Context MDPs

Christos Dimitrakakis
FIAS, University of Frankfurt, Germany
christos.dimitrakakis@gmail.com

September 9, 2010

Abstract

This paper presents a simple method for exact online inference and approximate decision making, applicable to large or partially observable Markov decision processes. The approach is based on a closed form Bayesian inference procedure for a class of context models which contains variable order Markov decision processes. The models can be used for prediction, and thus for decision theoretic planning. The other novel step of this paper is to use the belief (context distribution) at any given time as a compact representation of system state, in a manner similar to predictive state representations. Since the belief update is linear time in the worst case, this allows for computationally efficient value iteration and reactive learning algorithms such as Q-learning for this class of models.

1 Introduction

We consider estimation of a class of context models that can approximate either large or partially observable Markov decision processes (MDPs). This is closely related to the context tree weighting algorithm (CTW) for discrete sequence prediction (Willems et al., 1995). We present a *constructive* definition of a context process, extending the one proposed in (Dimitrakakis, 2010) to the estimation of variable order MDPs. With a suitable choice of context structure, the construction is applicable to large or continuous MDPs as well. We introduce two simple algorithms, the weighted context value iteration and weighted context Q-learning, for decision making in unknown environments with continuous or partially observable state. Finally, we provide preliminary experimental results on the predictive, state representation and decision making capabilities of the methods.

We consider discrete-time decision problems in unknown environments, with a known set of actions $\mathcal{A}$ chosen by the decision maker, and a set of observations $\mathcal{Z}$ drawn from some unknown process μ, to be made precise later. At each time step $t \in \mathbb{N}$, the decision maker observes $z_t \in \mathcal{Z}$, selects an action $a_t \in \mathcal{A}$ and receives a reward $r_t \in \mathbb{R}$.

The environment μ is a (partially observable) *Markov decision process* ((PO)MDP) with state $s_t \in \mathcal{S}$. The process is defined by the following conditional distributions: the set of transition and reward distributions

$$\mathcal{T}_\mu \triangleq \{\mathbb{P}_\mu(s_{t+1} \mid s_t = i, a_t = j) : i \in \mathcal{S}, j \in \mathcal{A}\}$$

and

$$\mathcal{R}_\mu \triangleq \{\mathbb{P}_\mu(r_{t+1} \mid s_t = i, a_t = j) : i \in \mathcal{S}, j \in \mathcal{A}\}.$$

For POMDPs, observations z_t are sampled from $\mathcal{O}_\mu^{i,j} \triangleq \mathbb{P}_\mu(z_{t+1} \mid s_t = i, a_t = j)$, while for MDPs, $\mathcal{Z} = \mathcal{S}$ and $z_t = i$ iff $s_t = i$.

The decision maker has a policy π for choosing actions, which indexes a set of probability measures on actions. Jointly π and μ index a set of probability measures $\mathbb{P}_{\mu,\pi}(z_{t+1}, s_{t+1}, r_{t+1}, a_t \mid s_t)$ on actions, states, rewards and observations. The agents goals are expressed in terms of the agent's utility function:

$$U_t \triangleq \sum_{k=1}^{T-t} \gamma^k r_{t+k}, \qquad \text{(utility)}$$

where $\gamma \in [0, 1]$ is a discount factor and T is a reward horizon.

The decision maker is usually uncertain about the true MDP μ. We adopt a subjective decision-theoretic viewpoint (DeGroot, 1970) and assume a set $\mathcal{M}$ of MDPs contains μ, then define a *prior* probability measure ξ_0 on $(\mathcal{M}, \mathfrak{B}_\mathcal{M})$, such that for any $M \in \mathfrak{B}_\mathcal{M}$:

$$\xi_{t+1}(M) \triangleq \xi_t(M \mid z_{t+1}, r_{t+1}, a_t, z_t) \tag{1}$$

is defined for all t and sequences of s_t, a_t, r_t. We now must find a policy π maximising the expected utility under our current belief ξ_t.

$$\mathbb{E}_{\xi_t,\pi} U_t = \int_\mathcal{M} \mathbb{E}_{\mu,\pi}(U_t)\xi_t(\mu)\,\mathrm{d}\mu, \tag{2}$$

The decision problem can be seen as an MDP whose state space is the product of $\mathcal{S}$ and the set of probability measures on $(\mathcal{M}, \mathfrak{B}_\mathcal{M})$. However, since there is an infinite number of beliefs, approximations are required even under full observability (Duff, 2002; Wang et al., 2005; Dimitrakakis, 2009). Nevertheless, such methods are also extensible to the partially observable case (Veness et al., 2009; Ross et al., 2008).

When dealing with large or partially observable MDPs, even (1) is not closed-form. Section 2 extends a specific formulation of variable order Markov model estimation (Dimitrakakis, 2010) to variable order *or* large MDPs. Section 3 discusses decision-theoretic planning and derives a value iteration (WCVI) and a Q-learning algorithm (WCQL) for the model, that utilise the context distribution as a representation of system state. Related work is discussed in Sec. 4. Section 5 experimentally shows that the context model not only can provide accurate predictions, but that the internal state of the process closely tracks the state of the system, even though no explicit state estimation is being performed. Experiments with the extremely simple WCQL algorithm demonstrate the effectiveness of the model for online decision making in unknown POMDPs.

2 Inference for context MDPs

Context models enable closed-form inference for either discrete variable order MDPs or for continuous MDPs. This is done by constructing a context tree, such that for each history $x^t = (x_k)_{k=1}^t$, there exists a set of contexts forming a chain on the tree. We can then perform a walk which starts at the deepest context, stops with probability w_k^t on the k-th node of the chain (and always stops at the root node) and generates an observation from the corresponding model.

More precisely, let $\mathcal{X} \triangleq \mathcal{Z} \times \mathcal{A}$ be the action-observation product space, let $\mathcal{X}^* = \bigcup_{k=0}^{\infty} \mathcal{X}^k$ be the set of all possible histories. Finally, let $\mathcal{F}$ be a σ-algebra on $\mathcal{X}^*$ and the context set $\mathscr{F} \triangleq \mathcal{F}^*$ be the set of all sequences of elements in $\mathcal{F}$. Consider a function $\mathcal{C} : \mathcal{X}^* \to \mathscr{F}$, mapping any history x^t into a sequence of contexts $\mathbf{c}^t = \mathcal{C}(x^t)$, such that: (a) $\mathbf{c}^t$ has at most t elments for any t, i.e. if $x \in \mathcal{X}^t$, $\mathcal{C}(x) \in \mathcal{F}^t$. (b) $\mathcal{C}$ induces a tree, i.e. if $x, x' \in \mathcal{X}^*$ are such that $\mathcal{C}_k(x) = \mathcal{C}_k(x')$ then $\mathcal{C}_{k-1}(x) = \mathcal{C}_{k-1}(x')$. Each context $\mathbf{c}_k^t \in \mathcal{F}$ is associated to a sequence of probability measures:

$$\phi_k^t(z_{t+1} \in Z) = \mathbb{P}(z_{t+1} \in Z \mid \mathbf{c}_k^t, x^t), \tag{3}$$

defined on some appropriate σ-algebra on $\mathcal{Z}$. We wish to perform online estimation of the marginal predictive distribution:

$$\mathbb{P}(z_{t+1} \mid x^t) = \sum_k \phi_k^t(z_{t+1}) \, \mathbb{P}(\mathbf{c}_k^t \mid x^t). \tag{4}$$

For a given x^t, let B_k denote the event that the next observation will be generated by one of the contexts in $\{\mathbf{c}_1^t, \dots, \mathbf{c}_k^t\}$. Then it holds that:

$$\mathbb{P}(z_{t+1}|B_k, x^t) = \phi_k^t(z_{t+1}) w_k^t + \mathbb{P}(z_{t+1}|B_{k-1}, x^t)(1 - w_k^t), \tag{5}$$

where the weight

$$w_k^t \triangleq \mathbb{P}(\mathbf{c}_k^t \mid x^t, B_k), \tag{6}$$

is a stopping probability. To perform inference, we only need to update ϕ and the weights. The former depends on the details of the model at each context. The weights can be updated via a simple recursive procedure, shown in Theorem 1. In general x^t is a concatenation of observation-action pairs, i.e. $(z_k, a_k) \circ (z_{k+1}, a_{k+1})$. The main question is what the context structure should be.

Example 1 (MVMDP)**.** *If, for any sequence $x^t \in \mathcal{X}^*$, $\mathbf{c}^t = \mathcal{C}(x^t)$ is such that $\mathbf{c}_{k+1}^t \subset \mathbf{c}_k$, then the random walk starts from the smallest matching context. If in addition, $\mathcal{X}$ is discrete, the contexts correspond to suffixes of $\mathcal{X}^*$ and we use Dirichlet-multinomial models at each context, then we obtain a mixture of variable order Markov decision processes (MVMDP). To see this, consider replacing each weight w with a sampled value $\hat{w}$ such that $\mathbb{P}(\hat{w} = 1) = 1 - \mathbb{P}(\hat{w} = 0) = w$. The resulting model is a VMDP.*

Example 2 (CMDP)**.** *One may alternatively consider fully observable but large spaces. Let us restrict $\mathcal{F}$ to an algebra generated by some subsets of $\mathcal{X}$. Let $X(x^t) \triangleq \{\mathbf{c} \in \mathcal{F} : x_t \in \mathbf{c}\}$, and let $\mathcal{C}(x^t) = (\mathbf{c}_k^t : k = 1, \ldots)$ be such that $\mathbf{c}_k \in X(x^t)$ and $\mathbf{c}_{k+1}^t \subset \mathbf{c}_k^t$ for all k. Now $\mathcal{C}$ defines a chain of contexts for each observation, where each deeper context is a smaller subset of $\mathcal{X}$.*[1] *The resulting model is a context MDP (CMDP). Since in many reinforcement learning problems $\mathcal{A}$ is discrete, the main difficulty is how to partition the state space $\mathcal{S}$. However, once this (admittedly hard) obstacle has been overcome, perhaps with some heuristics, it is straightforward to update conditional probabilities in the same manner as for discrete, partially observable problems. We, however, are* not *tackling this problem explicitly in this paper.*

2.1 Recursive Bayesian inference

We now derive a closed-form recursion for updating the parameters. We use a superscript t to denote the value of parameters at time t. Thus w_k^t and ϕ_k^t will denote the weights and the marginal predictive distribution of the k-th context at time t respectively. Using (5), we can write the marginal predictive distribution (4) at time t, in terms of w_k^t as:

$$\mathbb{P}(z_{t+1} = j | x^t) = \sum_{k=1}^{t} \phi_k^t(z_{t+1} = j) w_k^t \prod_{n=k+1}^{t} (1 - w_n^t).$$

If there are only $n < t$ contexts in $\mathcal{C}(x^t)$, then we set $w_k^t = 0$ for all $k > n$. The following theorem, which is analogous to Th. 1 in (Dimitrakakis, 2010), gives a closed-form procedure for updating w_k^t:

Theorem 1. *The weight parameters w_k^t can be updated according to:*

$$w_k^{t+1} = \mathbb{P}(\mathbf{c}_k^t | x_{1:t+1}, B_k) = \frac{\phi_k^t(z_{t+1}) w_k^t}{\phi_k^t(z_{t+1}) w_k^t + \mathbb{P}(z_{t+1} | x^t, B_{k-1})(1 - w_k^t)}$$

where k indexes the active contexts $\mathcal{C}(x^t)$.

Proof. First of all, note that B_t is trivially true at time t, since there are at most t contexts in $\mathcal{C}(x^t)$. For B_k with $k < t$, it is easy to see that the following recursions hold:

$$\mathbb{P}(B_{k-1} | x^t) = \mathbb{P}(B_k | x^t)(1 - w_k^t) \tag{7a}$$

$$\mathbb{P}(z_{t+1} | x^t, B_k) = \phi_k(z_{t+1}) w_k^t + \mathbb{P}(z_{t+1} | x^t, B_{k-1})(1 - w_k^t), \tag{7b}$$

where we used (6) and that $\mathbb{P}(z_{t+1} \mid \mathbf{c}_k^t, x^t, B_k) = \mathbb{P}(z_{t+1} \mid \mathbf{c}_k^t, x^t) = \phi_k^t(z_{t+1})$, as given the k-th context, the next observations do not depend on previous

[1] This is different from simply discretising the space and using VMDP estimation.

contexts. Using (6), (7) and Bayes' theorem, we have:

$$w_k^{t+1} = \frac{\mathbb{P}(z_{t+1}|\mathbf{c}_k^t, x^t, B_k)\,\mathbb{P}(\mathbf{c}_k^t|x^t, B_k)}{\mathbb{P}(z_{t+1}|\mathbf{c}_k^t, x^t, B_k)w_k^t + \mathbb{P}(z_{t+1}|x^t, B_{k-1})(1-w_k^t)}$$
$$= \frac{\phi_k^t(z_{t+1})w_k^t}{\phi_k^t(z_{t+1})w_k^t + \mathbb{P}(z_{t+1}|x^t, B_{k-1})(1-w_k^t)}.$$

□

Example 3. *For the MVMDP and discrete $\mathcal{Z}, \mathcal{A}$, we use an n-ary (with $n = |\mathcal{Z} \times \mathcal{A}|$) tree of suffixes on observation-action histories to generate the contexts and Dirichlet-multinomial models for ϕ to predict next observations. We use $\alpha_i^t \triangleq (\alpha_{i,j}^t)_{j=1}^K$ to denote the vector of Dirichlet parameters for context i, at time t. The corresponding marginal probability distribution is given by: $\phi_i^t(z_{t+1} = k) = \alpha_{i,k}^t / \sum_{j=1}^K \alpha_{i,j}^t$, for all $k \in \{1, \dots, Z\}$. Given a sequence x^t, the parameters each context $\mathbf{c}_i$ are $\alpha_{i,k}^T = \alpha_{i,k}^0 + \sum_{t=1}^T \mathbb{I}\{\mathbf{c}_i \prec x^t \wedge z_{t+1} = k\}$, where $\{\alpha_{i,k}^0\}$ is a set of non-negative prior parameters. In addition, each context can include a model for the reward distribution.*

Example 4. *For the CMDP and continuous $\mathcal{Z}$, we consider a partition tree. In some cases this can be chosen* a priori. *For instance if $\mathcal{Z} = [0,1]^n$, a n-ary tree of cubes forms a "natural" partition tree. In other cases, dynamically created structure such as cover trees (Beygelzimer et al., 2006) may be useful. The final question concerns the class of models used for ϕ. Naively, one would have to do density estimation at each context, but we do not deal with this problem in this paper.*

3 Action selection

Example 3 mentioned in passing that one could incorporate a reward model. To do this, we shall simply add a reward distribution $\mathbb{P}(r_{t+1} \mid \mathbf{c})$ to each context $\mathbf{c} \in \mathscr{T}$.[2] In the following discussion, we shall assume that we wish to take actions maximising the expectation of the utility U_t (defined in p. 2). In our model, we maintain a distribution $\mathbb{P}(\cdot|x^t)$ over contexts and parameters, with respect to which this expectation is calculated. It follows by elementary probability, that the expected utility can be written in terms of the utility of each context:

$$\mathbb{E}(U_t \mid x^t) = \sum_{\mathbf{c}} \mathbb{E}(U_t \mid \mathbf{c}, x^t)\,\mathbb{P}(\mathbf{c} \mid x^t). \tag{8}$$

We now define $Q_t(\mathbf{c}) \triangleq \mathbb{E}(U_t \mid \mathbf{c}, x^t)$, which can be expanded as:

$$Q_t(\mathbf{c}) = \mathbb{E}(r_{t+1} \mid \mathbf{c}, x^t) + \gamma \sum_{z_{t+1}} \mathbb{P}(z_{t+1} \mid \mathbf{c}, x^t) \max_{a_{t+1}} \mathbb{E}(U_{t+1} \mid a_{t+1}, z_{t+1}, x^t). \tag{9}$$

[2] In this section we omit context and weight subscripts to simplify the exposition, unless necessary.

The context value function (9) supplies a method to select the optimal (in a decision-theoretic sense) action and is the analogue of (2). The solution, however, requires, as in the standard MDP case, the creation of an augmented Markov decision process. This will take the form of a (pseudo) tree, whose every node corresponds to a particular history, rewards and actions and a set of model parameters. Since ususally the horizon T is too large to employ a full look-ahead, it becomes necessary to only partially construct this tree, via full expansion to a certain depth (Duff, 2002), sparse sampling (Wang et al., 2005), *Monte Carlo* tree search (Veness et al., 2009), or stochastic branch and bound methods (Dimitrakakis, 2009) and then approximate the value of each node. In this paper, we shall only look at methods for *approximating* the values of nodes by fixing the context parameters.

3.1 Approximate methods

Given x^t, we fix the context predictions to $\hat{\phi} = \phi^t$, so that for any $k > 0$ and $\mathbf{x} \in \mathcal{X}^*$, $\mathbb{P}(z_{t+k}, r_{t+k} \mid \mathbf{c}, \mathbf{x}) = \mathbb{P}(z_{t+k}, r_{t+k} \mid \mathbf{c}) = \hat{\phi}(z_{t+k}, r_{t+k})$, while we fix the context weights to $\hat{w} = w^t$, thus also fixing the *conditional* distribution over contexts to $\hat{\mathbb{P}}(\mathbf{c} \mid \mathbf{x})$.[3] Substituting the above in (8), we obtain a *weighted context value iteration* procedure (WCVI):

$$\hat{Q}_t(\mathbf{c}) \triangleq \mathbb{E}(r_{t+1}|\mathbf{c}) + \gamma \sum_{z_{t+1}} \hat{\mathbb{P}}(z_{t+1}|\mathbf{c}) \max_{a_{t+1}} \sum_{\mathbf{c}'} \hat{\mathbb{P}}(\mathbf{c}'|x^t, a_{t+1}, z_{t+1}) \hat{Q}_{t+1}(\mathbf{c}'). \quad (10)$$

This immediately defines a value iteration procedure, since were are only updating the Q_t. If, for all x^t, there is some a unique $\mathbf{c}$ such that $\hat{\mathbb{P}}(\mathbf{c} \mid x^t) = 1$, then this procedure becomes similar to the one proposed by McCallum (1995) for suffix trees. A weighted context Q-learning procedure (WCQL) can be obtained through stochastic gradient descent on the squared temporal-diffence error, shown in Algorithm 1. This leads to an extremely simple method for

Algorithm 1 Weighted context Q-learning with stochastic steepest gradient descent

WCQL($\mathscr{F}, \hat{w}, x^t, r_{t+1}, z_{t+1}, \hat{Q}_t, \eta$)
for $\mathbf{c} \prec x^t$ **do**
  $\zeta := \hat{\mathbb{P}}(\mathbf{c} \mid x^t)$.
  $\tilde{U}_t := r_{t+1} + \gamma \max_a \sum_{\mathbf{c}'} \hat{Q}_t(\mathbf{c}') \hat{\mathbb{P}}\left[\mathbf{c}' \mid x^t \circ (z_{t+1}, a)\right]$
  $\hat{Q}_{t+1}(\mathbf{c}) = \hat{Q}_t(\mathbf{c}) + \eta\zeta \left(\tilde{U}_t - \hat{Q}_t(\mathbf{c})\right)$
end for

acting in an unknown POMDP. The set of active contexts and their probabities $\mathbb{P}(\mathbf{c} \mid x^t)$ are always available due to the inference process, and thus the value function update only increases computation time by a constant factor.

[3] We are only fixing the parameters, so we still obtain a different context distribution for different $\mathbf{x}$.

4 Related models

The variable order Markov decision process presented in this paper is a direct extension of the Bayesian variable order Markov model (BVMM) introduced in (Dimitrakakis, 2010), and which used a similar closed-form recursive update rule with complexity $O(t)$ at each step t, and thus $O(T^2)$ for sequences of length T. This model is most closely related to context tree weighting Willems et al. (1995) and other methods for learning variable order Markov models (surveyed in Begleiter et al., 2004). Related methods include the infinite Markov model (IMM), introduced in (Mochihashi and Sumita, 2008), as well as the stochastic memoizer (Wood et al., 2009), both of which employ sampling. The IMM is related to the infinite hidden Markov model (Beal et al., 2001), which has recently been extended to the infinite partially observable Markov decision process (Doshi-Velez, 2009). Finally, expectation maximization procedures for learning tree mixtures has been reported in (Meila and Jordan, 2001).

Specifically for POMDPs, predictive approaches have been considered also by Wiewiora (2008), who not only examined predictive state representations (Littman et al., 2001), but also standard variable order Markov model algorithms. The CTW algorithm (Willems et al., 1995) has also been extended recently (Veness et al., 2009) to controlled sequences. This model, just as the context model presented here, allows decision-theoretic treatment of the problem and consequently, near-optimal decision making. The particular advantage of the explicit construction presented herein for reinforcement learning, other than its increased generality to continuous MDPs, is that the distribution over contexts can be used to perform approximate decision making without explicit planning. In this sense, it is also an extension of the approach of McCallum (1995) from suffix trees to probabilistic context models.

Finally, for continuous state spaces there is a relation to Ernst et al. (2005), who explored the use of tree representations to perform fitted Q-iteration. The model presented in this paper could also be used in conjuction with fitted Q-iteration. However, the main advantage of the presented approach is that we are no longer restricted to batch settings.

5 Experiments

We performed two types of experiments. Firstly, prediction experiments, where the online predictive accuracy was estimated over a number of runs. This included a small experiment to see whether the distribution over contexts is an effective representation of the hidden state of a POMDP. Secondly, we used the state representation generated by the model as a way to implement simple reactive planning algorithms (in this case, Q-learning (Watkins and Dayan, 1992)). These were evaluated in decision making tasks in a partially observable discrete environment and a discretised continuous environment.

5.1 Practical considerations

The context structure as defined in Sec. 2 can be created dynamically as a tree. In applications, we would need to limit its depth and add mechanisms to avoid creating a large number of branches. In the reported experiments, a leaf node was expanded only when it was at a depth smaller than D and when it had been reached at least once before in the past. The branching factor of the tree is n (see Ex. 3,4), so for a sequence x^t, a model of depth D requires $O\left(n \min(T, |\mathcal{X}|^D)\right)$ space, as there can be at most T unique contexts, while there are at most $|\mathcal{X}|^D$ contexts, each with $n+1$ parameters. At each step t, (7a) can be calculated with a forward and backward pass, while for the discrete case, at each active context $\mathbf{c}_i$, a Dirchilet model can be calculated in constant time, while a density can be calculated in $\mathcal{O}(\log t)$ time (Hutter, 2005). The cost at time t is $O\left(\min(D, t)\right)$, so the total cost after T steps is $O\left(\min(DT, T^2)\right)$, times a $\log t$ factor for the continuous case.

An important question is the choice of prior weights w_0. A natural method is to choosem so that, for all $k > 0$, $\mathbb{P}(\bigvee_{i=k}^{D} \mathbf{c}_i) = \mathbb{P}(\bigvee_{i=k}^{D'} \mathbf{c}_i) = 2^{-k}$ for any $D, D' > k$, where $\mathbf{c}_i$ denotes an context of size i. This ensures that the initial probability of all contexts beyond a certain depth k is always the same no matter how much we increase the maximum depth of the tree.

5.2 Mixture of k-order MDPs

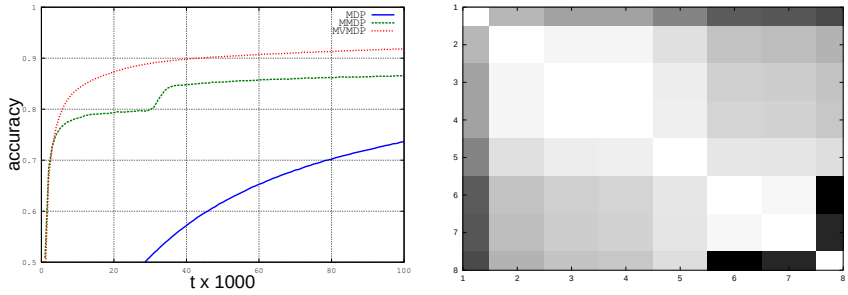
As a baseline for comparisons, we used a mixture of over Markov decision processes of different orders (henceforth MMDP). Let $\mathcal{M} = \{\mu_k : k = 1, \dots, D\}$, be a set of models, such that the model μ_k is a Markov decision process of order $k-1$, that is process for which $\mathbb{P}_{\mu_k}(s_{t+1}, r_{t+1} | a_{1:t}, s_{1:t}) = \mathbb{P}_{\mu_k}(s_{t+1}, r_{t+1} | a_{t+1-k:t}, s_{t+1-k:t})$. We can perform inference on this mixture by maintaining a distribution ψ_t over $\mu_k \in \Theta$. Each model μ_k contains all Markov chains of order k for a discrete observation set $\mathcal{Z}$ and is modelled using a product-of-Dirichlets as a conjugate prior, with parameters $\left\{\alpha_{i,z}^t : z \in \mathcal{Z}^k\right\}$, updated according to: $\alpha_{i,z}^T = \alpha_{i,z}^0 + \sum_{t=1}^{T} \mathbb{I}\left\{z \prec x^t \wedge z_{t+1} = k\right\}$, and with (marginal) predictive distribution $\mathbb{P}_{\mu_k^t}(x_{t+1} = i | z \prec x^t) = \alpha_{i,z}^t / \sum_j \alpha_{j,z}^t$. The mixture can be updated simply by $\psi_{t+1}(\mu_k) \triangleq \mathbb{P}_{\mu_k^t}(x_{t+1}) \psi_t(\mu_k) / \sum_{i=1}^{D} \mathbb{P}_{\mu_i^t}(x_{t+1}) \psi_t(\mu_i^t)$. The first problem with this model is that a large amount of data is required for larger models to start making globally better predictions smaller ones. The second problem is that for non-stationary policies, the distribution of observations is non-stationary. These problems should be significantly alleviated by context models.

5.3 Prediction

We compared the accuracy of the predictions of a MVMDP, an MMDP, as well as a single k-order MDP on a number of tasks. Each task is an unknown POMDP μ. There were $n = 10^3$ runs performed to a horizon $T = 10^6$ for each μ. For the i-th run, we select a policy π and generate a sequence of observations $z^t(i)$ and actions $a_1^t(i)$ with distribution $\mathbb{P}_{\mu,\pi}$. For any model ν, with posterior

predictive distribution $\mathbb{P}_\nu(z_{t+1}|x^t)$ at time t we calculate the average accuracy at time t.

$$u_t(\nu) \triangleq \frac{1}{n} \mathbb{P}_\nu(z_{t+1} = z_{t+1}^{(i)} \mid x_{1,t} = x_{1,t}^{(i)}) \tag{11}$$



(a) Prediction (b) Similarity

Figure 1: Prediction and state representation. Fig. 1(a) shows redictive accuracy on mazes averaged over 10^3 runs and smoothed over 10^3 steps. Unless the order is chosen correctly, the D-order MDP model (**MDP**) is very slow to converge. The mixture of MDP orders (**MMDP**) does achieve good convergence, in some cases matching that of the mixture of variable order Markov model (**MVMDP**). However, it exhibits a step-like behaviour, since a model of higher order needs to be consistently better than other models for a long time before it switches to it. Fig. 1(b) shows the state similarity matrix of an 8-state $1D$-maze problem, obtained by calculating the L_1 distance of the MVMDP context distribution at each actual state. Similar states are white, dissimilar states are darker. Neighbouring states have neighbouring indexes. The closer states are, the closer their context distributions, with the two endpoint states being significantly different from all others.

Figure 1 shows results on prediction tasks. In particular, Fig. 1(a) shows the average accuracy of 10^3 runs over 10^6 time steps on a stochastic maze task with $Z = 16$ observations, which represent a binary encoding of the occupancy of neighbouring grid-points by a wall. This task uses no rewards and a fixed policy.[4] In this and other cases, we found the MVMDP and MMDP to be superior to the MDP, apart from trivial environments where all performances were equal. For some environments, the MMDP approach was approximately as good as the MVMDP, but in general, the MVMDP approach performed best for the largest environments.

In all cases, the MMDP exhibits step-wise performance increases. Each step corresponds approximately to the time where a model of higher order has performed better on average than a model of lower order. This behaviour is well known in Bayesian prediction and could perhaps be rectified with the use

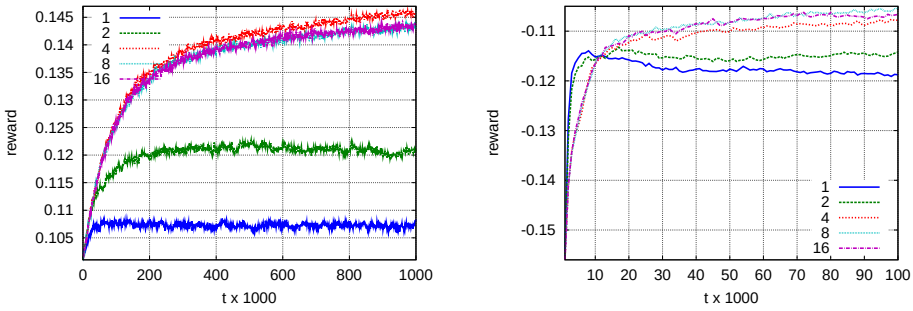
[4]The policy, with some probability $\epsilon > 0$, or whenever a wall was detected, took a random action, and otherwise took the same action as in the previous time step.

of switching-time priors (van Erven et al., 2008). However, even then the MMDP cannot work well when the policy is not fixed.

5.4 State representation

As a side effect, the model creates an internal representation of the current system state. To see this, consider the probability of each context conditioned on the current history, $\mathbb{P}(\mathbf{c}|x^t)$. This will be zero for inactive contexts, and will depend on the weights w_k^t for all the active contexts. Thus, if the MVMDP contains N contexts, its effective state space will be a simplex in $\mathbb{R}_+^N$. Figure 1(b) shows the L_1 distance between context distributions between each state for a corridor task with 8 states.[5] To further test the hypothesis that the context distribution is an approximately sufficient statistic for decision making, we examined the performance of WCQL in the following experiment.

5.5 Decision making



(a) Maze task

(b) Discretised inverted pendulum

Figure 2: Reward per time step, averaged over 10^3 time runs, for $D \in \{1, 2, 4, 8, 16\}$ for two different tasks.

In this paper we do not examine decision-theoretic planning (examined by Veness et al. (2009) using the related CTW model). Our main result is an easily implementable set of value-function-based algorithms (such as WCVI and WQCL), which use the state representation *implicitly defined* by the context distribution. The result is a computationally simple method for acting in partially observable or large environments.

Figure 5.5 shows reward averaged over 10^3 runs on two tasks with increasing depth D (see Sec. 5.1) of the context tre. Figure 2(a) shows results for a POMDP

[5] Briefly, there is a corridor with a finite number of positions. There are two actions, one for moving left and one for moving right. A leftwards movement is not possible at the leftmost end of the corridor and likewise, a rightwards movement is impossible at the rightmost end. With probability 0.01, a random observation $z_t \in \{0, 1\}$ is given. Otherwise $z_t = 0$ unless the action results in hitting a wall, in which case $z_t = 1$.

maze task, where the observation is $z_t = 1$ when a wall is hit and 0 otherwise. There are four actions, one for each cardinal direction, but with probability 0.1 they have no effect. Figure 2(b) shows average reward in an inverted pendulum task with three actions, and a state space $\mathcal{S} \subset \mathbb{R}^2$ which has been discretised in 25 subsets so that $\mathcal{Z} = \{1, \ldots, 25\}$. In both cases, an increased depth results in increased long-term performance.

6 Conclusion

We described a class of context models that can be used to perform efficient, online, closed-form inference for estimating variable order MDPs (the MVMDP model) and large MDPs (the CMDP model). Both models use the same inference procedure on contexts, but a different type of context structure and local context models. We furthermore presented a value iteration algorithm (WCVI) and a Q-learning algorithm (WCQL) that can be implemented with little overhead: The context models maintain a simple representation of state, which can be used to implement classical dynamic programming algorithms. We demonstrated the MVMDP's and the WCQL's capabilities on a number of prediction, state representation and decision making tasks in unknown POMDPs.

The actual structure used for inference is an extension of (Dimitrakakis, 2010), while the proposed value-based algorithms could be seen as an extension of the procedure proposed by McCallum (1995) to more general models. The context tree structure is also close to the BayesTree (Hutter, 2005), which used a random walk that started from the complete set and branched out to subsets. For that reason, the latter approach seems more suitable for density estimation. It seems promising, however, to combine the two approaches for *conditional* density estimation, by using a BayesTree for each ϕ_k model in a CMDP. Such an approach should remain tractable and could form the basis for closed-form non-parametric Bayesian inference in continuous MDPs.

Nevertheless, the *crucial* problem is how to partition a space when no "natural" partitioning (such as the tree of suffixes for discrete sequences, or the binary partition for intervals) exists. This is more pronounced for *controlled* processes, because one cannot rely on the statistics of the observations to create an effective partition. For such problems, entirely new methods may have to be developed.

The simplicity of the inference also makes application of approximate decision-theoretic action selection methods (DeGroot, 1970) possible. In the past point-based methods (Poupart et al., 2006), planning in an augmented-action MDP (Auer et al., 2008; Asmuth et al., 2009), sparse sampling (Wang et al., 2005), *Monte Carlo* tree search (Veness et al., 2009) or stochastic branch and bound (Dimitrakakis, 2009) methods have been suggested. It is an open question which of these, if any, is best for such planning problems.

References

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

Peter Auer, Thomas Jaksch, and Ronald Ortner. Near-optimal regret bounds for reinforcement learning. In *Proceedings of NIPS 2008*, 2008.

Matthew J. Beal, Zoubin Ghahramani, and Carl Edward Rasmussen. The infinite hidden Markov model. In Thomas G. Dietterich, Suzanna Becker, and Zoubin Ghahramani, editors, *NIPS*, pages 577–584. MIT Press, 2001.

Ron Begleiter, Ran El-Yaniv, and Golan Yona. On prediction using variable order Markov models. *Journal of Artificial Intelligence Research*, pages 385–421, 2004.

Aline Beygelzimer, Sham Kakade, and John Langford. Cover trees for nearest neighbor. In *ICML 2006*, 2006.

Morris H. DeGroot. *Optimal Statistical Decisions*. John Wiley & Sons, 1970.

Christos Dimitrakakis. Complexity of stochastic branch and bound for belief tree search in Bayesian reinforcement learning. In *2nd international conference on agents and artificial intelligence (ICAART 2010)*, pages 259–264, Valencia, Spain, 2009. ISNTICC, Springer.

Christos Dimitrakakis. Bayesian variable order Markov models. In Yee Whye Teh and Mike Titterington, editors, *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 9 of *JMLR : W&CP*, pages 161–168, Chia Laguna Resort, Sardinia, Italy, 2010.

Finale Doshi-Velez. The infinite partially observable Markov decision process. In *Advances in Neural Information Processing Systems 21*, Cambridge, MA, 2009. MIT Press.

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

Damien Ernst, Pierre Geurts, and Louis Wehenkel. Tree-based batch mode reinforcement learning. *Journal of Machine Learning Research*, 6:503–556, 2005.

Marcus Hutter. Fast non-parametric Bayesian inference on infinite trees. In *AISTATS 2005*, 2005.

M. L. Littman, R. S. Sutton, and S. Singh. Predictive representations of state. In *Advances in Neural Information Processing Systems 14*, 2001.

Andrew McCallum. Instance-based utile distinctions for reinforcement learning with hidden state. In *ICML*, pages 387–395, 1995.

M. Meila and M.I. Jordan. Learning with mixtures of trees. *The Journal of Machine Learning Research*, 1:1–48, 2001.

D. Mochihashi and E. Sumita. The infinite Markov model. In *Advances in Neural Information Processing Systems*, pages 1017–1024. MIT Press, 2008.

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

Stephane Ross, Brahim Chaib-draa, and Joelle Pineau. Bayes-adaptive POMDPs. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, Cambridge, MA, 2008. MIT Press.

T. van Erven, P. D. Grünwald, and S. de Rooij. Catching up faster by switching sooner : a prequential solution to the AIC-BIC dilemma. *arXiv*, 2008. A preliminary version appeared in NIPS 2007.

J. Veness, K.S. Ng, M. Hutter, and D. Silver. A Monte Carlo AIXI approximation. Arxiv preprint arXiv:0909.0801, 2009.

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

Christopher J.C.H. Watkins and Peter Dayan. Technical note: Q-learning. *Machine Learning*, 8:279, 1992.

Eric Walter Wiewiora. *Modelling probability distributions with predictive state representations*. PhD thesis, University of California, San Diego, 2008.

F.M.J. Willems, Y.M. Shtarkov, and T.J. Tjalkens. The context tree weighting method: basic properties. *IEEE Transactions on Information Theory*, 41(3):653–664, 1995.

F. Wood, C. Archambeau, J. Gasthaus, L. James, and Y.W. Teh. A stochastic memoizer for sequence data. In *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM New York, NY, USA, 2009.