi_\mu^*(s_t)$ using equation (1.2), the agent needs the knowledge of P_μ and current state s_t. This setting is known as the *Planning* problem.

1.2.2 Types of problem setting

In what follows, if the agent can observe state s_t but is unaware of the transition function P_μ, the setting is called the *Reinforcement Learning* problem. On the contrary, if the environment dynamics P_μ are known but the state s_t is hidden, the setting is called a *Partially Observable* MDP or the POMDP problem. Finally, the Minimax setting extends the Bayesian RL problem with an adversarially chosen prior. We detail each of the settings next.

[3] In general, $\mathbb{E}_b^a$ denotes expectation operator conditioned on $A = a$ and $B = b$.

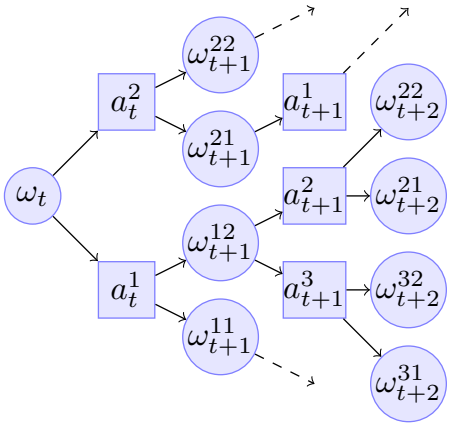

Figure 1.1: Visualising the BAMDP tree. ω_t^{ij} denotes the state at time t given action i and having observed state j.

1.2.3 Bayes-Adaptive MDP (BAMDP)

For the RL setting, we follow the Bayesian formulation [Duff, 2002] and maintain a belief distribution β_t over the possible MDP models $\mu \in \mathcal{M}$.[4] Using an appropriate prior $\beta_0(\mu)$, we obtain a sequence of posterior beliefs $\beta_t(\mu)$ that represents our subjective belief over the MDPs at each time t, depending on the latest observation. We obtain the posterior belief at time $t+1$ using the Bayes rule:

$$\beta_{t+1}(\mu) \triangleq \frac{\mathbb{P}_\mu(s_{t+1}|s_t, a_t)\beta_t(\mu)}{\int_{\mathcal{M}} \mathbb{P}_{\mu'}(s_{t+1}|s_t, a_t)\beta_t(\mu')d\mu'} \tag{1.3}$$

For each (s_{t+1}, s_t, a_t) tuple, the next β_{t+1} is uniquely determined by the equation (1.3). Therefore, we denote the Bayes rule as a mapping $\beta_{t+1} \triangleq \mathcal{B}^{\beta_t}_{s_t,a_t}(s_{t+1})$ to highlight its dependence on the next state.

The Bayes-utility v, can be defined analogously to the MDP value function:

$$v^\pi_\beta(s) \triangleq \int_{\mathcal{M}} V^\pi_\mu(s)\beta(\mu)d\mu \tag{1.4}$$

The Bayes-optimal policy has the corresponding Bayes-optimal utility.

$$v^*_\beta(s) \triangleq \max_{\pi\in\Pi} \int_{\mathcal{M}} V^\pi_\mu(s)\beta(\mu)d\mu \tag{1.5}$$

[4]More precisely, one can define a measurable space $(\mathcal{M}, \mathfrak{M})$, where $\mathcal{M}$ is the possible set of MDPs and $\mathfrak{M}$ is a suitable σ-algebra.

It is well known [Duff, 2002, Guez et al., 2012] that by considering the original MDP's state s_t and the belief β_t together as a hyper-state ω_t, one can obtain another MDP called the Bayes Adaptive MDP (BAMDP), whose optimal policy is the Bayes-optimal policy.

Definition 2. A Bayes Adaptive Markov Decision Process (BAMDP), denoted by $\tilde{\mu} \triangleq (\Omega, A, \nu, \tau)$ is a representation for MDP $\mu = (S, A, P_\mu, R_\mu)$ whose transition function is unknown. Its state space is a set of hyper states $\Omega = S \times \mathfrak{B}$, where $\mathfrak{B}$ is a space of probability distributions (called beliefs) over the set of possible MDP models $\mathcal{M}$. It has a common action space A while the transition $\nu(\omega_{t+1}|\omega_t, a_t)$ and the reward $\tau(\omega_t, a_t)$ functions are defined as follows:

$$\begin{aligned}
&\nu(\omega_{t+1}|\omega_t, a_t) \equiv \mathbb{P}(s_{t+1}, \beta_{t+1}|s_t, \beta_t, a_t) \\
&= \underbrace{\mathbb{P}(\beta_{t+1}|s_{t+1}, s_t, a_t, \beta_t)}_{\text{Bayes rule}} \int_{\mathcal{M}} \underbrace{\mathbb{P}_\mu(s_{t+1}|s_t, a_t)}_{\text{MDP kernel}} \; \beta_t(\mu) d\mu \\
&\tau(\omega_t, a_t) \triangleq \int_{\mathcal{M}} R_\mu(s_t, a_t) \; \beta_t(\mu) d\mu
\end{aligned}$$

At time t, an agent observes the state $\omega_t = (s_t, \beta_t)$, takes an action $a_t \in A$ and transitions into a new state $\omega_{t+1} = (s_{t+1}, \beta_{t+1})$, where β_{t+1} is computed using $\mathcal{B}^{\beta_t}_{s_t, a_t}(s_{t+1})$ (eq. 1.3).[5] The hyper-state of the BAMDP has the Markov property since each hyper-state $\omega_{t+1} = (s_{t+1}, \beta_{t+1})$ is uniquely determined by the previous ω_t. Because the BAMDP is simply an MDP over the space of hyperstates, backward induction can be used starting from the set of terminal hyperstates Ω_T and proceeding backwards to T-1, T-2, $\ldots$, t as follows:

$$V^*_t(\omega) = \max_{a \in A} \left[\tau(\omega, a) + \gamma \int_{\omega' \in \Omega_{t+1}} \nu(\omega'|\omega, a) V^*_{t+1}(\omega') \right] \tag{1.6}$$

where Ω_{t+1} is the set of reachable hyperstates from ω. It is important to note here that the size of the reachable hyperstates $|\Omega_{t+1}|$ is at most equal to the size of state space $|S|$ since the Bayes rule mapping $\mathcal{B}$ is one-to-one for all possible next state s_{t+1}.

[5] In particular, since the next belief is uniquely determined from the current and next state, the action and the previous belief, $\mathbb{P}(\beta_{t+1}|s_{t+1}, s_t, a_t, \beta_t)$ is a Dirac distribution.

1.2.4 Partially Observable MDP (POMDP)

For the POMDP setting, the agent can only indirectly estimate the state s_t via a set of observations $o_t \in O$ and their (state-dependent) emission probability. Formally,

Definition 3 (POMDP)**.** A Partially Observable MDP, or POMDP, denoted by $(S, A, P, R, \gamma, O, Z)$ tuple is an MDP model augmented with observations $o_t \in O$ and their corresponding emission probabilities $Z \triangleq \mathbb{P}(o_t|s_t)$. The state $s_t \in S$ is not directly observable by an agent.

In such cases, the best an agent could hope to achieve is to act optimally in an expected sense (averaging over its estimation of the current state). Let b_t denotes the current state distribution at time t, known as the *belief*. It is a probability distribution over the state space whose element $b_t(s)$ gives the probability that the system's true state is s.[6] It is a compact representation of the agent's complete history of interaction with its environment, i.e, the action-observation sequence it followed. For an initial belief b_0, policy π and horizon H, the corresponding value function $V_H^\pi(b_0)$, is defined as follows

$$V_H^\pi(b_0) = \mathbb{E}(\sum_{k=0}^{H-1} \gamma^k r_{k+1}|b_0, \pi) \tag{1.7}$$

$$b_0 \in \Delta^{|S|}$$

$$\Delta^{|S|} \triangleq \{b : S \to [0,1] \;\Big|\; \int_s b(s)\, \mathrm{d}s = 1\} \tag{1.8}$$

Let V_H^* denote the optimal value function. In the discounted setting, the objective is to find a policy π maximizing $V^\pi(b_0)$, which is defined as

$$V^\pi(b_0) \triangleq \lim_{H\to\infty} V_H^\pi(b_0) \tag{1.9}$$

Inference equations

Next, we describe the standard computational equations useful in the context of POMDP. To compute a new belief b_{k+1}, given the current belief b_k, action a

[6] For continuous state spaces, b is instead a density.

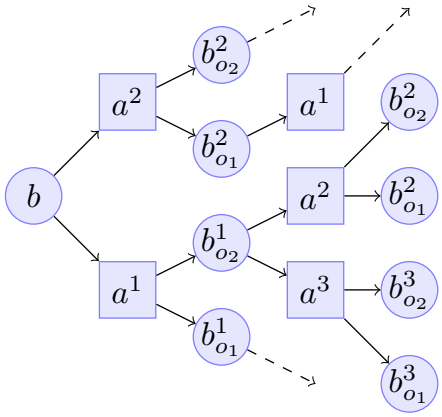

Figure 1.2: Visualising the POMDP tree. The posterior belief $b^a_{o_i}$ is inferred from the prior (parent belief) by observing action a and the i^{th} observation.

and next observation o_{k+1}, one uses the following set of equations:

$$b_{k+1}(s_{k+1}) = \frac{1}{o_{\text{prob}}} Z(o_{k+1}|s_{k+1})\, \mathbb{P}(s_{k+1}) \tag{1.10}$$

$$\mathbb{P}(s_{k+1}) = \int_{s'} P(s_{k+1}|s', a) b_k(s')\, \mathrm{d}s' \tag{1.11}$$

$$o_{\text{prob}} \triangleq \int_{s_{k+1}} Z(o_{k+1}|s_{k+1})\, \mathbb{P}(s_{k+1})\, \mathrm{d}s \tag{1.12}$$

collectively referred to as the *Bayes filter* [Thrun, 2002]. Equation (1.10) gives the Bayesian posterior of the states, also called the next belief b_{k+1}. Equation (1.11) is the state marginal distribution after the transition has occurred under the given action, while equation (1.12) is the marginal probability or likelihood of the observation o_{k+1}.

Solving a POMDP involves generating the belief tree and then doing backward induction [Bellman, 1952] on it to compute the optimal action at the root node. The corresponding Bellman equation is:

$$V(b_k) = \max_a \mathbb{E}_{\mathbb{P}(o_{k+1}|a,b_k)}[\, r_{k+1} + \gamma \mathrm{V}(b_{k+1})\,] \tag{1.13}$$

Where

$$r_{k+1} = \int_s R(s,a) b_k(s) \, \mathrm{d}s$$
$$\mathbb{P}(o_{k+1}|a, b_k) = o_{\text{prob}}$$

Note that b_{k+1} is uniquely determined by o_{k+1} using the Bayes filter. Note that in this setting, the likelihood function Z is assumed to be known.

1.2.5 Bayesian Planning

For both the BAMDP and POMDP settings, solving the corresponding Bellman equations (1.6 and 1.13) requires knowledge of the reachable states ω_H and b_H respectively. To this end, a tree-like structure can be associated with each problem, called the *Belief tree*. This tree is rooted at the current belief (or hyperstate) and is shown in Figure (1.1) and Figure (1.2) respectively.

The EXPAND method (algorithm 1) is used to generate the BAMDP belief tree shown in Figure (1.1). Algorithm 1 then solves the planning problem via backward induction (Line 17-21). A similar algorithm can be used for the POMDP setting although it is important to note that such exact tree expansion algorithms are only valid for the discrete state-action setting. For the continuous setting, we need certain approximations to the belief tree as the branching factor is infinite[7]. Controlling the size of the belief tree forms the basis of the contributions of this thesis.

1.2.6 Minimax Bayesian RL

In the Bayesian setting, the agent has a belief about the MDP, for which it tries to find the optimal policy. There are two possible ways to interpret the distribution β, depending on how it is chosen. If β is chosen by the agent itself, then it corresponds to his subjective belief about the most likely MDP a priori. In that case, $\mathcal{U}(\pi, \beta)$ corresponds to the expected utility of a particular policy under this belief.

[7] For continuous-state in BAMDP and continuous-observation in POMDP.

Algorithm 1 FHTS (Finite Horizon Tree Search)

Global: Horizon H

Function EXPAND($\omega_h = (s_h, \beta_h), h$)
  if $h > H$ **then**
    return
  end if
  for all actions a **do**
    for all next states s_{h+1} **do**
      $\beta_{h+1} = \mathcal{B}^{\beta_h}_{s_h,a}(s_{h+1})$ (eq. 1.3)
      ω_{h+1} = (s_{h+1}, β_{h+1})
      EXPAND($\omega_{h+1}, h+1$)
    end for
  end for
end Function

Function FHTS(ω_0)
  EXPAND(ω_0)
  $Q(\omega_H, a) = 0 \; \forall \; a \in A$
  for all h in $H-1, \ldots, 0$ **do**
    for all actions a **do**
      $Q(\omega_h, a) \mathrel{+}= \tau(r|\omega_h, a) + \nu(\omega_{h+1}|\omega_h, a) \times \max_a Q(\omega_{h+1}, a)$
    end for
  end for
  return $\max_a Q(\omega_0, a)$
end Function

The second view of β is to assume that the MDP is actually drawn randomly from the distribution β. If this is known, then the subjective value of a policy is equal to its true expected value. However, it is more interesting to consider the case where nature selects β in an arbitrary way from a set of possible priors $\mathcal{B}$. In this setting, the agent does not have a particular reason to select one prior over another but merely wants to select a prior that gives robustness guarantees for its adaptive policy.

We can then model this as a zero-sum game against nature, where the agent's

utility is the sum of rewards

$$\mathcal{U} \triangleq \sum_{t}^{T} r_t \ ,$$

and where the expected utility for any fixed agent policy π and MDP μ is

$$\mathcal{U}(\pi, \mu) \triangleq \mathbb{E}_{\mu}^{\pi}(\mathcal{U})$$

under some arbitrary starting state distribution. The corresponding utility for the agent when nature chooses the MDP according to some probability distribution β is then:

$$\mathcal{U}(\pi, \beta) \triangleq \int_{\mathcal{M}} \mathcal{U}(\pi, \mu)\beta(\mu)d\mu.$$

A natural question is how to select the worst-case prior β^* and the corresponding agent policy π^*

$$\beta^*, \pi^* \triangleq \arg\max_{\pi \in \Pi} \arg\min_{\beta \in \mathcal{B}} \ \mathcal{U}(\pi, \beta)$$

1.3 Contributions

Our main contributions can be summarized as follows:

- In the 1^{st} paper, we propose a novel Bayesian planning algorithm for the BAMDP in the discrete setting. We provide concrete finite-sample performance bounds relative to the algorithm parameters. Performance in experiments beats the state-of-the-art. My contribution is the development of the algorithm, part development of the proof strategy, selection of appropriate prior and policy-optimizer pairs as well as complete implementation and experiments.
- In the 2^{nd} paper, We provide concrete implementations of our meta algorithm for continuous state settings. This involves proposing a suitable Bayesian dynamics model as well as the policy optimizer. The proofs are non-trivially generalized to the continuous setting with appropriate

assumptions and lemmas. A study over important hyperparameters, lacking in the previous version is done. All contributions are my own.

- In the 3^{rd} paper, we propose a novel analysis of the PODMP planning problem. Insights from this analysis allow us to propose a novel algorithm applicable to both discrete and continuous state settings. We show its benefits experimentally compared to the current state-of-the-art. All contributions are my own.
- In the 4^{th} paper, we answer some of the questions about the existence of a solution in the Minimax setting and how to compute it. My contribution constitutes the convergence rate of a Gradient Descent style algorithm and the conditions under which it holds.

1.4 Related work

In this section, we first discuss the relevant POMDP literature. Section 1.4.2 discusses algorithms that directly attack the BAMDP problem followed by section 1.4.3 where methods casting the BAMDP as a POMDP are discussed. Finally, related work on the Minimax formulation is discussed.

1.4.1 POMDP

The current state-of-the-art POMDP solvers are AdaOPS [Wu et al., 2021], DESPOT [Ye et al., 2017] and POMCP [Silver and Veness, 2010]. They rely on some mix of three main strategies in the POMDP planning literature:

1. **Promising action:** Expanding and evaluating only the promising nodes of the planning tree.
2. **Inference:** Using a particle filter for inference, combined with intelligent resampling to take care of the particle degeneracy problem.
3. **Covering:** Merging nodes with similar beliefs by keeping a finite cover over the continuous belief space.

Promising action: While building the lookahead tree, all three solvers keep the tree size in check by only considering 'promising' sequences of actions. This

is done by only expanding the action nodes with the highest difference between their upper and lower bound values.

Inference: DESPOT uses resampling to overcome the sample degeneracy problem faced by particle filters, where most particles become highly improbable and have negligible weights. AdaOPS uses a posterior dependent resampling scheme, which bounds [Fox, 2001] the error between the approximate and true distribution. An orthogonal approach for approximate inference in POMDP includes low dimensional belief representation methods like [Roy et al., 2005].

Covering: One key concept is to pack similar beliefs together, to keep the exponential growth of the nodes in check. This idea is linked to the concept of the covering number. Informally, it is the number of smaller spaces $\mathcal{Y}_i \subset \mathcal{X}$, centred at points y_i, needed to cover all of $\mathcal{X}$. The placement of points y_i as well as shapes of the sets $\mathcal{Y}_i, \mathcal{X}$ are important to get a minimal cover. Only recently has it been used as a complexity measure for POMDP [Zhang et al., 2012, Lemma 1] planning. The idea is to choose a representative set of beliefs, covering as much of the belief space as possible. It was also used in the analysis of SARSOP [Kurniawati et al., 2008] by Hsu et al. [2007]. Note that PGVI [Zhang et al., 2014], SARSOP and AdaOPS all derive the planning error in terms of an unknown covering number (denoted by $\mathcal{C}$ and $P^{\delta}_{\max}$ respectively) associated with their proposed uniform δ-cover. They also don't give an upper bound on the value of their covering number.

1.4.2 BAMDP

BAMDPs were initially investigated by Silver [1963] and Martin [1967]. The problem of computational intractability of the Bayes-optimal solution motivated researchers to design approximate techniques. These are referred to as Bayesian RL (BRL) algorithms. Ghavamzadeh et al. [2015] compile a survey of BRL algorithms. BRL algorithms discussed here are all model-based. They can be further classified based on whether they directly approximate the belief tree structure (lookahead) or not (myopic). Therefore we first discuss BRL algorithms based on their design and then their theoretical motivation.

We classify them into various categories based on their functioning.

Myopic: Myopic algorithms do not explicitly take into account the information to be gained by future actions, and yet may still be able to learn efficiently. The simplest algorithm is the one taking the mean MDP estimate and playing according to its optimal policy. However, this is not even the best non-adaptive policy in many cases. A highly successful myopic algorithm is Thompson sampling [Thompson, 1933], which maintains a posterior distribution over models, samples one of them and then chooses the optimal policy for the sample. A reformulation of this for BRL was investigated in [Strens, 2000]. The Best Of Sampled Set (BOSS) [Asmuth et al., 2009] algorithm generalizes this idea to a multi sample optimistic approach. BEB [Kolter and Ng, 2009] at the first look, seems to work directly on the Bayesian value function (eq. 1.6), but it simply adds an explicit bonus term to the mean MDP estimates of the state-action value. Similar to BEB, MMBI [Dimitrakakis, 2011] assumes constant belief and therefore rolls in the hyper-state value into the state-action value, but unlike BEB, it directly approximates the Bayesian value function. This assumption removes the exponential dependence (due to path dependent belief) on the planning horizon. Then, backward induction is performed using the value of the next-step optimal adaptive policy. The final output is the stationary policy obtained at the root through backward induction[8].

Lookahead: Lookahead algorithms take into account the effects of their future action on their knowledge about the environment and quantify its benefit for current decision making. The simplest algorithm is to calculate and solve the BAMDP up to some horizon H, as outlined in Algorithm 1 and illustrated in Figure 1.1. Sparse sampling [Kearns et al., 1999] is a simple modification to it, which instead only iterates over a set of sampled states. Kearns' algorithm, when applied to the BAMDP belief tree,[9] would still have to consider all primitive actions. Wang et al. [2005] improved upon this by using Thompson sampling to only consider a subset of promising actions. The high branching factor of the belief tree still makes planning with a deep horizon computationally expensive. Thus more scalable algorithms, such as BFS3 [Asmuth and

[8] Although it shouldn't be hard to store the intermediate optimal constant-belief adaptive policy, since it is computed (step-8, Algo.1) anyway.

[9] We freely use the term 'tree' or 'belief tree' to denote the planning tree generated by the algorithms in the hyper-state space of BAMDP.

Littman, 2011], BOLT [Araya et al., 2012] and BAMCP [Guez et al., 2012], were proposed. Similar to [Wang et al., 2005], BFS3 also selects a subset of actions but with an optimistic action selection strategy, though the backups are still performed using the Bellman equation. BOLT includes optimism in the transition function instead. BAMCP takes a Monte-Carlo approach to the sparse-lookahead idea by using approximate inference via particle filters. It also uses optimism for action selection. Unlike BFS3, the next set of hyper-states is sampled from an MDP sampled at the root[10]. Since posterior inference is expensive for any non-trivial belief model, BAMCP applies a technique called lazy sampling. It involves sampling particles (MDPs) at the root and building the tree using their transitions instead of the BAMDP transition function. Finally, leaf values are initialized using rollouts. Both techniques are inspired by their previous application to the tree search problem [Kocsis and Szepesvári, 2006].

VariBAD: A recent neural network based BRL approach by Zintgraf et al. [2021], called VariBAD, proposes to do two things simultaneously. Firstly, it learns the inference mechanism by learning a sufficient statistic, by minimizing the prediction error over the seen data. In particular, the statistic is computed by an RNN encoder with (action,next-observation) tuple[11] as input, which outputs a posterior embedding to be fed to the decoder. Secondly, this posterior embedding/sufficient statistic is used as an input to a policy network in hopes of learning the Bayes-optimal policy using the Temporal Difference (TD) [Sutton and Barto, 1998] update rule for BAMDP's Bellman equation (eq. 1.6). Hence, gradients for the policy network are derived from the bootstrapped value function using a TD variant Schulman et al. [2015]). We use VariBAD as a benchmark for the continuous setting.

Analytical guarantees: We first discuss BEB and BOLT, which have theoretical results similar to ours. Both are PAC-BAMDP and derive their result by achieving a certain level of certainty about (s, a) tuples, similar to Kearns and Singh [1998] who define such tuples as 'known'. BEB's authors rely on how

[10]Note that ideally the next observations should be sampled from the $P(s_{t+1}|\omega_t)$ instead of $P(s_{t+1}|\omega_{t_o})$, i.e. the next-state marginal at the root belief.

[11]Note that the prior should also be a separate input to the encoder, to properly approximate the posterior, but it is only implicitly computed within the RNN.

many (s, a) tuples are already 'known' to prove their result. They go on to prove that both finite horizon Bayesian-optimal policy and BEB's policy, decrease the exploration rate so fast that they are not PAC-MDP[12], providing a 3-state MDP counter-example. BOLT's authors prove that if the probability of unknown states is small enough, their value function is close to optimal, if not, then such events (of seeing 'unknown' (s, a) tuples) occur only a limited number of times (by contradiction). The required amount of exploration is ensured by BOLT's optimism. They also extend BEB's PAC-BAMDP result to the infinite horizon case. The main problem with BOLT's approach of 'knowing' all the (s, a) tuples enough is that the Bayes-optimal policy no longer remains interesting; Martin [1967] has shown that in such a case, the Bayes-optimal policy approaches the optimal policy of underlying MDP. This leads to an unfaithful approximation of the true Bayes-optimal policy. A better approach is to prove approximate optimality without such an assumption. For example, BOP [Fonteneau et al., 2013] uses an upper bound on the Bayesian value function for their branch-and-bound tree search. In practice, the exponential dependence on the branching factor is still quite strong (proposition 2 of their work). No analytical results exist for the Neural Network based VariBAD algorithm.

1.4.3 POMDP perspective on BAMDP

A natural idea would be to apply the already existing literature on POMDP to the BAMDP setting since both are Bayesian planning problems. Significant effort was made by Duff [2002], where he shows an almost mechanical translation of the POMDP alpha-vector[13] formulation to BAMDP (sec. 5.3). He notes that due to belief having continuous support in the BAMDP, in contrast to the discrete support (over states) in the POMDP, fundamental differences[14] arise in the application of Monahan's algorithm [Monahan, 1982]. He comments (sec. 5.3.3) how the alpha functions due to backward induction are just a mixture of alpha functions at previous iteration[15], by extension of which any closed set of functions representing the Bayesian value initially will imply a

[12] This is the first of any result on the Frequentist nature of Bayes-optimal policy.
[13] A vector, $\alpha : [0, 1]^{|S|} \to \mathbb{R}$, compactly represents the Bayesian value over all the belief space.
[14] Alpha vectors become alpha functions and their dot product with belief becomes integral.
[15] More precisely, the reward is added to this mixture by due to the Bellman operator definition.

function from the same family locally at the root belief. Therefore the "idea of characterizing the value function in terms of a finite set of elements generalizes from the POMDP case". Although he quickly points out how this approach is computationally infeasible: Exact methods for PODMP [Sondik, 1978, Kaelbling et al., 1998] crucially depend on eliminating the exponentially growing alpha vectors with respect to the planning horizon, and for this, they solve a set of linear equation constraints. These constraints in the BAMDP case turn out to be integral constraints, which usually don't have any easy tractable solution. Hence, the curse of exponentially memory usage with respect to the planning depth still remains. Poupart et al. [2006] claim to show that alpha functions in BAMDP are multivariate polynomials in shape, but their main theorem only relies on backward induction for the proof. Therefore it is unclear how the initial alpha functions should indeed be multivariate polynomials. Duff [2002] goes on to argue and develop a general finite-state (memory) controller method for both POMDP and BAMDP problems. This approach holds much promise and should be investigated further. One key observation usually missed by the 'BAMDP as POMDP' literature is that the convergence property of the belief doesn't exist for POMDP[16] and hence we miss out by not fully exploiting this property of BAMDP.

1.4.4 Minimax Bayesian RL

Berger [2013] has extensively discussed Minimax-Bayes decision problems in his work. The problem is to find a worst-case prior so as to obtain guarantees in terms of the expected loss in a Bayesian decision making process. Grünwald and Dawid [2004] have also discussed the problem of Bayesian experiment design in this context. However, the problem has not been addressed in the Bayesian reinforcement learning literature.

[16] In BAMDP, the belief over transition probabilities converge as we plan, while no such analogy exists for belief simply over the MDP states.

1.5 Concluding remarks and future directions

In this thesis, we presented Bayesian planning algorithms for two widely studied settings of BAMDP and POMDP. In both settings, many early algorithms focused on leaf approximations while later ones on Monte-carlo inference. Our proposed algorithms use systematic approximation to the belief tree by fully utilizing the underlying structure of the problems. This is in contrast to both type of past approaches and directly tackle the fundamental problem of exponentially increasing belief tree. Our analysis in both cases is general and novel enough to be applicable to both discrete and continuous state settings. Experiments demonstrate the performance of algorithms with better or comparable results to the current state-of-the-art in their respective domains. Finally, we answer some of the questions about the existence of a solution in the Minimax setting and how to compute it.

Future directions

We now discuss potential research directions that can be studied and built upon our work. The proposed ideas are of both theoretical and practical interest.

Minimax Convergence results for specific cases of prior and policy space could be established for the proposed algorithms. Analytical properties of the Bayesian utility/regret could be studied more deeply, especially its gradient.

BAMDP A direct extension would be the use of a more general dynamics model and policy generator such as neural networks. The analysis could be specialized by considering specific priors from the exponential family distributions. A regret bound could also be proposed by studying the rate of convergence of the posterior under the Bayes-optimal policy, possible in part due to our value function bound at the root. We *conjecture* that the Bayes-optimal policy would inherently explore enough without the need for explicit optimism [17]. Studying

[17] This connection may also be seen in a previous attempt by [Duff and Barto, 1997], where they try to use the Gittins index for BAMDP.

the effects of adaptive vs non-adaptive policies by establishing some kind of submodularity property would also be an interesting avenue.

POMDP In future work, there is a possibility to extend the analysis by more tightly coupling the POMDP parameters and the belief discretization. A more interesting direction would be to extend the problem to a likelihood-free POMDP setting (Z unknown).

Finally, the unknown-dynamics plus unobservable-state setting could be simply formulated as an RL problem, since the model space (say a neural network) could be made to learn features representing the state directly, bypassing the observation to state ($f_\theta : o \to s$) modelling.

Bibliography

Mauricio Araya, Olivier Buffet, and Vincent Thomas. Near-optimal brl using optimistic local transitions. *arXiv preprint arXiv:1206.4613*, 2012.

J. Asmuth, L. Li, M. L. Littman, A. Nouri, and D. Wingate. A Bayesian sampling approach to exploration in reinforcement learning. In *UAI 2009*, 2009.

John Asmuth and Michael L Littman. Approaching bayes-optimalilty using monte-carlo tree search. In *Proc. 21st Int. Conf. Automat. Plan. Sched., Freiburg, Germany*, 2011.

Richard Bellman. On the theory of dynamic programming. *Proceedings of the National Academy of Sciences of the United States of America*, 38(8):716, 1952.

James O Berger. *Statistical decision theory and Bayesian analysis*. Springer Science & Business Media, 2013.

Christos Dimitrakakis. Robust bayesian reinforcement learning through tight lower bounds. In *European Workshop on Reinforcement Learning*, page arXiv:1106.3651v2. Springer, 2011.

Michael O Duff and Andrew G Barto. Local bandit approximation for optimal learning problems. In *Advances in Neural Information Processing Systems*, pages 1019–1025, 1997.

Michael O'Gordon Duff. *Optimal Learning Computational Procedures for Bayes-adaptive Markov Decision Processes*. PhD thesis, University of Massachusetts at Amherst, 2002.

Richard E Fikes and Nils J Nilsson. Strips: A new approach to the application of theorem proving to problem solving. *Artificial intelligence*, 2(3-4):189–208, 1971.

Raphael Fonteneau, Lucian Buşoniu, and Rémi Munos. Optimistic planning for belief-augmented markov decision processes. In *2013 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL)*, pages 77–84. IEEE, 2013.

Dieter Fox. Kld-sampling: Adaptive particle filters. *Advances in neural information processing systems*, 14, 2001.

Mohammad Ghavamzadeh, Shie Mannor, Joelle Pineau, Aviv Tamar, et al. Bayesian reinforcement learning: A survey. *Foundations and Trends® in Machine Learning*, 8(5-6):359–483, 2015.

Peter D Grünwald and A Philip Dawid. Game theory, maximum entropy, minimum discrepancy and robust bayesian decision theory. *the Annals of Statistics*, 32(4):1367–1433, 2004.

Arthur Guez, David Silver, and Peter Dayan. Efficient bayes-adaptive reinforcement learning using sample-based search. In *Advances in Neural Information Processing Systems*, pages 1025–1033, 2012.

David Hsu, Wee Lee, and Nan Rong. What makes some pomdp problems easy to approximate? *Advances in neural information processing systems*, 20: 689–696, 2007.

Leslie Pack Kaelbling, Michael L Littman, and Anthony R Cassandra. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1-2):99–134, 1998.

Michael Kearns and Satinder Singh. Near-optimal reinforcement learning in polynomial time. In *Proc. 15th International Conf. on Machine Learning*, pages 260–268. Morgan Kaufmann, San Francisco, CA, 1998. URL `citeseer.ist.psu.edu/kearns98nearoptimal.html`.

Michael J. Kearns, Yishay Mansour, and Andrew Y. Ng. A sparse sampling algorithm for near-optimal planning in large Markov decision processes. In

Thomas Dean, editor, *IJCAI*, pages 1324–1231. Morgan Kaufmann, 1999. ISBN 1-55860-613-0.

Levente Kocsis and Csaba Szepesvári. Bandit based monte-carlo planning. In *European conference on machine learning*, pages 282–293. Springer, 2006.

J Zico Kolter and Andrew Y Ng. Near-bayesian exploration in polynomial time. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 513–520. ACM, 2009.

Hanna Kurniawati, David Hsu, and Wee Sun Lee. Sarsop: Efficient point-based pomdp planning by approximating optimally reachable belief spaces. In *Robotics: Science and systems*, volume 2008. Zurich, Switzerland., 2008.

James John Martin. *Bayesian decision problems and Markov chains*. Wiley, 1967.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

George E Monahan. State of the art—a survey of partially observable markov decision processes: theory, models, and algorithms. *Management science*, 28 (1):1–16, 1982.

P. Poupart, N. Vlassis, J. Hoey, and K. Regan. An analytic solution to discrete Bayesian reinforcement learning. In *ICML 2006*, pages 697–704. ACM Press New York, NY, USA, 2006.

Marting L. Puterman. *Markov Decision Processes : Discrete Stochastic Dynamic Programming*. John Wiley & Sons, New Jersey, US, 1994.

Nicholas Roy, Geoffrey Gordon, and Sebastian Thrun. Finding approximate pomdp solutions through belief compression. *Journal of artificial intelligence research*, 23:1–40, 2005.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation, 2015.

EH Shortliffe and BG Buchanan. A model of inexact reasoning in medicine, mathem. *Biosc*, 23(3/4), 1975.

David Silver and Joel Veness. Monte-carlo planning in large pomdps. In *Advances in neural information processing systems*, pages 2164–2172, 2010.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *nature*, 550(7676):354–359, 2017.

Edward A Silver. Markovian decision processes with uncertain transition probabilities or rewards. Technical report, MASSACHUSETTS INST OF TECH CAMBRIDGE OPERATIONS RESEARCH CENTER, 1963.

Edward J Sondik. The optimal control of partially observable markov processes over the infinite horizon: Discounted costs. *Operations research*, 26(2):282–304, 1978.

Malcolm Strens. A bayesian framework for reinforcement learning. In *ICML*, pages 943–950, 2000.

Richard S. Sutton and Andrew G. Barto. *Introduction to reinforcement learning*, volume 135. MIT press Cambridge, 1998.

Gerald Tesauro. Td-gammon, a self-teaching backgammon program, achieves master-level play. *Neural computation*, 6(2):215–219, 1994.

W.R. Thompson. On the Likelihood that One Unknown Probability Exceeds Another in View of the Evidence of two Samples. *Biometrika*, 25(3-4):285–294, 1933.

Sebastian Thrun. Probabilistic robotics. *Communications of the ACM*, 45(3): 52–57, 2002.

Tao Wang, Daniel Lizotte, Michael Bowling, and Dale Schuurmans. Bayesian sparse sampling for on-line reward optimization. In *ICML '05*, pages 956–963, New York, NY, USA, 2005. ACM. ISBN 1-59593-180-5. doi: http://doi.acm.org/10.1145/1102351.1102472.

Chenyang Wu, Guoyu Yang, Zongzhang Zhang, Yang Yu, Dong Li, Wulong Liu, and Jianye Hao. Adaptive online packing-guided search for pomdps. *Advances in Neural Information Processing Systems*, 34, 2021.

Nan Ye, Adhiraj Somani, David Hsu, and Wee Sun Lee. Despot: Online pomdp planning with regularization. *Journal of Artificial Intelligence Research*, 58: 231–266, 2017.

Zongzhang Zhang, Michael Littman, and Xiaoping Chen. Covering number as a complexity measure for pomdp planning and learning. In *Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.

Zongzhang Zhang, David Hsu, and Wee Sun Lee. Covering number for efficient heuristic-based pomdp planning. In *International conference on machine learning*, pages 28–36, 2014.

Luisa Zintgraf, Sebastian Schulze, Cong Lu, Leo Feng, Maximilian Igl, Kyriacos Shiarlis, Yarin Gal, Katja Hofmann, and Shimon Whiteson. Varibad: Variational bayes-adaptive deep rl via meta-learning. *Journal of Machine Learning Research*, 22(289):1–39, 2021. URL `http://jmlr.org/papers/v22/21-0657.html`.