Expectation-maximization for Bayes-adaptive POMDPs

A Dissertation

Presented to

the faculty of the School of Engineering and Applied Science

University of Virginia

---

in partial fulfillment

of the requirements for the degree

Doctor of Philosophy

by

Erik P. Vargo

December

2013

APPROVAL SHEET

The dissertation

is submitted in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

AUTHOR

The dissertation has been read and approved by the examining committee:

Randy Cogill

Advisor

Stephen Patek

Alfredo Garcia

Peter Beling

Gang Tao

Accepted for the School of Engineering and Applied Science:

Dean, School of Engineering and Applied Science

December

2013

The University of Virginia

Doctoral Dissertation

Expectation-maximization for Bayes-adaptive POMDPs

Author:

Erik P. Vargo

Committee:

Dr. Randy Cogill (Advisor)

Dr. Stephen Patek (Chair)

Dr. Peter Beling

Dr. Alfredo Garcia

Dr. Gang Tao

A dissertation submitted in fulfilment of the requirements

for the degree of Doctor of Philosophy

in the

Department of Systems & Information Engineering

December 2013

Declaration of Authorship

I, Erik P. VARGO, declare that this dissertation titled, "Expectation-maximization for Bayes-adaptive POMDPs" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this dissertation has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this dissertation is entirely my own work.
- I have acknowledged all main sources of help.
- Where the dissertation is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

THE UNIVERSITY OF VIRGINIA

Abstract

Department of Systems & Information Engineering

Doctor of Philosophy

Expectation-maximization for Bayes-adaptive POMDPs

by Erik P. Vargo

Partially observable Markov decision processes, or POMDPs, are used extensively in modeling the complex interactions between an agent and a dynamic, stochastic environment. When all model parameters are known, near-optimal solutions to the reward maximization problem can be obtained through approximate value iteration. Unfortunately, in many real-world applications a POMDP formulation may not be justified due to uncertainty in the underlying hidden Markov model parameters. However, if model uncertainty can be characterized by a prior distribution over the state-transition and observation-emission probabilities, it is natural to seek *Bayes optimal* policies which maximize the expected reward subject to this distribution. The coupling of a POMDP with a model prior was recently formalized as the Bayes-adaptive POMDP (BAPOMDP) and various online and offline algorithms have since been proposed for this class of problems, the most popular of which are inspired by approximate POMDP value iteration. Despite its success when applied to small benchmark BAPOMDPs, empirical results suggest that value iteration may be inadequate as the degree of model uncertainty increases. As an alternative, in this dissertation we explore expectation-maximization approaches to solving BAPOMDPs, which have the potential to scale more gracefully with both the number of uncertain model parameters and their assumed variability.

Acknowledgements

I would like to thank my committee for their service during the completion of my dissertation. In particular, the direction and guidance of my advisor, Dr. Randy Cogill, has been indispensable both in regards to my research endeavors and broader postgraduate goals. The freedom I have had to explore unfamiliar yet exciting topics has undoubtedly contributed to my growth as an independent researcher. I also extend my gratitude to Dr. Ellen J. Bass, who served as my advisor and mentor during my first two years at the University of Virginia, helped me to develop as a technical writer, and played a major role in my decision to enroll in the program. Last but not least, I would like to thank my parents, Tom and Jo, for their unconditional support throughout my academic career and beyond.

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

Markov decision processes, or MDPs, provide a general framework for modeling the complex interactions between an agent and a dynamic, stochastic environment. As such, the applications of MDPs are broad, spanning the disciplines of machine learning, artificial intelligence, operations research, and many others. Informally, a Markov decision process is a discrete-time, stochastic process that transitions among a finite number of states subject to the control of an agent. The agent's action and system's state parameterize the immediate state-transition probabilities, which by assumption of the *Markov property* are independent of all previously visited states. Furthermore, during each time period the agent's action and system's state generate a reward via a deterministic reward function. With this in mind, an optimal MDP policy is a state-to-action mapping that maximizes the discounted expected reward over a possibly infinite horizon. In this dissertation we restrict our attention to *stationary* Markov processes, for which the state-transition probabilities are time-invariant.

Efficient polynomial-time algorithms, such as value iteration and policy iteration, exist for solving large-scale MDPs (Bertsekas, 1995). However, in many applications the

system's state is not directly visible to the agent and hence an MDP formulation is not appropriate. Consider the manufacturing problem of tool replacement, for example, where the goal is to replace a machining tool only once accrued wear significantly affects product quality (Fish, 2001; Fish et al., 2003; Cetin and Ostendorf, 2004; Kunpeng, 2007; Wang and Wang, 2012). While a tool's condition at any time is not visible to the agent—and is therefore said to be "hidden"—observations that are correlated with tool condition can usually be derived from force, acoustic, and temperature signals captured in real-time from the machining surface. Subsequently, these observations can be used to infer a tool's condition via statistical methods in an online setting. In a more general sense, if the relationships between states and observations can be formally characterized by probability densities, then the resulting observation-emission and state-transition distributions define a hidden Markov model (HMM); and when these distributions are allowed to vary subject to the control of an agent and a reward function is introduced, the result is a partially observable Markov decision process (POMDP). Because POMDPs can be interpreted as MDPs with a continuous *belief space*, the value iteration and policy iteration algorithms of MDPs have natural extensions to the POMDP setting. However—with more to be said in the literature review of Chapter 2—the introduction of a continuous belief space significantly complicates the analysis of POMDPs, which have been classified as PSPACE-complete (Papadimitriou and Tsitsiklis, 1987). In particular, exact implementations of POMDP policy iteration and value iteration have a worst-case exponential complexity and are only tractable for problems of a trivial size, so that approximate variations must be employed.

To further complicate matters, the underlying HMM parameters—that is, the state-transition and observation-emission probabilities—are generally not known *a priori*. A common simplification is to form an HMM point estimate given historical training data

and a prior belief over the unknown model parameters. Once a point estimate is obtained, a policy is then derived via standard POMDP methods under the assumption that the fitted model is true. This approach has two primary shortcomings. First, estimating the usual mean and mode point estimates is itself non-trivial: In partially observable settings, training data will often be in the form of unsupervised observation-action trajectories. As such, the posterior probability function over HMM parameters will generally be multimodal so that hill climbing procedures for computing the mode—such as gradient ascent (Baldi and Chauvin, 1994; Bagos et al., 2004) and expectation-maximization (Baum et al., 1970; Dempster et al., 1977)—will only return locally optimal parameter estimates. Alternatively, Gibbs' procedure (Cappé et al., 2005; Rydén, 2008) can be used to estimate the mean HMM given a prior distribution and training data, but such Markov chain Monte-Carlo algorithms are known for their notoriously slow convergence. Second, and perhaps more importantly, the "point-based" policies generated via this two-stage approach of (i) model fitting and (ii) subsequent policy optimization lack robustness with respect to prior model uncertainty. In particular, a point-based policy will only perform well when its corresponding point estimate is "close" to the true model. A natural alternative that fully accounts for model uncertainty in the optimization phase is to seek the *Bayes optimal* policy with respect to the full model prior. The resulting decision problem—recently formalized as the *Bayes-adaptive* POMDP (BAPOMDP)—is the primary focus of this dissertation.

1.1 Problem statement

In this section we define the BAPOMDP, justify our convention for evaluating BAPOMDP policies, and discuss why we have chosen expectation-maximization as our approach to solving this class of decision problems.

1.1.1 Formalizing the BAPOMDP

BAPOMDPs are POMPDs coupled with a prior distribution over the HMM parameters. Whereas optimal POMDP policies can be sufficiently defined by maintaining a belief over the state space, optimal BAPOMDP policies—or Bayes optimal policies—must operate on a belief over both the state space *and* the model space. Intuitively, this is so because process information acquired over time updates the model posterior (or model belief), which in turn influences the perceived effect of future actions on expected reward. In this sense, Bayes optimal policies are *model-adaptive*. By quantifying the value of information in this way, Bayes optimal policies provide an unambiguous answer to the fundamental problem of reinforcement learning, that of balancing "exploration" against "exploitation". Here exploration refers to decision-making with the goal of increasing one's knowledge of the system, and exploitation refers to decision-making with the immediate goal of maximizing one's reward given current system knowledge. While there exist both offline and online approaches to reinforcement learning under conditions of model uncertainty, only offline policies—which can potentially plan for all possible histories *a priori*—are capable of achieving a truly optimal balance in the Bayesian sense. As such, we devote our attention to offline approaches in this dissertation, but for the interested reader we provide a brief review of online methods for BAPOMDPs in Section 2.3.3.

To construct a BAPOMDP, we begin with a POMDP for which the underlying hidden Markov model is composed of discrete state, action, and observation spaces denoted by X, A, and O, respectively. The collection of all parameters for the underlying HMM is denoted by the vector θ, which consists of state-transition and observation-emission probabilities such that $\theta_{x,a}^{x'} = p_\theta(x_{t+1} = x' | x_t = x, a_t = a)$ and $\theta_{x',a}^{o'} = p_\theta(o_{t+1} = o' | x_{t+1} = x', a_t = a)$. Here the subscript represents a conditional dependence on θ, and we use $p_\theta(\cdot)$ to denote a prior distribution over models $\theta \in \Theta$. We assume that the initial state distribution $p_0(\cdot)$ is fixed, although the extension to the more general case is straightforward. In each time

period, a reward $u(x_t, a_t)$ is earned based on the current state and action. We assume, without loss of generality, that the reward function $u : A \times X \mapsto \mathbb{R}$ is nonnegative. We denote an instance of a BAPOMDP by $\mathcal{P} = \{X, A, O, u, \gamma, T, p_\theta\}$, where $0 \leq \gamma \leq 1$ is a discount factor that will be applied when evaluating the reward earned over the finite planning horizon $T \geq 0$.

The objective of $\mathcal{P}$ is to find a policy that maximizes the discounted, finite-horizon expected reward for the stochastic process. A sufficient statistic for optimal behavior in this domain is the history of observations and actions realized by the current time. Let

$$o^t = (o_1, o_2, \ldots, o_t) \quad \text{and} \quad a^t = (a_0, a_1, \ldots, a_t)$$

denote sequences of observations and actions up to time t, and let O^t and A^t denote the set of all possible observation and action sequences up to time t. In the most general setting, a policy is a sequence of mappings $\pi = \{\pi_0, \ldots, \pi_T\}$, where $\pi_t : O^t \times A^{t-1} \to A$ is the observation-action mapping used at time t. Therefore, the objective is to find a policy π that maximizes

$$\begin{aligned} J(\pi) &= \sum_{t=0}^{T} \mathbf{E}_\theta \left[\gamma^t u \left(\pi_t(o^t, a^{t-1}), x_t \right) \right] \\ &= \int_\theta p_\theta(\theta) \sum_{t=0}^{T} \mathbf{E} \left[\gamma^t u \left(\pi_t(o^t, a^{t-1}), x_t \right) | \, \theta \right] \, d\theta. \end{aligned} \tag{1.1}$$

Of course, it is well-known that optimal (BA)POMDP policies can be represented more compactly, for example, by collections of α-vectors over the corresponding belief space (see Section 2.2.1). Note that when $\gamma < 1$ we can also consider the infinite-horizon variation to the above problem by letting $T \to \infty$. Furthermore, given this definition of

BAPOMDP, a POMDP can be interpreted as a special case where the model prior p_θ places all mass on a single HMM θ, which we denote by $\mathcal{P} = \{X, A, O, u, \gamma, T, \theta\}$.

1.1.2 Evaluating BAPOMDP policies

Recall that a BAPOMDP arises when a prior distribution p_θ is used to represent a belief over the unknown model parameters θ. Historically, the standard for evaluating BAPOMDP policies is to solve the BAPOMDP subject to the prior p_θ, and then evaluate via Monte-Carlo simulation how well the resulting policy performs over time with respect to a *single* chosen model $\theta^* \in \Theta$, by comparing its performance to that of the optimal policy under θ^* over a succession of "episodes" (Wang et al., 2012; Ross et al., 2011). However, the choice of θ^* here is rather arbitrary and a positive result only indicates adaptivity with respect to θ^*, rather than adaptivity with respect to all models possible under p_θ. In other words, this metric is generally a poor indicator of robustness against model uncertainty, which is the desired characteristic of BAPOMDP policies. We do concede that empirical evaluation with respect to a single model θ^* is justified when θ^* and the BAPOMDP prior p_θ are coupled in a natural way, for example, when p_θ is conditioned on historical training data generated by θ^*. In fact, we adopt this convention in the case study introduced in Section 1.2. Elsewhere, though, our convention will instead be to evaluate policies with respect to the full model prior p_θ, that is, subject to the actual BAPOMDP objective (1.1).

1.1.3 Why expectation-maximization?

Notably, any BAPOMDP can be formulated as an equivalent POMDP in which the uncertain model parameters are explicitly embedded in the POMDP state space, a construction first proposed by Duff (2002) in the context of Bayes-adaptive MDPs and later extended

to partially observable domains by Ross et al. (2008). It follows that, at least in theory, POMDP value iteration can be adapted to solve BAPOMDPs. Unfortunately, exact value iteration has a worst-case exponential complexity that is further compounded by the augmented state space introduced in the BAPOMDP-to-POMDP conversion. Still, the underlying principles of dynamic programming have motivated both offline and online algorithms for approximate planning in BAPOMDPs, such as point-based value iteration (PBVI) (Wang et al., 2012) and related heuristics which operate on local approximations to the model posterior (Doshi et al., 2008; Dallaire et al., 2009; Dearden et al., 1999; Duff and Barto, 1997; Ross et al., 2011). A more thorough review of these approaches can be found in Chapter 2.

A property shared by the vast majority of approximate value iteration algorithms is that the underlying (BA)POMDP belief space must be discretized to maintain tractability, thus requiring the modeler to define a mechanism for sampling beliefs given a predefined granularity. This leads to a general class of algorithms in which the value iteration backup is performed over a finite set of beliefs. Belief sampling poses a significant challenge since performance will likely be sensitive to the chosen mechanism of discretization, the most appropriate of which is certain to vary from one application to another. As evidence to this, in the standard POMDP setting an empirical study comparing various belief point expansion heuristics for PBVI algorithms (Pineau et al., 2006) suggests that a heuristic's performance will be a function of both the problem domain and various tunable parameters, but not necessarily in a predictable way.

An alternative to POMDP value iteration is motivated by the fact that optimal finite-horizon POMDP policies can be encoded as finite-state controllers, or FSCs (Kaelbling et al., 1998; Hansen, 1998b), which provide an automated approach to belief space discretization. Loosely speaking—with a formal definition provided in Chapter 2—a *bounded* finite-state controller is a graphical model defined on a finite set of abstract "belief nodes" that are

analogous to a discrete approximation of the continuous POMDP belief space. The agent's current position in the graph dictates the (possibly stochastic) mechanism by which the next action is chosen, and each new observation results in a (possibly stochastic) transition from the current node to an adjacent node. FSC optimization, then, amounts to jointly optimizing the conditional action and transition distributions at each node in the graphical model. Importantly, the behavior at each node in an optimal FSC is determined solely by the POMDP parameters and requires no additional belief sampling, parameter tuning, or other modeler intervention. Furthermore, stochastic finite-state controllers are generally more parsimonious than deterministic value iteration-based policies, which could require hundreds (if not thousands) of multi-dimensional α-vectors to achieve near-optimal performance. Because the optimization of bounded FSCs is NP-hard in the POMDP setting (Meuleau et al., 1999), existing approaches limit their search to locally optimal controllers. The most popular approaches of this type are policy-gradient (that is, gradient ascent with respect to the FSC parameters) (Meuleau et al., 1999) and expectation-maximization (Toussaint et al., 2010; Barber and Furmston, 2009), which casts the reward maximization problem as one of likelihood maximization. While both EM and policy-gradient have a polynomial-time iteration complexity, Toussaint et al. (2010) show that EM scales more gracefully with problem size in an empirical study. Furthermore, EM guarantees a monotonic improvement to the objective without the additional overhead of a line search.

Recently, EM has emerged as a scalable, lightweight approach to solving moderately sized benchmark POMDPs that is competitive with state-of-the-art PBVI algorithms when coupled with subroutines for escaping local optima (Poupart et al., 2011b). This result, and the ability to convert any BAPOMDP into an equivalent POMDP, suggest that EM could also compete with value iteration in the more general BAPOMDP setting. While PBVI is arguably the preferred method for solving *standard* POMDPs—owing to the success of heuristic belief sampling strategies in large domains—we will show that EM offers

several advantages when applied to BAPOMDPs. In particular, by casting the reward maximization problem as one of likelihood maximization, various probabilistic inference techniques can be applied to ensure the comparative scalability of EM with respect to the degree of model uncertainty.

We now take a step back and introduce a manufacturing case study that shows the unreliability of point-based policies and supports our adoption of the Bayes-adaptive POMDP framework. After developing theoretical tools to solve BAPOMDPs via expectation-maximization in subsequent chapters, we return to this case study in Section 4.8 to demonstrate the success of our algorithms in a more realistic setting.

1.2 A case study in manufacturing

In this case study we were tasked with improving the current *broach* replacement strategy for a machining process at the Rolls-Royce facility in Indianapolis. A broach is a sophisticated tool used to machine complex patterns in metal with a single cutting pass. The development of successful broach replacement strategies is a non-trivial task: direct assessment of wear after each cutting pass is not practical due to the high cost of stopping production, by which the decision to replace a broach must be made subject to uncertainty in the broach's condition. The current replacement strategy at Rolls-Royce uses a broach for an *a priori* fixed number of cutting passes, K, before replacing it with a new piece. This deterministic strategy has two notable shortcomings. First, the broach is often in a good condition after K cutting passes and does not need to be replaced. Second, it is possible for the broach to become sufficiently worn (e.g., chipping, dulling) prior to the completion of K cuts, leading to an unacceptable final product with continued use. Fortunately, sensor measurements in the form of signals (e.g., force, vibration, acoustic emissions) can be collected from the machining surface during each cutting pass, from

which various useful *features* can be derived. Features could include a signal's root mean square value, maximum absolute value, or other statistics derived from the frequency domain, such as the sum of log energies or maximum energy. If correlations can be established between features and broach quality, then these features can be used to infer the broach's condition in real-time and inform more sophisticated adaptive replacement strategies, which can potentially maximize a broach's yield by replacing it only once it has outlived its useful life.

Initially, we sought to construct adaptive broach replacement strategies by casting the problem as a POMDP. Motivated by recent tool condition monitoring studies (Fish, 2001; Fish et al., 2003; Cetin and Ostendorf, 2004; Kunpeng, 2007; Wang and Wang, 2012), we modeled broach wear via a left-to-right hidden Markov model on n states—which enforces the physical constraint of nondecreasing wear—and derived sensor-based features to serve as informative observation-emissions. A reward structure was imposed such that: a reward of $r > 0$ is received when the broach is used in one of the first $n - 1$ states, all of which indicate an acceptable condition; a cost of $c_1 > 0$ is incurred when the broach is replaced; and a cost of $c_2 > c_1$ is incurred when the broach is used in the n^{th} state, which indicates an unacceptable condition. When the broach is replaced the Markov chain transitions deterministically to state 1, indicating that a new broach has been mounted. Assuming an infinite horizon and a discount factor of $\gamma < 1$, the HMM and reward function define a POMDP, the solution of which is an optimal broach replacement policy. See Figure 1.1 for a graphical illustration of the broach replacement problem. Initially, we adopted the ubiquitous approach of forming point estimates of the unknown HMM and subsequently deriving policies from these fitted models, as we now describe.

To formulate a POMDP for the replacement problem we trained an HMM from historical broaching data. While the initial data provided by our sponsor was inadequate for this

purpose, it inspired a simulation-based approach to generating synthetic data more appropriate for model training and validation in the short-term. The synthetic data consisted of feature sequences collected over K cutting passes—as required by current operating procedures at Rolls-Royce—where the same broach (initially fresh) was used for each cut. Furthermore, each sequence was partially supervised with a single binary value indicating whether the broach was in the unacceptable wear state ($x_{K-1} = n$) during the K^{th} and final cutting pass. In each trial the same generative HMM θ^*—naturally, hidden from our learning algorithm—was used to produce the training data. Furthermore, we imposed uninformative, independent Dirichlet priors on the state-transition and observation-emission probability vectors. Given the prior distribution and training data, two approaches were considered for learning an HMM: (i) we employed a variation of Baum-Welch (a form of expectation-maximization) to approximate the maximum *a posteriori* (MAP) HMM from the model posterior; and (ii) we used Gibbs' procedure to approximate the mean HMM from the model posterior. Both training algorithms were modified to account for the partial supervision with respect to the K^{th} cutting pass in each trial.

Next, we used a standard implementation of point-based value iteration (Cassandra, 2009) to approximate optimal policies under the mean and mode point estimates assuming an infinite horizon and a discount factor of $\gamma = 0.99$, with $r = 1$, $c_2 \in \{5, 10, 15\}$ and $c_1 = \frac{1}{2}c_2$. The values of r, c_1, and c_2 used here had no real-world origin, but were otherwise reasonable given the application. Importantly, the infinite planning horizon allowed for adaptive replacement strategies in which the broach could be used beyond the default K cutting passes. In this study the model posterior (and hence the mean and mode point estimates) were naturally coupled to the generative model θ^* via training data, which justified evaluating policy performance with respect to the single model θ^* (recall the discussion of Section 1.1.2).

The generative HMM θ^* for our empirical study was defined as follows. First, we set

$n = 7$ and $p_{x|x} = 0.8$ for all $x < n$. For simplicity, we assumed a single feature that could take on $|O| = |X|$ discrete values, and for each state $x' = 1, 2, \ldots, n$ the feature density $p(\cdot|x')$ was distributed according to the rule $p(o'|x') \propto n - |o' - x'|$ for each possible feature value $o' = 1, 2, \ldots n$. This convention served to correlate more highly states and features that were closer in index. As indicated above, only the final observation in each sequence was supervised to indicate whether the final broach condition was acceptable ($x_{K-1} < n$) or unacceptable ($x_{K-1} = n$).

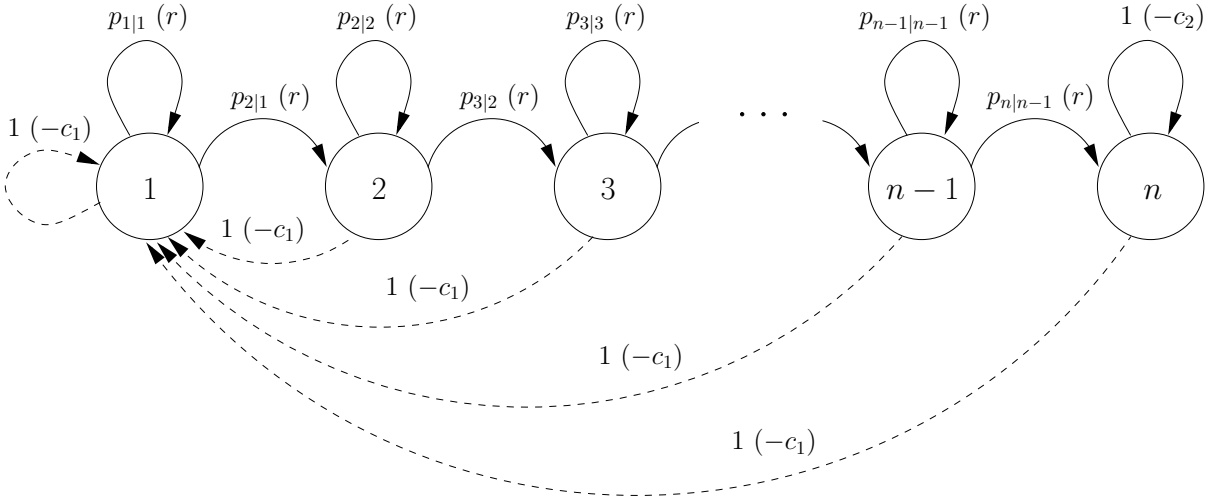

Figure 1.1: Diagram of the POMDP from the Rolls-Royce case study. Arcs are labeled with their corresponding state-transition probabilities and the associated rewards follow in parentheses. Solid arcs indicate state-transitions without replacement, and dashed arcs indicate broach replacement. For simplicity, the observation-emission distributions are not included.

Figure 1.2 contains performance plots for nine independent trials of the experiment, where in each case the same generative HMM θ^* was used to produce $L = 20$ independent observation sequences of length $K = 30$ for model training. The plots in Figure 1.2 indicate that both the mean and mode point-based policies are generally inadequate when compared to those derived from the true HMM θ^* (with their performance indicated by dashed, dotted, and solid lines, respectively), and that neither the mean nor the mode point estimate is exclusively preferred over the other. The poor performance of the mean

and mode point-based policies suggests that the modeling error introduced by the *a priori* resolution of parameter uncertainty onto a single HMM can be prohibitive. As a result, our attention turned towards the construction of model-adaptive policies which—given a model prior—resolve uncertainty through an optimal Bayesian interaction with the system.

1.3 Outline

The outline of this dissertation is as follows: In Chapter 2 we provide relevant background, including a summary of progress in the area of BAPOMDP research and a discussion on the limitations of existing approaches. In Chapter 3 we demonstrate through a tractable yet non-trivial BAPOMDP that even the *best* point-based policy can significantly underperform the Bayes optimal, model-adaptive policy. Having argued for the superiority of Bayes optimal policies in this way, we proceed to derive an EM algorithm for BAPOMDPs in Chapter 4. Due to the intractability of the M-step in the general case, we propose two efficient alternatives. The first is a sampling-based EM algorithm that operates on a discrete subset $\tilde{\Theta}$ of the model space Θ, and the second is a variational Bayes EM algorithm that accommodates a model prior given by a product of independent Dirichlet distributions. Furthermore, these approaches are evaluated against approximate value iteration in an empirical study, which includes a return to the Rolls-Royce broach replacement problem. We consider various practical techniques for addressing the slow convergence rate of EM and the local optimality of EM fixed points in Chapter 5. Finally, we close with a summary and suggestions for future work in Chapter 6.

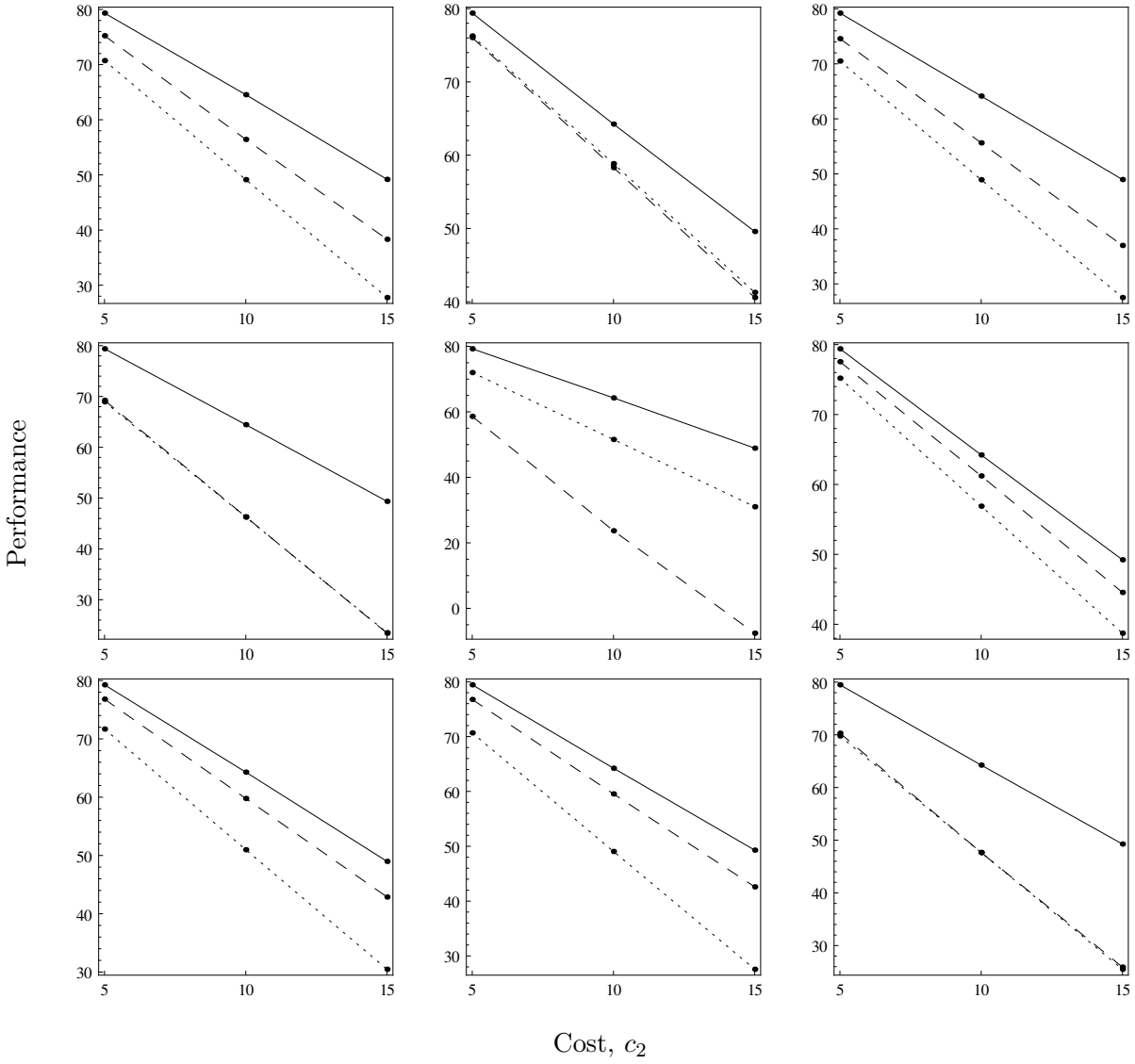

Figure 1.2: Performance plots for the Rolls-Royce case study using synthetic broaching data for model training. Solid lines indicate performance of policies derived from the true model θ^*, dashed lines indicate performance of policies derived from the mode point estimates, and dotted lines indicate performance of policies derived from the mean point estimates. Each plot corresponds to a unique set of training data generated by θ^*.

Chapter 2

Background

In this chapter we present background material which motivates our approach and provides a suitable context for the dissertation. To begin, we review the expectation-maximization algorithm in its most general form, as it features prominently in our original contributions. Then, we summarize existing POMDP solution techniques before examining the current state-of-the-art for solving BAPOMDPs. We close by placing our contributions within the context of existing literature.

2.1 Expectation-maximization

Expectation-maximization is an iterative algorithm for maximizing a likelihood that is marginalized over a set of latent variables. EM has been used extensively in statistical inference applications—such as computing the maximum likelihood parameters for hidden Markov models (Rabiner, 1989; Cappé et al., 2005) and Gaussian mixture models (Bilmes, 1998)—and is often preferred to other approaches owing to its simplicity and ease of implementation. In particular, EM offers an efficient alternative to gradient ascent,

which requires expensive gradient computations and a line search procedure to guarantee monotonicity. During each iteration, EM forms a lower bound to the likelihood in the neighborhood of the current parameter estimate via Jensen's inequality (E-step) and then maximizes this lower bound to obtain parameter updates that monotonically improve the original objective (M-step). Importantly, analytical parameter updates can often be derived when the lower bound is concave. We now provide a more detailed description of the EM algorithm along with a self-contained proof of monotonicity.

2.1.1 Monotonicity of convergence

Consider a statistical process characterized by latent variables z, observables x, and a collection of free parameters π that describe how x and z are generated. Suppose that our goal is to maximize the likelihood

$$L(x;\pi) = p(x|\pi)$$

with respect to the free parameters π, and that direct maximization of this quantity is not tractable. Let $q(z)$ be an arbitrary distribution over the latent variables z, and consider the Kullback-Leibler (KL) divergence (Kullback, 1968) between $q(z)$ and $p(z|x,\pi)$. Noting the non-negativity of KL divergence, we have

$$\begin{aligned}
\mathrm{KL}(q(z), p(z|x,\pi)) &= \langle \log q(z) \rangle_q - \langle \log p(x,z|\pi) \rangle_q + \langle \log p(x|\pi) \rangle_q \\
&= \langle \log q(z) \rangle_q - \langle \log p(x,z|\pi) \rangle_q + \log p(x|\pi) \qquad (2.1) \\
&\geq 0.
\end{aligned}$$

We can rearrange the terms in the above inequality to obtain

$$\log p(x|\pi) \geq -\langle \log q(z) \rangle_q + \langle \log p(x, z|\pi) \rangle_q. \tag{2.2}$$

This suggests that in place of optimizing $L(x; \pi) = p(x|\pi)$ directly, we can instead optimize the lower bound (2.2). To this end, we adopt the coordinate ascent approach of Neal and Hinton (1998) and iteratively (i) maximize the lower bound (2.2) with respect to q for fixed π (E-step), and then (ii) maximize the lower bound (2.2) with respect to π for fixed q (M-step).

When carrying out the E-step for fixed π, maximizing the lower bound is equivalent to minimizing $\mathrm{KL}(q(z), p(z|x, \pi))$ with respect to q. By properties of KL divergence, the global minimizer is $q(z) = p(z|x, \pi)$. In the M-step, the goal is to maximize the lower bound (2.2) with respect to π given fixed q, that is, we solve $\max_\pi \langle \log p(x, z|\pi) \rangle_q$, which often admits analytical solutions when the objective is concave in π. Furthermore, when the E-step and M-step are carried out exactly, EM guarantees a monotonically increasing objective with respect to consecutive parameter updates π' and π''.

Proposition 1: *If π' and π'' are consecutive parameter updates generated by the* EM *algorithm, then* $\log L(x; \pi'') \geq \log L(x; \pi')$.

Suppose that π' is the current parameter estimate, and let $q_{\pi'}$ denote the subsequent E-step update to q. Given an arbitrary π, let $p_\pi(z) = p(z|x, \pi)$ and define

$$H(\pi, \pi') = \mathrm{KL}(q_{\pi'}, p_\pi) \tag{2.3}$$

$$Q(\pi, \pi') = \langle \log p(x, z|\pi) \rangle_{q_{\pi'}}. \tag{2.4}$$

From the identity in equation (2.1) we can write

$$\begin{aligned}
\log p(x|\pi) - \log p(x|\pi') &= [\langle \log p(x, z|\pi) \rangle_{q_{\pi'}} - \langle \log p(x, z|\pi') \rangle_{q_{\pi'}}] + \\
&\qquad [\mathrm{KL}(q_{\pi'}, p_\pi) - \mathrm{KL}(q_{\pi'}, p_{\pi'})] \\
&= [Q(\pi, \pi') - Q(\pi', \pi')] + [H(\pi, \pi') - H(\pi', \pi')] \\
&= [Q(\pi, \pi') - Q(\pi', \pi')] + H(\pi, \pi'),
\end{aligned}$$

where we use the fact that $H(\pi', \pi') = 0$. The M-step maximizes $Q(\pi, \pi')$ with respect to π, so that by definition $\pi'' = \mathrm{argmax}_\pi \, Q(\pi, \pi')$. Clearly $Q(\pi'', \pi') - Q(\pi', \pi') \geq 0$, and $H(\pi, \pi') \geq 0$ by properties of KL divergence, so that $\log L(x; \pi'') \geq \log L(x; \pi')$. Therefore each iteration of EM results in a monotonic increase to the objective $L(x; \pi)$. ■

This result is quite general, and in cases where $\log L(x; \pi)$ satisfies mild smoothness conditions we can often make the stronger claim that $\log L(x; \pi'') > \log L(x; \pi')$ (Wu, 1983; Little and Rubin, 1987). For example, we will find the EM algorithm for BAPOMDPs to have this stronger characterization (Chapter 4). Unfortunately, even when a strict improvement is guaranteed the convergence rate can be slow and, moreover, the corresponding EM fixed point may be of poor quality relative to the global optimum. Regarding the latter concern, a *multi-start* approach in which EM is initialized in different regions of the parameter space can be useful for uncovering multiple fixed points, from which the best is then selected. While to our knowledge multi-start is the only *general* approach for addressing the local optima issue, we will later consider subroutines for escaping local

optima in the specific context of EM for BAPOMDPs (Chapter 5). On the other hand, there do exist a variety of general strategies for accelerating EM convergence, which we now take a moment to review.

2.1.2 Acceleration methods

Various methods have been developed to accelerate the slow convergence rate of EM. Arguably the simplest, *parameterized* EM updates the current parameter estimate $\pi^{(k)}$ by first computing the subsequent EM update $\pi_{\text{EM}}^{(k+1)}$ and selecting $\pi^{(k+1)}$ such that

$$\pi^{(k+1)} = \pi^{(k)} + \Delta^{(k)}(\pi_{\text{EM}}^{(k+1)} - \pi^{(k)}), \tag{2.5}$$

for some appropriately chosen positive scalar $\Delta^{(k)}$ (Ortiz and Kaelbling, 1999). (Note that $D^{(k)} = 1$ in (2.5) recovers the ordinary EM algorithm.) In this way, the difference $\pi_{\text{EM}}^{(k+1)} - \pi^{(k)}$ approximates the true gradient of the likelihood in the neighborhood of $\pi^{(k)}$. While the optimal step-size at each iteration can be obtained via spectral analysis of the EM update's Jacobian matrix, the required computations are generally intractable (Roland, 2010).

The expectation-conjugate gradient (ECG) algorithm (Jamshidian and Jennrich, 1993) refines this approach by using a generalized conjugate gradient to choose the optimal step-size $\Delta^{(k)}$ in the direction of the EM update $\pi_{\text{EM}}^{(k)}$. While conjugate gradient methods typically require the computation of second-order derivatives, ECG avoids this complication by leveraging the EM search direction. However, a line search is still required to optimize $\Delta^{(k)}$, which can be prohibitive. As a result, it is of interest to determine when the solution quality of an ECG update outweighs the efficiency of an EM update.

Salakhutdinov et al. (2003) identified a connection between the convergence rate of EM and the ratio of missing information to complete information in the neighborhood of a local optimum. When the ratio is large, EM convergence will be slow; when the ratio is small, EM will exhibit super-linear, Quasi-Newton convergence. With this in mind, a hybrid EM-ECG algorithm is proposed that switches from ordinary EM updates to conjugate gradient updates when the ratio of missing information exceeds a threshold. An alternative to hybrid EM-ECG is *scaled* ECG (Fischer and Kersting, 2003), which circumvents the complexity of the ECG line search via the introduction of a scaled conjugate gradient. In particular, scaled ECG requires only a single evaluation of the likelihood during each iteration. Empirical results suggest that scaled ECG compares favorably to EM-ECG in solution quality while offering considerable gains in efficiency. Ortiz and Kaelbling (1999) offer a more complete review and empirical comparison of accelerated EM algorithms. To our knowledge, no substantial effort has been made to accelerate the convergence of EM for (BA)POMDPs, and we address this matter in Section 5.1.

2.2 Solving POMDPs

While solving BAPOMDPs is our goal, it is worth reviewing standard POMDP solution techniques due to the close relationship between the two problem types. Generally, existing approaches for solving POMDPs can be grouped into two categories: (i) value function optimization, and (ii) direct policy optimization.

2.2.1 Value function optimization

Value function approaches seek to determine or approximate the optimal POMDP value function, from which a policy can then be extracted. Let $\mathcal{P} = \{X, A, O, u, \gamma, T, \theta\}$ be

a standard POMDP corresponding to a known HMM θ. We use b to denote the belief state vector of dimension $|X|$, so that $b(x)$ is the probability that the system is in state $x \in X$ given the history of observations and actions. It is well-known that the belief state is a sufficient statistic for computing optimal POMDP policies. Furthermore, let $\bar{u}(b,a) = \sum_x b(x)u(x,a)$ be the immediate expected reward associate with taking action a in belief state b.

The Bellman optimality equations associated with the T-horizon POMDP $\mathcal{P}$ are given by

$$V_t(b) = \max_a \ \bar{u}(b,a) + \gamma \sum_{o'} \tau(o'|b,a) V_{t-1}(b_a^{o'}), \tag{2.6}$$

where $\tau(o'|b,a)$ is the probability of observing o' after taking action a in belief state b, $b_a^{o'}$ is the updated belief after taking action a and observing o', and $V_t(b)$ is the expected reward associated with following the optimal policy for the t-horizon subproblem when starting in belief state b. More explicitly,

$$\tau(o'|b,a) = \sum_x b(x) \sum_{x'} p(x'|x,a) p(o'|x',a) \tag{2.7}$$

and

$$b_a^{o'}(x') = \frac{p(o'|x',a) \sum_x b(x) p(x'|x,a)}{\tau(o'|a,b)}. \tag{2.8}$$

The goal of value function optimization is to compute the optimal t-horizon value functions V_t via the dynamic programming recursions (2.6). Existing methods can be categorized as either exact or approximate.

2.2.1.1 Exact methods

Given the initial zero-horizon value function V_0, the exact dynamic programming backups (2.6) can be executed recursively for each $1 \leq t \leq T$, and the optimal actions enumerated over the set of reachable beliefs. Unfortunately, this brute force approach is intractable in the general case, as the set of reachable beliefs grows exponentially in the time horizon. The efficiency of the backup operations can be improved considerably by noting that the optimal finite-horizon value functions V_t will be piecewise linear and convex (PWLC) over the space of beliefs. In particular, if V_{t-1} is PWLC then it follows from the backup operation (2.6) that V_t can be represented as the upper envelope of a collection of α-vectors Γ_t, thus establishing the piecewise-linear convexity of V_t. See Figure 2.1 for an illustration. Each linear segment of V_t corresponds to a unique action, and this action is optimal for all belief states that fall within the segment's support. As a result, instead of explicitly computing the optimal action for all possible belief states in each time period, it is sufficient to compute the collection of α-vectors that define Γ_t along with the support over which each is active. Generally, the α-vector representation will be much more parsimonious than enumeration over all reachable beliefs.

Various subroutines exist for computing V_t from V_{t-1}, such as incremental pruning (Cassandra et al., 1997), Monahan's algorithm (Monahan, 1982), Witness (Littman, 1994), Sondik's one-pass algorithm (Sondik, 1971), and Cheng's linear support (Cheng, 1988). These are all exact approaches, so that despite leveraging the PWLC nature of the optimal value function, they are only tractable for POMDPs of a trivial size. In particular, if Γ_{t-1} is the current set of α-vectors then the backup operation (2.6) has a worst-case

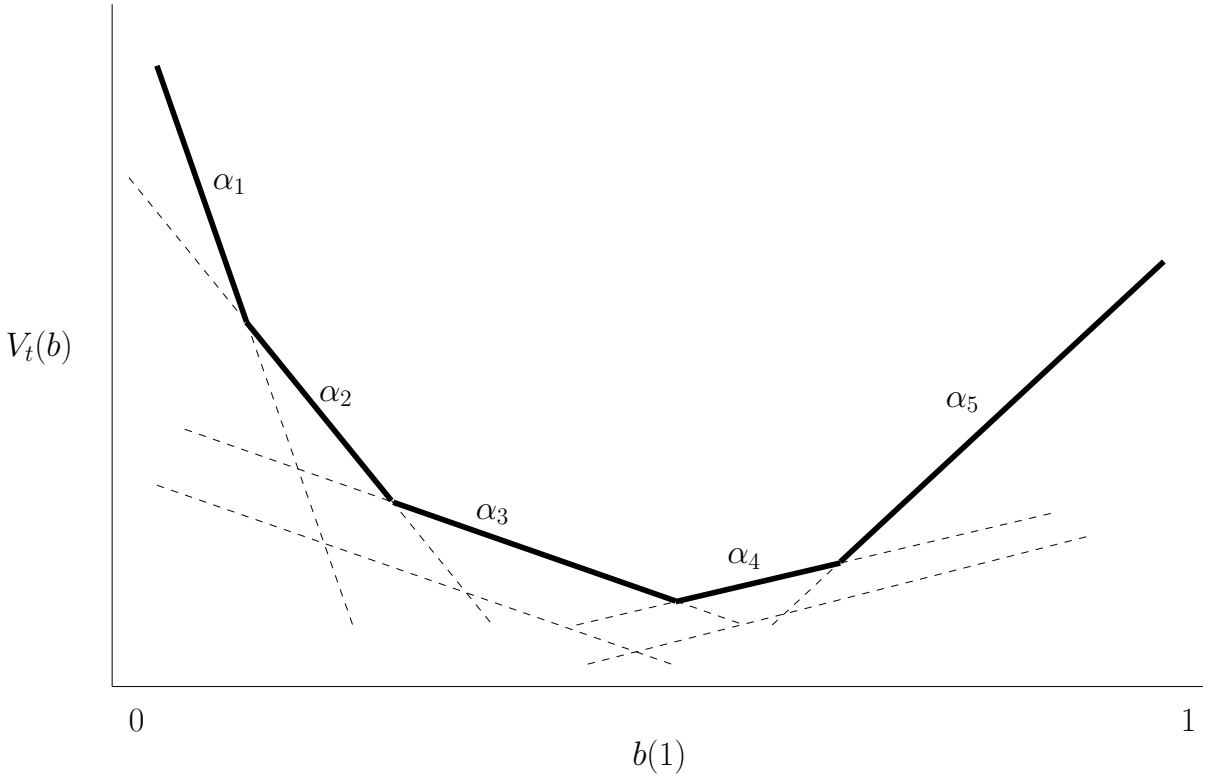

Figure 2.1: Representation of an optimal value function V_t when $|X| = 2$. The value function (in bold) is the upper envelope of a collection of linear segments, which establishes its piecewise-linear convexity.

exponential complexity of $O(|X|^2|A||\Gamma_{t-1}|^{|O|})$, which corresponds to the addition of at most $|A||\Gamma_{t-1}|^{|O|}$ α-vectors to the set Γ_t (Pineau et al., 2003).

2.2.1.2 Approximate methods

There exist numerous approximate variations to value iteration that scale more gracefully with problem size. Generally, all such approaches simplify the recursions of equation (2.6) by replacing the full set of belief states with a finite approximation. Grid-based approaches sample a finite set B of points from the belief simplex and the backup operation is limited to updating the value function at each point in the grid, and hence gradient information typically encoded by the α-vectors is not computed. As a result, interpolation must be used to approximate the value function at all non-grid points. The efficiency

and accuracy of interpolation are related to the grid's *regularity* and *resolution*, respectively. Brafman (1997) proposes a variable resolution, non-regular grid, which allows for a higher concentration of grid points in targeted regions of the belief simplex. This approach was later improved via the introduction of a variable resolution, regular grid (Zhou and Hansen, 2001).

Point-based value iteration (PBVI), unlike grid-based approaches, leverages the convexity of the optimal value function in its backup operation by updating both the value function *and* gradient information at each belief state in B. While the resulting backup is more expensive than the grid-based alternative, good results can be obtained with a significantly smaller set B. As one might expect, the error of approximating the optimal value function is related to density of B within the belief simplex (Pineau et al., 2003). In PBVI, B is usually constructed to include belief states that are encountered during a Monte-Carlo exploration of the system. The rationale is that the space of reachable beliefs is typically sparse, so that regions of the belief simplex that are unlikely to be encountered can be ignored without significant consequence to performance. Although the belief set B can be fully constructed prior to running PBVI, more sophisticated "anytime" variants begin with a small set B and intermittently expand its size with provable reductions in the error between the PBVI value function and the optimal value function (Pineau et al., 2006). A standard PBVI implementation is described by Pineau et al. (2003). Here, a backup operation is performed on all belief states in B during each iteration, resulting in the construction of $|B|$ α-vectors in polynomial time (in contrast to the exponential complexity of the exact backup operation). Alternatively, it has been shown that value iteration for infinite-horizon POMDPs will still converge to the optimal value function provided that V_t is an upper bound to V_{t-1} from the previous iteration (Zhang and Zhang, 2001). Leveraging this fact, PBVI variations have been developed that succeed in computing an upper bound $V_t \geq V_{t-1}$ by performing backups on a small subset of B

(Spaan and Spaan, 2004; Vlassis and Spaan, 2004; Spaan and Vlassis, 2005).

Lastly, heuristic search value iteration (HSVI) (Smith and Simmons, 2004) is an approach that, like PBVI, maintains a lower bound to the optimal value function by performing backups over a finite set of beliefs B. Unlike PBVI, HSVI also maintains an upper bound to the optimal value function. Whereas the lower bound encodes policy information, the upper bound is used to guide exploration and expansion of the belief set B. Furthermore, from the nature of this expansion various convergence results can be derived. While improvements to the upper bound are expensive and require solving linear programs, empirical results comparing solution time and quality suggest that HSVI is a viable alternative to PBVI for larger POMDPs. A more detailed description of PBVI and related algorithms is beyond the scope of this dissertation, so we refer the reader to a survey by Pineau et al. (2006) for an in-depth review and empirical comparison of popular PBVI algorithms and strategies for belief selection and expansion.

2.2.2 Policy optimization

The above approaches search the space of value functions, and once the optimal value function (or an approximation) is obtained, a policy can then be extracted from the value function representation. Alternatively, one may search the policy space directly. We now review some common approaches to direct policy optimization.

2.2.2.1 Finite-state controllers

Most policy iteration algorithms for POMDPs constrain their search to policies with a special structure, namely finite-state controllers (FSCs). FSCs use a finite set of belief nodes, N, to summarize the information conveyed by historical observation and action

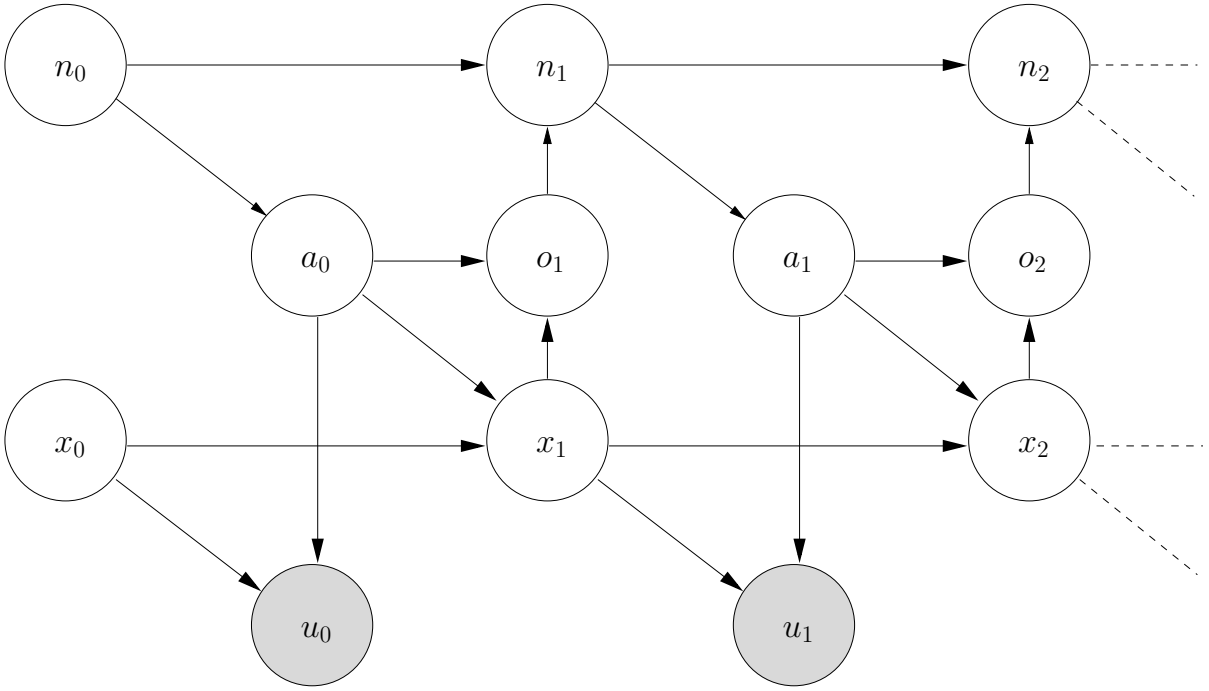

Figure 2.2: Bayesian network of a POMDP subject to a finite-state controller.

sequences about the hidden state at each time t. In time period $t = 0$, the initial belief node n_0 is drawn from a probability mass function (PMF) $\nu(\cdot)$ with support N. More generally, the belief node n_t at time $t > 0$ is drawn from the conditional PMF $\lambda(\cdot \,|\, n_{t-1}, o_t)$, which depends on both the previous belief node n_{t-1} and the most recent observation o_t. Once a belief node has been selected, an action a_t is then drawn from the conditional PMF $\pi(\cdot \,|\, n_t)$. Therefore, a complete finite-state controller is specified by $\Lambda = (\nu, \pi, \lambda)$. Although an FSC may generate belief nodes and actions randomly, this class of policies includes all possible deterministic finite-state controllers as well. In fact, FSCs are sufficient for the optimal control of any finite-horizon POMDP, although the controller's size, $|N|$, may be intractable. Figure 2.2 shows the Bayesian network of a POMDP subject to a finite-state controller.

2.2.2.2 Exact methods

Sondik (1978) presented the first exact policy iteration algorithm for infinite-horizon POMDPs. His approach—which is prohibitively complicated—iteratively refines a partition of the belief simplex, where each region corresponds to a unique action. Hansen (1998a,b) noted that any PWLC value function can be encoded as an equivalent finite-state controller, and used this fact to develop a more tractable policy iteration algorithm. Hansen's policy iteration converges to an optimal FSC through an iterative process of policy evaluation and policy improvement. The policy evaluation step reduces to solving a system of linear equations, which generates an α-vector representation of the current FSC's PWLC value function. The bottleneck of Hansen's algorithm is the policy improvement step, which transforms the current FSC into an improved FSC by backing up the current value function V to produce V', and then adding a new node to the FSC for each linear segment in V'. After this update, dominated nodes can be pruned to help moderate the FSC's size. Further details are beyond the scope of this dissertation, but monotonic convergence to an ε-optimal FSC is guaranteed after a finite number of iterations.

Hansen found policy iteration to outperform exact value iteration on a number of benchmark POMDPs. Generally speaking, value iteration requires more dynamic programming backups than policy iteration before convergence is achieved. In the MDP setting, the efficiency of the backup operation relative to policy evaluation makes value iteration the more efficient alternative. However, in the POMDP setting the worst-case exponential complexity of the backup operation, compared to the polynomial time complexity of policy evaluation, suggests that policy iteration should be preferred. Despite the improved efficiency of Hansen's approach over Sondik's, exact policy iteration—much like exact value iteration—is only tractable for prohibitively small POMDPs. Next, we discuss approximate alternatives to Hansen's algorithm that scale more gracefully with problem size.

2.2.2.3 Approximate methods

We now review approximate methods for optimizing finite-state controllers. The optimization of bounded FSCs for POMDPs is NP-hard (Meuleau et al., 1999) so that existing methods limit their search to locally optimal controllers. Given an FSC $\Lambda = \{\nu, \pi, \lambda\}$, the gradient of the objective—as a function of the parameters of Λ—can be computed in polynomial time for the purpose of gradient ascent (Meuleau et al., 1999). This *policy-gradient* approach has been used to generate locally optimal FSCs to problems that are well beyond the reach of exact value iteration and policy iteration algorithms, for example where $|X| = O(1000)$. Poupart and Boutilier (2003) supplement policy gradient with a policy iteration subroutine, which escapes local optima by adding nodes to the current FSC in a manner similar to Hansen's policy improvement step.

An EM approach to optimizing bounded FSCs was recently explored by Toussaint et al. (2010). So that EM can be applied, policy optimization must be recast as probabilistic inference. To this end, the total discounted reward for an infinite-horizon POMDP is expressed as an infinite mixture of finite-horizon POMDPs, where a single binary reward is received (with some probability) at the termination of each process. The resulting mixture can be interpreted as a likelihood over the space of FSCs. It is shown that the maximum-likelihood FSC is optimal for the original infinite-horizon POMDP, and an EM algorithm is derived for maximizing this likelihood. An alternative, but related, approach to solving finite-horizon POMDPs constructs a *reward-weighted path distribution*, the normalizing constant of which is equal to the performance of the parameterizing FSC (Barber and Furmston, 2009). A lower bound to this normalizing constant is obtained by applying Jensen's inequality, and an EM algorithm naturally arises by maximizing this lower bound, analogously to the coordinate ascent approach used in our proof of Proposition 1. While both EM and policy gradient have a polynomial-time iteration complexity, Toussaint et al. (2010) show that EM scales more gracefully with problem

size in an empirical study. Furthermore, EM guarantees a monotonic improvement to the objective without the additional overhead of a line search. We adopt the EM approach of Barber and Furmston (2009) in this dissertation due to its comparative simplicity and because it is easily extended to the infinite-horizon case (Appendix, Proposition A). Because EM for POMDPs is a special case of EM for BAPOMDPs, we postpone a detailed account of this approach until Chapter 4.

2.3 Existing BAPOMDP algorithms

We now review the current state-of-the-art for solving BAPOMDPs and highlight the limitations of existing methods. This serves to both motivate and justify our adoption of EM-based alternatives.

2.3.1 Point-based solutions

The most common approach to generating policies for BAPOMDPs is composed of two stages. The first stage artificially resolves model uncertainty by choosing a point estimate from the model prior (e.g., mean, mode). In other words, an operator M maps the model prior p_θ to a single model $\hat{\theta} = Mp_\theta$. For example, the operator $\bar{M}$ defined such that $\bar{M}p_\theta = \mathbf{E}_{p_\theta}[\theta]$ returns the mean model $\bar{\theta}$. In general, the result is a standard POMDP characterized by the HMM $\hat{\theta}$. The second stage then selects the policy that is optimal for this POMDP via existing POMDP algorithms.

The point-based policies constructed in this manner will by definition operate on a belief over the state alone and hence will not be model-adaptive. Of course, the appeal of this two-stage approach is its comparative tractability, that is, the POMDP characterized by $\hat{\theta}$ is easier to solve than the original BAPOMDP. However, as we demonstrate in Chapter 3,

even the *best* point-based policy can significantly underperform the Bayes optimal, model-adaptive policy. We now review some existing approaches that are capable of generating model-adaptive policies for BAPOMDPs, albeit at a greater computational expense.

2.3.2 BAPOMDPs as augmented POMDPs

State space augmentation is an exact approach to solving BAPOMDPs that explicitly embeds model uncertainty into the state space itself (Wang et al., 2012). This is done by constructing a new state space $Y = X \times \Theta$ that is the cross-product of the original state space X and the space of model parameters Θ. The cross-product state space imposes transformed state-transition and observation-emission distributions, which define a new "augmented" POMDP $\tilde{\mathcal{P}}$. To see how, let $y, y' \in Y$ such that $y = (x, \theta)$, $y' = (x', \theta')$, and let $o' \in O, a \in A$. Then,

$$\tilde{p}(x', \theta' | x, \theta, a) = \begin{cases} p_\theta(x'|x, a), & \theta = \theta' \\ 0, & \text{otherwise} \end{cases} \tag{2.9}$$

$$\tilde{p}(o'|x', \theta, a) = p_\theta(o'|x', a) \tag{2.10}$$

are the corresponding state-transition and observation-emission distributions. In this way, the belief vector over the augmented state space Y naturally maintains a joint belief over the state $x \in X$ *and* the model $\theta \in \Theta$. This ensures the model-adaptivity of solutions to $\tilde{\mathcal{P}}$.

For the sake of tractability, Wang et al. (2012) restrict their attention to a finite set of models $\Theta = \{\theta_1, \theta_2, \ldots, \theta_K\}$, so that the resulting augmented POMDP $\tilde{\mathcal{P}}$ can be solved with PBVI. However, state space augmentation is limited in scalability: if the θ_k are a discrete approximation to a continuous prior p_θ, it is not clear how large K must be so that the sampled models adequately reflect prior uncertainty; and as K grows, the computational expense of using PBVI to solve $\tilde{\mathcal{P}}$ will quickly become infeasible. As evidence to this, an empirical study showed that policies generated via this sampling-based PBVI approach were competitive with existing offline and online reinforcement learning algorithms when only a few model parameters were unknown, but underperformed as the number of uncertain model parameters increased, indicating that state space augmentation does not scale to domains with large amounts of parameter uncertainty (Wang et al., 2012). This is expected, since the size of the cross-product state space Y limits the number of sampled models that can be considered during the offline planning phase.

Ross et al. (2008) consider a variation to the augmentation approach for when the model prior given by a product of independent Dirichlet distributions. Because the Dirichlet is the conjugate prior to the categorical distribution, the Θ component of the cross-product state space can be represented via a compact factored form, involving pseudo-count vectors for the state-transitions and observation-emissions. Even though the resulting augmented POMDP has a countably infinite state space in the infinite horizon (with the number of reachable states growing exponentially with time), it is shown that ε-optimal solutions can be obtained by approximating the infinite state space with a finite set. As the accuracy goal increases, however, the size of this finite set grows considerably so that exact offline planning must be replaced by *belief tracking*—a form of online Monte-Carlo reinforcement learning—for the sake of tractability.

2.3.3 Online reinforcement learning

In addition to the belief tracking approach of Ross et al. (2008) for the special case of Dirichlet priors, a number of additional online reinforcement learning algorithms have been proposed for BAPOMDPs. We now briefly describe two of the more well-known approaches of this type.

Dallaire et al. (2009) assume that the BAPOMDP is a Gaussian process, for which the state, action, and observation spaces are represented by Gaussian mixtures. At each decision epoch, the MAP state sequence is computed given appropriately chosen priors and a history of observations and actions, and the final state in the sequence is used to approximate the current belief. While the underlying state-transition and observation-emission kernels are unknown, the Gaussian process model admits MAP estimates of these quantities as well. Given the MAP estimates, a sampling-based look-ahead is used to approximate the effect of immediate actions on future reward and hence inform the agent's decision. Naturally, this approach is limited to BAPOMDPS with continuous state, action, and observation spaces; and even in cases where the Gaussian process model is relevant, we suspect that using a MAP state estimate for the current belief and Gaussian process kernels would not sufficiently capture model uncertainty, a concern expressed by the authors themselves. Finally, there is significant overhead associated with the look-ahead procedure, rendering this algorithm inappropriate for time-sensitive applications.

A similar approach is taken by Doshi et al. (2008), but instead of representing the BAPOMDP as a Gaussian process, the algorithm operates on a finite number of POMDPs sampled from an unconstrained model posterior. Given a history of actions and observations, the immediate action is chosen by minimizing the Bayes risk over the set of all possible actions. To derive the Bayes risk criterion, one must be able to solve (or approximate) an optimal policy for each of the sampled POMDPs. As exploration proceeds and

new data is acquired, the posterior is intermittently updated and a new set of POMDPs is sampled. However, the convergence properties of this method require that the agent can query an oracle (at a cost) to reveal the optimal action at any time, which is not consistent with our general BAPOMDP framework. Furthermore, the computational expense of intermittently re-solving a collection of POMDPs would be prohibitive in an online setting.

2.4 Our contributions in context

As first stated in Section 1.1.1, our focus in this dissertation is on offline policies for BAPOMDPs. By planning for all possible histories *a priori*, offline policies can—at least in theory—achieve Bayes optimality, in contrast to online planning. Of course, the offline computation of Bayes optimal policies is not tractable in the general case so that online reinforcement learning merits consideration, especially for larger problems when scalability is a concern. While an empirical comparison of offline and online approaches is necessary, such a study is beyond the scope of this dissertation. Rather, our aim is to show that FSCs optimized via expectation-maximization can outperform common point-based policies and also address the deficiencies of existing offline approaches for solving BAPOMDPs. In particular, we extend the EM algorithm for standard POMDPs (Toussaint et al., 2010; Barber and Furmston, 2009) to the BAPOMDP setting by introducing the uncertain model parameters θ as a latent variable in the underlying stochastic process, in a manner similar to Furmston and Barber (2010) in the more restricted BAMDP setting. We also show that the M-step computations (the algorithm's bottleneck) can be naturally distributed over multiple concurrent threads. With this in mind, we present a sampling-based EM algorithm that is amenable to a higher discretization granularity than the sampling-based PBVI approach of Wang et al. (2012), the primary limitation of which is scalability with respect to the model space. We stress that while model discretization

is not an ideal approach to solving BAPOMDPs, it is likely unavoidable in certain cases. In the manufacturing case study of Section 1.2, for example, the posterior model distribution that arises from unsupervised training data (which would serve as the BAPOMDP prior p_θ) has no analytical form, and hence discretization of the posterior via Gibbs' sampling is one of the few viable alternatives.

Recall that when p_θ is given by a product of independent Dirichlet distributions, a compact state representation—consisting of pseudo-count vectors for the state-transitions and observation-emissions—can be adopted to simplify the value iteration backup procedure (Ross et al., 2008). Still, the offline version of this approach is only tractable for small problems due to the exponential growth of reachable states in the planning horizon T. To address this deficiency, we derive a variational Bayes EM algorithm—inspired by the work of Furmston and Barber (2010)—that efficiently handles the continuous prior while avoiding model discretization. Despite the strong assumptions required by the variational Bayes formulation, we show that it can outperform common point-based methods and compete with the sampling-based versions of EM and PBVI.

Expectation-maximization algorithms are often criticized for their slow convergence rates and the local optimality of their fixed points, and this dissertation would not be complete without addressing these concerns in the context of EM for BAPOMDPs. To this end, we show that the convergence rate of ordinary EM can be accelerated by adjusting the EM step-size via a simple parameterized update. In addition, we extend the forward-search procedure of Poupart et al. (2011b) for escaping locally optimal FSCs in the POMDP setting to the more general BAPOMDP setting and, furthermore, derive a finite-horizon analog to the original infinite-horizon subroutine.

In the following chapter, we further justify the consideration of model-adaptive policies by demonstrating that even the *best* point-based policy can significantly underperform the Bayes optimal policy.

Chapter 3

An argument for the Bayesian control of POMDPs

An operative assumption in the BAPOMDP literature is that the expense of computing model-adaptive policies is justified by their superior performance. However, to our knowledge no example has been formally proposed that illustrates a significant performance gap between simple point-based policies and Bayes optimal, model-adaptive policies. Our contribution in this chapter is to demonstrate, through a tractable yet nontrivial example, that even the *best* point-based policy can significantly underperform the Bayes optimal policy. As a result, we can make the stronger claim that point-based policies are not only suboptimal, but can be prohibitively so. This chapter is outlined as follows: We introduce a two-state Bayes-adaptive POMDP in Section 3.1, derive optimal point-based policies in Section 3.2, and demonstrate how Bayes optimal policies can be computed via dynamic programming in Section 3.3. We conclude with a brief discussion in Section 3.4.

3.1 Problem description

In this section we introduce a tractable yet nontrivial Bayes-adaptive POMDP. We begin by defining the standard POMDP $\mathcal{P}$, and then construct the Bayes-adaptive POMDP $\mathcal{P}'$ by introducing uncertainty into the model parameter space.

Consider a Markov decision process that is characterized by two states (1 and 2) and left-to-right transition dynamics. Under the left-to-right assumption, state 1 can transition to either state, and state 2 is an absorbing state. The process always begins in state 1 at time $t = 0$, and this is known to the agent. At each discrete time, the agent must take one of two possible actions: "stop" or "go". The "stop" action terminates the process, and the "go" action allows one more transition to occur before the next action is taken. A reward $r > 0$ is received when the "go" action is taken in state 1, and a cost of $c > 0$ is incurred when the "go" action is taken in state 2.

The goal is to maximize the expected reward that is accumulated prior to the process' termination via the "stop" action. In a fully observable environment, the agent would simply stop the process after state 2 was entered for the first time, but here we assume that the agent is limited by imperfect state information: While $x_0 = 1$ is known to the agent, the current state x_t at time $t \geq 1$ is hidden. However, upon transitioning from the current state, the subsequent state x_{t+1} emits a scalar observation y_{t+1}, which follows a Gaussian distribution with common standard deviation σ and means μ_1 and μ_2 for states 1 and 2, respectively. As such, the observation y_{t+1} can be used to infer the current state x_{t+1} and inform the action a_{t+1} taken at time $t + 1$. Without loss of generality, we assume that $\mu_1 < \mu_2$. In addition to these observations, the agent also has delayed state information of one time unit. In other words, the true state of the system at time t is revealed to the agent at time $t + 1$. Note that a discount factor is not required due to the absorbing nature of state 2 and the effect of action "stop". The policy that maximizes the

infinite-horizon expected reward for this process can be recovered by solving the POMDP $\mathcal{P} = \{X, A, \mathbb{R}, u, \theta\}$, where $\theta = (p, \mu_1, \mu_2, \sigma)$,

$$\begin{aligned} X &= \{1, 2\} \text{ is the set of states,} \\ A &= \{g, s\} \text{ is the set of available actions: “go” and “stop”,} \\ p &= p_{11} \text{ is the probability of self-transition for state 1,} \\ f_1 &\sim N(\mu_1, \sigma) \text{ is the observation distribution for state 1,} \\ f_2 &\sim N(\mu_2, \sigma) \text{ is the observation distribution for state 2} \end{aligned} \tag{3.1}$$

and $u : A \times X \to \mathbb{R}$ is the reward function. By the above description, $u(g, 1) = r$, $u(g, 2) = -c$, and $u(s, \cdot) = 0$.

If the model parameters $\theta = (p, \mu_1, \mu_2, \sigma)$ are known, then the optimal policy can be obtained by applying standard POMDP algorithms to $\mathcal{P}$. However, our problem is more complex than $\mathcal{P}$ since we assume the model parameters θ are not known to within a sufficient degree of certainty. Fundamental to our approach is the assumption of a prior distribution $h(\theta)$ on θ, so that a discussion of model uncertainty is meaningful. This prior implies a Bayes-adaptive form of the POMDP $\mathcal{P}$ which we will refer to as the BAPOMDP $\mathcal{P}'$.

Let $y^t = (y_1, y_2, \ldots, y_t)$ and $a^t = (a_0, a_1, \ldots, a_t)$ denote sequences of observations and actions up to time period t, and let Y^t and A^t denote the sets of all possible observation and action sequences up to time period t. In the most general setting, a policy is a sequence of mappings $\pi = \{\pi_0, \pi_1, \ldots\}$, where $\pi_t : Y^t \times A^{t-1} \to A$ is the mapping used in each time period. A Bayes optimal policy for $\mathcal{P}'$ solves the following:

$$\max_{\pi} \mathbf{E}_{\theta}[R|\pi] \;=\; \max_{\pi} \int_{\theta} h(\theta)\mathbf{E}[R|\theta,\pi]\; d\theta, \tag{3.2}$$

where R is a random variable representing the total accumulated reward until the stopping action s is taken. For the remainder of this paper, we limit uncertainty to the transition probability p and assume that σ, μ_1, and μ_2 are known to the agent. Note that if the agent has not taken action s by time $t \geq 1$, the vector $(y_t, t-1)$—consisting of the current observation y_t and the number $t-1$ of self-transitions from state 1 that have been revealed via delayed state information—defines a sufficient statistic for $\mathcal{P}'$, which simplifies the form of an optimal policy for the control problem.

3.2 An argument for Bayesian control

The objective is to determine the stopping policy that maximizes the expected reward (3.2). In Section 3.2.1 we assume that no observations are available for inferring the true system state, and we show that there does in fact exist a Bayes optimal point-based policy. However, in Section 3.2.2 we reintroduce the observation distributions f_1 and f_2 and demonstrate that even the *best* point-based policy can significantly underperform the Bayes optimal policy.

Hereafter, our uncertainty regarding p is characterized by a $Beta(\alpha, \beta)$ prior with $\alpha > 1$ and $\beta > 1$, so that

$$h(p) = \frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha,\beta)}, \quad 0 < p < 1, \tag{3.3}$$

where $B(\alpha, \beta) = \Gamma(\alpha)\Gamma(\beta)/\Gamma(\alpha + \beta)$ is the Beta function and Γ is the Gamma function (Davis, 1972). The Beta distribution—a special case of the Dirichlet—is the conjugate prior to the Bernoulli (Johnson et al., 2002). As such, imposing a Beta prior on p simplifies the posterior update given observed self-transitions from state 1.

3.2.1 Case 1: No observations

In this case we simplify the analysis by assuming that no observations are available for inferring the true system state. For the moment, let us assume a *known* p. Because $x_0 = 1$ is also known, an optimal policy will certainly take action $a_0 = g$ at time $t = 0$. Now consider the following two policies: (a) take action g and then immediately take action s; and (b) take action g until, via delayed state information, it is revealed that the process has entered state 2. We now show that one of these two policies must be optimal.

Proposition 2: *If p is known, then either policy (a) or policy (b) is optimal.*

We wish to show that either policy (a) or policy (b) is optimal. Equivalently, we show that if policy (a) is not optimal, then policy (b) must be optimal. To this end, suppose that policy (a) is not optimal.

By definition of $\mathcal{P}$ we have $x_0 = 1$. Because policy (a) is not optimal, it must be optimal to take action g at time $t = 1$. Action g results in a transition from state x_1 to state x_2, after which x_1 is revealed to the agent. If $x_1 = 2$, then action s is optimal at time $t = 2$, which agrees with policy (b). Otherwise, $x_1 = 1$ is revealed at time $t = 2$ and the agent's belief at time $t = 2$ is identical to its belief at time $t = 1$, by which action g is optimal at time $t = 2$. Therefore by induction it is optimal to take action g at all times $t \geq 1$ until delayed state information reveals that the system has entered state 2. This is precisely

what policy (b) prescribes. Therefore either policy (a) or policy (b) must be optimal. ■

The expected reward V^a associated with policy (a) is trivially $V^a = r$. The expected reward $V^b(p)$ associated with policy (b) is

$$V^b(p) = \frac{r}{1-p} - c, \tag{3.4}$$

so that policy (b) is optimal if and only if $V^b(p) > V^a$, that is,

$$p \ > \ \frac{c}{r+c}. \tag{3.5}$$

Now let us consider the Bayes-adaptive case where $p \sim h(p)$. In fact, just as in the case where p is known, either policy (a) or policy (b) must be optimal.

Proposition 3: *If $p \sim Beta(\alpha, \beta)$, then either policy (a) or policy (b) is optimal.*

We first note the following useful fact. Under any optimal policy, if action s has not been taken prior to time $t \geq 1$, then $t-1$ self-transitions from state 1 will have been revealed via delayed state information. The $Beta(\alpha, \beta)$ prior implies that the model posterior at time t can be compactly summarized by the $Beta(\alpha + t - 1, \beta)$ distribution.

Let V_t denote the expected reward associated with following the optimal policy from time $t \geq 1$, assuming that action s has not been taken prior to time t. Note that by definition V_t conditions on the event that $t-1$ self-transitions from state 1 have been

recorded, which implies $x_{t-1} = 1$. Furthermore, let $V_t(s)$ and $V_t(g)$ denote the expected rewards associated with taking immediate actions s and g, respectively, at time t and following the optimal policy thereafter.

We wish to show that either policy (a) or policy (b) is optimal. Equivalently, we show that if policy (a) is not optimal, then policy (b) must be optimal. To this end, suppose that policy (a) is not optimal. Then necessarily action g is optimal at time $t = 1$, which agrees with policy (b). Furthermore, this implies $V_1 = V_1(g) > 0$, or more specifically,

$$V_1 = V_1(g) = (r + V_2)\frac{\alpha}{\alpha + \beta} - c\frac{\beta}{\alpha + \beta} > 0.$$

Now we consider the optimal action at time $t = 2$. If $x_1 = 2$ is revealed at time $t = 2$, then action s is clearly optimal, which agrees with policy (b). Suppose instead that $x_1 = 1$ is revealed at time $t = 2$. Our goal is to show that action g is optimal. We assume to the contrary that action s is optimal and proceed by contradiction. The optimality of s implies $V_2 = V_2(s) = 0$. This and the above condition on V_1 imply that $r\alpha > c\beta$. But then, noting the updated $Beta(\alpha + 1, \beta)$ posterior on p,

$$\begin{aligned} V_2(g) &= (r + V_3)\frac{\alpha + 1}{\alpha + \beta + 1} - c\frac{\beta}{\alpha + \beta + 1} \\ &\geq r\frac{\alpha + 1}{\alpha + \beta + 1} - c\frac{\beta}{\alpha + \beta + 1} && (\text{by } V_3 \geq 0) \\ &> 0 && (\text{by } r\alpha > c\beta), \end{aligned}$$

so that $V_2(g) > V_2(s) = 0$, which contradicts the optimality of action s. Therefore action g is optimal at time $t = 2$ given delayed state information $x_1 = 1$, which agrees with policy (b). By induction we can extend this result to all t. To summarize, when policy

(a) is not optimal, we have shown that the optimal policy agrees with policy (b). It follows that either policy (a) or policy (b) must be optimal. ■

We now compare the expected rewards associated with policies (a) and (b). The expected reward $\bar{V}^a$ associated with policy (a) is trivially $\bar{V}^a = r$. The expected reward $\bar{V}^b$ associated with policy (b) is, using the result (3.4) for fixed p,

$$\begin{aligned} \bar{V}^b &= \int_p h(p) V^b(p)\ dp \\ &= -c + r\frac{\alpha+\beta-1}{\beta-1}. \end{aligned} \tag{3.6}$$

Therefore policy (b) is Bayes optimal if and only if $\bar{V}^b > \bar{V}^a$, or equivalently

$$\frac{\alpha}{\alpha+\beta-1} > \frac{c}{r+c}. \tag{3.7}$$

Comparing (3.7) to (3.5), we see that the point-based policy corresponding to point estimate $\hat{p} = \alpha/(\alpha+\beta-1)$ is, in fact, Bayes optimal. In other words, the optimal policy obtained by assuming $p = \hat{p}$ is also the Bayes optimal policy for the case of $p \sim Beta(\alpha, \beta)$. It is interesting to note that this point estimate is different than the more common mean and mode point estimates of the Beta distribution, given by $\alpha/(\alpha+\beta)$ and $(\alpha-1)/(\alpha+\beta-2)$, respectively. We now consider a more realistic case in which no point-based policy is Bayes optimal.

3.2.2 Case 2: Informative observations

Now we consider the case where observations are available for inferring the true system state. For the moment, assume that p is *known*. As with POMDPs in general, a sufficient statistic for defining an optimal policy is the belief state. Because $|X| = 2$, we may simply maintain the state 2 belief, call it δ_t, for time $t \geq 0$. For example, $\delta_0 = 0$ since the process is known to begin in state 1. Recall that the observations follow Gaussian distributions with means $\mu_1 < \mu_2$ and common standard deviation σ. This fact and the assumption of delayed state information together imply a one-to-one correspondence between beliefs δ_t and observations y_t for all times $t \geq 1$. To see why, suppose that at time t the agent observes y_t and it is revealed that $x_{t-1} = 1$ at time $t-1$. Computing the state 2 belief δ_t at time t, we find

$$\delta_t = \frac{(1-p)f_2(y_t)}{(1-p)f_2(y_t) + pf_1(y_t)} = \frac{1}{1 + \frac{p}{1-p}e^{-\frac{(y_t-\mu_1)^2}{2\sigma^2} + \frac{(y_t-\mu_2)^2}{2\sigma^2}}} = \frac{1}{1 + \frac{p}{1-p}e^{-\frac{(\mu_2-\mu_1)(2y_t-\mu_1-\mu_2)}{2\sigma^2}}},$$

which is strictly increasing as a function of y_t since $\mu_2 > \mu_1$. This establishes the one-to-one correspondence between δ_t and y_t. Note that as $y_t \to \infty$ the belief δ_t approaches 1, so that action s is optimal. Likewise, as $y_t \to -\infty$ the belief δ_t approaches 0, so that action g is optimal. Finally, the convexity of optimal POMDP value functions over the belief space (Cassandra, 1998) implies that an optimal policy will be a "threshold" policy on the current belief δ_t—or equivalently the current observation y_t—such that for the optimal threshold y^*, action g is taken when $y_t < y^*$ and action s is taken when $y_t > y^*$.

Let $V(p, y)$ denote the expected reward associated with threshold policy y and fixed probability p. By this definition, $V(p, y)$ has an equivalent interpretation as the expected reward of taking action g when $x = 1$, and following threshold policy y thereafter, given

p. As such, we can write $V(p,y) = r + pF_1(y)V(p,y) - c(1-p)F_2(y)$, where F_1 and F_2 are the CDFs of the conditional observation distributions for states 1 and 2, respectively. Solving for $V(p,y)$, we obtain

$$V(p,y) = \frac{r - c(1-p)F_2(y)}{1 - pF_1(y)}. \tag{3.8}$$

Therefore, given a point estimate $\hat{p}$ from the model prior $h(p)$, the corresponding point-based policy maximizes the expected reward (3.8) for $p = \hat{p}$ and so is given by the threshold $\hat{y} = \operatorname{argmax}_y V(\hat{p}, y)$. Using (3.8), the point-based performance $\bar{V}(y)$ associated with threshold y when $p \sim h(p)$ is

$$\begin{aligned} \bar{V}(y) &= \int_p h(p)V(p,y)\ dp \\ &= \int_p \frac{p^{\alpha-1}(1-p)^{\beta-1}}{B(\alpha,\beta)} \left\{ \frac{r - c(1-p)F_2(y)}{1 - pF_1(y)} \right\}\ dp. \end{aligned} \tag{3.9}$$

In Figures 3.1–3.4 we plot the expected reward associated with a range of point-based (threshold) policies when $p \sim h(p)$ over four problem instances. Plot windows were chosen to include the best achievable point-based performance. Note that in Figures 3.1 and 3.4, point-based performance is bounded by 0.138 and 0.105, respectively. These bounds were obtained by taking the limit of (3.9) as $y \to \infty$, or equivalently evaluating (3.6). We highlight three point-based policies, which are derived from the mean estimator $\hat{p} = \alpha/(\alpha+\beta)$, the mode estimator $\hat{p} = (\alpha-1)/(\alpha+\beta-2)$, and the optimal estimator from Case 1, $\hat{p} = \alpha/(\alpha+\beta-1)$. We also indicate Bayes optimal performance, the computation of which is described in Section 3.3. We see that in each problem instance even the *best*

point-based policy can significantly underperform the Bayes optimal policy due to a lack of model-adaptivity.

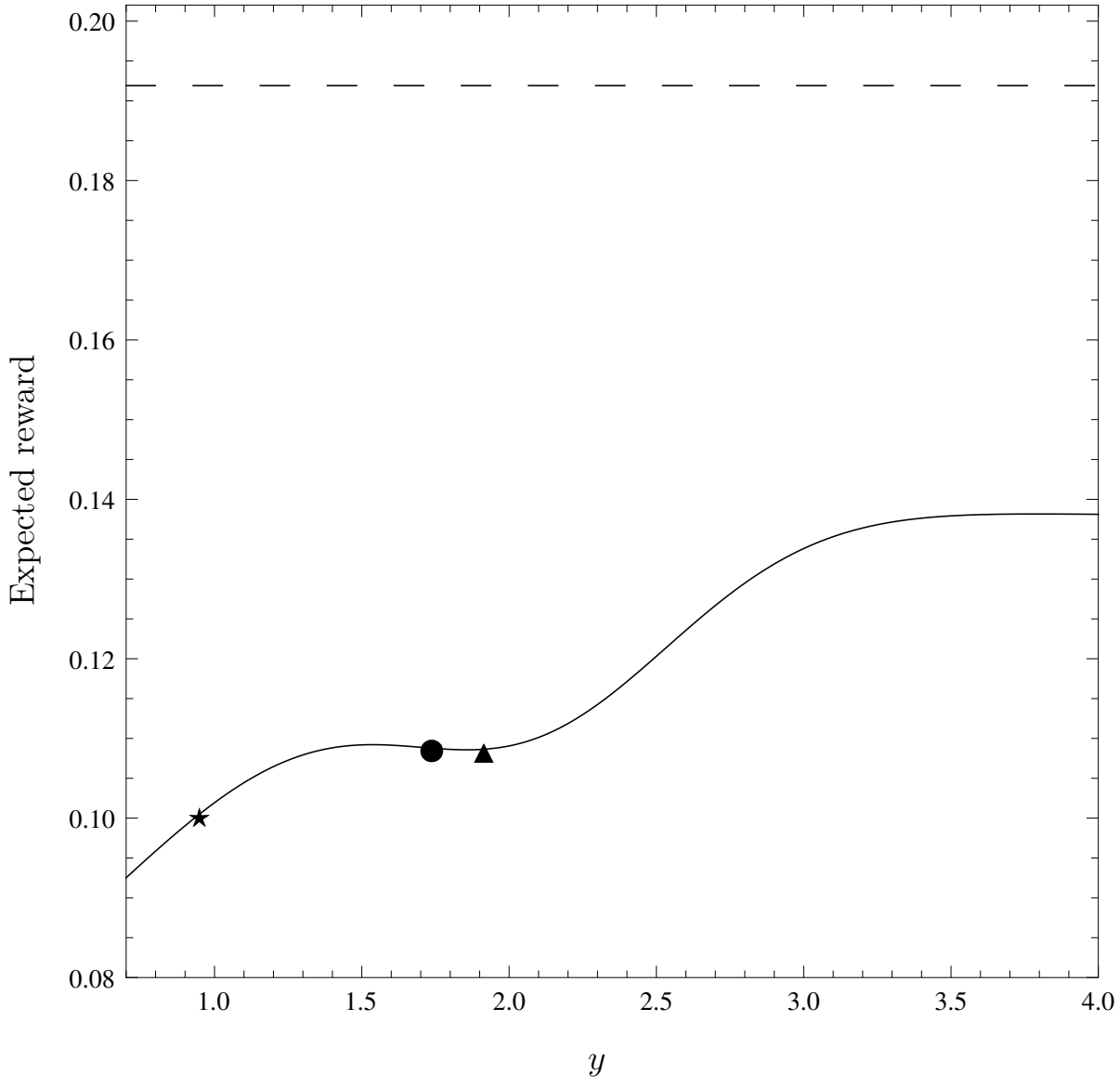

Figure 3.1: A comparison of point-based (threshold) policy performance to Bayes optimal performance with $\alpha = 10$, $\beta = 2$, $r = 0.058$, $c = 0.5$, $\mu_1 = 0.5$, $\mu_2 = 1.5$, $\sigma = 0.8$. We highlight three threshold policies in particular, corresponding to the mean estimator ($\star$), mode estimator ($\bullet$), and the estimator that assumes uninformative features from Case 1 ($\blacktriangle$). Dashing indicates Bayes optimal performance (Section 3.3).

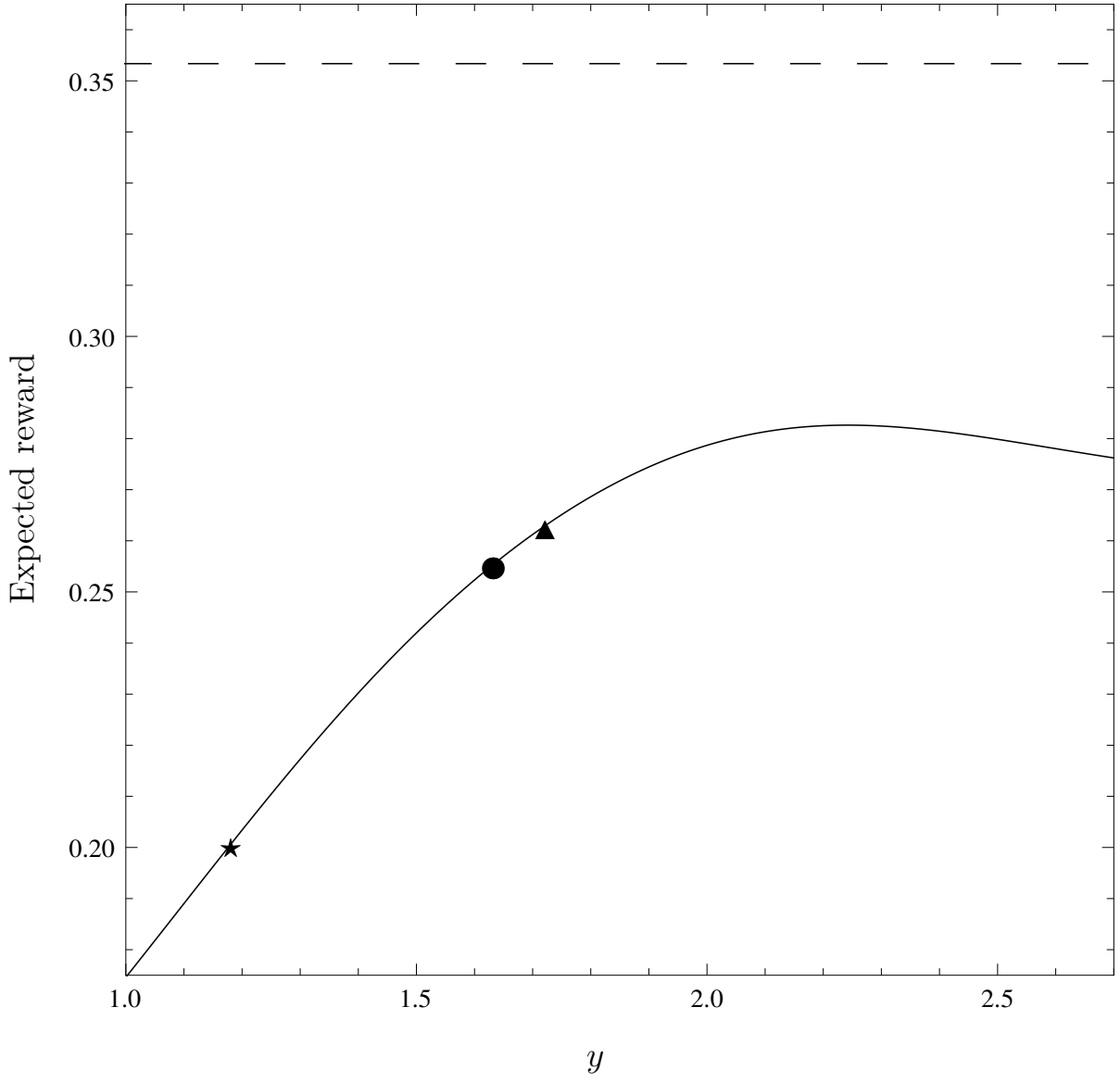

Figure 3.2: A comparison of point-based (threshold) policy performance to Bayes optimal performance with $\alpha = 10$, $\beta = 2$, $r = 0.07$, $c = 0.5$, $\mu_1 = 0.5$, $\mu_2 = 1.5$, $\sigma = 0.6$. We highlight three threshold policies in particular, corresponding to the mean estimator ($\star$), mode estimator ($\bullet$), and the estimator that assumes uninformative features from Case 1 ($\blacktriangle$). Dashing indicates Bayes optimal performance (Section 3.3).

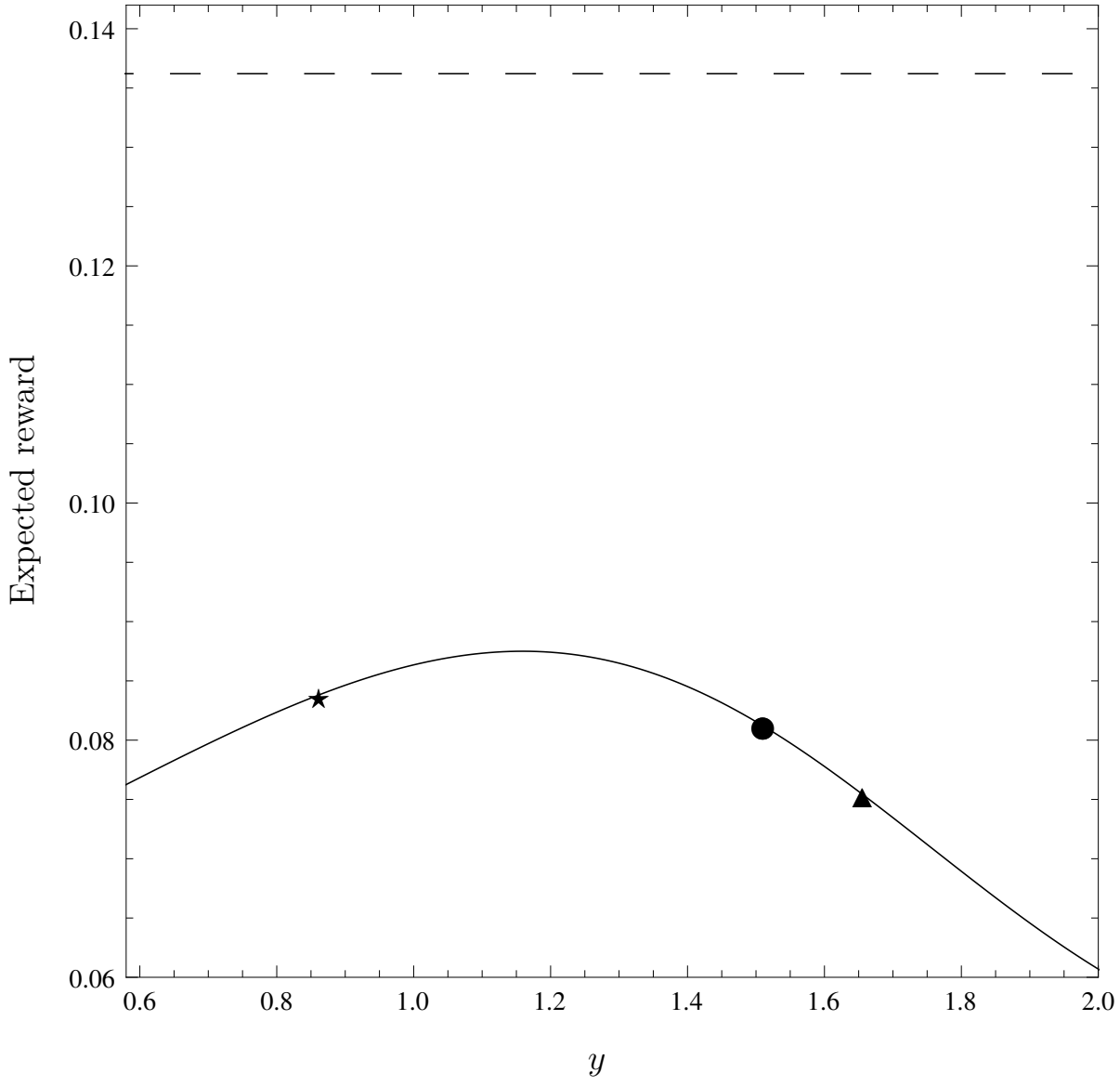

Figure 3.3: A comparison of point-based (threshold) policy performance to Bayes optimal performance with $\alpha = 10$, $\beta = 2$, $r = 0.05$, $c = 0.5$, $\mu_1 = 0.5$, $\mu_2 = 1.5$, $\sigma = 0.75$. We highlight three threshold policies in particular, corresponding to the mean estimator ($\star$), mode estimator ($\bullet$), and the estimator that assumes uninformative features from Case 1 ($\blacktriangle$). Dashing indicates Bayes optimal performance (Section 3.3).

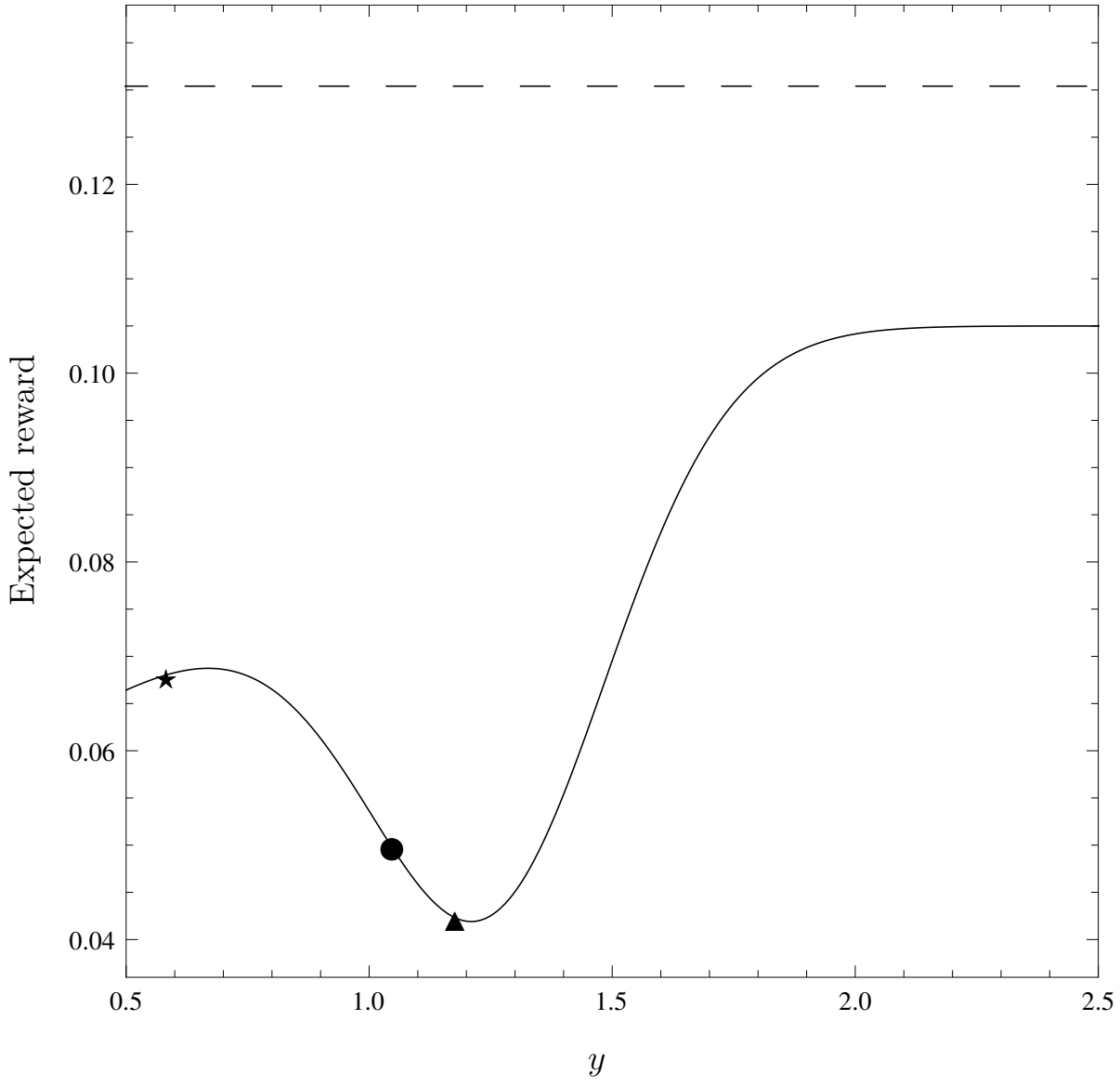

Figure 3.4: A comparison of point-based (threshold) policy performance to Bayes optimal performance with $\alpha = 10$, $\beta = 2$, $r = 0.055$, $c = 0.5$, $\mu_1 = 0.6$, $\mu_2 = 0.9$, $\sigma = 0.35$. We highlight three threshold policies in particular, corresponding to the mean estimator ($\star$), mode estimator ($\bullet$), and the estimator that assumes uninformative features from Case 1 ($\blacktriangle$). Dashing indicates Bayes optimal performance (Section 3.3).

3.3 Bayes optimal control via dynamic programming

Recall from Section 3.1 that a sufficient statistic for optimal control within the Bayes-adaptive POMDP $\mathcal{P}'$ is $(y_t, t-1)$. As a result, the Bayes optimal policy can be characterized as a "time-varying" threshold policy on the most recent observation y_t. To see why, suppose that at time t it is revealed that $x_{t-1} = 1$, so that the posterior belief on p follows a $Beta(\alpha + t - 1, \beta)$ distribution. Let V_t denote the expected reward associated with following the optimal policy from time $t \geq 1$, and let $V_t(g|y)$ denote the expected reward associated with taking action g when $y_t = y$ and following the optimal policy thereafter. If we define $\tilde{\delta}_t = P(x_t = 2|y_t, \alpha + t - 1, \beta)$ then

$$V_t(g|y) = (1 - \tilde{\delta}_t)(r + V_{t+1}) - c\tilde{\delta}_t.$$

Note that V_{t+1} is independent of y_t since the revelation of $x_t = 1$ decouples y_t from this future expectation. Now suppose for some $y_t = y'$ that $V_t(g|y') > 0$, by which action g is optimal at time t given y'. Analogously to the *known* p case, it is straightforward to show that $\tilde{\delta}_t$ is a strictly increasing function of y. Therefore $V_t(g|y)$ is a strictly decreasing function of y, by which $V_t(g|y) > V_t(g|y') > 0$ for all $y < y'$, and hence action g is optimal given all $y < y'$. Since action s is clearly optimal as $y_t \to \infty$, it follows that the optimal decision rule at time t is a threshold on the observation y_t. However, the optimal threshold will now vary as a function of the current time.

With this in mind, let $\mathbf{y} = (y^1, y^2, \ldots)$ denote an arbitrary time-varying threshold policy, such that the agent takes action g at time $t \geq 1$ if $y_t < y^t$, and takes action s otherwise. Recall that the agent will always take action g at time $t = 0$ since the system is known to begin in state 1. While it is not obvious how to compute the Bayes optimal $\mathbf{y}$ in the infinite-horizon, we can compute the optimal $\mathbf{y}$ for an arbitrarily large finite-horizon T,

as we now describe. Let $T > 0$ be a finite planning horizon, and fix $y^t = \infty$ for $t > T$. We say that this time-varying threshold policy is "truncated", since the thresholds are fixed for $t > T$. The optimal thresholds y^t for $1 \leq t \leq T$ can be computed in a finite number of recursions. Let J_t denote the expected reward associated with following the optimal (truncated) time-varying threshold policy $\mathbf{y} = (y^1, y^2, \ldots, y^T, \infty, \ldots)$ beginning at time $t \geq 1$, after $t-1$ self-transitions from state 1 have been recorded. It follows that

$$J_{T+1} = \int_p \tilde{h}(p) \left\{ \frac{rp - c(1-p)}{1-p} \right\} \, dp, \tag{3.10}$$

where $\tilde{h}(p) \sim Beta(\alpha + T, \beta)$. Then, we can recursively compute

$$J_t = \max_{y^t} \; \frac{\alpha + t - 1}{\alpha + t - 1 + \beta} F_1(y^t)(r + J_{t+1}) - \frac{\beta}{\alpha + t - 1 + \beta} F_2(y^t)c \tag{3.11}$$

for $1 \leq t \leq T$, where the maximizing argument y^t is the optimal threshold on the observation at time t, given that $t-1$ self-transitions from state 1 have been recorded. Because the system is known to start in state 1, it follows that the expected reward J^* associated with the optimal (truncated) time-varying threshold policy is $J^* = r + J_1$. This procedure is summarized in Algorithm 3.1. As $T \to \infty$, J^* will converge to the Bayes optimal expected reward for $\mathcal{P}'$. This limiting process was used to compute the Bayes optimal performance in Figures 3.1–3.4. In all four of these experiments, $T > 10$ was sufficient for J^* to reach ninety-nine percent of the Bayes optimal expected reward.

Algorithm 3.1 Approximating the Bayes optimal policy for the two-state problem

1: Initialize $J_{T+1} = \int_p \tilde{h}(p) \left\{ \frac{rp - c(1-p)}{1-p} \right\} \, dp.$
2: $t \leftarrow T$
3: **while** $t \geq 1$ **do**
4: Compute y_t and J_t from:

$$J_t = \max_{y^t} \; \frac{\alpha+t-1}{\alpha+t-1+\beta} F_1(y^t)(r + J_{t+1}) - \frac{\beta}{\alpha+t-1+\beta} F_2(y^t)c.$$

5: $t \leftarrow t - 1$
6: **end while**
7: $J^* \leftarrow r + J_1$

3.4 Summary

Through a tractable yet nontrivial example, we justified the search for Bayes optimal policies under conditions of model uncertainty by demonstrating that that even the *best* point-based policy can significantly underperform the Bayes optimal, model-adaptive policy. While exact BAPOMDP solutions are unobtainable in the general case, our results suggest that even suboptimal BAPOMDP solutions could dominate point-based policies in a more general setting. In the following chapter, we explore expectation-maximization as a scalable approach to approximating model-adaptive policies through the optimization of a bounded finite-state controller.

Chapter 4

Expectation-maximization for BAPOMDPs

4.1 Introduction

In this chapter we present an expectation-maximization approach to solving BAPOMDPs via finite-state controller optimization. However, in the most general case the M-step requires the evaluation of an intractable integral. Recently this issue was addressed in the more restricted Bayes-adaptive MDP setting by introducing a variational Bayes (VB) approximation to the EM algorithm under the assumption of independent Dirichlet priors on the model parameters (Furmston and Barber, 2010). Here, we extend VB-EM to the full BAPOMDP case and derive a novel *constrained* VB-EM algorithm, which addresses an unfavorable preference that can arise toward a certain class of FSCs in the standard VB-EM implementation. In addition, we present a sampling-based EM algorithm for the case when the model prior does not have this convenient Dirichlet form. Through an empirical study we show that finite-state controllers generated by EM can compete with

more conventional value iteration-based policies, and we argue that EM has the potential to scale more gracefully as model uncertainty increases.

4.2 EM for BAPOMDPs

We proceed by deriving a general EM algorithm for maximizing the expected reward associated with the BAPOMDP $\mathcal{P}$ over the set of all possible bounded FSCs Λ of some *a priori* fixed size, $|N|$, subject to a finite planning horizon $T \geq 0$. The approach taken here can be seen as a generalization of EM for Bayes-adaptive MDPs (Furmston and Barber, 2010). First, we define the reward-weighted path distribution

$$\hat{p}(t, \theta, x^t, n^t, a^t, o^t|\Lambda) = \frac{u_t(a_t, x_t)p(\theta, x^t, n^t, a^t, o^t|\Lambda)}{J(\Lambda)}, \tag{4.1}$$

where $u_t(a, x) = \gamma^t u(a, x)$, $J(\Lambda)$ is a normalizing constant equal to the expected reward associated with FSC Λ, $x^t = (x_0, x_1, \ldots, x_t)$ is a trajectory of hidden states $x \in X$, and n^t, a^t, and o^t are defined similarly. If we let $z = (t, x^t, n^t, a^t, o^t)$, it is straightforward to verify that $\hat{p}$ is a valid density function over the latent variables (z, θ) by computing $\int_\theta \sum_z \hat{p}(z, \theta) = 1$.

Let q be an arbitrary distribution over the latent variables (z, θ) and for notational convenience define $\tilde{p}(z, \theta|\Lambda) = \hat{p}(z, \theta|\Lambda)J(\Lambda)$. Consider the KL divergence $\mathrm{KL}(q, \hat{p})$ between q and $\hat{p}$:

$$\mathrm{KL}(q, \hat{p}) = \langle \log q(z, \theta) \rangle_q - \langle \log \hat{p}(z, \theta|\Lambda) \rangle_q$$

$$= \langle \log q(z, \theta) \rangle_q - \langle \log \tilde{p}(z, \theta | \Lambda) \rangle_q + \log J(\Lambda)$$
$$\geq 0. \tag{4.2}$$

Rearranging terms in the above inequality, we find

$$\log J(\Lambda) \geq -\langle \log q(z, \theta) \rangle_q + \langle \log \tilde{p}(z, \theta | \Lambda) \rangle_q. \tag{4.3}$$

The direct optimization of $\log J(\Lambda)$ is not tractable in general, so a natural approach is to instead optimize the lower bound (4.3). An EM algorithm arises if we alternate the following two steps: (i) optimize (4.3) with respect to q given a fixed Λ (E-step), and (ii) optimize (4.3) with respect to Λ for fixed q (M-step). If both the E-step and M-step are executed exactly, then it can be shown—analogously to the coordinate ascent interpretation of EM (Neal and Hinton, 1998)—that each update of Λ will result in a strict increase in the objective $J(\Lambda)$ until convergence to a local optimum.

4.2.1 E-step

In the E-step the lower bound (4.3) is maximized with respect to q given a fixed Λ. This is equivalent to minimizing $\mathrm{KL}(q, \hat{p})$ with respect to q. Therefore, by properties of KL divergence $q(z, \theta) = \hat{p}(z, \theta | \Lambda)$ is the unique maximizer of the lower bound.

4.2.2 M-step

Subsequently, the M-step maximizes the lower bound (4.3) with respect to Λ given a fixed q. This reduces to maximizing $\langle \log \tilde{p}(z, \theta | \Lambda) \rangle_q$ over the parameters ν, π, and λ. We can

write

$$\langle \log \tilde{p}(z, \theta | \Lambda) \rangle_q = \int_\theta \sum_{t=0}^{T} \sum_{x^t, n^t, a^t, o^t} q(z, \theta) \log \nu_{n_0} \, d\theta + \int_\theta \sum_{t=0}^{T} \sum_{x^t, n^t, a^t, o^t} q(z, \theta) \sum_{\tau=0}^{t} \log \pi_{a_\tau n_\tau} \, d\theta + \int_\theta \sum_{t=1}^{T} \sum_{x^t, n^t, a^t, o^t} q(z, \theta) \sum_{\tau=0}^{t-1} \log \lambda_{n_{\tau+1} n_\tau o_{\tau+1}} \, d\theta + C,$$

where C is a constant independent of Λ. From the above form, it is clear that we can optimize the lower bound independently with respect to ν, π, and λ. The component dependent on ν can be written as

$$\int_\theta \sum_{t=0}^{T} \sum_{x^t, n^t, a^t, o^t} q(z, \theta) \log \nu_{n_0} \, d\theta = \sum_n \log \nu_n \int_\theta \sum_{t=0}^{T} q(t, n_0 = n, \theta) \, d\theta.$$

Similarly, for the components dependent on π and λ we can write

$$\int_\theta \sum_{t=0}^{T} \sum_{x^t, n^t, a^t, o^t} q(z, \theta) \sum_{\tau=0}^{t} \log \pi_{a_\tau n_\tau} \, d\theta = \sum_{a,n} \log \pi_{an} \int_\theta \sum_{t=0}^{T} \sum_{\tau=0}^{t} q(t, a_\tau = a, n_\tau = n, \theta) \, d\theta$$

and

$$\int_\theta \sum_{t=1}^{T} \sum_{x^t, n^t, a^t, o^t} q(z, \theta) \sum_{\tau=0}^{t-1} \log \lambda_{n_{\tau+1} n_\tau o_{\tau+1}} \, d\theta =$$

$$\sum_{n',n,o'} \log \lambda_{n'no'} \int_\theta \sum_{t=1}^{T} \sum_{\tau=0}^{t-1} q(t, n_{\tau+1} = n', n_\tau = n, o_{\tau+1} = o', \theta)\ d\theta,$$

respectively. Subject to normalization constraints of the form $\sum_n \nu_n = 1$, $\sum_a \pi_{an} = 1$, and $\sum_{n'} \lambda_{n'no'} = 1$, the method of Lagrange multipliers can be used to solve for the optimal parameter updates. We find that

$$\nu_n^* \propto \int_\theta \sum_{t=0}^{T} q(t, n_0 = n, \theta)\ d\theta \tag{4.4}$$

$$\pi_{an}^* \propto \int_\theta \sum_{t=0}^{T} \sum_{\tau=0}^{t} q(t, a_\tau = a, n_\tau = n, \theta)\ d\theta \tag{4.5}$$

$$\lambda_{n'no'}^* \propto \int_\theta \sum_{t=1}^{T} \sum_{\tau=0}^{t-1} q(t, n_{\tau+1} = n', n_\tau = n, o_{\tau+1} = o', \theta)\ d\theta, \tag{4.6}$$

so that the FSC updates require computing marginals of the distribution q obtained in the E-step.

4.3 A sampling-based approach

While the integrals required by the M-step in (4.4)–(4.6) are not tractable in the general case, a smaller representative subset of models can be sampled to approximate the true prior as done in the PBVI approach of Wang et al. (2012). Sampling-based approaches are not ideal—particularly when the dimensionality of uncertain parameters is large—but may be unavoidable when the BAPOMDP prior is not of a convenient form. Suppose, for example, that our initial uncertainty with respect to the model parameters θ is expressed as a product of independent Dirichlet distributions, and that additional training data is

available in the form of unsupervised action-observation trajectories. The resulting posterior distribution—which serves as the BAPOMDP prior—will generally be multi-modal with no analytical expression. However, a Markov chain Monte-Carlo algorithm such as Gibbs' procedure (Cappé et al., 2005) could be used to sample models from the posterior, which would then serve as a finite approximation to the BAPOMDP prior.

In light of this, let us reconsider the M-step updates when Θ contains a finite number of models. From (4.4)–(4.6) and the definition of $\tilde{p}$ we immediately obtain

$$\nu_n^* \propto \sum_\theta p_\theta(\theta) \sum_{t=0}^{T} \tilde{p}(t, n_0 = n | \theta, \Lambda) \tag{4.7}$$

$$\pi_{an}^* \propto \sum_\theta p_\theta(\theta) \sum_{t=0}^{T} \sum_{\tau=0}^{t} \tilde{p}(t, a_\tau = a, n_\tau = n | \theta, \Lambda) \tag{4.8}$$

$$\lambda_{n'no'}^* \propto \sum_\theta p_\theta(\theta) \sum_{t=1}^{T} \sum_{\tau=0}^{t-1} \tilde{p}(t, n_{\tau+1} = n', n_\tau = n, o_{\tau+1} = o' | \theta, \Lambda). \tag{4.9}$$

Note that if the models Θ are sampled from some other continuous prior, then we should have $p_\theta(\theta) = 1/|\Theta|$ in the above expressions. From the finite-horizon update equations, we find that the required marginals of $\tilde{p}$ for fixed $\theta \in \Theta$ can be computed independently from all other $\theta' \in \Theta$—for instance, by running the forward-backward algorithm on the factor graph of $\tilde{p}$—and hence can be distributed over multiple concurrent threads. This is a useful insight, since the forward-backward procedure used to compute the marginals for fixed θ is the bottleneck of the EM algorithm. It is also worth noting that there is no clear analogous parallelization within the PBVI framework, since all model parameters jointly mix within the same augmented belief vector.

We now describe how the marginals of $\tilde{p}$ required by the M-step updates (4.7)–(4.9) can be computed efficiently. To this end, we begin by running sum-product (Kschischang

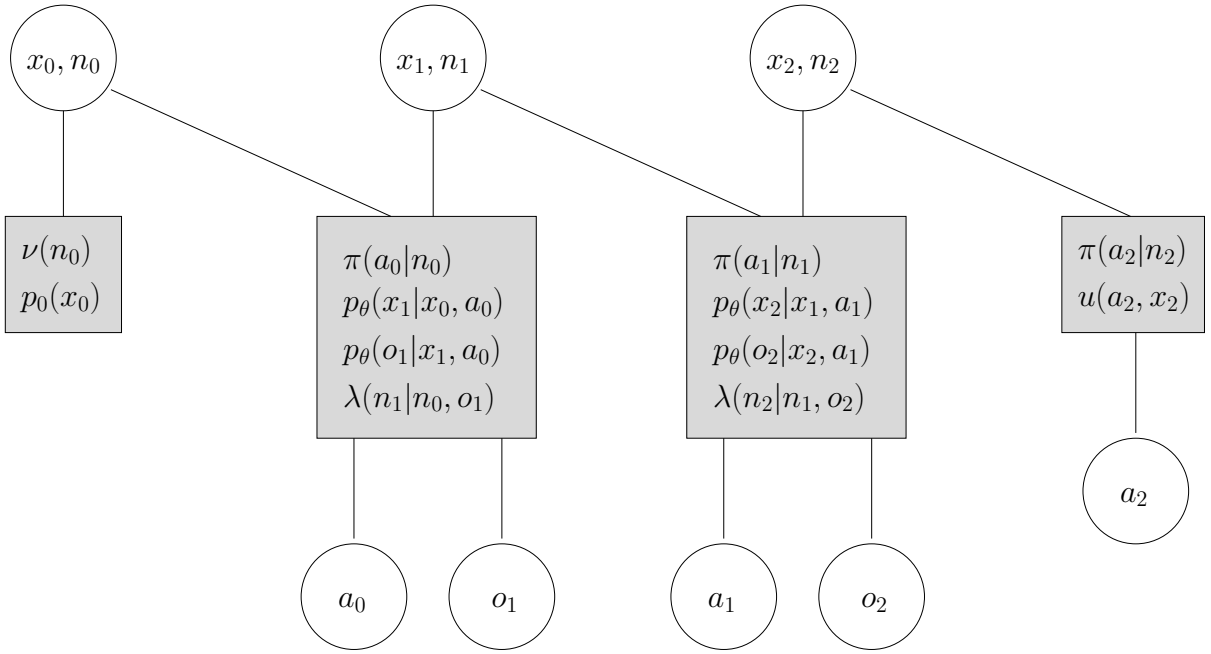

Figure 4.1: A factor graph representation of $\tilde{p}(T, x^T, n^T, a^T, o^T|\theta, \Lambda)\gamma^{-T}$ when $T = 2$.

et al., 2001) on the factor graph of $\tilde{p}(T, x^T, n^T, a^T, o^T|\theta, \Lambda)\gamma^{-T}$ for each $\theta \in \Theta$. As an illustration, Figure 4.1 contains the corresponding factor graph when $T = 2$. Note that because the factor graph is a chain, sum-product reduces to the familiar forward-backward algorithm. Algorithms 4.1 and 4.2 summarize how the forward messages μ_θ and the backward messages $\bar{\mu}_\theta$ are computed for fixed $\theta \in \Theta$.

Algorithm 4.1 Computing the forward messages μ_θ for fixed $\theta \in \Theta$

Input: FSC Λ

Output: Forward messages μ_θ

1: Initialize $\mu_{0,\theta}(x, n) = \nu_n p_0(x)$ for all $x \in X, n \in N$.
2: **for** $t = 1$ to $t = T$ **do**
3: **for all** $x' \in X, n' \in N$ **do**
4: Set $\mu_{t,\theta}(x', n') = \sum_{x,n,a,o'} \mu_{t-1,\theta}(x, n)\pi_{an}\lambda_{n'no'}p_\theta(x'|x, a)p_\theta(o'|x', a)$.
5: **end for**
6: **end for**

Given a fixed $\theta \in \Theta$, the marginals of $\tilde{p}$ can then be derived from the forward messages μ_θ and backward messages $\bar{\mu}_\theta$. We obtain

Algorithm 4.2 Computing the backward messages $\bar{\mu}_\theta$ for fixed $\theta \in \Theta$

Input: FSC Λ

Output: Backward messages $\bar{\mu}_\theta$

1: Initialize $\bar{\mu}_{0,\theta}(x,n) = \sum_a \pi_{an} u(a,x)$ for all $x \in X, n \in N$.
2: **for** $t = 1$ to $t = T$ **do**
3: **for all** $x \in X, n \in N$ **do**
4: Set $\bar{\mu}_{t,\theta}(x,n) = \sum_{x',n',a,o'} \bar{\mu}_{t-1,\theta}(x',n')\pi_{an}\lambda_{n'no'}p_\theta(x'|x,a)p_\theta(o'|x',a)$.
5: **end for**
6: **end for**

$$\tilde{p}(t, n_0 = n|\theta, \Lambda) = \sum_x \mu_0(x,n)\bar{\mu}_{t,\theta}(x,n)\gamma^t \tag{4.10}$$

$$\tilde{p}(t, a_\tau = a, n_\tau = n|\theta, \Lambda) = \begin{cases} \sum\limits_{x,x',n',o'} \mu_{\tau,\theta}(x,n)\pi_{an}p_\theta(x'|x,a)p_\theta(o'|x',a)\lambda_{n'no'}\bar{\mu}_{t-\tau-1,\theta}(x',n')\gamma^t, & \tau < t \\ \sum\limits_x \mu_{\tau,\theta}(x,n)\pi_{an}u(a,x)\gamma^\tau, & \tau = t \end{cases} \tag{4.11}$$

$$\tilde{p}(t, n_{\tau+1} = n', n_\tau = n, o_{\tau+1} = o'|\theta, \Lambda) = \sum_{x,x',a} \mu_{\tau,\theta}(x,n)\pi_{an}p_\theta(x'|x,a)p_\theta(o'|x',a)\lambda_{n'no'}\bar{\mu}_{t-\tau-1,\theta}(x',n')\gamma^t. \tag{4.12}$$

FSC performance can then be computed via the formula

$$J(\Lambda) = \sum_\theta p_\theta(\theta) \sum_{t=0}^{T} \sum_{x,n} \nu_n p_0(x)\bar{\mu}_{t,\theta}(x,n)\gamma^t. \tag{4.13}$$

Furthermore, the infinite-horizon FSC updates can be derived by taking the limit of the summations in (4.7)–(4.9) as $T \to \infty$ (Appendix, Proposition A). We obtain

$$\sum_{t=0}^{\infty} \tilde{p}(t, n_0 = n|\theta, \Lambda) = \sum_{x} \mu_0(x, n)\mathcal{B}_\theta(x, n) \tag{4.14}$$

$$\sum_{t=0}^{\infty}\sum_{\tau=0}^{t} \tilde{p}(t, a_\tau = a, n_\tau = n|\theta, \Lambda) = \pi_{an}\sum_{x} u(a, x)\mathcal{F}_\theta(x, n) + \gamma \sum_{x,x',n',o'} \pi_{an} p_\theta(x'|x, a)p_\theta(o'|x', a)\lambda_{n'no'}\mathcal{F}_\theta(x, n)\mathcal{B}_\theta(x', n') \tag{4.15}$$

$$\sum_{t=1}^{\infty}\sum_{\tau=0}^{t-1} \tilde{p}(t, n_{\tau+1} = n', n_\tau = n, o_{\tau+1} = o'|\theta, \Lambda) = \gamma \sum_{x,x',a} \pi_{an} p_\theta(x'|x, a)p_\theta(o'|x', a)\lambda_{n'no'}\mathcal{F}_\theta(x, n)\mathcal{B}_\theta(x', n'), \tag{4.16}$$

where $\mathcal{F}_\theta(x, n) = \lim_{t\to\infty} \mathcal{F}^t_\theta(x, n)$ and $\mathcal{B}_\theta(x, n) = \lim_{t\to\infty} \mathcal{B}^t_\theta(x, n)$ can be computed for fixed $\theta \in \Theta$ by initializing $\mathcal{F}^0_\theta(x, n)$ and $\mathcal{B}^0_\theta(x, n)$ arbitrarily and applying the recursive formulas

$$\mathcal{F}^t_\theta(x', n') = \nu_{n'} p_0(x') + \gamma \sum_{x,n,a,o'} \mathcal{F}^{t-1}_\theta(x, n)\pi_{an}\lambda_{n'no'} p_\theta(x'|x, a)p_\theta(o'|x', a) \tag{4.17}$$

$$\mathcal{B}^t_\theta(x, n) = \sum_{a} \pi_{an} u(a, x) + \gamma \sum_{x',n',a,o'} \mathcal{B}^{t-1}_\theta(x', n')\pi_{an}\lambda_{n'no'} p_\theta(x'|x, a)p_\theta(o'|x', a), \tag{4.18}$$

until convergence to a unique fixed point. FSC performance in the infinite horizon can be computed via the following formula:

$$J(\Lambda) = \sum_{\theta} p_\theta(\theta) \sum_{x,n} p_0(x) \nu_n \mathcal{B}_\theta(x, n). \tag{4.19}$$

4.4 Variational Bayes EM for BAPOMDPs

We now consider the case where the model prior p_θ is given by a product of independent Dirichlet distributions (Johnson et al., 2002), i.e.,

$$p_\theta(\theta) \propto \prod_{x,a} \mathrm{Dir}(\boldsymbol{\theta}^{x'}_{x,a} | \boldsymbol{\alpha}^{x'}_{x,a}) \prod_{x',a} \mathrm{Dir}(\boldsymbol{\theta}^{o'}_{x',a} | \boldsymbol{\alpha}^{o'}_{x',a}), \tag{4.20}$$

where the vectors $\boldsymbol{\alpha}$ parameterize the Dirichlet distributions and hence can be interpreted as pseudo-counts. Generally speaking, if $\theta = (\theta_1, \theta_2, \ldots, \theta_n) \sim \mathrm{Dir}(\alpha)$ for $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ with $n \geq 2$, then the multivariate density of θ is given by

$$\mathrm{Dir}(\theta | \alpha) = \frac{\Gamma(\Sigma \alpha_i)}{\prod \Gamma(\alpha_i)} \prod_i \theta_i^{\alpha_i - 1}$$

for all θ such that $0 < \theta_i < 1$ and $\sum_i \theta_i = 1$, where Γ is the Gamma function (Davis, 1972). As before, we begin by considering the T-horizon problem since the infinite-horizon version can be derived by taking the limit as $T \to \infty$.

In the more restricted Bayes-adaptive MDP setting (Furmston and Barber, 2010), the following assumption is made to obtain a tractable M-step: constrain the distribution q to have the factored representation $q = q_z q_\theta$, that is, q can be written as a product of distributions over the latent variables z and the unknown model parameters θ. Recall

that when q is unconstrained, choosing q to be the maximizer of the lower bound (4.3) in the E-step guarantees that any increase to the lower bound in the M-step likewise implies a strict increase to $J(\Lambda)$. Once we constrain $q = q_z q_\theta$, however, we lose this guarantee. Given the general complexity of BAPOMDPs, we see this as an acceptable concession for the sake of tractability.

We now derive the EM algorithm resulting from this factorization of q in the BAPOMDP case, and highlight an unfavorable preference toward a certain class of FSCs that can arise.

4.4.1 Variational Bayes E-step

We now constrain q to have the factored form

$$q(z, \theta) = q_z(z) q_\theta(\theta). \tag{4.21}$$

The lower bound of $\log J(\Lambda)$ in (4.3) then becomes

$$L(q_z, q_\theta) = -\langle \log q_z(z) \rangle_{q_z} - \langle \log q_\theta(\theta) \rangle_{q_\theta} + \langle \log p_\theta(\theta) \rangle_{q_\theta} + \langle \log \tilde{p}(z|\Lambda, \theta) \rangle_{q_z q_\theta}. \tag{4.22}$$

The E-step consists of optimizing (4.22) with respect to the factored distribution q given a fixed FSC Λ. We adopt the coordinate ascent approach of Furmston and Barber (2010) and alternate between (i) maximizing (4.22) with respect to q_θ for fixed q_z and Λ, and (ii) maximizing (4.22) with respect to q_z for fixed q_θ and Λ.

Maximizing (4.22) with respect to q_θ using the calculus of variations, we find the optimal q_θ update to be

$$q_\theta(\theta) \;\propto\; p_\theta(\theta) \exp\langle \log \tilde{p}(z|\theta, \Lambda)\rangle_{q_z}. \tag{4.23}$$

The details of this derivation are provided in the Appendix (Lemma A). Expanding the right-hand side of (4.23),

$$\begin{aligned}
q_\theta(\theta) &\propto p_\theta(\theta) \exp\langle \log \tilde{p}(z|\theta, \Lambda)\rangle_{q_z} \\
&\propto p_\theta(\theta) \exp\left(\sum_z q_z(z) \log \tilde{p}(z|\theta, \Lambda) \right) \\
&\propto p_\theta(\theta) \exp\left(\sum_z q_z(z) \sum_{\tau=0}^{t-1} \log \theta^{x_{\tau+1}}_{x_\tau, a_\tau} \right) \exp\left(\sum_z q_z(z) \sum_{\tau=0}^{t-1} \log \theta^{o_{\tau+1}}_{x_{\tau+1}, a_\tau} \right) \\
&\propto p_\theta(\theta) \exp\left(\sum_{x',x,a} \log \theta^{x'}_{x,a} \sum_{t=1}^{T} \sum_{\tau=0}^{t-1} q_z(t, x_{\tau+1} = x', x_\tau = x, a_\tau = a) \right) \\
&\qquad \exp\left(\sum_{o',x',a} \log \theta^{o'}_{x',a} \sum_{t=1}^{T} \sum_{\tau=0}^{t-1} q_z(t, x_{\tau+1} = x', o_{\tau+1} = o', a_\tau = a) \right).
\end{aligned} \tag{4.24}$$

For fixed x', o', x, a let

$$\begin{aligned}
q^{x'}_{x,a} &= \sum_{t=1}^{T} \sum_{\tau=0}^{t-1} q_z(t, x_{\tau+1} = x', x_\tau = x, a_\tau = a) \\
q^{o'}_{x',a} &= \sum_{t=1}^{T} \sum_{\tau=0}^{t-1} q_z(t, x_{\tau+1} = x', o_{\tau+1} = o', a_\tau = a),
\end{aligned} \tag{4.25}$$

which can be computed by applying the forward-backward algorithm to the factor graph of q_z. Then, working with the final line of (4.24) we obtain

$$\begin{aligned} q_\theta(\theta) &\propto p_\theta(\theta) \exp\left(\sum_{x',x,a} q^{x'}_{x,a} \log \theta^{x'}_{x,a} \right) \exp\left(\sum_{o',x',a} q^{o'}_{x',a} \log \theta^{o'}_{x',a} \right) \\ &\propto p_\theta(\theta) \prod_{x',x,a} {\theta^{x'}_{x,a}}^{q^{x'}_{x,a}} \prod_{o',x',a} {\theta^{o'}_{x',a}}^{q^{o'}_{x',a}} \\ &\propto \prod_{x,a} \mathrm{Dir}(\boldsymbol{\theta}^{x'}_{x,a} | \boldsymbol{\alpha}^{x'}_{x,a} + \mathbf{q}^{x'}_{x,a}) \prod_{x',a} \mathrm{Dir}(\boldsymbol{\theta}^{o'}_{x',a} | \boldsymbol{\alpha}^{o'}_{x',a} + \mathbf{q}^{o'}_{x',a}) \\ &= \prod_{x,a} \mathrm{Dir}(\boldsymbol{\theta}^{x'}_{x,a} | \boldsymbol{\beta}^{x'}_{x,a}) \prod_{x',a} \mathrm{Dir}(\boldsymbol{\theta}^{o'}_{x',a} | \boldsymbol{\beta}^{o'}_{x',a}), \end{aligned} \tag{4.26}$$

where for notational convenience we have defined $\boldsymbol{\beta}^{x'}_{x,a} = \boldsymbol{\alpha}^{x'}_{x,a} + \mathbf{q}^{x'}_{x,a}$ and $\boldsymbol{\beta}^{o'}_{x',a} = \boldsymbol{\alpha}^{o'}_{x',a} + \mathbf{q}^{o'}_{x',a}$. Therefore, the update to $q_\theta(\theta)$ reduces to computing Dirichlet pseudo-count vectors of the form $\mathbf{q}$, which are derived from the current estimate of q_z.

Next, we maximize (4.22) with respect to q_z using the calculus of variations and find the optimal q_z update to be

$$q_z(t, x^t, n^t, a^t, o^t) \;\propto\; \exp\langle \log \tilde{p}(t, x^t, n^t, a^t, o^t | \theta, \Lambda) \rangle_{q_\theta}. \tag{4.27}$$

The details of this derivation are provided in the Appendix (Lemma A). Expanding the right-hand side of (4.27),

$$q_z(t, x^t, n^t, a^t, o^t) \propto \exp \langle \log \tilde{p}(t, x^t, n^t, a^t, o^t | \theta, \Lambda) \rangle_{q_\theta}$$

$$= p_0(x_0)\nu_{n_0}\pi_{a_t n_t} u_t(a_t, x_t) \prod_{\tau=0}^{t-1} \pi_{a_\tau n_\tau} \lambda_{n_{\tau+1} n_\tau o_{\tau+1}}$$
$$\prod_{\tau=0}^{t-1} \exp \langle \log \theta^{x_{\tau+1}}_{x_\tau, a_\tau} \rangle_{q_\theta} \prod_{\tau=0}^{t-1} \exp \langle \log \theta^{o_{\tau+1}}_{x_{\tau+1}, a_\tau} \rangle_{q_\theta}. \tag{4.28}$$

In particular, we see that the only quantities dependent on q_θ—and hence must be updated with each iteration on q_z—are

$$\exp \langle \log \theta^{x'}_{x,a} \rangle_{q_\theta} \qquad \text{and} \qquad \exp \langle \log \theta^{o'}_{x',a} \rangle_{q_\theta}. \tag{4.29}$$

In the Appendix (Lemma B), we prove that these state-transition and observation-emission parameters are sub-stochastic. For simplicity, we will refer to these quantities collectively as the "q_z parameters".

Because the update to q_θ can be written as a product of independent Dirichlets (4.26) with pseudo-count parameters $\boldsymbol{\beta}$, the q_z parameters can be computed efficiently using the Digamma function ψ (Medina and Moll, 2009). Without loss of generality, if $\theta \sim \mathrm{Dir}(\alpha_1, \alpha_2, \ldots, \alpha_n)$, then

$$\mathbf{E}[\log \theta_i] = \psi(\alpha_i) - \psi(\Sigma \alpha_j).$$

It follows that

$$\exp \langle \log \theta^{x'}_{x,a} \rangle_{q_\theta} = \frac{\exp \psi(\beta^{x'}_{x,a})}{\exp \psi(\sum_{x''} \beta^{x''}_{x,a})} \qquad \text{and} \qquad \exp \langle \log \theta^{o'}_{x',a} \rangle_{q_\theta} = \frac{\exp \psi(\beta^{o'}_{x',a})}{\exp \psi(\sum_{o''} \beta^{o''}_{x',a})}. \tag{4.30}$$

The conditional updates to q_z (4.27) and q_θ (4.23) are iterated until sufficient convergence is obtained, thus concluding the E-step.

4.4.2 Variational Bayes M-step

Given the factorization $q = q_z q_\theta$, the M-step updates to Λ take the following form:

$$\nu_n^* \propto \sum_{t=0}^{T} q_z(t, n_0 = n) \tag{4.31}$$

$$\pi_{an}^* \propto \sum_{t=0}^{T} \sum_{\tau=0}^{t} q_z(t, a_\tau = a, n_\tau = n) \tag{4.32}$$

$$\lambda_{n'no'}^* \propto \sum_{t=1}^{T} \sum_{\tau=0}^{t-1} q_z(t, n_{\tau+1} = n', n_\tau = n, o_{\tau+1} = o'), \tag{4.33}$$

which can be derived by maximizing the lower bound (4.22) with respect to Λ, given fixed $q = q_z q_\theta$. Note that since we have approximated q by a factored distribution of the form $q_z q_\theta$, the marginalization of q_z is now trivial in contrast to the updates (4.4)–(4.6) when q is not constrained, which require integrating over the continuous model parameters θ.

4.5 A critique of VB-EM

We now offer a critique of the VB-EM algorithm just presented that is consistent with observations made by Furmston and Barber (2010) pertaining to the more restricted Bayes-adaptive MDP setting.

When the exact M-step updates are carried out as in (4.4)–(4.6), the marginals of q are reward-weighted and capture the relative worth of FSC policy decisions, thereby

informing FSC updates that are guaranteed to improve the controller with each iteration. In VB-EM, however, the FSC updates are computed using marginals of q_z. While these marginals are also reward-weighted—as can be seen from (4.28)—they also differ in an important way. Recall that q_z is defined in terms of sub-stochastic state-transition and observation-emission parameters (4.30). In other words,

$$\sum_{x'} \frac{\exp \psi(\beta^{x'}_{x,a})}{\exp \psi(\sum_{x''} \beta^{x''}_{x,a})} < 1 \qquad \text{and} \qquad \sum_{o'} \frac{\exp \psi(\beta^{o'}_{x',a})}{\exp \psi(\sum_{o''} \beta^{o''}_{x',a})} < 1, \tag{4.34}$$

where the level of sub-stochasticity will be greater for smaller sums $\sum_{x'} \beta^{x'}_{x,a}$ and $\sum_{o'} \beta^{o'}_{x',a}$, corresponding to larger variances in the independent Dirichlets that define q_θ. Due to the form of q_z (4.28), computing its marginals requires partitioning over all possible state-transitions and observation-emissions whenever an action a_t is drawn from $\pi_{\cdot n_t}$. Because the state-transition and observation-emission parameters are sub-stochastic, this partitioning effectively discounts the future expected reward associated with the current controller Λ; and the greater the sub-stochasticity, the larger this discount will be. As a result, controllers Λ that tend to "activate" model parameters with greater sub-stochasticity are discriminated against in the M-step updates, and hence an unfavorable preference can arise toward controllers Λ' that activate fewer sub-stochastic model parameters yet perform worse, that is, $J(\Lambda') < J(\Lambda)$. This phenomenon—which we will hereafter refer to as "variance-based discounting"—is particularly problematic when $\mathcal{P}$ contains some actions that cause deterministic state-transitions and/or observation-emissions and others that do not. In the Appendix (Example A), we show that VB-EM's variance-based discounting can lead to exceptionally poor FSCs even in the case of a trivial Bayes-adaptive MDP.

One approach to eliminating the undesirable effects of variance-based discounting is to simply choose a single model $\hat{\theta}$ (e.g., mean, mode) from the continuous prior p_θ and solve

the resulting standard POMDP with an existing POMDP algorithm. However, empirical results presented by Wang et al. (2012) and Ross et al. (2008, 2011), along with our analysis of the two-state problem in Chapter 3, suggest that such point-based policies can significantly underperform model-adaptive policies in a Bayesian setting. In the following section we propose a novel generalization of VB-EM that allows for tunable control over the effect of model variance on the M-step updates. Our approach—which we refer to as *constrained* VB-EM (CVB-EM)—strikes a balance between point-based methods and the VB-EM algorithm of Section 4.4. In particular, we (i) retain the general structure of VB-EM so that robustness against model uncertainty is preserved via the continuous q_θ distribution; and (ii) constrain the E-step so that q_θ is more concentrated on a single model, thus reducing the influence of variance-based discounting in the M-step.

4.6 Constrained VB-EM

For clarity of presentation, we now assume that the uncertain model parameters θ consist of a single probability vector $\theta = (\theta_1, \theta_2, \ldots, \theta_n)$ such that $n \geq 2$, with prior distribution $p_\theta(\theta) \sim \mathrm{Dir}(\theta|\alpha)$. In this context, θ could represent either a state-transition distribution or an observation-emission distribution. Note that the extension to the full BAPOMDP case is straightforward due to the independence assumption on the full set of Dirichlets that define the model prior p_θ in (4.20). Furthermore, we now explicitly constrain the update q_θ to have the convenient Dirichlet form, that is, $q_\theta \sim \mathrm{Dir}(\beta)$. For convenience, we reproduce the lower bound $L(q_z, q_\theta)$ originally defined in (4.22):

$$L(q_z, q_\theta) = -\langle \log q_z(z) \rangle_{q_z} - \langle \log q_\theta(\theta) \rangle_{q_\theta} + \langle \log p_\theta(\theta) \rangle_{q_\theta} + \langle \log \tilde{p}(z|\Lambda, \theta) \rangle_{q_z q_\theta}.$$

Let C denote any constant independent of β. Due to the $\mathrm{Dir}(\beta)$ form of q_θ, we can write

$$\begin{aligned}
\langle \log q_\theta(\theta) \rangle_{q_\theta} &= \int_\theta q_\theta(\theta) \log \left(\frac{\Gamma(\Sigma\beta_i)}{\prod \Gamma(\beta_i)} \prod_i \theta_i^{\beta_i - 1} \right) \, d\theta \\
&= \int_\theta q_\theta(\theta) \sum_i (\beta_i - 1) \log \theta_i \, d\theta + \log \frac{\Gamma(\Sigma\beta_i)}{\prod \Gamma(\beta_i)} \\
&= \sum_i (\beta_i - 1) \int_\theta q_\theta(\theta) \log \theta_i \, d\theta + \log \frac{\Gamma(\Sigma\beta_i)}{\prod \Gamma(\beta_i)} \\
&= \sum_i (\beta_i - 1) \big(\psi(\beta_i) - \psi(\Sigma\beta_j) \big) + \log \frac{\Gamma(\Sigma\beta_i)}{\prod \Gamma(\beta_i)}.
\end{aligned}$$

In a similar fashion, we find that

$$\begin{aligned}
\langle \log p_\theta(\theta) \rangle_{q_\theta} &= \sum_i (\alpha_i - 1) \big(\psi(\beta_i) - \psi(\Sigma\beta_j) \big) + C \\
\langle \log \tilde{p}(z|\Lambda, \theta) \rangle_{q_z q_\theta} &= \sum_i q_i \big(\psi(\beta_i) - \psi(\Sigma\beta_j) \big) + C,
\end{aligned}$$

where the q_i are uniquely determined scalars such that $\exp\langle \log \tilde{p}(z|\Lambda, \theta) \rangle_{q_z} \propto \prod_j \theta_j^{q_j}$, analogous to (4.25), with the specific formula depending on whether β corresponds to a state-transition distribution or an observation-emission distribution. Given the above expressions, the lower bound $L(q_z, q_\theta)$ can be expressed as

$$L(q_z, q_\theta) = \sum_i (\alpha_i + q_i - \beta_i) \big(\psi(\beta_i) - \psi(\Sigma\beta_j) \big) - \log \frac{\Gamma(\Sigma\beta_i)}{\prod \Gamma(\beta_i)} + C. \tag{4.35}$$

4.6.1 Constrained E-step

Recall that the M-step of VB-EM favors policies that tend to activate lower-variance model parameters. A natural approach to reduce this effect is to reward updates to q_θ that result in lower degrees of sub-stochasticity in q_z. In the most extreme case, we want to enforce the constraint $\sum_i e^{\psi(\beta_i)-\psi(\Sigma\beta_j)} = 1$. This gives rise to a *constrained* E-step, where the update to q_θ solves the following optimization problem:

$$\begin{aligned} \max_{\beta} \quad & \sum_i (v_i - \beta_i)\big(\psi(\beta_i) - \psi(\Sigma\beta_j)\big) - \log \frac{\Gamma(\Sigma\beta_i)}{\prod \Gamma(\beta_i)} \\ \text{s.t.} \quad & \sum_i e^{\psi(\beta_i)-\psi(\Sigma\beta_j)} = 1 \\ & \beta_i > 0 \ \text{ for all } i, \end{aligned} \tag{4.36}$$

where we have defined $v_i = \alpha_i + q_i$ for notational convenience. However, by Lemma B in the Appendix the strict inequality $\sum_i e^{\psi(\beta_i)-\psi(\Sigma\beta_j)} < 1$ holds for all $\beta > 0$, so that (4.36) has no feasible solutions. As an alternative, we introduce a Lagrangian relaxation to (4.36) by incorporating the stochasticity constraint as a penalty term in the objective. Let

$$\tilde{L}(\beta) = \sum_i (v_i - \beta_i)\big(\psi(\beta_i) - \psi(\Sigma\beta_j)\big) - \log \frac{\Gamma(\Sigma\beta_i)}{\prod \Gamma(\beta_i)} - \kappa\Big(1 - \sum_i e^{\psi(\beta_i)-\psi(\Sigma\beta_j)}\Big), \tag{4.37}$$

where the scalar multiplier $\kappa > 0$ indicates the strength of our aversion to sub-stochasticity in the q_z parameters. The Lagrangian relaxation to (4.36) is then

$$\begin{aligned} \max_{\beta} \quad & \tilde{L}(\beta) \\ \text{s.t.} \quad & \beta_i > 0 \ \text{ for all } i. \end{aligned} \tag{4.38}$$

A necessary condition for optimality in (4.38) is given by $\partial\tilde{L}/\partial\beta_i = 0$ for all $i = 1, \ldots, n$. From these conditions, it can be shown (Appendix, Lemma C) that any optimal β must satisfy either

$$\beta_i = v_i + \kappa e^{\psi(\beta_i) - \psi(\Sigma\beta_j)} \ \text{ for all } i, \tag{4.39}$$

or

$$\psi'(\Sigma\beta_j) \sum_j \frac{1}{\psi'(\beta_j)} = 1. \tag{4.40}$$

Furthermore, it can be shown that condition (4.40) will never hold since $\psi'(\Sigma\beta_j) \sum_j \frac{1}{\psi'(\beta_j)} < 1$ for all $\beta > 0$ (Appendix, Lemma D). Therefore any critical point must satisfy the system of equations in (4.39). In the following section we prove that there is a unique solution β^* to this system, and that it can be obtained via fixed-point iteration. Once established, the uniqueness of β^* implies that it must also be a global maximizer of (4.38), that is, $q_\theta \sim \text{Dir}(\beta^*)$ is the optimal constrained Dirichlet update in the CVB-EM algorithm.

4.6.2 Computing β^* via fixed-point iteration

From this point forward, we restrict our consideration to v such that $v_i \geq 1$ for all i. This is a very mild restriction, since the components of v are at least as large as the corresponding pseudo-counts α_i from the model prior p_θ, which we should expect to be no less than one in any instance we consider. We now prove that there exists a unique Dirichlet distribution satisfying the system of equations (4.39), and that this solution can be obtained via fixed-point iteration.

Theorem: *The system of equations given by* $\beta_i = v_i + \kappa e^{\psi(\beta_i) - \psi(\Sigma\beta_j)}$ *for all* $1 \leq i \leq n$ *has a unique solution, which can be obtained via fixed-point iteration.*

Proof outline: We express the system of equations as $G(\beta) = \beta$, where $G : B \mapsto B$ is a vector-valued function satisfying $G(\beta) = (g_1(\beta), g_2(\beta), \ldots, g_n(\beta))$ with $g_i(\beta) = v_i + \kappa e^{\psi(\beta_i) - \psi(\Sigma\beta_j)}$. Here B is an appropriately defined compact, convex set that is closed under the continuous mapping G, and includes all possible solutions to the nonlinear system. By Brouwer's fixed-point theorem (Istratescu, 2002) there must exist at least one solution $\beta^* \in B$ such that $G(\beta^*) = \beta^*$. We then show that the spectral radius $\rho(G'(\beta))$ of the Jacobian matrix $G'(\beta)$ satisfies $\rho(G'(\beta)) < 1$ for all $\beta \in B$. This proves that G is a contraction (Schwarz and Waldvogel, 1989) so that the fixed-point iteration $\beta^{(k+1)} = G(\beta^{(k)})$ converges to a unique solution β^*. □

The full proof of this theorem can be found in the Appendix (Theorem A).

4.6.3 Characterization of β^*

In the Appendix (Lemma E), we show that any β satisfying the system of equations (4.39) also satisfies

$$\beta_i = v_i + \kappa \frac{\beta_i}{\Sigma \beta_j} - d_i \tag{4.41}$$

for all $1 \leq i \leq n$, where each d_i is a scalar (not necessarily unique) such that $0 < d_i < 1/2$. Solving (4.41) for κ, we obtain

$$\kappa = \Sigma \beta_j - \frac{\Sigma \beta_j}{\beta_i}(v_i - d_i).$$

Equating the right-hand side of the above expression for i and some $l \neq i$, we find that

$$\beta_l = \beta_i \left(\frac{v_l - d_l}{v_i - d_i} \right),$$

and therefore by substitution

$$\frac{\beta_i}{\Sigma \beta_j} = \frac{v_i - d_i}{\sum_j v_j - d_j}. \tag{4.42}$$

From this expression, we see that the mean of the updated Dirichlet $q_\theta \sim \mathrm{Dir}(\beta^*)$ has a consistent and non-degenerate form for any $\kappa > 0$.

Let us consider the behavior of CVB-EM as $\kappa \to \infty$, corresponding to a tightening of the soft constraint on q_θ. First, it is straightforward to show via summation of the right-hand and left-hand sides of (4.41) that the fixed point β^* satisfies $\Sigma\beta_j^* > \kappa$. Therefore $\kappa \to \infty$ implies $\Sigma\beta_j^* \to \infty$, by which $\mathrm{Var}[q_\theta] \to 0$. As a result, as κ tends to infinity the updated $q_\theta \sim \mathrm{Dir}(\beta^*)$ will become increasingly concentrated towards a mean of the form (4.42).

4.6.4 Practical considerations

We have described how the unfavorable variance-based discounting of certain policies can be mitigated by imposing a soft constraint on the form of the update to q_θ. The soft constraint is tightened by increasing the penalty multiplier κ, which can be distinct for each of the independent Dirichlets in the model prior p_θ. Furthermore, as an alternative to simply choosing a point estimate $\hat{\theta}$ from the model prior, the effect of variance-based discounting can be eliminated completely by considering the limiting case as $\kappa \to \infty$.

The CVB-EM algorithm of Section 4.6 requires the modeler to select a κ for each Dirichlet composing the distribution p_θ *a priori*, but does not prescribe how the κ parameters should be chosen to maximize the performance $J(\Lambda)$ of the resulting FSC Λ. As a rule of thumb, the κ parameters should be large enough to mitigate the undesirable effect of variance-based discounting, while not so large that the peakedness of q_θ fails to capture the uncertainty given by the model prior p_θ. When the standard VB-EM algorithm fails to generate good policies, this is likely because increases in the VB-EM lower bound (4.22) do not correspond to significant increases in $\log J(\Lambda)$. This suggests that CVB-EM will perform best when the κ parameters are chosen to more tightly couple the CVB-EM lower bound (4.37) to the true objective $\log J(\Lambda)$. Because there is no obvious mechanism for the optimal selection of these parameters, in the empirical study that follows we simply run CVB-EM over a range of multipliers κ, where the same κ is used for each Dirichlet

distribution in the prior. In Section 4.9 we revisit this issue and outline a possible heuristic for the more sophisticated selection of the κ parameters.

4.7 An empirical study

In this section we apply the CVB-EM algorithm to a pair of test problems. We compare the performance of FSCs generated by CVB-EM to policies derived from the mean and mode point estimates of the prior p_θ, and we compare the sampling-based EM algorithm of Section 4.3 to the sampling-based PBVI approach of Wang et al. (2012). Historically, BAPOMDP policies have been evaluated assuming that a *single* model from the prior p_θ is true, so that empirical performance captures how well the BAPOMDP policy adapts to this single model over time. However, the selection of a single model from the prior is somewhat arbitrary and policy performance will likely be sensitive to this choice (recall the discussion of Section 1.1.2). As a result, we adopt the actual BAPOMDP objective (1.1) as our performance criterion, which we believe to be a more robust measure of policy performance given prior model uncertainty.

4.7.1 Problem definitions

Let $\mathcal{P} = \{X, A, O, u, \gamma, p_\theta\}$ be an infinite-horizon BAPOMDP with the following properties: First, the state space is defined such that $X = \{1, \ldots, n\}$, and the process always begins in state 1. The agent has two actions at its disposal, "go" (g) and "shuffle" (s), so that $A = \{g, s\}$. Under action g, each model $\theta \in \Theta$ defines a left-to-right HMM on the states $1 \leq x \leq n$, where n is an absorbing state. In particular, for states $1 \leq x < n$, the probability of self-transition under action g is p^g and the probability of transitioning to state $x+1$ is $1-p^g$. Furthermore, $u(g, x) = r > 0$ for all $1 \leq x < n$ and $u(g, n) = -c_2 < 0$.

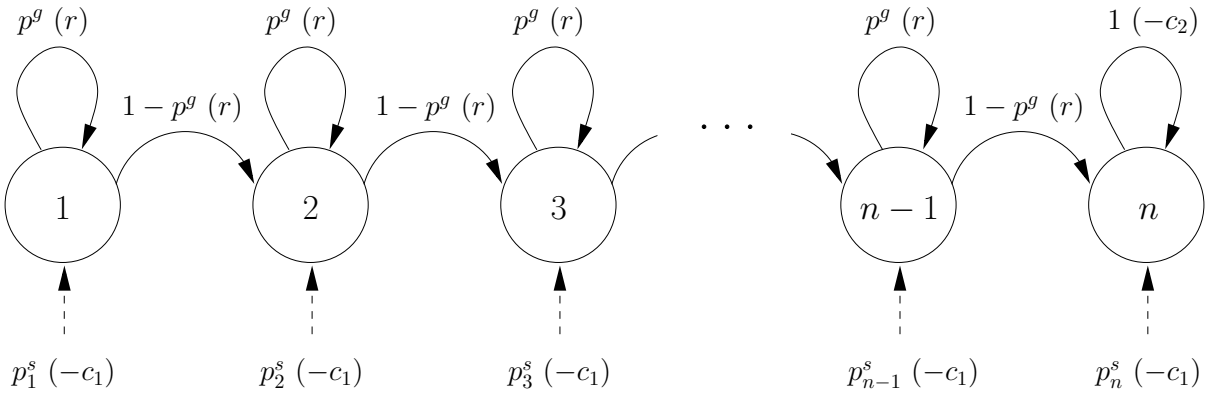

Figure 4.2: Diagram of the BAPOMDP *Shuffle* from Section 4.7.1. Rewards associated with each transition are indicated in parentheses. For simplicity, the observation-emission distributions are not included.

When action s is taken, the current state is "shuffled" by transitioning from this state to any of the n states with probabilities p_x^s for $1 \leq x \leq n$, and a reward of $u(s, \cdot) = -c_1$ is received for taking this action. Generally, we assume that $-c_2 \leq -c_1 \leq 0$.

When the goal is to maximize the discounted reward over an infinite horizon, we refer to this problem as *Shuffle*. In a fully observable setting, an optimal *Shuffle* policy would take action g in all states $1 \leq x < n$ and take action s in state n. However, to make the problem challenging we assume that the true state is not directly visible to the agent. Rather, each state x emits a discrete observation $o \sim p(\cdot|x)$ that can be used to infer the hidden state. Problem *Shuffle* is summarized by the diagram in Figure 4.2. Note that the state-transition and observation-emission probabilities described above are implicitly parameterized by the model $\theta \sim p_\theta(\cdot)$. We will also consider a T-horizon variation of problem *Shuffle*, which we refer to as *Stop*. Problem *Stop* differs from problem *Shuffle* in that action s now terminates the reward process with an immediate reward of 0.

Recall that u must satisfy $u \geq 0$ to apply the EM algorithms. A *Shuffle* BAPOMDP, for example, can be transformed into the equivalent BAPOMDP $\mathcal{P}' = \{X, A, O, u', \gamma, p_\theta\}$, where we set $u' = u + c_2 \geq 0$. In the following section we conduct an algorithm performance comparison by choosing specific inputs to parameterize *Shuffle* and *Stop*

BAPOMDPs, including model priors p_θ given by products of independent Dirichlet distributions.

4.7.2 Numerical results

The instances of *Stop* and *Shuffle* that we examine include a larger number of uncertain model parameters than the problems typically encountered in the BAPOMDP literature, and by necessity are generally smaller in size than the problems typically encountered as well (as judged by the number of states, actions, and observations). By restricting our attention to smaller BAPOMDPs, our performance comparison is more attuned to evaluating the overarching goal of BAPOMDP policies, that is, robustness against model uncertainty.

First, we consider a BAPOMDP parameterization of *Stop* given by $n = 5$, $r = 1$, $c = 5$, $T = 20$, and $\gamma = 1$. The self-transition probability p^g follows a $\text{Beta}(4, 2)$ prior and is shared by all states $1 \leq x < n$. Furthermore, we assume $|O| = |X|$ and that for each state $x' = 1, \ldots, n$ the observation-emission probabilities $p(\cdot|x')$—which are here independent of the action—follow a $\text{Dir}(\alpha)$ prior, where $\alpha = (\alpha_1, \ldots, \alpha_n)$ and $\alpha_i = n - |i - x'|$. This serves to correlate more highly states and observations that are closer in index.

Next, we consider a BAPOMDP parameterization of *Shuffle* given by $n = 5$, $r = 5$, $c_1 = c_2 = 10$, $\gamma = 0.95$. As with problem *Stop*, the self-transition probability p^g under action g follows a $\text{Beta}(4, 2)$ prior and is shared by all states $1 \leq x < n$. Also as before, we assume $|O| = |X|$ and that for each state $x' = 1, \ldots, n$ the observation-emission probabilities $p(\cdot|x')$ follow a $\text{Dir}(\alpha)$ prior, where $\alpha = (\alpha_1, \ldots, \alpha_n)$ and $\alpha_i = n - |i - x'|$. Under action "shuffle" (s), the conditional transition probabilities follow a $\text{Dir}(n, n-1, \ldots, 1)$ prior that is shared by all n states.

We now apply the CVB-EM algorithm to the instances of *Stop* and *Shuffle* defined above. To this end, we run the CVB-EM algorithm over a range of κ values, using the same κ for each Dirichlet in the model prior. For each $\kappa \geq 0$ considered, CVB-EM is run to convergence over five independent trials, each seeded with a random FSC $\Lambda^{(0)}$ where $|N| = 30$ for *Stop* and $|N| = 60$ for *Shuffle*. To better gauge the effectiveness of CVB-EM, we compare its performance to that of existing BAPOMDP algorithms. As one alternative, we compute the mean and mode point estimates of the prior p_θ, and then approximate the optimal policy via a standard implementation of point-based value iteration (Cassandra, 2009), assuming that the point estimate is the true model (algorithms PBVI-MEAN and PBVI-MODE, respectively). Policies constructed in this way are by definition not model-adaptive and hence serve as crude lower bounds to Bayes optimal performance. Note that algorithms PBVI-MEAN and PBVI-MODE are run only once, since we found the policies generated to be relatively insensitive to random belief point selection. As a second alternative, we sample 10, 100, and 1000 models $\tilde{\Theta}$ from the prior p_θ and then apply the sampling-based EM algorithm of Section 4.3 (algorithm EM-SAMPLE) and the sampling-based PBVI approach of Wang et al. (2012) (algorithm PBVI-SAMPLE) to the resulting discrete BAPOMDP. For each finite sample $\tilde{\Theta}$ of size $|\tilde{\Theta}| = 10, 100, 1000$ drawn from p_θ, algorithms EM-SAMPLE and PBVI-SAMPLE are run to convergence using the same uniform discrete prior over $\tilde{\Theta}$, and this experiment is repeated five times for each sample size. As with CVB-EM, algorithm EM-SAMPLE is initialized with random FSCs of sizes $|N| = 30$ and $|N| = 60$ for *Stop* and *Shuffle*, respectively.

Additionally, we supplement EM-SAMPLE with a forward-search subroutine for escaping locally optimal FSCs, originally proposed by Poupart et al. (2011b) in the context of standard POMDPs and extended herein (with some additional modifications) to the more general BAPOMDP setting. Essentially, forward-search grows the current FSC via exploration from the initial belief, during which new nodes are added to the controller when

suboptimal actions and/or successor nodes are encountered. In this way, forward-search has the potential to improve locally optimal FSCs by inserting sequences of nodes that are explicitly model-adaptive. A formal statement of the forward-search subroutine can be found in Section 5.2. Our convention in this study was to run forward-search as an intermediate procedure during EM whenever a locally optimal FSC was reached, adding no more than $O(500)$ new nodes to the controller in total. As such, the FSCs generated by EM with forward-search were more parsimonious than the policies generated by PBVI-SAMPLE, which required $O(1000)$ α-vectors at each decision epoch to attain the levels of performance cited below.

It should also be noted that we adopt a *parameterized* M-step update to accelerate the convergence of all EM algorithms considered in this study. With a more detailed description to follow in Section 5.1—along with a performance comparison of parameterized EM to ordinary EM—the parameterized M-step essentially leverages the standard EM update to approximate the objective function's gradient in the neighborhood of the current solution $\Lambda^{(k)}$ during each iteration k, which is then used to potentially improve $\Lambda^{(k)}$ by pushing the update even further in the direction of the approximated gradient.

The performance of each policy generated in this study is approximated via Monte-Carlo simulation subject to the continuous model prior p_θ. Performance plots comparing algorithm CVB-EM to algorithms PBVI-MEAN and PBVI-MODE can be found in Figures 4.3 and 4.4 for problems *Stop* and *Shuffle*, respectively. For each κ, the CVB-EM performance curve is centered at the mean of five independent trials, with a vertical bar used to denote the best and worst controllers obtained over these trials. Table 4.1 contains various statistics comparing the performance of algorithms EM-SAMPLE and PBVI-SAMPLE.

First, let us consider the performance plot in Figure 4.3. While not pictured, when κ is sufficiently small (e.g., $\kappa < 200$) we find that CVB-EM converges to the trivial stopping policy in which action s is taken at time $t = 0$, corresponding to an expected

reward of zero. The explanation for this phenomenon is analogous to that of Example A in the Appendix: Low-variance priors on the state-transition and observation-emission probabilities lead to variance-based discounting of policies that include action g, so that the trivial stopping policy is preferred in the M-step updates. As κ increases, however, the variance-based discounting effect is dampened and CVB-EM performance improves dramatically before tapering off with performance comparable to that of PBVI-MEAN as $\kappa \to \infty$. CVB-EM performance attains a maximum of 6.56 near $\kappa = 340$ following a steep rise in performance after $\kappa = 220$. Notably, for $\kappa > 240$ CVB-EM dominates both PBVI-MEAN (6.3) and PBVI-MODE (5.32). Figure 4.3 suggests the general conclusion that CVB-EM is most successful when a balance is struck between ordinary VB-EM—which arises as $\kappa \to 0$ and maintains a robustness against uncertainty via q_θ—and point-based optimization—which is approximated as $\kappa \to \infty$ and does not suffer from the effects of variance-based discounting.

Comparing Figure 4.3 to Table 4.1, we find that CVB-EM outperforms EM-SAMPLE and PBVI-SAMPLE for $|\tilde{\Theta}| = 10$, but is inferior to the sampling-based approaches for $|\tilde{\Theta}| = 100$ and 1000. Intuitively, this is so because the variational Bayes approximation to the model prior is inadequate relative to the sampled approximation for larger $|\tilde{\Theta}|$. The superiority of PBVI-SAMPLE to EM-SAMPLE for $|\tilde{\Theta}| = 100$ and 1000 is an expected result: the α-vector policy generated by PBVI-SAMPLE is large relative to the FSC of sizes $|N| = 30$ and $|N| = 60$ optimized via EM-SAMPLE, and is also explicitly model-adaptive by virtue of the backup operation executed during each iteration over the joint model-state belief vector. While EM-SAMPLE lacks the explicit model-adaptivity of PBVI-SAMPLE, it succeeds in generating FSCs that display model-adaptive behavior and perform well "in the average case", as judged by their superiority to PBVI-MODE and PBVI-MEAN. Moreover, the statistics in Table 4.1 indicate that by injecting explicit model-adaptivity into the FSC via forward-search, FSC performance is comparable to that of PBVI-SAMPLE while

maintaining a parsimonious size. Of course, the performance of all EM-based algorithms can be marginally improved by increasing the number of belief nodes $|N|$, albeit at a greater computational cost.

The analysis of the infinite-horizon *Shuffle* instance is similar. Algorithm CVB-EM performance increases sharply near $\kappa = 20$ and reaches its peak at approximately $\kappa = 1100$ before leveling off as $\kappa \to \infty$ (see Figure 4.4). The CVB-EM algorithm achieves a maximum performance of 40.81, exceeding the PBVI-MODE and PBVI-MEAN performances of 39.0 and 40.7, respectively. Although, there is very little separating CVB-EM and PBVI-MEAN performance for $\kappa > 900$, indicating that the FSCs generated by CVB-EM may lack robustness against model uncertainty in this infinite-horizon setting. One likely explanation is that the infinite planning horizon of problem *Shuffle* magnifies the variance-based discounting effect, which can only be mitigated by choosing a large κ. (Here $\kappa \approx 1100$ produces the best performance, as compared to $\kappa \approx 340$ for the finite-horizon problem *Stop*.) Large κ implies that q_θ will be highly concentrated on a single model and hence FSCs generated by CVB-EM will effectively be point-based. Note, however, that CVB-EM remains a significant improvement to standard variational Bayes EM for problem *Shuffle*, which corresponds to $\kappa \to 0$ in Figure 4.4. As expected, the sampling-based algorithms perform better as $|\tilde{\Theta}|$ increases. EM-SAMPLE is generally superior to both point-based algorithms and CVB-EM, and inferior to the approaches that display explicit model-adaptivity, that is, PBVI-SAMPLE and EM-SAMPLE with forward-search.

4.8 Application: A case study in manufacturing

We now return to the manufacturing case study first introduced in Section 1.2. There we demonstrated the unpredictable performance of the mean and mode point-based policies,

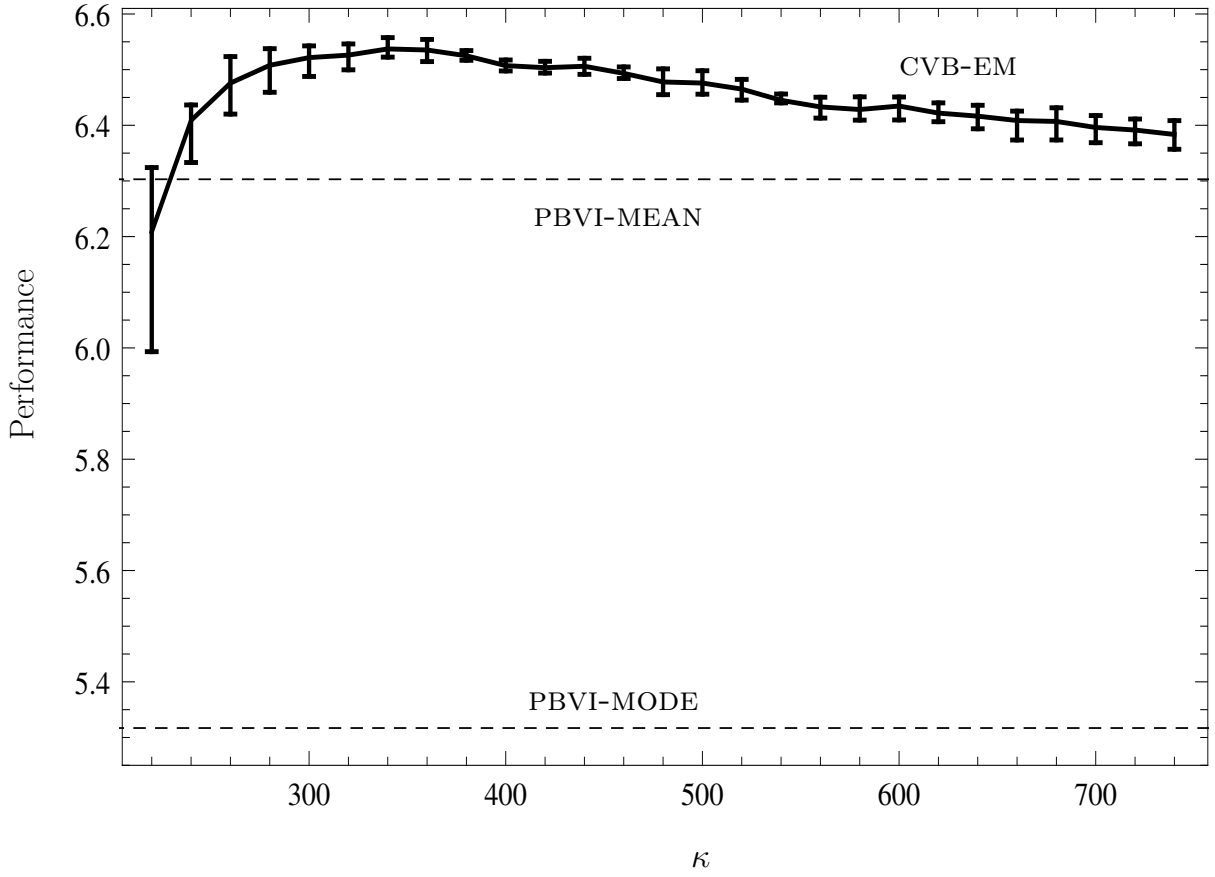

Figure 4.3: Performance plot for algorithms CVB-EM, PBVI-MEAN, and PBVI-MODE when applied to the *Stop* instance of Section 4.7.2.

Algorithm	$\lvert\tilde{\Theta}\rvert$	Min.	Mean	Max.
EM-SAMPLE	10	6.20	6.32	6.40
	100	6.49	6.54	6.60
	1000	6.55	6.58	6.60
EM-SAMPLE + forward-search	10	6.16	6.30	6.42
	100	6.58	6.60	6.65
	1000	6.57	6.63	6.67
PBVI-SAMPLE	10	6.10	6.29	6.44
	100	6.59	6.63	6.67
	1000	6.67	6.68	6.69

Problem *Stop*

Algorithm	$\lvert\tilde{\Theta}\rvert$	Min.	Mean	Max.
EM-SAMPLE	10	38.30	39.30	40.84
	100	41.41	41.45	41.49
	1000	41.49	41.52	41.55
EM-SAMPLE + forward-search	10	38.35	39.40	40.92
	100	41.52	41.60	41.64
	1000	41.62	41.70	41.81
PBVI-SAMPLE	10	37.94	39.08	40.80
	100	41.71	41.73	41.78
	1000	41.72	41.78	41.85

Problem *Shuffle*

Table 4.1: Performance comparison of the EM and PBVI sampling-based algorithms for BAPOMDPs.

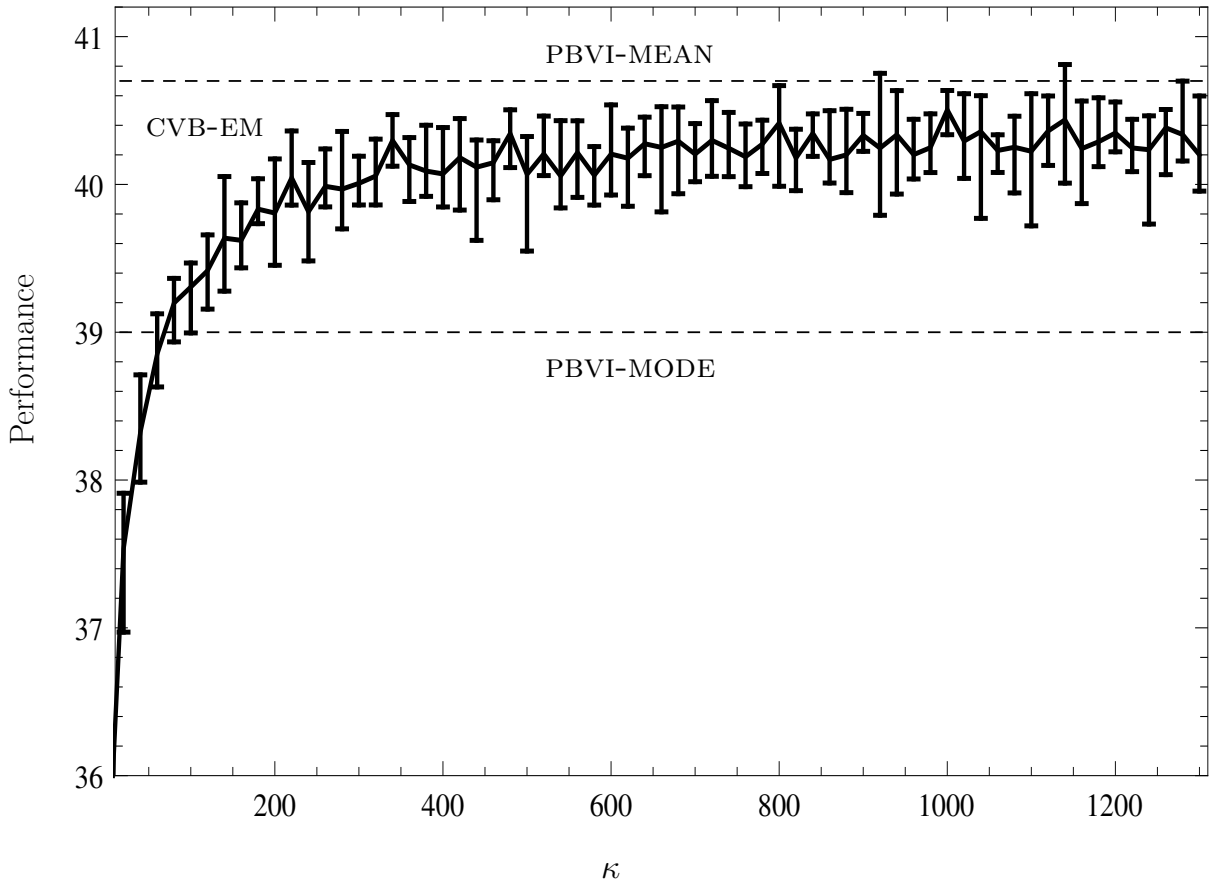

Figure 4.4: Performance plot for algorithms CVB-EM, PBVI-MEAN, and PBVI-MODE when applied to the *Shuffle* instance of Section 4.7.2.

and their inferiority to the optimal policy under the true, generative HMM θ^*. Our objective in this section is to show that FSCs generated by applying expectation-maximization to an appropriate BAPOMDP can outperform the mean and mode point-based policies on a consistent basis.

To this end, we must first define the model prior p_θ. Recall that our convention in this study was to assume an uninformed Dirichlet prior over the state-transition and observation-emission probabilities, which was then updated via Bayes rule given independent, partially supervised training sequences. A natural approach is to use the resulting posterior distribution as the model prior p_θ in a BAPOMDP formulation. Because the training sequences are only partially supervised—recall that each training sequence

consisted only of observed features and a binary value indicating whether the broach is in a good ($x_{K-1} < n$) or bad ($x_{K-1} = n$) condition during the K^{th} and final cutting pass—the model posterior will *not* be a product of independent Dirichlets and hence the constrained VB-EM algorithm of Section 4.6 is not applicable. However, we can randomly draw a large number of models $\tilde{\Theta} \subset \Theta$ from the posterior p_θ via Gibbs' procedure (Cappé et al., 2005; Rydén, 2008) and then apply the sampling-based EM algorithm to the discretized BAPOMDP using a discrete uniform prior over $\tilde{\Theta}$.

Recall that each panel in Figure 1.2 corresponds to a unique set of training data arising from the same generative tool wear HMM. Here we repeat these experiments and include performance of the sampling-based EM algorithm applied to 100 models $\tilde{\Theta}$ drawn via Gibbs' procedure from the model posterior, which is a function synthetic training data unique to the trial. In each case, the EM algorithm is initialized with a random FSC of size $|N| = 60$. We adopt the parameterized EM approach to accelerate convergence, but forego the application of forward-search both for simplicity and, more importantly, to demonstrate that EM alone is capable of dominating point-based performance in this context.

The results in Figure 4.5 show that controllers generated by sampling-based EM do, in fact, dominate the mean and mode point-based policies. While EM performance is only marginally superior to mean model performance in panels 3, 6, 7, and 8 (from left to right, and top to bottom) in the remaining five cases EM significantly outperforms both point-based policies and compares favorably to optimal performance under the generative model θ^*. Surprisingly, this holds despite the local optimality of EM solutions, the comparatively parsimonious FSC size, and the lack of *explicit* model-adaptivity inherent in the M-step updates. Of course, EM performance could be further improved in this application by increasing the controller size, $|N|$, and injecting explicit model-adaptivity via forward-search, as supported by the empirical results in Section 4.7.

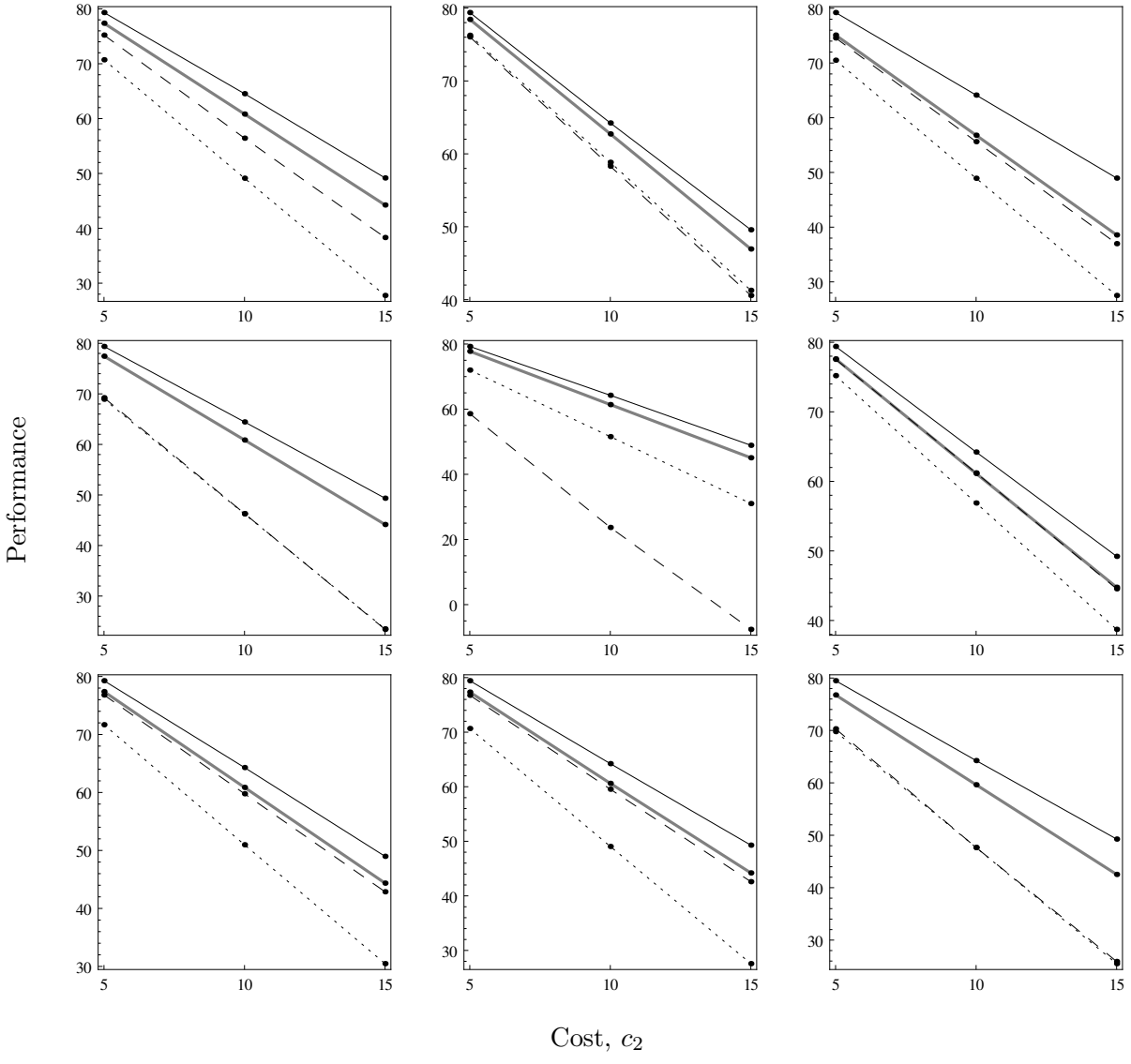

Figure 4.5: Performance plots for the Rolls-Royce case study using synthetic broaching data for model training. Solid lines indicate performance of policies derived from the generative model, dashed lines indicate performance of policies derived from the mode point estimates, dotted lines indicate performance of policies derived from the mean point estimates, and gray lines indicate FSC performance resulting from the sampling-based EM algorithm applied to 100 models drawn via Gibbs' procedure from the model posterior. Each plot corresponds to a unique set of training data generated by the true model θ^*.

4.9 Summary

In this chapter we presented a general expectation-maximization algorithm for solving BAPOMDPs via finite-state controller optimization and proposed two alternatives for addressing the intractable integral that arises in the M-step. The first is a sampling-based algorithm that optimizes an FSC subject to a finite number of models drawn from the BAPOMDP prior distribution. Here the integral is replaced by a summation that can be distributed over multiple concurrent threads. Notably, there is no analogous parallelization in the PBVI framework due to the mixing of model parameters in the augmented belief vector. As a second alternative, we derived a variational Bayes EM algorithm that permits efficient M-step updates when the model prior is given by a product of independent Dirichlet distributions. However, as described in Section 4.5 and demonstrated by example in the Appendix (Example A), the VB-EM factorization of $q = q_z q_\theta$ discriminates against policies that activate higher-variance model parameters, as a result of sub-stochasticity in the state-transition and observation-emission quantities defining q_z. This phenomenon can be detrimental, causing VB-EM to generate policies of monotonically decreasing reward before converging to a sub-optimal controller. To mitigate the effects of variance-based discounting we proposed a *constrained* VB-EM algorithm, which reduces sub-stochasticity in the q_z parameters by rewarding lower-variance updates to q_θ. By preserving the Dirichlet form in the update to q_θ—as opposed to optimizing with respect to a single point estimate from the model prior—the CVB-EM algorithm maintains a robustness against model uncertainty. Our empirical study supports this claim, demonstrating that CVB-EM can significantly outperform policies derived from the mean and mode of p_θ and is far superior to the standard VB-EM algorithm ($\kappa \to 0$). While the sampling-based approaches of EM and PBVI generally outperformed CVB-EM, their success owed in part to the relatively small problem instances considered, and as the space of uncertain model parameters increases, the curse of dimensionality will render the sampling-based algorithms

impractical. In particular, explicit model-adaptive control (achieved via the augmented model-state belief in the PBVI context and via forward-search in the EM context) will no longer be feasible. It is in these settings where the optimization of bounded FSCs via CVB-EM or sampling-based EM (without forward-search) will be most attractive, striking a balance between computational tractability and robustness against model uncertainty.

For simplicity, our CVB-EM implementation used same κ parameter for all Dirichlet distributions in the model prior p_θ. Our experiments suggest that CVB-EM will perform best when κ is large enough to mitigate the ill effects of variance-based discounting, while not so large that the update to q_θ resolves uncertainty in p_θ to a single point estimate. Unfortunately, this conclusion does not provide much insight into how the κ parameters should be chosen to *optimize* CVB-EM performance. At a high level, our goal should be to choose these parameters so that increases to the CVB-EM lower bound are more tightly coupled to increases in the true objective $\log J(\Lambda)$. While a rigorous analysis of this relationship could be fruitful, we leave this as question for future research and conclude the discussion by offering the following (untested) heuristic for adaptively setting the κ parameters:

Suppose that our goal is optimize a bounded FSC for some BAPOMDP. Let us consider the optimal FSC Λ^* and assess how VB-EM might discriminate against this policy, and how we could correct this behavior with CVB-EM. Say, for example, that Λ^* tends to take action a in state x, but that the corresponding prior on the state-transition distribution $p(\cdot|x,a)$ is highly variable. Consequently, the FSC returned by VB-EM might avoid this action altogether as a result of variance-based discounting. In theory, we could mitigate this unfavorable discounting by choosing a large enough κ multiplier for $p(\cdot|x,a)$ in a CVB-EM implementation. Assuming Λ^* was known, the relative frequency with which action a is taken in state x could be approximated by running forward-backward on the factor graph corresponding to Λ^* and, say, the mean HMM $\bar{\theta}$ of the prior p_θ. The κ multiplier for

each Dirichlet could be set proportionally to its corresponding frequency so that policies consistent with Λ^* are favored. With respect to the observation-emission distributions, we would similarly have to assess the frequency with which, for example, taking action a causes a transition to a state x', thereby activating the observation-emission distribution $p(\cdot|x', a)$. Of course, Λ^* will not be known and hence a heuristic alternative is required. One approach is to do what is prescribed above, but replace Λ^* with an FSC obtained by optimizing the POMDP corresponding to the HMM $\bar{\theta}$. If the resulting FSC is "close enough" to Λ^*, then a rough approximation of the desired frequencies could be obtained by running forward-backward on the factor graph corresponding to this FSC and the mean HMM $\bar{\theta}$. Open questions still remain, however, such as the specific mechanism by which the frequencies described above should be converted into proportional κ weights. As one alternative, the trace over κ in the experiments of Section 4.7 could be replaced by a trace over a proportionality constant that maps the distinct frequency for each state-transition and observation-emission distribution to a *unique* κ value. In this way—at the very least—we overcome the limitation of using the same κ for all Dirichlets, while still only having to trace over a single dimension.

Chapter 5

Improving EM performance

Expectation-maximization is often criticized for its slow rate of convergence and the local optimality of its fixed points. In this chapter we develop techniques to address these criticisms in the context of EM for (BA)POMDPs and provide empirical results to demonstrate their usefulness.

5.1 Accelerating convergence with parameterized EM

In this section we derive an efficient parameterized EM algorithm for (BA)POMDPs that improves the convergence rate by facilitating greedier updates to the FSC parameters. Parameterized EM for (BA)POMDPs updates the current FSC $\Lambda^{(k)}$ at iteration k by first computing the subsequent ordinary EM update $\Lambda_{\text{EM}}^{(k+1)}$, and then selecting $\Lambda^{(k+1)}$ such that

$$\Lambda^{(k+1)} = \Lambda^{(k)} + \Delta^{(k)}(\Lambda_{\text{EM}}^{(k+1)} - \Lambda^{(k)}), \tag{5.1}$$

for some appropriately chosen positive scalar $\Delta^{(k)}$. In this way, the difference $\Lambda_{\text{EM}}^{(k+1)} - \Lambda^{(k)}$ is used to approximate the true gradient of $\log J(\Lambda)$ in the neighborhood of $\Lambda^{(k)}$. Note that when $\Delta^{(k)} = 1$ we have $\Lambda^{(k+1)} = \Lambda_{\text{EM}}^{(k+1)}$, so that parameterized EM reduces to ordinary EM.

The success of parameterized EM is determined by the choice of step-size $\Delta^{(k)}$. When the ratio of missing information to complete information is small, EM will exhibit super-linear, Quasi-Newton convergence (Salakhutdinov et al., 2003), and hence executing an ordinary EM iteration with $\Delta^{(k)} = 1$ should be sufficient. When this ratio is large, however, EM will often require an inordinate number of iterations before a significant improvement to FSC performance is realized. In the latter case, consecutive iterations will tend to shift the FSC in a similar direction and by a similar magnitude, so that considerable gains in efficiency can be obtained by choosing $\Delta^{(k)} \gg 1$ with no negative effect on performance.

With this in mind, we now build a simple parameterized EM algorithm for FSC optimization. Let $\Lambda^{(k)}$ be the FSC at the beginning of some iteration $k \geq 0$. Our goal is to choose a step-size $\Delta^{(k)} > 0$ that can improve the resulting FSC $\Lambda^{(k+1)}$ beyond the ordinary EM update. To this end, we must first take care to disregard infeasible step-sizes $\Delta^{(k)}$. In particular, the components of $\nu^{(k+1)}$, $\pi_{\cdot n}^{(k+1)}$ and $\lambda_{\cdot no'}^{(k+1)}$ must be nonnegative and sum to one. Given the current FSC $\Lambda^{(k)}$ and a step-size $\Delta^{(k)} > 0$, it is straightforward to verify that the components of $\nu^{(k+1)}$ will sum to one:

$$\begin{aligned}
\sum_n \nu_n^{(k+1)} &= \sum_n \nu_n^{(k)} + \Delta^{(k)}\big(\nu_{\text{EM},n}^{(k+1)} - \nu_n^{(k)}\big) \\
&= \sum_n \nu_n^{(k)} + \Delta^{(k)} \sum_n \big(\nu_{\text{EM},n}^{(k+1)} - \nu_n^{(k)}\big) \\
&= 1 + \Delta^{(k)} \cdot 0 \\
&= 1,
\end{aligned}$$

and the same can be shown for all other PMFs defining $\Lambda^{(k+1)}$. Therefore, to maintain feasibility in the updated FSC it is sufficient to bound $\Delta^{(k)} < \bar{\Delta}^{(k)}$, where $\bar{\Delta}^{(k)}$ is the smallest step-size for which some component of the updated FSC $\Lambda^{(k+1)}$ becomes negative. With this in mind, the parameterized EM update at iteration $k \geq 0$ proceeds as follows: Given a set $D^{(k)}$ of feasible step-sizes, the subsequent FSC $\Lambda^{(k+1)}$ is chosen according to the rule

$$\begin{aligned}\Lambda^{(k+1)} &= \underset{\Delta \in D^{(k)}}{\operatorname{argmax}} \; J\big(\, \Lambda^{(k)} + \Delta(\Lambda_{\mathrm{EM}}^{(k+1)} - \Lambda^{(k)}) \,\big) \\ &= \underset{\Delta \in D^{(k)}}{\operatorname{argmax}} \; J(\Lambda_{\Delta}^{(k)}). \end{aligned} \tag{5.2}$$

The parameterized EM update (5.2) requires a comparison of $J(\Lambda_{\Delta}^{(k)})$ over all $\Delta \in D^{(k)}$. To this end, we run the forward algorithm (Algorithm 4.1) for each Δ and compute $J(\Lambda_{\Delta}^{(k)})$ using the forward messages μ as described by equation (4.13). Ideally, our choice of $D^{(k)}$ will (i) maintain the monotonicity property of EM, (ii) allow for accelerated iterations, and (iii) admit a tractable FSC update via (5.2). A simple choice of $D^{(k)}$ that satisfies all three conditions is

$$D^{(k)} = \{1, d \cdot \bar{\Delta}^{(k)}\} \tag{5.3}$$

for some $0 < d < 1$. Monotonicity is guaranteed by $1 \in D^{(k)}$, acceleration is permitted by inclusion of the feasible (and possibly large) step-size $d \cdot \bar{\Delta}^{(k)}$, and $|D^{(k)}| = 2$ ensures the tractability of the parameterized EM update. As an added benefit, parameterized EM can also be useful for avoiding locally optimal solutions. For instance, if the update at

iteration $k+1$ satisfies $J(\Lambda^{(k+1)}) > J(\Lambda_{\text{EM}}^{(k+1)})$, then the EM trajectory could "overshoot" the nearest local optimum in the direction of a more favorable solution. A simple approach to FSC optimization with parameterized EM is summarized in Algorithm 5.1.

Algorithm 5.1 A parameterized EM algorithm for BAPOMDPs.

Input: Initial FSC $\Lambda^{(0)}$, $\varepsilon > 0$
Output: Local optimum Λ^*

$k \leftarrow 0$
$converged \leftarrow false$
while $\neg converged$ **do**
  E-step: Compute the marginals of q given $\Lambda^{(k)}$
  M-step: Compute $\Lambda_{\text{EM}}^{(k+1)}$ using the marginals of q
  $\Lambda^{(k+1)} \leftarrow \text{argmax}_{\Delta \in D^{(k)}} J(\Lambda_{\Delta}^{(k)})$
  if $||J(\Lambda^{(k+1)}) - J(\Lambda^{(k)})|| < \varepsilon$ **then**
    $converged \leftarrow true$
  end if
  $k \leftarrow k+1$
end while

5.2 Escaping local optima with forward-search

Poupart et al. (2011b) evaluate two novel alternatives for escaping locally optimal FSCs in the POMDP setting. The first, *node-splitting*, is a subroutine for the EM algorithm that independently copies each node $n \in N$ and reruns EM to convergence by warm-starting from the current FSC. The split resulting in the best FSC is retained, thus adding a single node to the controller with each iteration. This process is repeated until the gain in performance associated with the most recent split is sufficiently small. The second, *forward-search*, is a subroutine for infinite-horizon POMDPs that performs a depth-constrained search through the FSC starting from the initial belief. When a suboptimal action or successor node is encountered, a new node is added to the FSC with the optimal action and successor node distributions. The authors evaluate both

node-splitting and forward-search as supplements to the ordinary EM algorithm. To this end, EM performance is compared to that of state-of-the-art PBVI algorithms, such as heuristic search value iteration (HSVI, see Section 2.2.1.2), when applied to a variety of challenging benchmark POMDPs. While both subroutines were successful in generating FSCs competitive with PBVI policies, the authors found node-splitting to be comparatively intractable due to the number of times, $|N|$, that EM must be run to convergence when the subroutine is called. For this reason we focus exclusively on forward-search from this point onward.

Forward-search can be used as an intermediate procedure to supplement the EM algorithm when a local optimum is encountered. In particular, the addition of a small number of well-chosen nodes may be sufficient to escape the optimum, after which EM can renew its ascent towards the globally optimal solution with ordinary updates. Unfortunately, this use of forward-search is less effective for larger BAPOMDPs, since an inordinate number of nodes may need to be added to achieve the same effect, thus rendering subsequent EM iterations intractable. In this case forward-search could be used as a terminal procedure to improve FSC performance through the one-time addition of a large number of nodes.

Owing to the success of forward-search for POMDPs, we have extended this subroutine (with some additional modifications) to the more general BAPOMDP setting and also derived a variation for finite-horizon problems, as this case is not addressed by Poupart et al. (2011b) but is still of general interest. The finite-horizon and infinite-horizon forward-search procedures for BAPOMDPs are given in Algorithm 5.2 and Algorithm 5.3, respectively. Note that both algorithms assume a model prior p_θ for which $|\Theta| < \infty$ and are therefore only suitable when used in conjunction with the sampling-based EM algorithm of Section 4.3.

The finite-horizon and infinite-horizon versions differ only in how the forward and backward messages are used to compute future expectations given our current position in the search tree, so that without loss of generality we limit the following description to the infinite-horizon forward-search procedure of Algorithm 5.3. In words, the procedure begins at the initial joint model-state belief vector $b = b_0$ and some FSC node n for which $\nu_n > 0$. All combinations of actions a that can be selected by node n and all observations o that can subsequently be observed at the next time-step are considered in turn. The updated model-state belief b' is computed, from which the value, $V_{\max}$, of the future expected reward given the current FSC is derived. The following questions are then asked: Is it possible to improve upon the current FSC at belief b' if we define a new node n' to be the immediate successor to n given observation o? And if so, what is the optimal action a^* to take at node n' and what are the optimal successor nodes to n'—belonging to the *initial* FSC Λ—given subsequent observations o'? Algorithm 5.3 provides answers to these questions, and a new node n' is added to the current controller only when doing so guarantees improvement to the objective (Line 18). Note that when successor nodes are evaluated for the candidate n', only those nodes included in the initial FSC Λ are considered, that is, nodes added to the controller during calls of a lesser depth are ignored (Line 12). This preserves the tractability of forward-search, since then (i) the backward messages need not be updated during the execution of forward-search, and (ii) the calculation of quantities $V(a')$ can be done without accounting for cycling back to candidate nodes n'. Furthermore, to increase the tractability of exploring greater depths, we can limit the nodes added at each depth to those K which produce the largest gains in controller performance.

Algorithm 5.2 Forward-search for finite-horizon BAPOMDPs

Input: FSC Λ, node indices N, node n, belief b, messages $\bar{\mu}_t = (\bar{\mu}_{t,\theta})_{\theta\in\Theta}$, depth $d \geq 1$
Output: Improved FSC Λ

for all $(a, o) \in A \times O$ such that $\pi_{an} > 0$ **do**
  $b'_\theta \leftarrow b^o_{\theta,a}$ for each $\theta \in \Theta$
  $b' \leftarrow (b'_\theta)_{\theta\in\Theta}$
  $V_{\max} \leftarrow \sum_{n'\in N} \sum_{t=0}^{T-d} \sum_{\theta,x} \lambda_{n'no} b'_\theta(x) \bar{\mu}_{t,\theta}(x, n') \gamma^t$
  $V^* \leftarrow -\infty$
  for all $a' \in A$ **do**
    $V(a') \leftarrow \sum_{\theta,x} u(a', x) b'_\theta(x)$
    for all $o' \in O$ **do**
      $b''_\theta \leftarrow b'^{o'}_{\theta,a'}$ for each $\theta \in \Theta$
      $b'' \leftarrow (b''_\theta)_{\theta\in\Theta}$
      $V(a') \leftarrow V(a') + \tau(o'|b', a') \max_{n'\in N} \sum_{\theta,x} b''_\theta(x) \sum_{t=0}^{T-d-1} \bar{\mu}_{t,\theta}(x, n') \gamma^{t+1}$
      $\text{successor}(a', o') \leftarrow \text{argmax}_{n'\in N} \sum_{\theta,x} b''_\theta(x) \sum_{t=0}^{T-d-1} \bar{\mu}_{t,\theta}(x, n') \gamma^{t+1}$
    end for
    if $V(a') > V^*$ **then**
      $V^* \leftarrow V(a')$
    end if
  end for
  if $V^* > V_{\max}$ s.t. $a^* = \text{argmax}_{a'} V(a')$ **then**
    Create a copy $\tilde{n}$ of node n in Λ
    Split $\tilde{n}$ into $\tilde{n}_1$ and $\tilde{n}_2$ s.t. $\pi_{a\tilde{n}_1} = 1$, $\pi_{a\tilde{n}_2} = 0$, and $\pi_{a''\tilde{n}_2} \propto \pi_{a''n}$ for $a'' \neq a$
    if $d = 0$ **then**
      $\nu_{\tilde{n}_1} \leftarrow \nu_n \pi_{an}$ and $\nu_{\tilde{n}_2} \leftarrow \nu_n (1 - \pi_{an})$
      $\nu_n \leftarrow 0$
    end if
    Add a node n' to Λ s.t. $\pi_{a^*n'} = 1$ and $\lambda_{n'n''o'} = 1$ where $n'' = \text{successor}(a^*, o')$
    $\lambda_{\tilde{n}_1 n' o} \leftarrow 1$ and $\lambda_{\tilde{n}_1 n'' o} \leftarrow 0$ for all $n'' \neq n'$
    $\Lambda \leftarrow$ `forward_search`$(\Lambda, N, n', b', \bar{\mu}_t, d + 1)$
  end if
end for

5.3 An empirical study

In this study we limit our consideration to parameterized EM, since forward-search for BAPOMDPs was already evaluated in Section 4.7. Here we apply parameterized EM to two POMDPs: the well-known *Machine* problem (Cassandra, 2009)—an infinite horizon POMDP with 256 states, 16 observations, and 4 actions—and the *Shuffle* problem from

Algorithm 5.3 Forward-search for infinite-horizon BAPOMDPs

Input: FSC Λ, node indices N, node n, belief b, messages $\mathcal{B} = (\mathcal{B}_\theta)_{\theta \in \Theta}$
Output: Improved FSC Λ

for all $(a, o) \in A \times O$ such that $\pi_{an} > 0$ **do**
  $b'_\theta \leftarrow b^o_{\theta,a}$ for each $\theta \in \Theta$
  $b' \leftarrow (b'_\theta)_{\theta \in \Theta}$
  $V_{\max} \leftarrow \sum_{n' \in N} \sum_{\theta,x} \lambda_{n'no} b'_\theta(x) \mathcal{B}_\theta(x, n')$
  $V^* \leftarrow -\infty$
  for all $a' \in A$ **do**
    $V(a') \leftarrow \sum_{\theta,x} u(a', x) b'_\theta(x)$
    for all $o' \in O$ **do**
      $b''_\theta \leftarrow b'^{o'}_{\theta,a'}$ for each $\theta \in \Theta$
      $b'' \leftarrow (b''_\theta)_{\theta \in \Theta}$
      $V(a') \leftarrow V(a') + \tau(o'|b', a') \max_{n' \in N} \sum_{\theta,x} b''_\theta(x) \mathcal{B}_\theta(x, n') \gamma$
      $\text{successor}(a', o') \leftarrow \text{argmax}_{n' \in N} \sum_{\theta,x} b''_\theta(x) \mathcal{B}_\theta(x, n') \gamma$
    end for
    if $V(a') > V_{\max}$ **then**
      $V^* \leftarrow V(a')$
    end if
  end for
  if $V^* > V_{\max}$ s.t. $a^* = \text{argmax}_{a'} V(a')$ **then**
    Create a copy $\tilde{n}$ of node n in Λ
    Split $\tilde{n}$ into $\tilde{n}_1$ and $\tilde{n}_2$ s.t. $\pi_{a\tilde{n}_1} = 1$, $\pi_{a\tilde{n}_2} = 0$, and $\pi_{a''\tilde{n}_2} \propto \pi_{a''n}$ for $a'' \neq a$
    if $d = 0$ **then**
      $\nu_{\tilde{n}_1} \leftarrow \nu_n \pi_{an}$ and $\nu_{\tilde{n}_2} \leftarrow \nu_n (1 - \pi_{an})$
      $\nu_n \leftarrow 0$
    end if
    Add a node n' to Λ s.t. $\pi_{a^* n'} = 1$ and $\lambda_{n'n''o'} = 1$ where $n'' = \text{successor}(a^*, o')$
    $\lambda_{\tilde{n}_1 n' o} \leftarrow 1$ and $\lambda_{\tilde{n}_1 n'' o} \leftarrow 0$ for all $n'' \neq n'$
    $\Lambda \leftarrow \texttt{forward\_search}(\Lambda, N, n', b', \mathcal{B})$
  end if
end for

Section 4.7.1, parameterized by $n = 5$, $r = 1$, $c = 10$, $\gamma = 0.99$, $p_{ii} = 0.75$ for all $i = 1, 2, \ldots, n-1$, and $p_0(1) = 1$. The discrete observation-emission distributions for the *Shuffle* instance are defined as follows: let $|O| = |X|$ and for each state $x' \in \{1, 2, \ldots, n\}$, we set $p(o'|x') \propto n - |o' - x'|$ for each $o' \in O$. In this way, states and features that are closer in index are more highly correlated. Moreover, the "shuffle" action (s) results in a deterministic transition to state $x = 1$.

We compared parameterized EM performance to that of ordinary EM when applied to the *Machine* and *Shuffle* POMDPs. Note that EM for POMDPs is a special case of EM for BAPOMDPs (Section 4.2), where all prior mass p_θ is placed on the known HMM θ. We considered finite-state controllers of sizes $|N| = 5, 10, \ldots, 30$ for *Shuffle* and $|N| = 2, 5, 10, 15, 20$ for *Machine*. For each version of EM and for each $|N|$, we ran EM to convergence over 10 independent trials. Here, convergence was defined such that the marginal improvement in FSC performance was less than $\varepsilon = 1 \times 10^{-5}$ units. To reduce the variance in our comparisons, the same random FSC $\Lambda^{(0),i}$ was used to initialize the i^{th} trial of both parameterized and ordinary EM. Furthermore, at each iteration k of parameterized EM, we set $D^{(k)} = \{1,\ 0.5 \cdot \bar{\Delta}^{(k)}\}$. For reference, we used the GAPMIN algorithm (Poupart et al., 2011a)—which is closely related to PBVI and iteratively minimizes the gap between upper and lower value function bounds for infinite-horizon POMDPs—to compute upper bounds to the *Shuffle* and *Machine* problem objectives. These bounds were found to be 22.17 and 64.24, respectively.

We compare terminal FSC performance and convergence time for both the ordinary and parameterized versions of EM applied to problem *Shuffle* in Figures 5.1 and 5.2, respectively. Each data point in the graphs is centered at the mean of ten independent trials, with vertical bars indicating the minimum and maximum values achieved for the statistic over all runs. From Figure 5.1, we see that parameterized EM performance dominates that of ordinary EM for all $|N|$. Because both EM variations are started from the same initial FSC in all trials, we conclude that parameterized EM succeeds in avoiding local optima in the direction of more favorable FSCs. Furthermore, from Figure 5.2 it is clear that parameterized EM converges more quickly than ordinary EM by orders of magnitude. To illustrate the contrast in convergence rates, in Figure 5.3 we plot FSC performance versus time for both EM algorithms over a single trial with $|N| = 10$. Note also that parameterized EM achieves near-optimal performance for $|N| \geq 20$, as indicated

by the upper bound of 22.17.

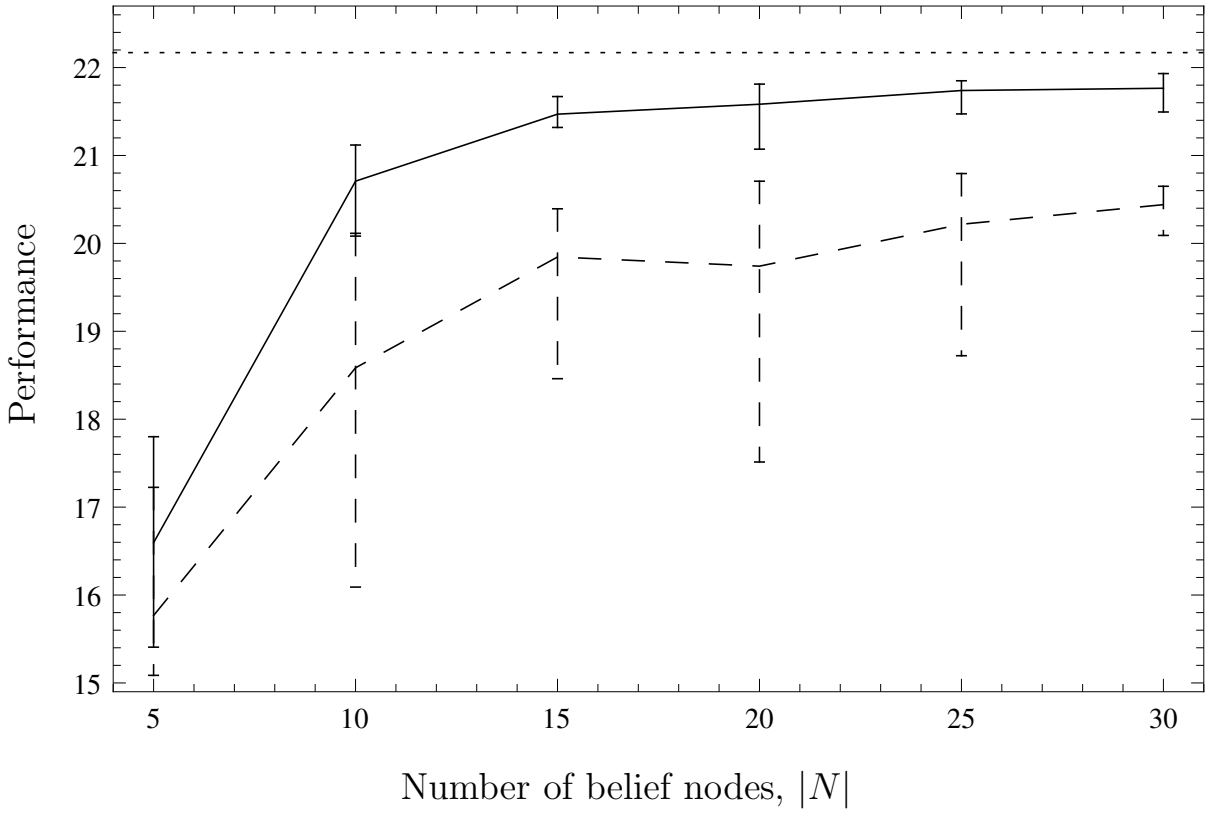

Figure 5.1: A performance comparison of parameterized EM and ordinary EM for an instance of the *Shuffle* POMDP. Solid lines indicate the use of parameterized EM, dashed lines indicate non-parameterized EM, and the dotted line denotes the GAPMIN upper bound.

Now we turn our attention to the *Machine* problem. In the performance comparison of Figure 5.4, we find that average parameterized EM performance dominates that of ordinary EM, although best-achieved performance is comparable for all $|N|$. Generally, we find that ordinary EM algorithm is more susceptible to becoming trapped at poor local optima, as evidenced by its worst-achieved performances in Figure 5.4. From Figure 5.5, we once again find that EM run-time can be reduced considerably by taking a parameterized approach, a point that is made more explicit in the comparison of EM performance trajectories over time in Figure 5.6, where $|N| = 10$.

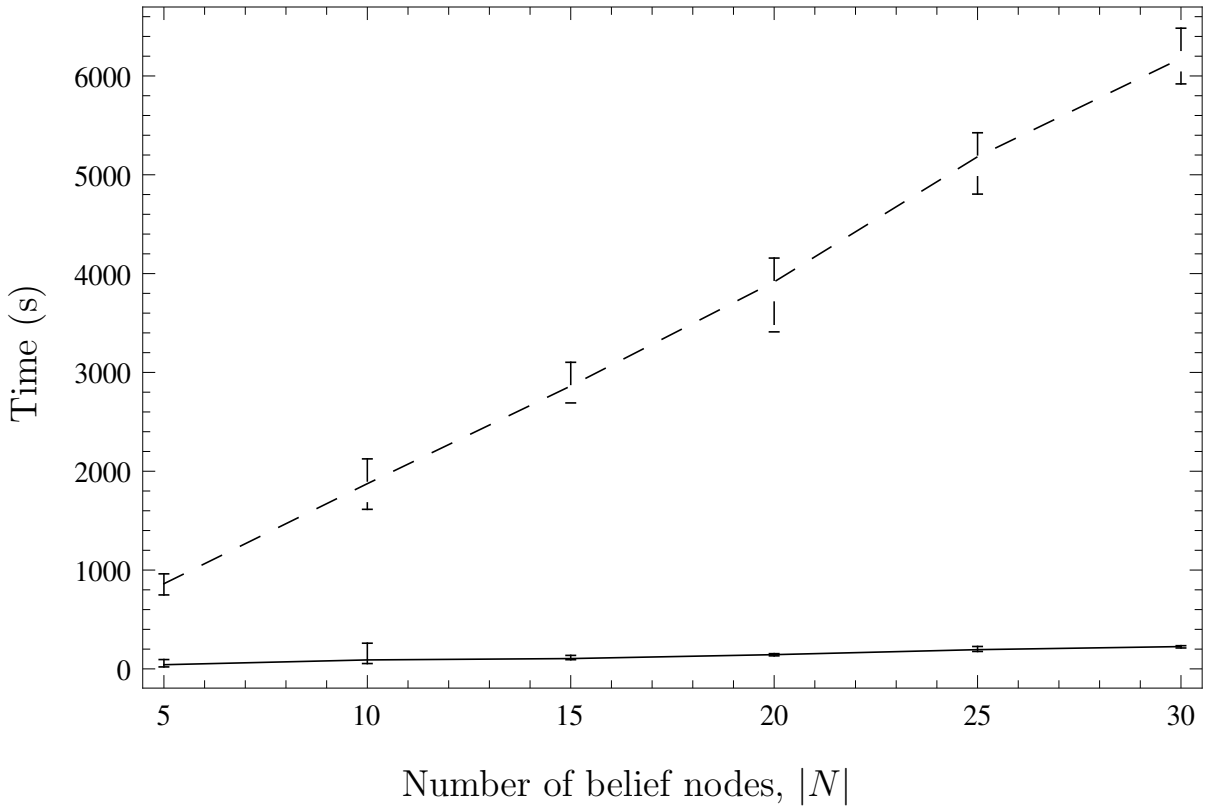

Figure 5.2: A run-time comparison of parameterized EM and ordinary EM for an instance of the *Shuffle* POMDP. Solid lines indicate the use of parameterized EM and dashed lines indicate non-parameterized EM.

5.4 Summary

In this chapter we explored techniques for addressing the slow convergence rate of EM and the local optimality of EM fixed points in the context of (BA)POMDPs. Through an empirical study, we demonstrated that parameterized EM can accelerate convergence by orders of magnitude in both the finite-horizon and infinite-horizon domains. Moreover, by taking larger step-sizes than ordinary EM, parameterized EM can avoid poor local optima by escaping towards more favorable regions of the parameter space.

Recall that in the above experiments we used a fixed step-size multiplier $d = 0.5$ for all iterations of parameterized EM. The convergence rate could be further improved by considering adaptive multipliers $d^{(k)}$ that vary from one iteration to the next. Related to

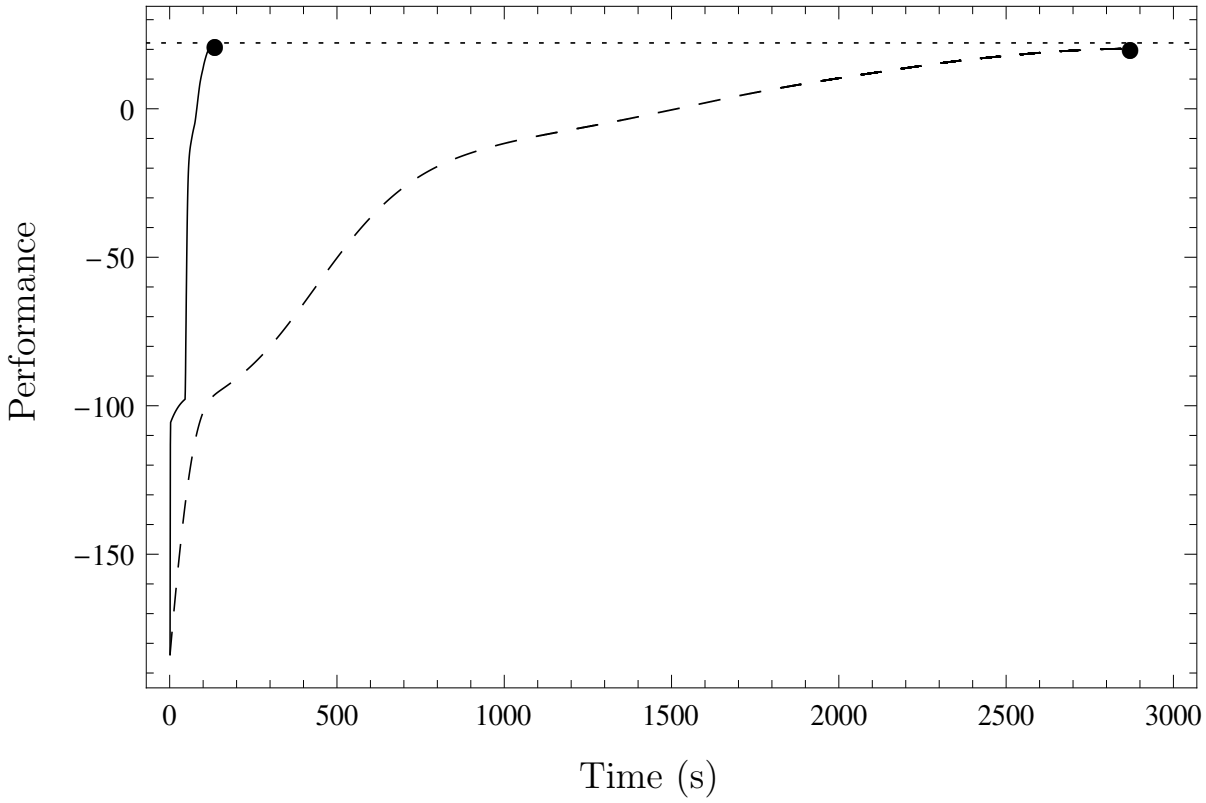

Figure 5.3: A comparison of FSC performance over time for an instance of the *Shuffle* POMDP with $|N| = 10$. Solid lines indicate the use of parameterized EM, dashed lines indicate non-parameterized EM, and the dotted line denotes the GAPMIN upper bound.

this idea, we have found that any multiplier d sufficiently greater than zero will accelerate EM to within a small percentage of the nearest local optimum very quickly. However, attaining these last few percentage points typically requires a disproportionate number of iterations even when parameterized EM is used, likely owing to the number of FSC parameters that are nearly zero at this juncture and hence restrict the parameterized EM step-size. To increase the convergence rate in this setting it might be useful to explicitly "zero out", and thereafter ignore, parameters that are within some threshold of zero so that future parameterized EM updates are less constrained. Of course, care must be taken to ensure that nodes which become unreachable as a result are removed from the FSC so that numerical instabilities (e.g., division by zero) do not occur during subsequent M-steps.

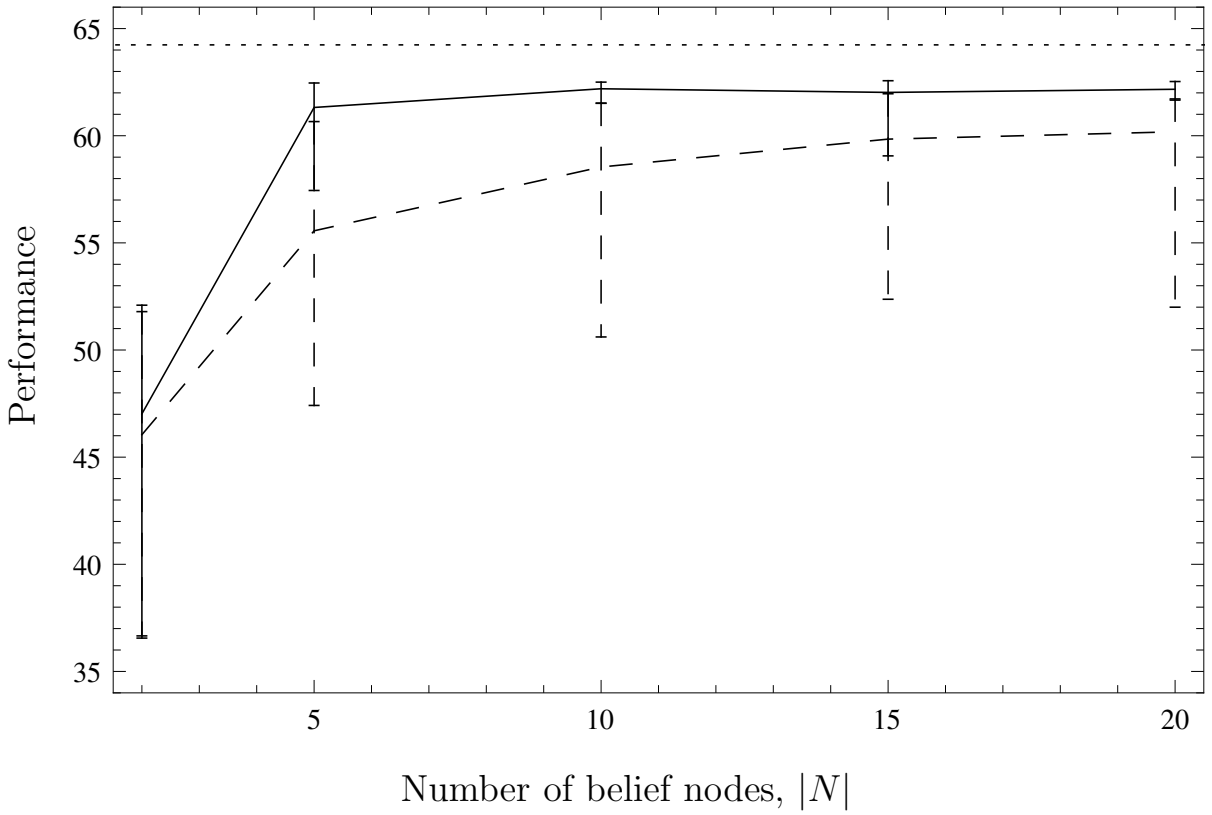

Figure 5.4: A performance comparison of parameterized EM and ordinary EM for the *Machine* POMDP. Solid lines indicate the use of parameterized EM, dashed lines indicate non-parameterized EM, and the dotted line denotes the GAPMIN upper bound.

In their current form, the forward-search procedures of Algorithm 5.2 and Algorithm 5.3 are quite useful (as indicated by the performance comparison of Table 4.1) but leave much room for improvement. Given an FSC Λ, the procedures add optimal action and successor nodes at increasing depths, which requires computing future expected rewards when positioned at newly added nodes. These computations can be done efficiently when cycles are prohibited among the nodes added during forward-search. To facilitate this, we require successor nodes to be members of the original FSC Λ, which guarantees that nodes added during the procedure will not be revisited. As such, it is sufficient to compute future expectations via a single-step look-ahead using only the backward messages $\bar{\mu}$ (finite-horizon) or $\mathcal{B}$ (infinite-horizon). If instead cycles were permitted among new nodes, then the conditional expectations at these nodes would have to be stored

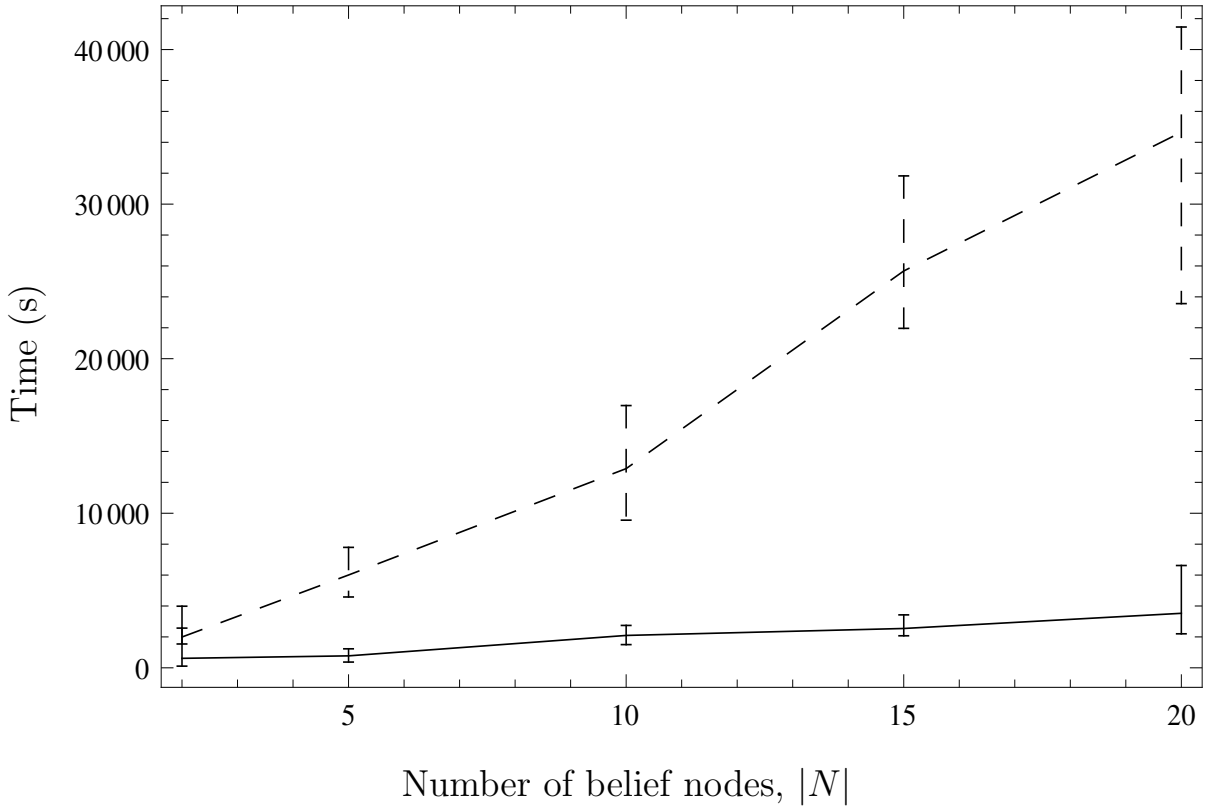

Figure 5.5: A run-time comparison of parameterized EM and ordinary EM for the *Machine* POMDP. Solid lines indicate the use of parameterized EM and dashed lines indicate non-parameterized EM.

and updated during the procedure, adding considerable overhead to the algorithm due to recursive dependencies introduced by the cyclic structure. We revisit this idea in the closing remarks of Chapter 6 and further discuss the issues that would have to be resolved if cycles were allowed.

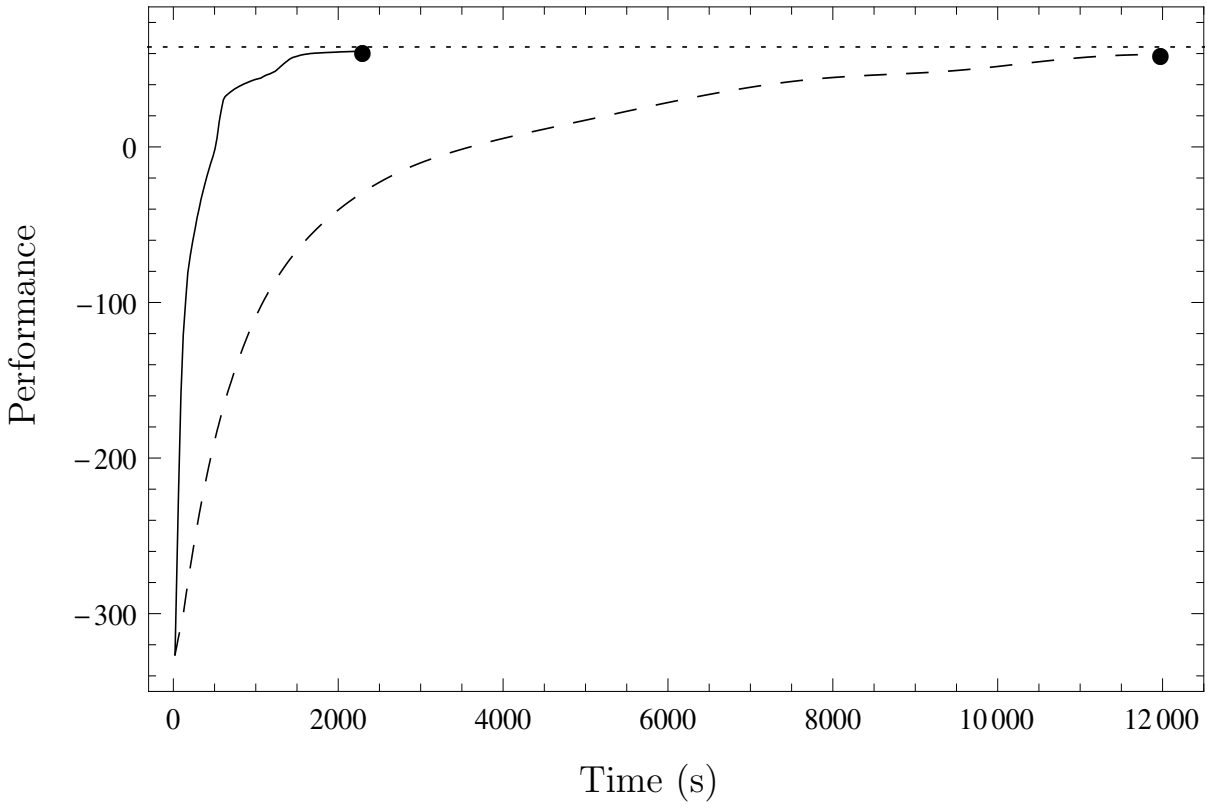

Figure 5.6: A comparison of FSC performance over time for the *Machine* POMDP with $|N| = 10$. Solid lines indicate the use of parameterized EM, dashed lines indicate non-parameterized EM, and the dotted line denotes the GAPMIN upper bound.

Chapter 6

Closing remarks

Broadly speaking, the aim of this dissertation was to both justify the superiority of model-adaptive policies under conditions of model uncertainty and, once established, develop algorithms for solving Bayes-adaptive POMDPs, which naturally arises when model uncertainty is characterized by a prior distribution over the underlying state-transition and observation-emission probabilities. Our focus was on *offline* policies in particular, since only by planning for all possible histories *a priori* can Bayes optimal control truly be achieved.

We first introduced a manufacturing case study to demonstrate the inconsistency and unpredictability of policies derived from the mean and mode point estimates of the model prior, which motivated the search for robust model-adaptive policies. To further justify this search, we analyzed a tractable two-state BAPOMDP and showed that even the *best* point-based policy can significantly underperform the Bayes optimal policy. Next, we derived a general framework for the EM-based optimization of FSCs in the BAPOMDP setting. Due to intractable integrals that arise in the M-step updates, we offered two tractable alternatives for approximate inference. The first is a sampling-based approach

that optimizes with respect to a finite subset of models $\tilde{\Theta} \subset \Theta$ that are randomly drawn from the BAPOMDP prior. As such, the M-step integral is replaced by summation, where the $|\tilde{\Theta}|$ summands can be computed in parallel with multiple concurrent threads, one for each model $\theta \in \tilde{\Theta}$. The second approach—a form a variational Bayes EM—replaces the EM lower bound with a variational approximation, such that the M-step integrals are replaced by an update corresponding to a single "model" q_z, which is composed of sub-stochastic state-transition and observation-emission parameters. Unfortunately, this sub-stochasticity can lead to an undesirable variance-based discounting of certain policies in the VB-EM algorithm. We introduced the novel *constrained* VB-EM algorithm, which mitigates the variance-based discounting phenomenon by reducing sub-stochasticity in the q_z parameters of the factored variational distribution $q = q_z q_\theta$. By altering the CVB-EM multipliers κ, a balance can be struck between ordinary VB-EM—which arises as $\kappa \to 0$ and maintains a robustness against uncertainty via q_θ—and point-based optimization—which is emulated as $\kappa \to \infty$ and does not suffer from the effects of variance-based discounting.

Through the empirical study of Section 4.7 and the manufacturing case study of Section 4.8, we demonstrated the superiority of model-adaptive FSCs generated by EM to the more ubiquitous point-based policies and, moreover, showed EM to be competitive with an existing offline value iteration algorithm. As the space of uncertain model parameters grows, value iteration may no longer be able to capture the nuances of the BAPOMDP objective when optimizing over a tractable subset of beliefs (Wang et al., 2012). In contrast, the scalability of sampling-based EM is limited only by the number of available computational threads, and the scalability of CVB-EM follows from the variational lower bound, which eliminates the M-step integral over model parameters. Unfortunately, the Monte-Carlo simulation used to evaluate the BAPOMDP objective is computationally infeasible for problems much larger than those considered in this dissertation, so that we are

currently unable to validate EM's scalability in an empirical sense, and we leave this as an open question. In future studies involving larger problem domains, it may be necessary to evaluate policies with respect to a single model. This approach is not ideal, however, because the choice of model is somewhat arbitrary and can have a profound impact on the performance evaluation (see Section 1.1.2). A more justified scalable alternative is to evaluate BAPOMDP policies as in our manufacturing case study—that is, evaluating with respect to a single model that generates training data to inform the BAPOMDP prior—so that there is some connection between the BAPOMDP prior and the generative model against which policies are judged.

Despite the encouraging results of EM, our current implementations can be improved. With respect to the sampling-based EM algorithm, the major limitations remain the slow rate of convergence and the local optimality of solutions. Both of these issues were addressed in Chapter 5, the former with a simple parameterized acceleration scheme and the latter with a forward-search subroutine. While convergence to the vicinity of a local optimum is accelerated considerably by parameterized EM, slow convergence often persists within a small neighborhood of the optimum. This behavior is due in part to FSC parameters that are driven toward extreme values (0 or 1) by early iterations and later constrain the parameterized EM step-size $\Delta^{(k)}$. One way to mitigate this behavior is to round such parameters to their nearest extreme value, effectively eliminating their influence on the choice of $\Delta^{(k)}$. As such, subsequent accelerated iterations will focus on adjusting those parameters in the interior of the simplex, which stand to have a greater marginal effect on FSC quality. As outlined in the literature review (Section 2.1.2), there are more sophisticated means of accelerating EM convergence that have not been formally evaluated within the BAPOMDP setting, for instance certain gradient-based approaches. In our opinion, the additional overhead of a gradient calculation coupled with a line search compromises EM's efficiency, thus rendering most gradient

ascent algorithms impractical. However, scaled expectation-conjugate gradient (Fischer and Kersting, 2003)—which requires only a single likelihood evaluation with each iteration and no gradient calculations in addition to the EM search direction—could prove useful for accelerating EM in the neighborhood of a local optimum, at least to an extent not achievable by our parameterized approach.

In Section 5.2 we presented forward-search subroutines for both finite-horizon and infinite-horizon BAPOMDPs. Forward-search can either be used as an intermediate procedure to escape a local optimum through the addition of a small number of well-chosen nodes, or as a terminal procedure to improve FSC performance through the addition of a large number of nodes. The forward-search implementations used in our experiments—see Algorithms 5.2 and 5.3—take as input a locally optimal FSC and the initial joint model-state belief b_0. Starting from b_0 and the initial node distribution ν, optimal action and successor nodes are added in an acyclic fashion when the current FSC can be improved. Otherwise, the current branch is terminated by returning control to the initial FSC. Despite the empirical success of forward-search in Section 4.7, open questions remain:

First, recall that future expected rewards are easily computed during forward-search due to the acyclic addition of nodes, which avoids circular dependencies in the expectation equations. On the other hand, this convention is somewhat inflexible and the introduction of cycles could potentially improve FSC performance while allowing for a more parsimonious controller. If cycling was permitted, however, then a node n added during forward-search could be revisited, by which its conditional expectation would have to be updated and stored whenever a component of the FSC reachable from n was altered, for example when a new successor node was appended to the search tree. In particular, the addition of a new node would require (i) the back-propagation of expectations to nodes at lesser depths, and (ii) the resolution of circular dependencies in the expectation

equations if a cycle was introduced. The former could be handled via message passing, and the latter could be accomplished by either solving a system of Bellman-like equations or through approximate iterative methods, but the computational expense of each procedure could be prohibitive if the current FSC is large. Second, in our empirical studies the forward-search parameters—namely, the number of nodes added, the search depth, and the threshold indicating when a new node should be added versus returning control to the initial FSC—were chosen on an *ad hoc* basis. While there is certainly no general rule for parameter selection that will always lead to the best performance, an additional sensitivity analysis of performance to these parameters might suggest a more informed heuristic.

Among the approaches considered in this dissertation, the CVB-EM algorithm has the greatest potential for scalability with respect both the number of uncertain model parameters and their assumed variability, since discretization of the model space is avoided by means of the factored variational distribution. In the empirical study of Section 4.7, CVB-EM performance was far superior to that of the point-based policies for the finite-horizon problem *Stop*. On the other hand, when applied to problem *Shuffle*, best-achieved CVB-EM performance was comparable to that of the mean model's point-based policy, likely due to the magnified effects of variance-based discounting in the infinite-horizon. However, it must be noted that the best-achieved CVB-EM performance was still far superior to standard VB-EM and, moreover, for both problem instances the best-achieved CVB-EM performance was within a few percentage points of the best-achieved performance among the sampling-based EM and PBVI algorithms. Also, recall that for simplicity we used the same weight κ for all Dirichlets in p_θ when running CVB-EM, even though optimal CVB-EM performance will surely be achieved when the κ weights are allowed to vary independently for each Dirichlet. Intuitively, these weights should be chosen so that increases to the CVB-EM lower bound are more tightly coupled to increases in the true

objective $\log J(\Lambda)$. At the very least, further analysis of the CVB-EM lower bound should suggest more sophisticated heuristics for choosing the κ weights (see Section 4.9 for a more detailed discussion).

To summarize, this dissertation concerns the justification of BAPOMDP policies and algorithms for computing such policies offline. Existing offline approaches are inspired by approximate value iteration for POMDPs and include the sampling-based approach of Wang et al. (2012) and the factored state approach of Ross et al. (2008, 2011) when the BAPOMDP prior is given by a product of independent Dirichlet distributions. Arguably, approximate value iteration scales more gracefully than EM with respect to the action, observation, and state space sizes—$|A|$, $|O|$, and $|X|$, respectively—owing to the success of approximating the convex optimal value function with a finite number of well-chosen belief points, and the comparatively large number of iterations required for EM to converge. However, the scalability of sampling-based PBVI does not extend to the number of uncertain model parameters and their assumed variability, due to the cross-product state space introduced during the BAPOMDP-to-POMDP conversion. Likewise, when the model prior is given by a product of independent Dirichlet distributions, the number of reachable factored states—represented by pseudo-count vectors—remains exponential in the time-horizon, so that optimal offline planning must be replaced by online heuristics for tractability. Offline sampling-based EM and CVB-EM, on the other hand, have the potential to scale more gracefully with respect to model uncertainty, the former owing to the distributed M-step and the latter owing to the factored variational distribution, which eliminates the M-step integral over the model prior.

Appendix

Example A: In Section 4.5 we describe how the VB-EM algorithm can discriminate against policies that activate high-variance model parameters. We now demonstrate this phenomenon by applying VB-EM to a Bayes-adaptive MDP with an analytically tractable M-step. We show that under certain conditions each update can produce a controller that is strictly worse than the previous iteration's. This result serves to motivate the constrained VB-EM approach of Section 4.6.

Consider an instance of the *Shuffle* BAPOMDP defined in Section 4.7.1 with $n = 2$ and a discount factor of $\gamma < 1$, such that a reward of $r > 0$ is received if action g is taken in state $x = 1$, a cost of $c_2 < 0$ is incurred if action g is taken in state $x = 2$, and the "shuffle" action (s) results in a deterministic transition to state $x = 1$ at a cost of $c_1 = 0$. Furthermore, we assume the states are *fully observable* so that the problem is more accurately classified as a BAMDP. Due to the full observability of states, the only uncertainty in our model is the probability p of self-transition from state 1, which we assume follows a Beta(α_1, α_2) prior. Despite this model uncertainty, the Bayes optimal policy is clear: When $x_t = 1$ take action g, and when $x_t = 2$ taken action s. This policy can be captured by the FSC π such that $\pi_{s1} = 0$ and $\pi_{s2} = 1$. Note that here the fully observable states, x, play the role of the belief nodes, n, from the more general BAPOMDP setting.

We initialize the VB-EM algorithm with an FSC π satisfying $\pi_{s2} = 1$ and $0 < \pi_{s1} < 1$. As such, the state 2 action is optimal and will not be altered by VB-EM so that the success of VB-EM is determined solely by the updates to π_{s1}. Given this initial π, we execute the E-step to optimize the factored distribution $q = q_z q_\theta$ via coordinate ascent as described in Section 4.4.1. Suppose that at the E-step's termination $q_\theta(p) \sim \text{Beta}(p|\beta_1, \beta_2)$ so that the relevant parameters in the corresponding distribution q_z are

$$\bar{p}_1 = e^{\psi(\beta_1)-\psi(\beta_1+\beta_2)} \quad \text{and} \quad \bar{p}_2 = e^{\psi(\beta_2)-\psi(\beta_1+\beta_2)},$$

which are the sub-stochastic state-transition quantities—that is, $\bar{p}_1+\bar{p}_2 < 1$—from state 1 to states 1 and 2, respectively. Here overbars are used to emphasize that these quantities are not true probabilities. The optimal M-step update to π_{s1} is given by

$$\pi^*_{s1} \propto \sum_{t=0}^{\infty} \sum_{\tau=0}^{t} q_z(t, a_\tau = s, x_\tau = 1).$$

Because this update is independent of q_θ, the required marginals can be computed using formulas from the EM-based FSC optimization of infinite-horizon BAPOMDPs presented in Section 4.3. While the details are beyond the scope of this example, it can be shown that

$$\pi^*_{s1} = \pi_{s1} \cdot \frac{r\gamma\pi_{g1} + c_2(1 - \gamma\pi_{g1}(1 - \bar{p}_1))}{r\pi_{g1} + c_2(1 - \gamma\pi_{g1}\bar{p}_2)}. \tag{6.1}$$

Since $\pi_{s1} = 0$ is optimal, the M-step will make an improving move with respect to the target objective $J(\pi)$ if and only if $\pi^*_{s1} < \pi_{s1}$. However, it is easy to choose problem parameters such that $\pi^*_{s1} > \pi_{s1}$, corresponding to a decrease in $J(\pi)$. For example, $r = 1$, $c_2 = 2$, $\pi_{g1} = \pi_{s1} = 1/2$, $\beta_1 = 10$, $\beta_2 = 2$, and $\gamma = 0.99$ yield $\pi^*_{s1} \approx 0.5054 > \pi_{s1} = 0.5$. In general, we find that when $\bar{p}_1 + \bar{p}_2$ is sufficiently small, the variance-based discounting of policy $\pi_{s1} = 0$ causes the M-step to favor the trivial, suboptimal stopping policy $\pi_{s1} = 1$, which is not affected by such a discount.

Now, let us consider the behavior of the update in the limit as $\bar{p}_1 + \bar{p}_2 \to 1$. By substitution, it is straightforward to verify that

$$\begin{aligned}
\lim_{\bar{p}_1+\bar{p}_2\to 1} \pi^*_{s1} &= \pi_{s1} \cdot \lim_{\bar{p}_1+\bar{p}_2\to 1} \frac{r\gamma\pi_{g1} + c_2(1 - \gamma\pi_{g1}(1-\bar{p}_1))}{r\pi_{g1} + c_2(1 - \gamma\pi_{g1}\bar{p}_2)} \\
&= \pi_{s1} \cdot \frac{r\gamma\pi_{g1} + c_2(1 - \gamma\pi_{g1}\bar{p}_2)}{r\pi_{g1} + c_2(1 - \gamma\pi_{g1}\bar{p}_2)} \\
&< \pi_{s1}.
\end{aligned}$$

This implies that improving updates to π_{s1} are guaranteed as the sub-stochasticity of the state-transition quantities $\bar{p}_1$ and $\bar{p}_2$ decreases.

Next, we explored this result numerically by running VB-EM to convergence on six problem instances. In each case, we set $r = 1$, $c_2 = 2$, $\gamma = 0.99$, and initialized the FSC with $\pi_{s1} = 0.5$. Of primary interest was investigating M-step behavior under varying conditions of sub-stochasticity in the parameters $\bar{p}_1$ and $\bar{p}_2$. Noting from (4.26) that $(\beta_1, \beta_2) \geq (\alpha_1, \alpha_2)$ for any update q_θ, the effect of variance-based discounting in the M-step will generally decrease as the prior $\text{Beta}(\alpha_1, \alpha_2)$ becomes more concentrated on a single model. With this in mind, we chose the (α_1, α_2) pseudo-count vectors of $(10, 2)$, $(35, 7)$, $(40, 8)$, $(45, 9)$, $(75, 15)$, and $(200, 40)$ for our experiments.

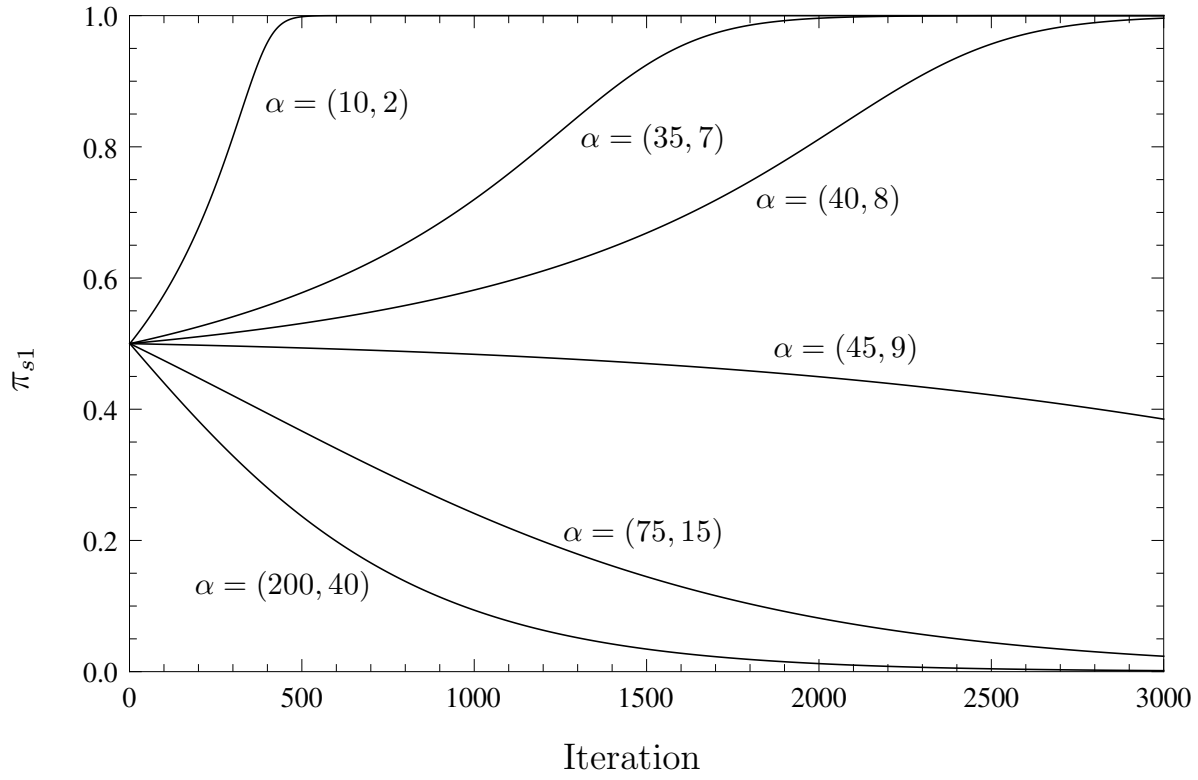

Figure 6.1: An illustration of VB-EM behavior when applied to the BAMDP of the Appendix (Example A). For this problem, $\pi_{s1} = 0$ is optimal, and VB-EM converges to the optimal policy only when the variance of the model prior $p \sim \text{Beta}(\alpha_1, \alpha_2)$ is sufficiently small, here corresponding to the $\text{Beta}(45, 9)$, $\text{Beta}(75, 15)$, and $\text{Beta}(200, 40)$ cases.

In Figure 6.1 we plot the value of π_{s1} recorded over 3000 iterations for each problem instance, and we find that π_{s1} converged to the least desirable policy $\pi_{s1} = 1$ when $p \sim \text{Beta}(10, 2)$, $\text{Beta}(35, 7)$, $\text{Beta}(40, 8)$, and converged to the optimal policy $\pi_{s1} = 0$ when $p \sim \text{Beta}(45, 9)$, $\text{Beta}(75, 15)$, $\text{Beta}(200, 40)$. This is intuitive given our discussion above, since the variance-based discounting of policy $\pi_{s1} = 0$ is smallest in the $p \sim \text{Beta}(45, 9)$, $\text{Beta}(75, 15)$, $\text{Beta}(200, 40)$ instances. However, in most real-world applications the model prior p_θ will have considerably more variance than these three Beta distributions, and hence the undesirable effects of variance-based discounting may persist. In Section 4.6, we introduce *constrained* VB-EM as one approach to addressing this issue.

□

Theorem A: *The system of equations given by* $\beta_i = v_i + \kappa e^{\psi(\beta_i) - \psi(\Sigma\beta_j)}$ *for all* $1 \leq i \leq n$ *has a unique solution, which can be obtained via fixed-point iteration.*

Let

$$B \;=\; \{\beta | \Sigma\beta_j \geq \kappa + 1\} \;\cap\; \{\beta | \beta \geq v\} \;\cap\; \{\beta | \beta \leq \kappa \mathbf{1} + v\}.$$

and let $G : \mathbb{R}^n \mapsto \mathbb{R}^n$ be a vector-valued function satisfying $G(\beta) = (g_1(\beta), g_2(\beta), \ldots, g_n(\beta))$ with $g_i(\beta) = v_i + \kappa e^{\psi(\beta_i) - \psi(\Sigma\beta_j)}$.

First we prove that B must contain all solutions to $G(\beta) = \beta$, and that B is closed under the mapping G. In Lemma E (see Appendix), we show that any β satisfying $G(\beta) = \beta$ also solves the system of equations $\beta_i = v_i + \kappa\beta_i / \Sigma\beta_j - d_i$ for $i = 1, 2, \ldots, n$, where each d_i is a scalar satisfying $0 < d_i < 1/2$. Summing up the left-hand and right-hand sides, we find that $\Sigma\beta_j = \Sigma v_j + \kappa - \Sigma d_j \geq \kappa + n/2$. Since $n \geq 2$, we obtain the bound $\Sigma\beta_j \geq \kappa + 1$. Given this bound, it should be clear that any solution β to $G(\beta) = \beta$ must be a member of the set B defined above. Now suppose that $\beta \in B$ and consider the vector $\beta' = G(\beta)$. We want to show $\beta' \in B$ so that $G : B \mapsto B$. Clearly we must have $\beta' \in \{\beta | \beta \geq v\} \cap \{\beta | \beta \leq \kappa \mathbf{1} + v\}$, so it remains to show $\Sigma\beta'_j \geq \kappa + 1$. Using the bounds $x - 1/2 \leq e^{\psi(x)} \leq x + e^{-\xi} - 1$ for $x \geq 1$ (Batir, 2005), where ξ is the Euler-Mascheroni constant and $-1 + e^{-\xi} \approx -0.44$, we have

$$\begin{aligned} \Sigma\beta'_j &= \Sigma v_j + \kappa e^{-\psi(\Sigma\beta_j)} \sum_j e^{\psi(\beta_j)} \\ &\geq n + \kappa \left(\frac{-n/2 + \Sigma\beta_j}{e^{-\xi} - 1 + \Sigma\beta_j} \right) \end{aligned}$$

$$
\begin{aligned}
&\geq n + \kappa\left(\frac{-n/2 + \Sigma\beta_j}{\Sigma\beta_j}\right) \\
&= n + \kappa - \frac{\kappa}{\Sigma\beta_j}\frac{n}{2} \\
&\geq \kappa + n/2,
\end{aligned}
$$

where the final line is obtained using the lower bound $\Sigma\beta_j \geq \kappa+1$. Therefore $\Sigma\beta_j' \geq \kappa+1$ for all $n \geq 2$, and hence $\beta' \in B$. This establishes that B is closed under the mapping G. Furthermore, note that $G(\beta) = \beta$ must have at least one solution $\beta \in B$ by Brouwer's fixed-point theorem (Istratescu, 2002), due to the compactness and convexity of B and the continuity of G.

Now we will show that the spectral radius of the Jacobian $G'(\beta)$ satisfies $\rho(G'(\beta)) < 1$ for all $\beta \in B$. Computing the entries of $G'(\beta)$, we find that

$$
\frac{\partial g_i}{\partial \beta_i} = \kappa e^{\psi(\beta_i)-\psi(\Sigma\beta_j)}[\psi'(\beta_i) - \psi'(\Sigma\beta_j)] \qquad \text{and} \qquad \frac{\partial g_i}{\partial \beta_j} = -\kappa e^{\psi(\beta_i)-\psi(\Sigma\beta_j)}\psi'(\Sigma\beta_j),
$$

so that—setting $w = \kappa e^{-\psi(\Sigma\beta_j)}$ for notational convenience—

$$
G'(\beta) = w \begin{pmatrix}
e^{\psi(\beta_1)}[\psi'(\beta_1) - \psi'(\Sigma\beta_j)] & -e^{\psi(\beta_1)}\psi'(\Sigma\beta_j) & \dots & -e^{\psi(\beta_1)}\psi'(\Sigma\beta_j) \\
-e^{\psi(\beta_2)}\psi'(\Sigma\beta_j) & e^{\psi(\beta_2)}[\psi'(\beta_2) - \psi'(\Sigma\beta_j)] & \dots & -e^{\psi(\beta_2)}\psi'(\Sigma\beta_j) \\
\vdots & \vdots & \ddots & \vdots \\
-e^{\psi(\beta_n)}\psi'(\Sigma\beta_j) & -e^{\psi(\beta_n)}\psi'(\Sigma\beta_j) & \dots & e^{\psi(\beta_n)}[\psi'(\beta_n) - \psi'(\Sigma\beta_j)]
\end{pmatrix}.
$$

Furthermore, we can express G' in the form $G' = D - H$, where

$$D = w \begin{pmatrix} e^{\psi(\beta_1)}\psi'(\beta_1) & 0 & \dots & 0 \\ 0 & e^{\psi(\beta_2)}\psi'(\beta_2) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & e^{\psi(\beta_n)}\psi'(\beta_n) \end{pmatrix}$$

$$H = w\psi'(\Sigma\beta_j) \begin{pmatrix} e^{\psi(\beta_1)} & e^{\psi(\beta_1)} & \dots & e^{\psi(\beta_1)} \\ e^{\psi(\beta_2)} & e^{\psi(\beta_2)} & \dots & e^{\psi(\beta_2)} \\ \vdots & \vdots & \ddots & \vdots \\ e^{\psi(\beta_n)} & e^{\psi(\beta_n)} & \dots & e^{\psi(\beta_n)} \end{pmatrix}.$$

As such, $G'(\beta)$ is given by a positive diagonal matrix D minus a rank-one adjustment H.

We now seek to bound the eigenvalues of $G'(\beta)$. We begin by proving that $G'(\beta)$ is an M-matrix, which guarantees that all real eigenvalues of $G'(\beta)$ are positive (Plemmons, 1977). To see this, first note that the entries of H are strictly positive so that the off-diagonals of $G'(\beta)$ are negative and therefore $G'(\beta)$ is a Z-matrix. It has been shown that if A is a Z-matrix, then A is an M-matrix if and only if A has a *convergent regular splitting*, i.e., A can be expressed in the form $M - N$, where M and N have nonnegative entries and $\rho(M^{-1}N) < 1$ (Plemmons, 1977). We claim that $D - H$ is a convergent regular splitting of $G'(\beta)$. Clearly, the entries of D and H are nonnegative, so it remains to show that $\rho(D^{-1}H) < 1$. We have

$$D^{-1}H = \psi'(\Sigma\beta_j) \begin{pmatrix} \frac{1}{\psi'(\beta_1)} & \frac{1}{\psi'(\beta_1)} & \cdots & \frac{1}{\psi'(\beta_1)} \\ \frac{1}{\psi'(\beta_2)} & \frac{1}{\psi'(\beta_2)} & \cdots & \frac{1}{\psi'(\beta_2)} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{\psi'(\beta_n)} & \frac{1}{\psi'(\beta_n)} & \cdots & \frac{1}{\psi'(\beta_n)} \end{pmatrix}$$

which has eigenvalues of zero (with multiplicity $n-1$) and $\psi'(\Sigma\beta_j)\sum_j \frac{1}{\psi'(\beta_j)}$, which we have proven to be strictly less than one in the Appendix (Lemma D). Therefore $\rho(D^{-1}H) < 1$ and hence $G'(\beta)$ is a M-matrix. By an equivalent characterization of M-matrices (Plemmons, 1977), it follows that the real eigenvalues of $G'(\beta)$ are positive.

We now show that all eigenvalues of $G'(\beta)$ are real and strictly less than one. Because $G'(\beta)$ is the difference of a diagonal matrix D and a rank-one adjustment $H = he^T$, where h is any column of H and e is a column vector of ones, the following holds (Anderson, 1996): If λ is not an eigenvalue of D, then λ is an eigenvalue of $G'(\beta)$ if and only if λ is a solution to the "secular equation"

$$1 = \sum_j \frac{h_j}{D_j - \lambda}, \tag{6.2}$$

where D_j is the j^{th} diagonal element of D and all solutions λ are real. Furthermore, any remaining eigenvalues of $G'(\beta)$ will necessarily be diagonals of D (possibly of multiplicity greater than one).

From this result, it immediately follows that all eigenvalues of $G'(\beta)$ are real. Recall that the diagonals D_i of D are of the form

$$D_i = \left(\kappa e^{-\psi(\Sigma\beta_j)}\right)\left(\psi'(\beta_i)e^{\psi(\beta_i)}\right).$$

It is known that $\psi'(x)e^{\psi(x)} < 1$ for all $x > 0$ (Batir, 2005), so that $\psi'(\beta_i)e^{\psi(\beta_i)} < 1$. Since $\beta \in B$, we have $\Sigma\beta_j \geq \kappa + 1$. By applying this bound and the Digamma bound $e^{-\psi(x)} < 1/(x - 1 + e^{-\xi})$ (Batir, 2005), we readily obtain $\kappa e^{-\psi(\Sigma\beta_j)} < 1$. As a result, $D_i < 1$ for all $i = 1, 2, \ldots, n$. By examination of the secular equation (6.2), we see that all solutions λ must be strictly less than one, otherwise the right-hand side of the equation would be negative. Furthermore, by the theorem cited above, all remaining eigenvalues of $G'(\beta)$ must be diagonals of D, which were just shown to be less than one. Since we have already established that the eigenvalues of $G'(\beta)$ are positive, it follows that $0 < \lambda < 1$ for all eigenvalues λ of $G'(\beta)$ and hence $\rho(G'(\beta)) < 1$. From a contraction result stated by Schwarz and Waldvogel (1989), it follows that—starting with any initial $\beta^{(0)} \in B$—the fixed-point iteration $\beta^{(k+1)} = G(\beta^{(k)})$ will converge to the unique $\beta^* \in B$ satisfying $G(\beta^*) = \beta^*$. ■

It is worth noting that the contraction result stated by Schwarz and Waldvogel (1989) is not proven; rather, the authors simply state that it follows "by means of sophisticated methods of linear algebra." For completeness, we provide an alternative proof that any fixed-point $\beta^* \in B$ of G must be unique.

The claim of Schwarz and Waldvogel (1989) aside, our work above still proves that starting with any initial $\beta^{(0)} \in B$, the fixed-point iteration $\beta^{(k+1)} = G(\beta^{(k)})$ will converge to a $\beta^* \in B$ satisfying $G(\beta^*) = \beta^*$, provided that $\beta^{(0)}$ is chosen sufficiently close to β^*. This follows because $\rho(G'(\beta^*)) < 1$ and the continuity of G imply that G is a contraction with respect to some matrix norm in a ball around β^*. We will now show that any $\beta \in B$ satisfying $G(\beta) = \beta$ is unique.

To this end, let the vector-valued function F be defined such that $F(\beta) = \beta - G(\beta)$. Then any solution to the system $G(\beta) = \beta$ is a solution to $F(\beta) = 0$. First, let us consider the Jacobian matrix $F'(\beta)$ of F. By definition, $F'(\beta) = I - G'(\beta)$. Recall that the off-diagonals of $G'(\beta)$ are strictly negative, and furthermore our analysis above implies that the diagonals of $G'(\beta)$ are both positive and less than the diagonals D_i of D, which are strictly less than 1. Therefore the entries of the Jacobian $F'(\beta)$ are strictly positive.

Now let $\beta, \tilde{\beta} \in B$ such that $F(\beta) = F(\tilde{\beta}) = 0$. We now show that $\beta = \tilde{\beta}$. We partition our proof over the cases $\Sigma\beta_j = \Sigma\tilde{\beta}_j$ and $\Sigma\beta_j > \Sigma\tilde{\beta}_j$. First, suppose that $\Sigma\beta_j = \Sigma\tilde{\beta}_j$ and define $C = \kappa e^{-\psi(\Sigma\beta_j)} = \kappa e^{-\psi(\Sigma\tilde{\beta}_j)}$. Then for all $i = 1, 2, \ldots, n$ we have

$$\beta_i = v_i + Ce^{\psi(\beta_i)} \qquad \text{and} \qquad \tilde{\beta}_i = v_i + Ce^{\psi(\tilde{\beta}_i)}.$$

Treating C as a fixed constant, the function $u(x) = x$ grows at a constant rate of $u'(x) = 1$, and the function $h(x) = v_i + Ce^{\psi(x)}$ grows at a rate of $h'(x) = C\psi'(x)e^{\psi(x)} < 1$, which follows from the Digamma bound $\psi'(x)e^{\psi(x)} < 1$ for $x > 0$ (Batir, 2005) and the fact that $C < 1$ since $\beta, \tilde{\beta} \in B$. Therefore $u(x)$ intersects $h(x)$ exactly once—that is, at $x = \beta_i$—after which $u(x) > h(x)$ for all larger x. In a general sense, this implies that the sum $\Sigma\beta_j$ of a solution β is sufficient to recover the individual components β_i of the solution. As a direct result of the assumption $\Sigma\beta_j = \Sigma\tilde{\beta}_j$, we must then have $\beta_i = \tilde{\beta}_i$ for all $i = 1, 2, \ldots, n$, by which $\beta = \tilde{\beta}$.

Now suppose, without loss of generality, that $\Sigma\beta_j > \Sigma\tilde{\beta}_j$. Then for all $i = 1, 2, \ldots, n$ we have

$$\beta_i = v_i + C_\beta e^{\psi(\beta_i)} \qquad \text{and} \qquad \tilde{\beta}_i = v_i + C_{\tilde{\beta}} e^{\psi(\tilde{\beta}_i)}$$

where $C_\beta = \kappa e^{-\psi(\Sigma\beta_j)}$ and $C_{\tilde{\beta}} = \kappa e^{-\psi(\Sigma\tilde{\beta}_j)}$. Note that because $\beta, \tilde{\beta} \in B$, we have $C_\beta < 1$ and $C_{\tilde{\beta}} < 1$. It follows (as in the previous case) that β_i and $\tilde{\beta}_i$ are the unique solutions to their respective equations above when C_β and $C_{\tilde{\beta}}$ are treated as fixed constants. Furthermore, by $\Sigma\beta_j < \Sigma\tilde{\beta}_j$ we have $C_{\tilde{\beta}} < C_\beta < 1$, so that necessarily $\beta_i > \tilde{\beta}_i$ for all $i = 1, 2, \ldots, n$ by which $\beta > \tilde{\beta}$. However, we have shown that $F'(\beta) > 0$ element-wise for all $\beta \in B$. Therefore $\beta > \tilde{\beta}$ implies $F(\beta) > F(\tilde{\beta}) = 0$, which contradicts our hypothesis that $F(\beta) = 0$. A similar contradiction arises when we assume that $\Sigma\beta_j > \Sigma\tilde{\beta}_j$, implying that we must have $\Sigma\beta_j = \Sigma\tilde{\beta}_j$, which we already showed to imply $\beta = \tilde{\beta}$. Therefore $\beta, \tilde{\beta} \in B$ such that $F(\beta) = 0$ and $F(\tilde{\beta}) = 0$ implies $\beta = \tilde{\beta}$. ■

Lemma A: *In the* VB-EM *algorithm's* E*-step, the optimal update to* q_z *in the* VB-EM *algorithm's* E*-step, given fixed* Λ *and* q_θ*, is*

$$q_z^*(z) \;\propto\; \exp\langle \log \tilde{p}(z|\theta, \Lambda)\rangle_{q_\theta},$$

and the optimal update to q_θ*, given fixed* Λ *and* q_z*, is*

$$q_\theta^*(\theta) \;\propto\; p_\theta(\theta) \exp\langle \log \tilde{p}(z|\theta, \Lambda)\rangle_{q_z}.$$

Proof: In the E-step of the VB-EM algorithm, the objective is to maximize the lower bound (4.22) with respect to the factored distribution $q = q_z q_\theta$, which we now restate for convenience as

$$L(q_z, q_\theta) = -\langle \log q_z(z)\rangle_{q_z} - \langle \log q_\theta(\theta)\rangle_{q_\theta} + \langle \log p_\theta(\theta)\rangle_{q_\theta} + \langle \log \tilde{p}(z|\Lambda, \theta)\rangle_{q_z q_\theta}.$$

To this end, we use a calculus of variations approach (Gelfand and Fomin, 1963).

First, we derive the optimal update q_z^* given fixed q_θ and Λ. Let $\mathcal{L}(q_z)$ denote those terms in $L(q_z, q_\theta)$ that are dependent on q_z, so that optimizing $L(q_z, q_\theta)$ with respect to q_z is equivalent to optimizing $\mathcal{L}(q_z)$ with respect to q_z. We can write

$$\begin{aligned}\mathcal{L}(q_z) &= \sum_z -q_z(z)\log q_z(z) + q_z(z)\langle \log \tilde{p}(z|\Lambda,\theta)\rangle_{q_\theta} \\ &= \sum_z S(q_z(z)). \end{aligned} \tag{6.3}$$

Furthermore, any feasible q_z must satisfy $\sum_z q_z(z) = 1$. Because the inputs to q_z are discrete, the equation

$$\frac{\partial S(q_z)}{\partial q_z} = \lambda,$$

is a necessary condition for optimality, where the left-hand side is also the *functional derivative* (Gelfand and Fomin, 1963) of $\mathcal{L}(q_z)$ with respect to q_z, and λ is the Lagrange multiplier corresponding to the unity constraint. By definition of q_z (6.3),

$$\frac{\partial S(q_z)}{\partial q_z} = -\log q_z(z) - 1 + \langle \log \tilde{p}(z|\Lambda,\theta)\rangle_{q_\theta}. \tag{6.4}$$

Setting (6.4) equal to λ and solving for q_z^*, we obtain

$$q_z^*(z) \;\propto\; \exp\langle \log \tilde{p}(z|\Lambda,\theta)\rangle_{q_\theta},$$

where λ is absorbed into the proportionality constant. It remains to show that q_z^* is the global optimizer of $\mathcal{L}(q_z)$. Because q_z has discrete inputs, it is sufficient to verify that

$$\left.\frac{\partial^2 S(q_z)}{\partial {q_z}^2}\right|_{q_z=q_z^*} < 0. \tag{6.5}$$

Differentiating (6.4) with respect to q_z, we obtain

$$\frac{\partial^2 S(q_z)}{\partial {q_z}^2} = -\frac{1}{q_z},$$

so that the second-order condition (6.5) is satisfied by noting $q_z^* > 0$. Therefore q_z^* is the global maximizer of $\mathcal{L}(q_z)$.

Now we derive the optimal update q_θ^* given fixed q_z and Λ. Let $\mathcal{L}(q_\theta)$ denote those terms in $L(q_z, q_\theta)$ that are dependent on q_θ, so that optimizing $L(q_z, q_\theta)$ with respect to q_θ is equivalent to optimizing $\mathcal{L}(q_\theta)$ with respect to q_θ. We can write

$$\mathcal{L}(q_\theta) = \int_\theta -q_\theta(\theta)\log q_\theta(\theta) + q_\theta(\theta)\log p_\theta(\theta) + q_\theta(\theta)\langle \log \tilde{p}(z|\Lambda,\theta)\rangle_{q_z}\; d\theta \tag{6.6}$$

$$= \int_\theta S(q_\theta(\theta))\; d\theta. \tag{6.7}$$

Furthermore, any feasible q_θ must satisfy $\int_\theta q_\theta(\theta)\, d\theta = 1$. Because $S(q_\theta)$ does not depend on derivatives of q_θ, the equation

$$\frac{\partial S(q_\theta)}{\partial q_\theta} = \lambda,$$

is a necessary condition for optimality (Gelfand and Fomin, 1963), where the left-hand side is also the functional derivative of $\mathcal{L}(q_\theta)$ with respect to q_θ, and λ is the Lagrange multiplier corresponding to the unity constraint. By definition of q_θ (6.6),

$$\frac{\partial S(q_\theta)}{\partial q_\theta} = -\log q_\theta - 1 + \log p_\theta(\theta) + \langle \log \tilde{p}(z|\Lambda, \theta) \rangle_{q_z}. \tag{6.8}$$

Setting (6.8) equal to λ and solving for q_θ^*, we obtain

$$q_\theta^*(\theta) \;\propto\; p_\theta(\theta) \exp\langle \log \tilde{p}(z|\theta, \Lambda) \rangle_{q_z},$$

where λ is absorbed into the proportionality constant. It remains to show that q_θ^* is the global optimizer of $\mathcal{L}(q_\theta)$. Because $S(q_\theta)$ does not depend on derivatives of q_θ, it is sufficient to verify (Gelfand and Fomin, 1963) that

$$\left. \frac{\partial^2 S(q_\theta)}{\partial {q_\theta}^2} \right|_{q_\theta = q_\theta^*} < 0. \tag{6.9}$$

Differentiating (6.8) with respect to q_θ, we obtain

$$\frac{\partial^2 S(q_\theta)}{\partial {q_\theta}^2} = -\frac{1}{q_\theta},$$

so that the second-order condition (6.9) is satisfied by noting $q_\theta^* > 0$. Therefore q_θ^* is the global maximizer of $\mathcal{L}(q_\theta)$. ■

Lemma B: *The q_z parameters of the variational Bayes* E*-step update are sub-stochastic.*

Proof: Here we show that

$$\sum_{x'} \frac{\exp\psi(\beta_{x,a}^{x'})}{\exp\psi(\sum_{x''}\beta_{x,a}^{x''})} < 1 \quad \text{and} \quad \sum_{o'} \frac{\exp\psi(\beta_{x',a}^{o'})}{\exp\psi(\sum_{o''}\beta_{x',a}^{o''})} < 1,$$

where $\boldsymbol{\beta}$ is defined in (4.30). Without loss of generality, we will show that

$$\sum_i e^{\psi(\beta_i)-\psi(\Sigma\beta_j)} < 1 \tag{6.10}$$

for any pseudo-count vector $\beta = (\beta_1, \beta_2, \ldots, \beta_n) > 0$ such that $n \geq 2$.

First, we show that $e^{\psi(\beta_i)-\psi(\Sigma\beta_j)} < \frac{\beta_i}{\Sigma\beta_j}$ for fixed i. By rearranging terms in this inequality, we will show the equivalent

$$\frac{e^{\psi(\beta_i)}}{\beta_i} < \frac{e^{\psi(\Sigma\beta_j)}}{\Sigma\beta_j} \quad \Longleftrightarrow \quad g(\Sigma\beta_j) - g(\beta_i) > 0, \tag{6.11}$$

where $g(x) = e^{\psi(x)}/x$. To show $g(\Sigma\beta_j) - g(\beta_i) > 0$, it is sufficient to establish that g is strictly increasing. To this end, we show that $(\log g(x))' = \psi'(x) - 1/x > 0$. The inequality $\psi'(x) - 1/x - 1/(2x)^2 > 0$ is proven for all $x > 0$ by Batir (2005), from which $(\log g(x))' > 0$ immediately follows. Therefore we can conclude $e^{\psi(\beta_i)-\psi(\Sigma\beta_j)} < \frac{\beta_i}{\Sigma\beta_j}$ for all i. Using this fact, we have

$$\begin{aligned}\sum_i e^{\psi(\beta_i)-\psi(\Sigma\beta_j)} &< \sum_i \frac{\beta_i}{\Sigma\beta_j} \\ &= 1,\end{aligned}$$

which completes the proof. ■

Lemma C: *In the Lagrangian relaxation* (4.38)*, the first-order optimality conditions* $\partial\tilde{L}/\partial\beta_i = 0$ *imply that either*

$$\beta_i = v_i + \kappa e^{\psi(\beta_i)-\psi(\Sigma\beta_j)} \text{ for all } i,$$

or

$$\psi'(\Sigma\beta_j)\sum_j \frac{1}{\psi'(\beta_j)} = 1.$$

Proof: Suppose that β satisfies $\partial\tilde{L}/\partial\beta_i = 0$ for all i. It is straightforward to show by the definition of $\tilde{L}$ in (4.37) that

$$\frac{\partial\tilde{L}}{\partial\beta_i} = \psi'(\beta_i)\Big(v_i - \beta_i + \kappa e^{\psi(\beta_i)-\psi(\Sigma\beta_j)}\Big) - \psi'(\Sigma\beta_j)\Big(\sum_m v_m - \beta_m + \kappa e^{\psi(\beta_m)-\psi(\Sigma\beta_j)}\Big). \tag{6.12}$$

Using (6.12), for any $l \neq i$ we obtain

$$\begin{aligned}\frac{\partial\tilde{L}}{\partial\beta_i} - \frac{\partial\tilde{L}}{\partial\beta_l} &= \psi'(\beta_i)\Big(v_i - \beta_i + \kappa e^{\psi(\beta_i)-\psi(\Sigma_j\beta_j)}\Big) - \psi'(\beta_l)\Big(v_l - \beta_l + \kappa e^{\psi(\beta_l)-\psi(\Sigma_j\beta_j)}\Big) \\ &= 0,\end{aligned} \tag{6.13}$$

so that

$$v_l - \beta_l + \kappa e^{\psi(\beta_l)-\psi(\Sigma_j\beta_j)} = \frac{\psi'(\beta_i)}{\psi'(\beta_l)}\Big(v_i - \beta_i + \kappa e^{\psi(\beta_i)-\psi(\Sigma_j\beta_j)}\Big), \tag{6.14}$$

noting that $\psi' > 0$. For fixed i, the optimality condition $\partial\tilde{L}/\partial\beta_i = 0$ can be rewritten by substituting the identities (6.14) for all $l \neq i$ into (6.12), leading to the condition

$$\psi'(\beta_i)\Big(v_i - \beta_i + \kappa e^{\psi(\beta_i)-\psi(\Sigma\beta_j)}\Big) - \psi'(\beta_i)\Big(v_i - \beta_i + \kappa e^{\psi(\beta_i)-\psi(\Sigma\beta_j)}\Big)\psi'(\Sigma\beta_j)\sum_l \frac{1}{\psi'(\beta_l)} = 0,$$

or equivalently

$$\Big(v_i - \beta_i + \kappa e^{\psi(\beta_i)-\psi(\Sigma\beta_j)}\Big)\Big(1 - \psi'(\Sigma\beta_j)\sum_l \frac{1}{\psi'(\beta_l)}\Big) = 0. \tag{6.15}$$

Noting that (6.15) must hold for all i completes the proof. ■

Lemma D: $\psi'(\Sigma\beta_j)\sum_j \frac{1}{\psi'(\beta_j)} < 1$ *for all* $\beta = (\beta_1, \beta_2, \ldots, \beta_n)$ *satisfying* $\beta > 0$ *and* $n \geq 2$.

Proof outline: Let $h(\beta) = \psi'(\Sigma\beta_j)\sum_j \frac{1}{\psi'(\beta_j)}$. We assume that $h(\beta) \geq 1$ and seek a contradiction. Given the vector β of size n, we construct a vector $\beta^{(n-1)}$ of size $n-1$ such that $h(\beta^{(n-1)}) > h(\beta) \geq 1$, so that $h(\beta^{(n-1)}) > 1$. Applying this construction recursively, we arrive at a scalar $\beta^{(1)}$ and conclude via induction that $h(\beta^{(1)}) > 1$. The desired contradiction is then established, since by definition $h(\beta^{(1)}) = \psi'(\beta^{(1)})/\psi'(\beta^{(1)}) = 1$. □

Proof: Suppose to the contrary that $h(\beta) \geq 1$. Let $\beta^{(n)} = \beta$. Given the vector $\beta^{(k)} = (\beta_1^{(k)}, \beta_2^{(k)}, \ldots, \beta_k^{(k)})$ for $1 < k \leq n$, we recursively construct $\beta^{(k-1)}$ such that

$$\beta_i^{(k-1)} = \begin{cases} \beta_i^{(k)}, & 1 \leq i \leq k-2 \\ \beta_{k-1}^{(k)} + \beta_k^{(k)}, & i = k-1. \end{cases}$$

In words, $\beta^{(k-1)}$ is constructed by (additively) collapsing the final two components of $\beta^{(k)}$ into a single component.

Let β' and β'' be consecutive vectors generated by the above procedure, such that $|\beta'| = k$ and $|\beta''| = k-1$. Then—where summations over l range from $1 \leq l \leq k-1$ and summations over j range from $1 \leq j \leq k$—we have

$$\begin{aligned}
h(\beta'') &= \psi'(\Sigma\beta''_l) \sum_l \frac{1}{\psi'(\beta''_l)} \\
&= \psi'(\Sigma\beta'_j) \sum_l \frac{1}{\psi'(\beta''_l)} \quad (\text{by } \Sigma\beta''_l = \Sigma\beta'_j) \\
&= \psi'(\Sigma\beta'_j) \left(\frac{1}{\psi'(\beta'_{k-1} + \beta'_k)} - \frac{1}{\psi'(\beta'_{k-1})} - \frac{1}{\psi'(\beta'_k)} + \sum_j \frac{1}{\psi'(\beta'_j)} \right) \quad (\text{by definition of } \beta'') \\
&= \psi'(\Sigma\beta'_j) \left(\frac{1}{\psi'(\beta'_{k-1} + \beta'_k)} - \frac{1}{\psi'(\beta'_{k-1})} - \frac{1}{\psi'(\beta'_k)} \right) + h(\beta') \quad (\text{by definition of } h(\beta')).
\end{aligned}$$

Noting that $\psi' > 0$, if we can show

$$\frac{1}{\psi'(\beta'_{k-1} + \beta'_k)} - \frac{1}{\psi'(\beta'_{k-1})} - \frac{1}{\psi'(\beta'_k)} > 0, \tag{6.16}$$

then we can conclude $h(\beta'') > h(\beta')$. To this end, we now prove

$$f(x, y) = \frac{1}{\psi'(x+y)} - \frac{1}{\psi'(x)} - \frac{1}{\psi'(y)} > 0 \tag{6.17}$$

for all $x, y > 0$. Without loss of generality, let us consider $f(x, y)$ as a function of x, while keeping y fixed. It is straightforward to verify that $\lim_{x\to 0} f(x, y) = 0$. From this point, to show $f(x, y) > 0$ it is sufficient to show that $f(x, y)$ is strictly increasing as a function of x. Taking the partial derivative of $f(x, y)$ with respect to x, we obtain

$$\frac{\partial f(x,y)}{\partial x} = \frac{\psi''(x)}{\psi'(x)^2} - \frac{\psi''(x+y)}{\psi'(x+y)^2}. \tag{6.18}$$

Let $g(x) = \psi''(x)/\psi'(x)^2$, so that $\partial f(x, y)/\partial x = g(x) - g(x + y)$, and hence showing $\partial f(x, y)/\partial x > 0$ for all $x, y > 0$ can be achieved by showing $g'(x) < 0$ for all $x > 0$. We have

$$g'(x) = \frac{-2\psi''(x)^2 + \psi'(x)\psi'''(x)}{\psi'(x)^3}. \tag{6.19}$$

Noting that $\psi' > 0$, the condition $g'(x) < 0$ is equivalent to

$$\psi''(x)^2 - 1/2\psi'(x)\psi'''(x) > 0, \tag{6.20}$$

which was established by English and Rousseau (1997) for all $x > 0$. See also Alzer and Wells (1998). This proves that $\partial f(x, y)/\partial x > 0$ for all $x, y > 0$. Recalling that $\lim_{x\to 0} f(x, y) = 0$, we can conclude that $f(x, y) > 0$ for all $x, y > 0$.

The above result establishes inequality (6.16), so that $h(\beta'') > h(\beta')$. Because $h(\beta) = h(\beta^{(n)}) \geq 1$ by assumption, it follows by induction that $h(\beta^{(k)}) > 1$ for all $1 \leq k \leq n - 1$.

In particular, $h(\beta^{(1)}) > 1$. This is a contradiction, however, since by definition $h(\beta^{(1)}) = \psi'(\beta^{(1)})/\psi'(\beta^{(1)}) = 1$. Therefore we must have $h(\beta) < 1$ for all β satisfying $\beta > 0$ and $|\beta| \geq 2$. ■

Lemma E: *There exist scalars d_i satisfying $0 < d_i < 1/2$ such that any β satisfying $\beta_i = v_i + \kappa e^{\psi(\beta_i) - \psi(\Sigma\beta_j)}$ for all $1 \leq i \leq n$ also satisfies $\beta_i = v_i + \kappa \frac{\beta_i}{\Sigma\beta_j} - d_i$ for all $1 \leq i \leq n$.*

Proof: Suppose that β satisfies $\beta_i = v_i + \kappa e^{\psi(\beta_i) - \psi(\Sigma\beta_j)}$ for all $1 \leq i \leq n$. We have

$$\begin{aligned} \beta_i &= v_i + \kappa e^{\psi(\beta_i) - \psi(\Sigma\beta_j)} && (6.21) \\ &= v_i + \kappa \left(\frac{\beta_i}{\Sigma\beta_j} - \varepsilon_i \right) \\ &= v_i + \kappa \frac{\beta_i}{\Sigma\beta_j} - \kappa\varepsilon_i, && (6.22) \end{aligned}$$

where

$$\varepsilon_i = \frac{\beta_i}{\Sigma\beta_j} - e^{\psi(\beta_i) - \psi(\Sigma\beta_j)}. \quad (6.23)$$

Using the bounds $x - 1/2 \leq e^{\psi(x)} \leq x + e^{-\xi} - 1$ for $x \geq 1$ (Batir, 2005), where ξ is the Euler-Mascheroni constant and $-1 + e^{-\xi} \approx -0.44$, we can write

$$
\begin{aligned}
\kappa\varepsilon_i &= \kappa\left(\frac{\beta_i}{\Sigma\beta_j} - e^{\psi(\beta_i)-\psi(\Sigma\beta_j)}\right) \\
&= \frac{\beta_i}{\Sigma\beta_j}(\beta_i - v_i)e^{-\psi(\beta_i)+\psi(\Sigma\beta_j)} - (\beta_i - v_i) \quad \text{(solving for } \kappa \text{ in (6.21), then substituting)} \\
&\leq \frac{\beta_i(\beta_i - v_i)(\Sigma\beta_j - 1 + e^{-\xi})}{\Sigma\beta_j(\beta_i - 1/2)} - (\beta_i - v_i) \quad \text{(applying the bounds on } e^{\psi}\text{)} \\
&= \frac{\beta_i(\beta_i - v_i)(\Sigma\beta_j - 1 + e^{-\xi}) - (\beta_i - v_i)\Sigma\beta_j(\beta_i - 1/2)}{\Sigma\beta_j(\beta_i - 1/2)} \\
&= \frac{(\beta_i - v_i)\Big(\Sigma\beta_j/2 - \beta_i(1 - e^{-\xi})\Big)}{\Sigma\beta_j(\beta_i - 1/2)} \quad \text{(simplifying the numerator)} \\
&= \frac{1}{2}\frac{(\beta_i - v_i)}{(\beta_i - 1/2)} - \frac{\beta_i(\beta_i - v_i)(1 - e^{-\xi})}{\Sigma\beta_j(\beta_i - 1/2)} \\
&< \frac{1}{2}.
\end{aligned}
$$

Let $d_i = \kappa\varepsilon_i$, and note that by the result above $d_i < 1/2$. In our proof of Lemma B, we showed that $\varepsilon_i > 0$ for all i, from which $d_i > 0$ immediately follows. By substitution of d_i into (6.22), we obtain $\beta_i = v_i + \kappa\frac{\beta_i}{\Sigma\beta_j} - d_i$. ■

Proposition A: *The infinite-horizon* EM *updates of the* FSC *parameters are given by*

$$
\begin{aligned}
\nu_n^* &\propto \sum_x \mu_0(x,n)\mathcal{B}(x,n) \\
\pi_{an}^* &\propto \pi_{an}\sum_x u(a,x)\mathcal{F}(x,n) + \gamma\sum_{x,x',n',o'} \pi_{an}p(x'|x,a)p(o'|x',a)\lambda_{n'no'}\mathcal{F}(x,n)\mathcal{B}(x',n') \\
\lambda_{n'no'}^* &\propto \sum_{x,x',a} \pi_{an}p(x'|x,a)p(o'|x',a)\lambda_{n'no'}\mathcal{F}(x,n)\mathcal{B}(x',n').
\end{aligned}
$$

Proof: Here we extend EM for finite-horizon POMDPs to the infinite-horizon case, noting that the further extension to BAPOMDPs is trivial given the results of Chapter 4. To

simplify our notation, we begin by defining

$$\mathcal{F}(x,n) = \sum_{t=0}^{\infty} \gamma^t \mu_t(x,n), \qquad \mathcal{B}(x,n) = \sum_{t=0}^{\infty} \gamma^t \bar{\mu}_t(x,n)$$

which can be computed by iteratively applying the recursive definitions of μ_t and $\bar{\mu}_t$ until convergence. Alternatively, the desired quantities satisfy $\mathcal{F}(x,n) = \lim_{t\to\infty} \mathcal{F}^t(x,n)$ and $\mathcal{B}(x,n) = \lim_{t\to\infty} \mathcal{B}^t(x,n)$ and can be computed by initializing $\mathcal{F}^0(x,n)$ and $\mathcal{B}^0(x,n)$ arbitrarily and applying the recursive formulas

$$\mathcal{F}^t(x',n') = \nu_{n'} p_0(x') + \gamma \sum_{x,n,a,o'} \mathcal{F}^{t-1}(x,n)\pi_{an}\lambda_{n'no'} p(x'|x,a)p(o'|x',a) \tag{6.24}$$

$$\mathcal{B}^t(x,n) = \sum_{a} \pi_{an} u(a,x) + \gamma \sum_{x',n',a,o'} \mathcal{B}^{t-1}(x',n')\pi_{an}\lambda_{n'no'} p(x'|x,a)p(o'|x',a). \tag{6.25}$$

until convergence.

Now, let us consider the M-step update of ν_n^* as $T \to \infty$. We have

$$\begin{aligned}
\nu_n^* &\propto \sum_{t=0}^{\infty} q(t, n_0 = n) \\
&= \sum_{t=0}^{\infty} \sum_{x} \mu_0(x,n)\bar{\mu}_t(x,n)\gamma^t \\
&= \sum_{x} \mu_0(x,n) \sum_{t=0}^{\infty} \bar{\mu}_t(x,n)\gamma^t \\
&= \sum_{x} \mu_0(x,n)\mathcal{B}(x,n).
\end{aligned}$$

Next, we consider the M-step update of the quantity π^*_{an} as $T \to \infty$. The update is

$$\begin{aligned}\pi^*_{an} &\propto \sum_{t=0}^{\infty} \sum_{\tau=0}^{t} q(t, a_\tau = a, n_\tau = n) \\ &= \sum_{t=0}^{\infty} q(t, a_t = a, n_t = n) + \sum_{t=1}^{\infty} \sum_{\tau=0}^{t-1} q(t, a_\tau = a, n_\tau = n),\end{aligned}$$

where we partition the sum over the cases of $\tau = t$ and $\tau < t$. Note that the above sum is guaranteed to converge provided that the discount γ is less than 1. Working with the term corresponding to $\tau = t$, we obtain

$$\begin{aligned}q(t, a_t = a, n_t = n) &\propto \sum_{t=0}^{\infty} \sum_{x} \mu_t(x, n) \pi_{an} u(a, x) \gamma^t \\ &= \pi_{an} \sum_{x} u(a, x) \sum_{t=0}^{\infty} \mu_t(x, n) \gamma^t \\ &= \pi_{an} \sum_{x} u(a, x) \mathcal{F}(x, n).\end{aligned}$$

For the term corresponding to $\tau < t$, we have

$$\begin{aligned}&\sum_{t=1}^{\infty} \sum_{\tau=0}^{t-1} q(t, a_\tau = a, n_\tau = n) \\ &\qquad \propto \sum_{t=1}^{\infty} \sum_{\tau=0}^{t-1} \sum_{x,x',n',o'} \mu_\tau(x, n) \pi_{an} p(x'|x, a) p(o'|x', a) \lambda_{n'no'} \bar{\mu}_{t-\tau-1}(x', n') \gamma^t \\ &\qquad = \sum_{x,x',n',o'} \pi_{an} p(x'|x, a) p(o'|x', a) \lambda_{n'no'} \sum_{t=1}^{\infty} \sum_{\tau=0}^{t-1} \mu_\tau(x, n) \bar{\mu}_{t-\tau-1}(x', n') \gamma^t\end{aligned}$$

$$
\begin{aligned}
&= \sum_{x,x',n',o'} \pi_{an} p(x'|x,a) p(o'|x',a) \lambda_{n'no'} \sum_{\tau=0}^{\infty} \mu_\tau(x,n) \sum_{t=\tau+1}^{\infty} \bar{\mu}_{t-\tau-1}(x',n') \gamma^t \\
&= \gamma \sum_{x,x',n',o'} \pi_{an} p(x'|x,a) p(o'|x',a) \lambda_{n'no'} \sum_{\tau=0}^{\infty} \mu_\tau(x,n) \gamma^\tau \sum_{t=0}^{\infty} \bar{\mu}_t(x',n') \gamma^t \\
&= \gamma \sum_{x,x',n',o'} \pi_{an} p(x'|x,a) p(o'|x',a) \lambda_{n'no'} \mathcal{F}(x,n) \mathcal{B}(x',n').
\end{aligned}
$$

We then obtain

$$
\pi^*_{an} \propto \pi_{an} \sum_x u(a,x) \mathcal{F}(x,n) + \gamma \sum_{x,x',n',o'} \pi_{an} p(x'|x,a) p(o'|x',a) \lambda_{n'no'} \mathcal{F}(x,n) \mathcal{B}(x',n').
$$

Finally, we consider the M-step update of the quantity $\lambda^*_{n'no'}$ as $T \to \infty$. The update is

$$
\begin{aligned}
\lambda^*_{n'no'} &\propto \sum_{t=1}^{\infty} \sum_{\tau=0}^{t-1} q(t, n_{\tau+1} = n', n_\tau = n, o_{\tau+1} = o') \\
&= \sum_{t=1}^{\infty} \sum_{\tau=0}^{t-1} \sum_{x,x',a} \mu_\tau(x,n) \pi_{an} p(x'|x,a) p(o'|x',a) \lambda_{n'no'} \bar{\mu}_{t-\tau-1}(x',n') \gamma^t \\
&= \sum_{x,x',a} \pi_{an} p(x'|x,a) p(o'|x',a) \lambda_{n'no'} \sum_{t=1}^{\infty} \sum_{\tau=0}^{t-1} \mu_\tau(x,n) \bar{\mu}_{t-\tau-1}(x',n') \gamma^t \\
&= \sum_{x,x',a} \pi_{an} p(x'|x,a) p(o'|x',a) \lambda_{n'no'} \sum_{\tau=0}^{\infty} \mu_\tau(x,n) \sum_{t=\tau+1}^{\infty} \bar{\mu}_{t-\tau-1}(x',n') \gamma^t \\
&= \gamma \sum_{x,x',a} \pi_{an} p(x'|x,a) p(o'|x',a) \lambda_{n'no'} \sum_{\tau=0}^{\infty} \mu_\tau(x,n) \gamma^\tau \sum_{t=0}^{\infty} \bar{\mu}_t(x',n') \gamma^t \\
&= \gamma \sum_{x,x',a} \pi_{an} p(x'|x,a) p(o'|x',a) \lambda_{n'no'} \mathcal{F}(x,n) \mathcal{B}(x',n') \\
&\propto \sum_{x,x',a} \pi_{an} p(x'|x,a) p(o'|x',a) \lambda_{n'no'} \mathcal{F}(x,n) \mathcal{B}(x',n').
\end{aligned}
$$

Summarizing the FSC parameter updates, we have

$$\nu_n^* \propto \sum_x \mu_0(x,n)\mathcal{B}(x,n)$$

$$\pi_{an}^* \propto \pi_{an} \sum_x u(a,x)\mathcal{F}(x,n) + \gamma \sum_{x,x',n',o'} \pi_{an} p(x'|x,a) p(o'|x',a) \lambda_{n'no'} \mathcal{F}(x,n)\mathcal{B}(x',n')$$

$$\lambda_{n'no'}^* \propto \sum_{x,x',a} \pi_{an} p(x'|x,a) p(o'|x',a) \lambda_{n'no'} \mathcal{F}(x,n)\mathcal{B}(x',n').$$

Finally, the infinite-horizon expected reward is given by

$$J(\Lambda) = \sum_{x,n} p_0(x)\nu_n \mathcal{B}(x,n).$$

Note that these results are analogous to those presented by Poupart et al. (2011b) for infinite-horizon POMDPs, subject to various changes of variable. ■

References

H. Alzer and J. Wells. Inequalities for the polygamma functions. *SIAM Journal on Mathematical Analysis*, 29(6):1459–1466, 1998.

J. Anderson. A secular equation for the eigenvalues of a diagonal matrix perturbation. *Linear algebra and its applications*, 246:49–70, 1996.

P. G. Bagos, T.D. Liakopoulos, and S.J. Hamodrakas. Faster gradient descent training of hidden Markov models, using individual learning rate adaptation. In *Grammatical Inference: Algorithms and Applications*, pages 40–52. Springer, 2004.

P. Baldi and Y. Chauvin. Smooth online learning algorithms for hidden Markov models. *Neural Computation*, 6(2):307–318, 1994.

D. Barber and T. Furmston. Solving deterministic policy (PO)MDPs using expectation-maximisation and antifreeze. In *European Conference on Machine Learning (LEMIR workshop)*, pages 50–64, 2009.

N. Batir. Some new inequalities for Gamma and Polygamma functions. *Journal of Inequalities in Pure and Applied Mathematics*, 6(4):1–9, 2005.

L.E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The annals of mathematical statistics*, 41(1):164–171, 1970.

D.P. Bertsekas. *Dynamic programming and optimal control*. Athena Scientific Belmont, MA, 1995.

J.A. Bilmes. A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models. *International Computer Science Institute*, 4(510):126, 1998.

R.I. Brafman. A heuristic variable grid solution method for POMDPs. In *Proceedings of the National Conference on Artificial Intelligence*, pages 727–733. Citeseer, 1997.

O. Cappé, E. Moulines, and T. Rydén. *Inference in hidden Markov models*. Springer, 2005.

A. Cassandra, M.L. Littman, and N.L. Zhang. Incremental pruning: A simple, fast, exact method for partially observable markov decision processes. In *Proceedings of the Thirteenth conference on Uncertainty in artificial intelligence*, pages 54–61. Morgan Kaufmann Publishers Inc., 1997.

A.R. Cassandra. *Exact and approximate algorithms for partially observable Markov decision processes*. PhD thesis, Brown University, Providence, RI, USA, 1998. AAI9830418.

A.R. Cassandra. The POMDP page. `http://www.pomdp.org/pomdp/code/index.shtml`, 2009.

O. Cetin and M. Ostendorf. Multi-rate hidden Markov models and their application to machining tool-wear classification. In *Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP'04). IEEE International Conference on*, volume 5, pages V–837. IEEE, 2004.

H.T. Cheng. *Algorithms for partially observable Markov decision processes*. PhD thesis, University of British Columbia, Vancouver, 1988.

P. Dallaire, C. Besse, S. Ross, and B. Chaib-draa. Bayesian reinforcement learning in continuous POMDPs with Gaussian processes. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2604–2609. IEEE, 2009.

P.J. Davis. Gamma function and related functions. *Handbook of mathematical functions*, pages 253–293, 1972.

R. Dearden, N. Friedman, and D. Andre. Model based Bayesian exploration. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, pages 150–159. Morgan Kaufmann Publishers Inc., 1999.

A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 1–38, 1977.

F. Doshi, J. Pineau, and N. Roy. Reinforcement learning with limited reinforcement: Using Bayes risk for active learning in POMDPs. In *Proceedings of the 25th international conference on Machine learning*, pages 256–263. ACM, 2008.

M.O. Duff. *Optimal Learning: Computational procedures for Bayes-adaptive Markov decision processes*. PhD thesis, University of Massachusetts Amherst, 2002.

M.O. Duff and A.G. Barto. Local bandit approximation for optimal learning problems. In *Advances in Neural Information Processing Systems 9*. Citeseer, 1997.

B.J. English and G. Rousseau. Bounds for certain harmonic sums. *Journal of Mathematical Analysis and Applications*, 206(2):428–441, 1997.

J. Fischer and K. Kersting. Scaled CGEM: A fast accelerated EM. In *Machine Learning: ECML 2003*, pages 133–144. Springer, 2003.

R.K. Fish. *Dynamic models of machining vibrations, designed for classification of tool wear*. PhD thesis, University of Washington, 2001.

R.K. Fish, M. Ostendorf, G.D. Bernard, and D.A. Castanon. Multilevel classification of milling tool wear with confidence estimation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(1):75–85, 2003.

T. Furmston and D. Barber. Variational methods for reinforcement learning. In *AISTATS*, volume 9, pages 241–248, 2010.

I.M. Gelfand and S.V. Fomin. *Calculus of variations*. Revised English edition translated and edited by Richard A. Silverman. Prentice-Hall Inc., Englewood Cliffs, NJ, 1963.

E.A. Hansen. An improved policy iteration algorithm for partially observable MDPs. *Advances in Neural Information Processing Systems*, pages 1015–1021, 1998a.

E.A. Hansen. Solving POMDPs by searching in policy space. In *Proceedings of the fourteenth conference on uncertainty in artificial intelligence*, pages 211–219. Morgan Kaufmann Publishers Inc., 1998b.

V.I. Istratescu. *Fixed point theory: An introduction*, volume 7. Springer, 2002.

M. Jamshidian and R.I. Jennrich. Conjugate gradient acceleration of the EM algorithm. *Journal of the American Statistical Association*, 88(421):221–228, 1993.

N.L. Johnson, S. Kotz, and N. Balakrishnan. *Continuous Multivariate Distributions, Volume 1, Models and Applications*. New York: John Wiley & Sons, 2002.

L.P. Kaelbling, M.L. Littman, and A.R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1):99–134, 1998.

F.R. Kschischang, B.J. Frey, and H.A. Loeliger. Factor graphs and the sum-product algorithm. *Information Theory, IEEE Transactions on*, 47(2):498–519, 2001.

S. Kullback. *Information theory and statistics*. Courier Dover Publications, 1968.

Z. Kunpeng. *Multi-categories tool wear classification in micro-milling*. PhD thesis, National University of Singapore, 2007.

R.J.A. Little and D.B. Rubin. *Statistical analysis with missing data*, volume 539. Wiley New York, 1987.

M.L. Littman. The Witness algorithm: Solving partially observable Markov decision processes. *Brown University, Providence, RI*, 1994.

L.A. Medina and V.H. Moll. The integrals in gradshteyn and ryzhik. part 10: The Digamma function. *Sci. Ser. A Math. Sci. (N.S.)*, 17:46–66, 2009.

N. Meuleau, K.E. Kim, L.P. Kaelbling, and A.R. Cassandra. Solving POMDPs by searching the space of finite policies. In *Proceedings of the fifteenth conference on uncertainty in artificial intelligence*, pages 417–426. Morgan Kaufmann Publishers Inc., 1999.

G.E. Monahan. State of the arta survey of partially observable markov decision processes: Theory, models, and algorithms. *Management Science*, 28(1):1–16, 1982.

R.M. Neal and G.E. Hinton. A view of the EM algorithm that justifies incremental, sparse, and other variants. *NATO ASI Series D: Behavioural and Social Sciences*, 89: 355–370, 1998.

L.E. Ortiz and L.P. Kaelbling. Accelerating EM: An empirical study. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, pages 512–521. Morgan Kaufmann Publishers Inc., 1999.

C.H. Papadimitriou and J.N. Tsitsiklis. The complexity of Markov decision processes. *Mathematics of operations research*, pages 441–450, 1987.

J. Pineau, G. Gordon, and S. Thrun. Point-based value iteration: An anytime algorithm for POMDPs. In *International joint conference on artificial intelligence*, volume 18, pages 1025–1032. Lawrence Erlbaum Associates Ltd., 2003.

J. Pineau, G. Gordon, and S. Thrun. Anytime point-based approximations for large POMDPs. *Journal of Artificial Intelligence Research*, 27(1):335–380, 2006.

R.J. Plemmons. M-matrix characterizations. I-nonsingular M-matrices. *Linear Algebra and its Applications*, 18(2):175–188, 1977.

P. Poupart and C. Boutilier. Bounded finite state controllers. *Advances in Neural Information Processing Systems*, 16, 2003.

P. Poupart, K.E. Kim, and D. Kim. Closing the gap: Improved bounds on optimal POMDP solutions. In *International Conference on Automated Planning and Scheduling (ICAPS)*, 2011a.

P. Poupart, T. Lang, and M. Toussaint. Analyzing and escaping local optima in planning as inference for partially observable domains. In *Machine Learning and Knowledge Discovery in Databases*, pages 613–628. Springer, 2011b.

L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

C. Roland. A note on the parameterized EM method. *Statistics & probability letters*, 80 (17):1354–1357, 2010.

S. Ross, B. Chaib-draa, and J. Pineau. Bayes-adaptive POMDPs. In *Advances in Neural Information Processing Systems 20 (NIPS)*, 2008.

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

T. Rydén. EM versus Markov chain Monte Carlo for estimation of hidden Markov models: A computational perspective. *Bayesian Analysis*, 3(4):659–688, 2008.

R. Salakhutdinov, S. Roweis, and Z. Ghahramani. Optimization with EM and expectation-conjugate-gradient. *ICML*, 20(2):672, 2003.

H.R. Schwarz and J. Waldvogel. *Numerical analysis: A comprehensive introduction*, volume 10. Wiley Chichester, New York, 1989.

T. Smith and R. Simmons. Heuristic search value iteration for POMDPs. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, pages 520–527. AUAI Press, 2004.

E.J. Sondik. *The optimal control of partially observable Markov decision processes*. PhD thesis, Stanford University, Palo Alto, CA, USA, 1971.

E.J. Sondik. The optimal control of partially observable Markov processes over the infinite horizon: Discounted costs. *Operations Research*, pages 282–304, 1978.

M.T.J. Spaan and N. Spaan. A point-based POMDP algorithm for robot planning. In *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, volume 3, pages 2399–2404. IEEE, 2004.

M.T.J. Spaan and N. Vlassis. Perseus: Randomized point-based value iteration for POMDPs. *Journal of artificial intelligence research*, 24(1):195–220, 2005.

M. Toussaint, A. Storkey, and S. Harmeling. Expectation-maximization methods for solving (PO)MDPs and optimal control problems. In S. Chiappa and D. Barber, editors, *Inference and Learning in Dynamic Models*. Cambridge University Press, 2010.

N. Vlassis and M.T.J. Spaan. A fast point-based algorithm for POMDPs. In *Benelearn 2004: Proceedings of the Annual Machine Learning Conference of Belgium and the Netherlands*, pages 170–176, 2004.

M. Wang and J. Wang. CHMM for tool condition monitoring and remaining useful life prediction. *The International Journal of Advanced Manufacturing Technology*, 59(5): 463–471, 2012.

Y. Wang, K.S. Won, D. Hsu, and W.S. Lee. Monte Carlo Bayesian reinforcement learning. In *Proceedings of the 29th International Conference on Machine Learning (ICML)*, 2012.

C.F. Wu. On the convergence properties of the EM algorithm. *The Annals of Statistics*, 11(1):95–103, 1983.

N.L. Zhang and W. Zhang. Speeding up the convergence of value iteration in partially observable Markov decision processes. *Journal of Artificial Intelligence Research*, 14: 29–51, 2001.

R. Zhou and E.A. Hansen. An improved grid-based approximation algorithm for POMDPs. *International Joint Conference on Artificial Intelligence*, 17(1):707–716, 2001.