A Non-Stationary Infinite Partially-Observable Markov Decision Process

Sotirios P. Chatzis[1] and Dimitrios Kosmopoulos[2]

[1]Department of Electrical Eng., Computer Eng., and Informatics
Cyprus University of Technology, Cyprus
[2]Department of Informatics Engineering, TEI Crete, Greece

Abstract. Partially Observable Markov Decision Processes (POMDPs) have been met with great success in planning domains where agents must balance actions that provide knowledge and actions that provide reward. Recently, nonparametric Bayesian methods have been successfully applied to POMDPs to obviate the need of a priori knowledge of the size of the state space, allowing to assume that the number of visited states may grow as the agent explores its environment. These approaches rely on the assumption that the agent's environment remains stationary; however, in real-world scenarios the environment may change over time. In this work, we aim to address this inadequacy by introducing a dynamic nonparametric Bayesian POMDP model that both allows for automatic inference of the (distributional) representations of POMDP states, and for capturing non-stationarity in the modeled environments. Formulation of our method is based on imposition of a suitable dynamic hierarchical Dirichlet process (dHDP) prior over state transitions. We derive efficient algorithms for model inference and action planning and evaluate it on several benchmark tasks.

1 Introduction

Reinforcement learning in partially observable domains is a challenging and attractive research area in machine learning. One of the most common representations used for partially-observable reinforcement learning is the partially observable Markov decision process (POMDP) [13]. POMDPs are statistical models postulating that emission of the observation $\boldsymbol{o}_t$ that an agent receives from the environment at time t follows a distribution $\Omega(\boldsymbol{o}_t|s_t, a_t)$ that depends on the value of some latent (hidden) world-state s_t, and the agent's most recent action a_t. In addition, each action a_t of the agent results in a reward $R(s_t, a_t)$ emitted from the environment, the value of which also depends on the current state s_t, and induces a change in the latent state of the environment, which transitions to a new state s_{t+1}, drawn from a transition distribution $T(s_{t+1}|s_t, a_t)$.

A significant drawback of POMDPs is the large number of parameters entailed from the postulated emission distribution models $\Omega(\boldsymbol{o}|s, a)$, state transition distribution models $T(s'|s, a)$, and reward models $R(s, a)$. These parameters must be learned using data obtained through interaction of the agent with its

environment, in an online fashion. However, the combination of the typically limited availability of training data with the large number of parameters may result in uncertain trained models, where planning becomes extremely computationally cumbersome, and the generated policies rather unreliable. Bayesian reinforcement learning approaches [9,7,10] resolve these issues by accounting for both uncertainty in the agent's model of the environment, and uncertainty within the environment itself. This is effected by maintaining distributions over both the parameters of the POMDP and the latent states of the world s.

Most approaches require *a priori* provision of the number of model states: even if the size of the state-space is actually known (which is seldom the case), training a large number of model parameters from the beginning of the learning process (when no data is actually available) might result in poor model estimates and overfitting. Recently, [3] proposed leveraging the strengths of Bayesian nonparametrics, specifically hierarchical Dirichlet process (HDP) priors [15], to resolve these issues. The so-obtained infinite POMDP (iPOMDP) postulates an infinite number of states, conceived as abstract entities whose sole function is to render the dynamics of the system Markovian, instead of actual physical aspects of the system. Note though that, despite the assumption of infinite model states, only a small number of (actually visited) effective states need to be instantiated with parameters at each iteration of the learning algorithm, rendering the model computationally efficient.

Despite these advances, a significant drawback of existing nonparametric Bayesian formulations of POMDPs consists in their lack of appropriate mechanisms allowing for capturing non-stationarity in the modeled environments, expressed in the form of time-adaptive underlying state transition distributions. Indeed, the problem of capturing time-varying underlying distributions in conventional POMDP model formulations has been considered by various researchers in the recent literature (e.g., [16,5]). In this work, we address this inadequacy by introducing a non-stationary variant of the iPOMDP. Formulation of our model is based on imposition of the dynamic hierarchical Dirichlet process (dHDP) prior [8] over the postulated state transitions in the context of our model. We derive efficient model inference and action planning algorithms.

The remainder of this paper is organized as follows: In Section 2, we introduce our proposed model, and derive its learning and action selection algorithms. In Section 3, we provide the experimental evaluation of our approach, and compare it to state-of-the-art alternatives. Finally, in the last section we summarize our results and conclude this paper.

2 Proposed Approach

2.1 Motivation

The iPOMDP model is based on utilization of an HDP prior to describe the state transition dynamics in the modeled environments. The HDP is a model that allows for linking a set of group-specific Dirichlet processes, learning the model

components jointly across multiple groups. Specifically, let us assume C latent model states, and A possible actions; let us consider that each possible state-action pair (s, a) defines a different scenario in the environment. The iPOMDP model, being an HDP-based model, postulates that the new state of the environment (after an action is taken) is drawn from a distribution with different parameters $\boldsymbol{\theta}^{sa}$, which are in turn drawn from a scenario-specific Dirichlet process. In addition, the base distribution of the scenario-specific Dirichlet processes is taken as a common underlying Dirichlet process. Under this construction, the following generative model is obtained

$$s'|s, a \sim T(\boldsymbol{\theta}^{s,a}) \tag{1}$$

$$\boldsymbol{\theta}^{s,a} \sim G_{s,a} \tag{2}$$

$$G_{s,a} \sim \mathrm{DP}(\alpha, G_0) \tag{3}$$

$$G_0 \sim \mathrm{DP}(\gamma, H) \tag{4}$$

As we observe, in the context of the HDP, different state transitions that refer to the same state-action pair (scenario) share the same parameters (atoms) that comprise $G_{s,a}$. In addition, transitions might also share parameters (atoms) across different state-action pairs, probably with different mixing probabilities for each $G_{s,a}$; this is a consequence of the fact that the Dirichlet processes $G_{s,a}$ pertaining to all the modeled state-action pairs share a common base measure G_0, which is also a discrete distribution.

Although the HDP introduces a dependency structure over the modeled scenarios, it does not account for the fact that, when it comes to modeling of sequential data, especially data the distribution of which changes over time, sharing of underlying atoms from the Dirichlet processes is more probable in proximal time points. Recently, [8] developed a dynamic variant of the HDP that allows for such a modeling capacity, namely the dynamic HDP (dHDP). Therefore, utilization of this prior emerges as a promising solution for us to effect our goals.

2.2 Model formulation

To introduce our model, we have to provide our prior assumptions regarding the state transition distributions, observation emission distributions, and reward emission distributions of our model. Let us begin with the state transition distributions of our model. As we have already discussed, to capture non-stationarity, we model state transitions using the dHDP prior [8].

Let us introduce the notation $\boldsymbol{\pi}_{\tau}^{s,a} = (\pi_{\tau l}^{s,a})_{l=1}^{\infty}$. $\pi_{\tau l}^{s,a}$ denotes the (prior) probability of transitioning at some time point t to state l from state s by taking action a, given that the distributions of the various state transitions at that time point are the same as they were at time $\tau = \phi_t$. In other words, the employed dHDP assumes that the dynamics of state transition may change over time, with different time points sharing common transition dynamics patterns. Specifically, following [8], we have

$$s_{t+1} = k|s_t = s, a_t = a \sim \mathrm{Mult}(\boldsymbol{\pi}_{\phi_t}^{s,a}) \tag{5}$$

$$\boldsymbol{\pi}_{\tau}^{s,a} \sim \mathrm{DP}(\alpha, G_0) \tag{6}$$

and

$$G_0 \sim \mathrm{DP}(\gamma, H) \tag{7}$$

whence

$$\pi_{tl}^{s,a} = \tilde{\pi}_{tl}^{s,a} \prod_{h=1}^{l-1} (1 - \tilde{\pi}_{th}^{s,a}) \tag{8}$$

$$\tilde{\pi}_{tl}^{s,a} \sim \mathrm{Beta}(\alpha_t \beta_l, \alpha_t (1 - \sum_{m=1}^{l} \beta_m)) \tag{9}$$

$$\beta_k = \varpi_k \prod_{q=1}^{k-1} (1 - \varpi_q) \tag{10}$$

and

$$\varpi_k \sim \mathrm{Beta}(1, \gamma) \tag{11}$$

In the above equations, the latent variables ϕ_t are indicators of state-transition distribution sharing over time. Following [8], their prior distributions take the form

$$\phi_t | \tilde{\boldsymbol{w}} \sim \mathrm{Mult}(\boldsymbol{w}_t) \tag{12}$$

with $\boldsymbol{w}_t = (w_{tl})_{l=1}^{t}$, and

$$w_{tl} = \tilde{w}_{l-1} \prod_{m=l}^{t-1} (1 - \tilde{w}_m),\ l = 1, \ldots, t \tag{13}$$

while $\tilde{w}_0 = 1$, and

$$\tilde{w}_t | a_t, b_t \sim \mathrm{Beta}(\tilde{w}_t | a_t, b_t),\ t \geq 1 \tag{14}$$

As observed from (13), this construction induces a proximity-inclined transition dynamics sharing scheme; that is, $w_{t1} < w_{t2} < \cdots < w_{tt}$. In other words, it favors sharing the same dynamics between proximal time points, thus enforcing our assumptions of transition dynamics evolving over time in a coherent fashion.

Finally, our observation emission distributions are taken in the form $\Omega(\boldsymbol{o}|s,a) \sim H$, and our reward emission distributions yield $R(r|s,a) \sim H_R$. The distributions H and H_R can have any form, with the choice depending on the application at hand. In this paper, we shall be considering discrete reward and action distributions; as such, a suitable conjugate selection for the priors over their parameters is the Dirichlet distribution.

This concludes the formulation of our model. We dub our model the infinite dynamic POMDP (iDPOMDP) model. Our model is a completely non-stationary POMDP model, formulated under the assumption of an infinite space of latent POMDP states, and treated under the Bayesian inference paradigm. Note also that limiting the generative non-stationarity assumptions to the transition functions of our model does not limit non-stationarity *per se* to state transitions. Indeed, the non-identifiability of the postulated model latent states results in the assumed generative non-stationarity of the transition functions being implicitly extended to the observation and reward functions of our model.

2.3 Inference algorithm

To efficiently perform inference for our model, we combine alternative application of a variant of the block Gibbs sampler of [4], and importance sampling [14], in a fashion similar to the iPOMDP model [3]. Our block Gibbs sampler allows for drawing samples from the true posterior. However, we limit ourselves to using our block Gibbs sampler only on a periodical basis, and not at each time point. In the meanwhile, we use instead an importance sampling algorithm, which merely reweighs the already drawn samples so as to reflect the current posterior as closely as possible. This way, we obtain a significant speedup of our inference algorithm, without compromising model accuracy, since the actual model posterior is not expected to undergo large changes over short time windows.

Block Gibbs sampler To make inference tractable, we use a truncated expression of the stick-breaking representation of the underlying shared Dirichlet process of our model, G_0 [12]. In other words, we set a truncation threshold C, and consider $\boldsymbol{\pi}_t^{s,a} = (\pi_{tl}^{s,a})_{l=1}^{C}$, $\forall t, s, a$ [4]. A large value of C allows for obtaining a good approximation of the infinite underlying process, since in practice the $\pi_{tl}^{s,a}$ are expected to diminish quickly with increasing l, $\forall t$ [4]. Note also that, as discussed in [8], drawing one sample from the dHDP model by means of the block Gibbs sampler takes similar time as drawing one sample from HDP.

Let us consider a time horizon T steps long. We have

$$p(\tilde{w}_t|\dots) = \text{Beta}(\tilde{w}_t|a + \sum_{j=t+1}^{T} n_{j,t+1}, b + \sum_{j=t+1}^{T}\sum_{h=1}^{t} n_{jh}) \tag{15}$$

where n_{th} is the number of time points such that $\phi_t = h$. Similar,

$$\begin{aligned} p(\tilde{\pi}_{tl}^{s,a}|\dots) = \text{Beta}\Big(& \tilde{\pi}_{tl}^{s,a}|\alpha_t\beta_l + \sum_{j=1}^{T}\mathbb{I}(n_{jt}\neq 0)\mathbb{I}(\nu_{jl}^{s,a}\neq 0), \\ & \alpha_t(1-\sum_{m=1}^{l}\beta_m) + \sum_{k=l+1}^{C}\sum_{j=1}^{T}\mathbb{I}(n_{jt}\neq 0)\mathbb{I}(\nu_{jk}^{s,a}\neq 0)\Big) \end{aligned} \tag{16}$$

where $\nu_{tk}^{s,a}$ is the number of training episodes where we had a transition from state s to state k, by taking action a at time t.

The updates of the set of indicator variables ϕ_t can be obtained by generating samples from multinomial distributions with entries of the form

$$\begin{aligned} p(\phi_t = \tau|s_{t-1}=s, a_{t-1}=a;\dots) \propto & \tilde{w}_{\tau-1}\prod_{m=\tau}^{t-1}(1-\tilde{w}_m)\tilde{\pi}_{\tau s_t}^{s,a}\prod_{q=1}^{s_t-1}(1-\tilde{\pi}_{\tau q}^{s,a}) \\ & \times p(\boldsymbol{o}_{t+1}|s_t,a_t)\, p(r_{t+1}|s_t,a_t), \quad \tau = 1,\dots,t \end{aligned} \tag{17}$$

Further, the posterior distribution over the latent model states yields

$$p(s_t = k|s_{t-1} = s, a_{t-1} = a; \dots) \propto \tilde{\pi}^{s,a}_{\phi_t k} \prod_{q=1}^{k-1} (1 - \tilde{\pi}^{s,a}_{\phi_t q}) p(\boldsymbol{o}_{t+1}|s_t, a_t)\; p(r_{t+1}|s_t, a_t) \tag{18}$$

As we observe, this expression entails Markovian dynamics. Thus, to sample from it, we have to resort to some method suitable for distributions with temporal interdependencies. In our work, we employ the forward filtering-backward sampling (FFBS) algorithm [1]; this way, we can efficiently obtain samples of the underlying latent state sequences.

Finally, the observation and reward distribution parameters of our model are sampled in a manner similar to the original iPOMDP model [3].

Importance sampling At time points when we substitute block Gibbs sampling from the true posterior with importance sampling, we essentially reweigh the samples previously drawn from the true posterior. Initially, all samples have equal weight as they are drawn from the true posterior; this changes when we apply importance sampling, so as to capture small changes in the actual posterior in a computationally efficient manner (possible within short time-windows).

Let us denote as μ a sample of our model with weight $w_t(\mu)$ at time t (all samples have initial weights equal to one). Similar to the iPOMDP model, the weight update at time $t+1$ yields [3]

$$w_{t+1}(\mu) \propto w_t(\mu) \sum_{\forall s_t} \Omega(\boldsymbol{o}_{t+1}|s_t, a_t) b_\mu(s_t) \tag{19}$$

where $b_\mu(s)$ is the belief (posterior probability) for state s, as determined in the sample μ of the model.

2.4 Action selection

Once we have obtained a set of samples from the posterior distribution of our model, we can use them to perform action selection. For this purpose, in this work we apply *stochastic forward search in the model-space*, as proposed in [3]. The main concept of forward search is to use a forward-looking tree to compute action-values [11]. Starting from the current posterior (belief) over the model parameters of the agent, the tree branches on each action the agent might take and each observation the agent might see. At each action node, the agent computes the (posterior) expectation of the immediate reward, given the drawn samples, in a standard Monte Carlo-type fashion.

3 Experimental Evaluation

We evaluate our method in several benchmark scenarios and compare its performance to related alternatives, namely Medusa [5] and iPOMDP. Medusa is

provided with the true number of states, while iPOMDP determines it automatically, similar to our approach. The first benchmark scenario considered here, namely Tiger-3, is adopted from [3]; it comprises an environment that changes over time, thus allowing for us to evaluate the capacity of our model to adapt to new situations. The rest of our considered benchmarks are well-known problems in the POMDP literature, namely, Tiger [6], Shuttle [2], Network [6], and Gridworld [6].

In our experiments, tests had 200 episodes of learning, which interleaved acting and resampling models, and 100 episodes of testing with the models fixed. Our results are provided in Table 1. As we observe, our approach is capable of inferring a smaller number of states than the true count, only retaining states for which adequate information can be derived from the accrued experiences (training episodes); this is attained without any compromises in the yielded accumulated rewards in all scenarios. Given the fact that, as discussed in Section 2.3, drawing one sample from the dHDP by means of the block Gibbs sampler takes similar time as drawing one sample from the HDP, we deduce that our approach allows for obtaining improved total reward compared to the iPOMDP for decreased model complexity and resulting computational costs. Note also that the obtained performance improvement is more prominent in the case of the Tiger-3 problem, where the environment changes over time, thus posing greater learning challenges to the postulated agents. This finding vouches for the capacity of our model to capture non-stationarities in the modeled environments, which is the ultimate goal of this work.

Table 1. Experimental Evaluation: Number of inferred states and total obtained reward.

	#States			Total Reward		
Problem	Actual	iPOMDP	iDPOMDP	Medusa	iPOMDP	iDPOMDP
Tiger-3	4	4.1	3.8	-40.26	-42.07	-35.19
Tiger	2	2.1	2.1	0.83	4.06	4.64
Shuttle	8	2.1	2.1	10	10	10
Network	7	4.36	4.07	6671	6508	6749
Gridworld	26	7.36	6.82	-49	-13	-12

4 Conclusions

In this paper, we proposed a nonparametric Bayesian formulation of POMDPs that addressed the problem of capturing non-stationarities in the modeled environments. Formulation of our model was based on the imposition of a suitable dynamic prior over the state transitions of our model, namely the dHDP prior. We devised efficient learning and planning algorithms for our model, based on a combination of block Gibbs sampling and importance sampling. We showed that

our method outperforms related alternatives, namely Medusa and iPOMDP, in several benchmark tasks, combining increased reward performance with shorter model sizes, and, hence, better computational complexity.

Acknowledgments

This work was implemented under the Operational Program "Education and Lifelong Learning" action Archimedes III, co-financed by the European Union (European Social Fund) and Greek national funds (National Strategic Reference Framework 2007 - 2013).

References

1. Carter, C.K., Kohn, R.: On Gibbs sampling for state space models. Biometrika 81, 541–553 (1994)
2. Chrisman, L.: Reinforcement learning with perceptual aliasing: The perceptual distinctions approach. In: Proc. AAAI. pp. 183–188 (1992)
3. Doshi-Velez, F.: The infinite partially observable Markov decision process. In: Proc. NIPS (2009)
4. Ishwaran, H., James, L.F.: Gibbs sampling methods for stick-breaking priors. Journal of the American Statistical Association 96, 161–173 (2001)
5. Jaulmes, R., Pineau, J., Precup, D.: Learning in non-stationary Partially Observable Markov Decision Processes. In: ECML Workshop on Reinforcement Learning in Non-Stationary Environments (2005)
6. Littman, M.L., Cassandra, A.R., Kaelbling, L.P.: Learning policies for partially observable environments: scaling up. In: Proc. ICML (1995)
7. Poupart, P., Vlassis, N., Hoey, J., Regan, K.: An analytic solution to discrete Bayesian reinforcement learning. In: Proc. ICML. pp. 697–704 (2006)
8. Ren, L., Carin, L., Dunson, D.B.: The dynamic hierarchical Dirichlet process. In: Proc. International Conference on Machine Learning (ICML) (2008)
9. Ross, S., Chaib-draa, B., Pineau, J.: Bayes-adaptive POMDPs. In: Proc. NIPS (2008)
10. Ross, S., Chaib-draa, B., Pineau, J.: Bayesian reinforcement learning in continuous POMDPs with application to robot navigation. In: Proc. ICRA (2008)
11. Ross, S., Pineau, J., Paquet, S., Chaib-Draa, B.: Online planning algorithms for pomdps. Journal of Artificial Intelligence Research 32, 663–704 (2008)
12. Sethuraman, J.: A constructive definition of the Dirichlet prior. Statistica Sinica 2, 639–650 (1994)
13. Shani, G., Pineau, J., Kaplow, R.: A survey of point-based POMDP solvers. Auton Agent Multi-Agent Syst 27(1), 1–51 (2012)
14. Siegmund, D.: Importance sampling in the Monte Carlo study of sequential tests. The Annals of Statistics 4, 673–684 (1976)
15. Teh, Y.W., Jordan, M.I., Beal, M.J., Blei, D.M.: Hierarchical Dirichlet processes. Journal of the American Statistical Association 101, 1566–1581 (2006)
16. Theocharous, G., Kaelbling, L.P.: Approximate planning in POMDPs with macro-actions. In: Proc. NIPS (2003)