Efficient Planning and Tracking in POMDPs with Large Observation Spaces

Amin Atrash
School of Computer Science
McGill University
Montreal, QC H3A 2A7
aatras@cs.mcgill.ca

Joelle Pineau
School of Computer Science
McGill University
Montreal, QC H3A 2A7
jpineau@cs.mcgill.ca

Abstract

Planning in partially observable MDPs is computationally limited by the size of the state, action and observation spaces. While many techniques have been proposed to deal with large state and action spaces, the question of automatically finding good low-dimensional observation spaces has not been explored as thoroughly. We show that two different reduction algorithms, one based on clustering and the other on a modified principal component analysis, can be applied directly to the observation probabilities to create a reduced feature observation matrix. We apply these techniques to a real-world dialogue management problem, and show that fast and accurate tracking and planning can be achieved using the reduced observation spaces.

Introduction

The design of a good dialogue manager is a key to the deployment of language-based interactive agents, be they robotic assistants, automated answering systems, or online interactive agents. It has been proposed that the dialogue management problem can be cast in the Partially Observable Markov Decision Process (POMDP) framework, such that the agent can track the conversation over time and make an optimal choice of responses to the user's utterances (Singh *et al.* 2002; Roy, Pineau, & Thrun 2000; Williams, Poupart, & Young 2005).

While many recent algorithms have been proposed for finding good control policies in this framework, their efficiency typically depends on the size of the state, action and observation spaces. This is a severe limitation for many applications, including dialogue domains, where the state space spans the list of conversation topics, the action space is defined by the set of possible response, and the observation space corresponds to the space of possible user utterances.

Significant efforts have been devoted to developing MDP and POMDP solving techniques which can deal with large state and action spaces. Common methods for handling large state spaces include function approximation, factorization and dimensionality reduction (Sutton & Barto 1998; Poupart & Boutilier 2003; Roy, Gordon, & Thrun 2005). For large action spaces, hierarchical decomposition techniques have also been studied (Theocharous, Rohanimanesh, & Mahadevan 2001; Pineau, Gordon, & Thrun 2003b).

However there has been comparatively little work on automatically finding good low-dimensional observation spaces. This may be in part because the simpler MDP framework (in which much of the state/action abstraction has been studied) assumes that the state is fully observable (i.e. one-to-one match between states and observations). Nonetheless finding well-behaved compact observation spaces is of great importance in POMDPs where the complexity of both exact and approximate algorithms generally depends on the number of possible belief states (and thus on the number of distinct observations). The only work we are aware of in this direction looks at finding exact (lossless) observation abstractions (Hoey & Poupart 2005; Pineau 2004). While this is valuable, it is unlikely to be applicable in real-world domains with complex sensor data.

In this paper, we focus on the problem of approximately reducing the observation space to provide efficient POMDP planning in domains with rich input features. In particular, we study the applicability of two well-known classes of data summarization techniques to this problem. We present a clustering algorithm which, through a series of EM iterations, finds a small set of summary observations. To provide a comparison against the clustering algorithm, we then present a dimensionality reduction algorithm along the lines of Principal Component Analysis (with a few added constraints) which finds a compressed version of the observation probability model. We describe how the reduced observation space obtained by each of these techniques can be used instead of the full observation space when solving the POMDP model.

We present a full validation of these ideas on a real-world dialogue management problem. Our results show that the accuracy of belief tracking and POMDP planning can be preserved despite an aggressive reduction of the observation space.

POMDPs

Partially Observable Markov Decision Processes (POMDPs) are stochastic models used to model non-deterministic decision-making problems. POMDPs consist of a set of states, S, a set of actions, A, and a set of observations, Z (throughout the paper we assume all of these to be finite).

When an action, a, is executed in state s, the system transitions to state s' with probability $P(s'|s,a)$. The agent then receives a reward, $R(s,a)$ and an observation z is emitted with probability $Pr(z|s')$.[1] The agent has an initial belief distribution across the states, $Pr(s_{t=0})$.

At any point in time, the underlying state, s, is not necessarily observable by the agent. Therefore a distribution across all states must be maintained. The belief distribution $Pr(s_t)$ is updated recursively each time the agent executes an action a and receives an observation z:

$$Pr(s_t = s') = \frac{\sum_s Pr(z|s')Pr(s'|s,a)Pr(s_{t-1} == s)}{\sum_{s''}\sum_s Pr(z|s'')Pr(s''|s,a)Pr(s_{t-1} == s)} \tag{1}$$

Given a POMDP problem, an action-selection policy π can be determined which maps belief states to actions. Generally, this is a difficult problem, and finding an exact solution is at best PSPACE-complete (assuming a finite horizon) and at worse undecidable (assuming an infinite horizon) (Madani, Hanks, & Condon 1999). Efficient approximate solution methods exist, though details of these algorithms are beyond the scope of this paper. For our experiment, we use the Point-Base Value Iteration (PBVI) algorithm which approximates the policy by using stochastic trajectories to select belief points (Pineau, Gordon, & Thrun 2003a). This method allows us to solve relatively large POMDPs in a reasonable amount of time. However in reality, both exact and approximate methods suffer greatly when the number of distinct observations is large. The main problem is that the space of reachable beliefs grows exponentially, as a function of the number of observations, with the planning horizon. Thus POMDPs cannot practically be used for problems with more than a few dozen distinct observations. This clearly precludes the use of rich input modalities such as images and speech. The work we present below attempts to overcome this by adapting standard data reduction techniques to the task of observation abstraction in POMDPs, thus opening the door to solving POMDP problems with much richer observation spaces.

Observation Abstraction

Given an observation matrix for a POMDP with observations $Z = \{z_1, z_2, ..., z_d\}$, the goal is to perform a feature reduction to determine a new set of observations $Z' = \{z'_1, z'_2, ...z'_{d'}\}$ where $d' < d$. As a demonstrative example, the left matrix below may represent the original observation matrix for the POMDP. The goal is to generate a new matrix which has fewer observations (as shown on the right). This new matrix can then be used for tracking and planning. Ideally, this reduction will have little effect on the expected reward for the policy.

$$\begin{array}{cccc} & s_1 & s_2 & s_3 \\ z_1 & 0.60 & 0.30 & 0.05 \\ z_2 & 0.05 & 0.30 & 0.05 \\ z_3 & 0.10 & 0.05 & 0.40 \\ z_4 & 0.05 & 0.25 & 0.20 \\ z_5 & 0.15 & 0.05 & 0.20 \\ z_6 & 0.05 & 0.05 & 0.10 \end{array} \rightarrow \begin{array}{cccc} & s_1 & s_2 & s_3 \\ z'_1 & 0.65 & 0.60 & 0.10 \\ z'_2 & 0.30 & 0.35 & 0.80 \\ z'_3 & 0.05 & 0.05 & 0.10 \end{array}$$

Two feature reduction methods are examined in this paper: an explicit grouping of observations using EM clustering, and an implicit transform using principal component analysis. In both cases, we are faced with the additional constraint of maintaining probabilities conditions: the sum of observations per state of the reduced observation matrix must sum to one, and each value in the matrix must be between zero and one. Feature reduction techniques typically treat the input space as data, where only the relationship between the points themselves is significant, not the location of the data in the entire space. For example, PCA translates the data to an area around the origin. This problem is addressed differently for each method.

Explicit Observation Clustering

Our first method is a simple unsupervised clustering algorithm in the tradition of the K-means algorithm. The idea is to cluster the natural observations Z into the clusters Z', such that observations with similar emission probabilities over all states are clustered together. We denote $g()$ the function that maps observation $z \in Z$ into cluster $z' \in Z'$, as in $z' = g(z)$. [2] Note that observations are clustered based on similarity between their *normalized* emission probabilities:

$$Pr(g(z) = z') = \frac{\sum_{s \in S} |Pr(z|s)/Pr(z) - Pr(z'|s)/Pr(z')|}{\sum_{z'' \in Z}\sum_{s \in S} |Pr(z|s)/Pr(z) - Pr(z''|s)/Pr(z'')|} \tag{2}$$

It is crucial to use the normalized observation probabilities $Pr(z|s)/Pr(z)$ (rather than the unnormalized $Pr(z|s)$) to ensure that observations that are clustered together provide (near)-equivalent inference information over the set of states. Figure 1 shows an example of this normalization. Observations which are twice as likely to occur in s_1 as s_2 are closer after projection onto the $x + y = 1.0$ line, even through the values in the original space are further apart. This means that observations which behave the same way across the states will be grouped together.

The parameters of each cluster can be learned iteratively through the EM algorithm. In the E-step, observations are assigned to clusters as defined in Equation 2. We typically use L2-norm to measure distance between normalized observations. In the M-step, cluster parameters are estimated as follows:

$$\begin{aligned} Pr(z'|s) &= \sum_{z \in Z} Pr(z|s)Pr(g(z) == z'), \qquad (3) \\ Pr(z') &= \sum_{z \in Z} Pr(z)Pr(g(z) == z'). \end{aligned}$$

These two steps are repeated until there is no change in the estimation of the cluster location.

[1] We can assume more generally that observations are conditioned on both state and action: $P(z|s', a)$, however we ignore the dependency on actions throughout this paper for the sake of clarity.

[2] We have considered both **hard** cluster assignments, as is traditional in K-means and **soft** (probabilistic) cluster assignments as is the norm in EM. Our empirical results show no difference between the two.

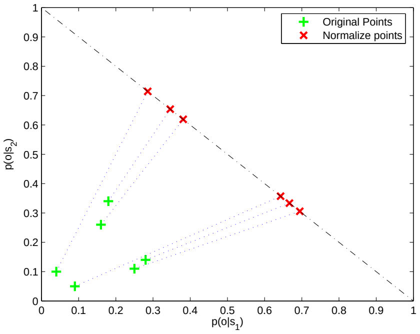

Figure 1: Example of 2-d normalized observations

This EM algorithm was used as an explicit method to group similar observations. The algorithm is run on the input observation matrix, with each observation acting as a sample, e.g. $(z_1 = (0.60, 0.30, 0.05), z_2 = (0.05, 0.30, 0.05), \ldots)$. The observations are clustered in d' clusters, where d' is the desired number of features. Once clustering is completed, all the samples in each clusters are summed together to create a new observation, e.g. $(z_1 + z_2 = z'_1 = (0.65, 0.60, 0.1))$. The advantages of this technique are numerous: (1) probability constraints for the observation matrix are maintained; (2) it generally converges very rapidly (local minima can be a problem, as in all EM procedures, but this has not been observed in practice); (3) the clusters have a natural interpretations in many real-world observation spaces.

Principal Component Analysis

Principal Component Analysis (PCA) is a commonly-used technique which reduces the dimensionality of data by projecting into a lower dimensional space. When applying PCA to reduce the observation matrix, we treat each state as a sample and each observation as a feature. The transformation to the reduced observation (or feature) space is determined by minimizing the sum of squares error between the original data points and the projected points. This can be formulated according to the following objective function.

$$J_{d'} = \sum_{i=1}^{n} \| Pr(Z|s_i) - (Pr(Z) + \alpha_i \cdot \nu) \|^2 \qquad (4)$$

where d' is the dimension of the reduced observation space, n is the number of states, $Pr(Z|s_i) = [Pr(z_1|s_i) \ldots Pr(z_d|s_i)]$ is the vector of original observation probabilities, $Pr(Z) = [Pr(z_1) \ldots Pr(z_d)]$ is the mean probability of each observation (marginalizing over states), α_i is a d'-length vector corresponding to the projection of $Pr(Z|s_i)$ into the reduced space, and ν is a $d' \times d$ projection matrix containing the top d' eigenvectors.

In general, PCA does not maintain any particular constraints (other than that of finding a linear projection) when determining the best transform from the original space to the projected space. And so when applied as described above, PCA produces projected observation vectors $[\alpha_1 \ldots \alpha_n]$ which are not necessarily valid multinomial distributions. If our subsequent goal is to do POMDP planning in the compressed space, this presents an important problem. Specifically, even though our original observation matrix conforms to probability properties, with all values $Pr(z|s) \geq 0$, and $\forall_{s \in S} \sum_{j=1}^{d} Pr(z_j|s) = 1$, the projected observation matrix may not necessarily conform.

A naive approach to handle such a problem is to simply normalize the projected observation matrix:

$$Pr(z'_j|s_i) = \frac{\alpha_i(z'_j)}{\sum_{z' \in Z'} \alpha_i(z')}, \quad \forall z'_j \in Z', \quad \forall s_i \in S \qquad (5)$$

Alternately, the Boltzmann equation can be used to renormalize the projected observation matrix:

$$Pr(z'_j|s_i) = \frac{e^{\alpha_i(z'_j)}}{\sum_{z' \in Z'} e^{\alpha_i(z')}}, \quad \forall z'_j \in Z', \quad \forall s_i \in S \qquad (6)$$

However, we show through experimental results that these types of normalizations do not work particularly well.

A third approach to overcoming this problem is to augment the objective function in Equation 4 with a set of additional constraints:

$$\begin{aligned} &\forall i = 1..n, \ \forall z'_j \in Z' && \alpha_i(z'_j) \geq 0 \qquad (7) \\ &\forall i = 1..n && \sum_{z' \in Z'} \alpha_i z'_j = 1 \end{aligned}$$

The objective function in Equation 4, combined with these constraints, can be formulated as a quadratic programming problem. Appropriate solution techniques can then be used to determine an α_i that satisfies these constraints, as well as the original input observation matrix.

Planning and tracking with reduced observation spaces

Given a reduced observation set Z' and associated probabilities $Pr(z'|s)$ constructed as described above, we can solve the POMDP using any standard algorithm, including the one described in Section . It is worth noting that the complexity of the solution, assuming a point-based approximation, is reduced in two ways by using a compressed observation set. First, the expectation over observations can be taken over the reduced set; second, the set of reachable beliefs is reduced by considering the compact observation set. This second factor is usually more important in terms of scalability.

Finally, a note on belief tracking. It is worth noting that a policy created using a reduced observation matrix can still be executed using the full observation matrix. The mapping of the belief state to the action via the policy does not depend on the observation set used to maintain the belief state, but rather is conditioned at every time step on the observation actually perceived. Note that we could map the received natural observation to the reduced set, and do tracking with the estimated probabilities in the compressed observation space. There is no particular computational advantage to doing this, and it can introduce a loss of information. The magnitude of this loss reflects the quality of the learned compression, as we show in some of our empirical results.

Dialogue

We now study the question of observation abstraction in a more realistic problem domain. We use the SACTI dialogue management dataset (Williams & Young 2004), which consists of a set of conversations between a user and a help wizard over the telephone. Both the user and wizard are human operators. The audio is recorded and transcribed. The speech input is processed through a speech recognizer. The dataset contains 168 conversations.

The task domain is one in which user can request information concerning nine different topics, including restaurants, movies, bus schedules, hotel locations, etc. The user can also request information concerning up to two topics concurrently. For example, he may be asking for the directions to a hotel, thus requesting map information and hotel information. Overall, twenty-five instances of single or pairs of topics appear in the dataset. These constitute the state space for the dialogue management problem. The set of words emitted by the user define the natural observation set. In total, 448 unique words occur in the data set. We assume the user can continue discussing the same topic or change topics after every word.

Throughout the recorded conversations, the wizard responds with the appropriate information. For the purposes of our experiment, we focus on optimizing a policy over which visual information to present to the wizard (e.g. list of hotels, map of the city, bus schedule, etc.) There is therefore one action per state (including pairs of topics in which 2 information windows can be presented side-by-side).

A portion of the data set covering 25 conversations (each consisting of 10-30 utterances) was annotated by hand to have accurate state labellings. The POMDP parameters (topic-to-topic transition probabilities, word emission probabilities) are estimated from the annotated dataset. The reward function was defined as positive when the correct action is taken (i.e. relevant information is displayed).

We apply our observation reduction techniques to this problem. The objective is to find a small reduced observation space which can compactly summarize the space of words used throughout the set of conversations. This domain is particularly interesting because defining a good reduced observation space is not nearly as intuitive as in domains where observations come from a physical environment (e.g. robot sensors). In particular, the notion of “distance” between words is highly context dependent and often stems from complex grammatical and semantic structure. Fortunately, as suggested by our methods, in the context of a planning problem it is sufficient to examine the relationship between observations (words) in the context of their emission probabilities and thus we can perform reduction based on the emission of each word conditioned on each state (topic).

Tracking

We begin our empirical investigation by considering the quality of the tracking under different reduced observation sets. Assuming a POMDP model built as described above, then simple Bayesian tracking (Eqn 1) can be used to determine the likelihood of each topic throughout the course of a conversation. Figure 2 shows the likelihood of six of the topics during the course of a conversation using the full observation set. We can see that different topics become dominant as the conversation progresses, corresponding to change in topic during the actual dialogue.

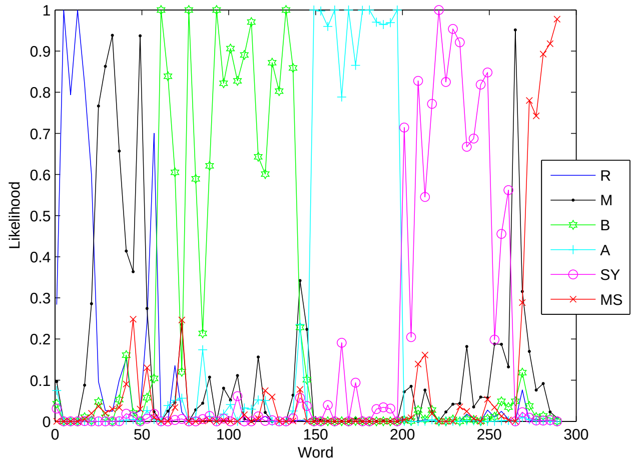

Figure 2: Topic Tracking with 448 features

EM clustering was then applied to reduce the number of observations and the conversation tracked with the reduced observation set. We used hard-EM (i.e. K-means) for this task, therefore we get a unique mapping from each word to its corresponding reduced observation. Figures 3 and 4 show the state (topic) likelihoods over time. For clarity, we only present a segment of the conversation and the top three most likely topics over that period. With 100 features (observation clusters), the system can still reasonably track the topics. Even reducing from 448 observations to 10, the tracking information is still usable and topics recognizable during the conversation. Some confusion does begin to appear. Around word 250, the correct topic is lost as very similar topics become more likely. In this case, MS represents discussions about a map and bus routes. This becomes confused with similar topics such as SY, which represents map and tram routes. As a comparison, random clustering was also applied. Figures 3 and 4 also show the results from random clustering with 100 and 10 features. At 100 features, there is some discrimination between the topics (especially in the case of the first topic). However, at 10 features, the likelihoods are more uniform and the information not usable (recall that we only show the 3 most-likely topics out of 25).

These results show the effects of observation reduction using an EM approach on the quality of state tracking. The ability to track successfully as the number of observations decreases is a promising indicator of the ability to plan with reduced observation spaces. We now investigate this question directly.

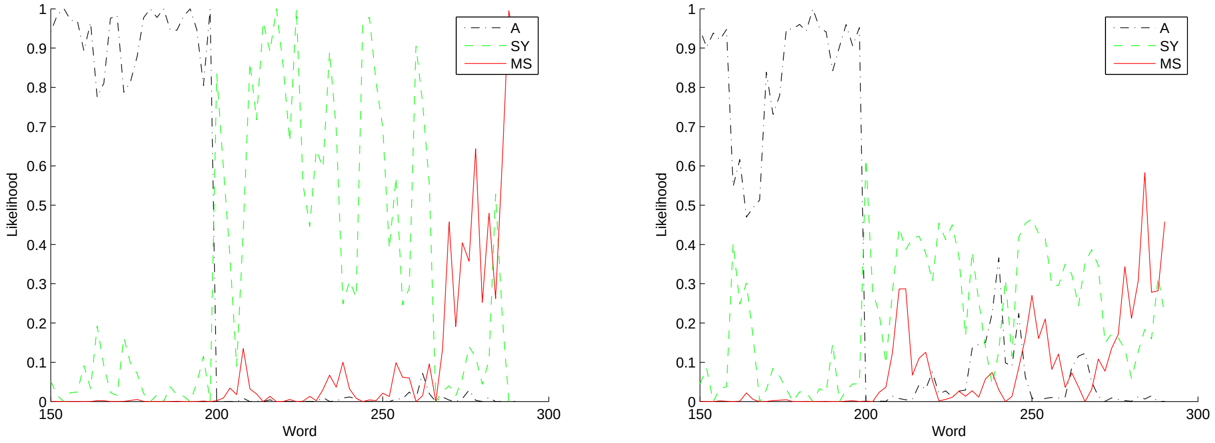

Figure 3: Topic Tracking with 100 features. K-means (on the left) vs. random clustering (on the right)

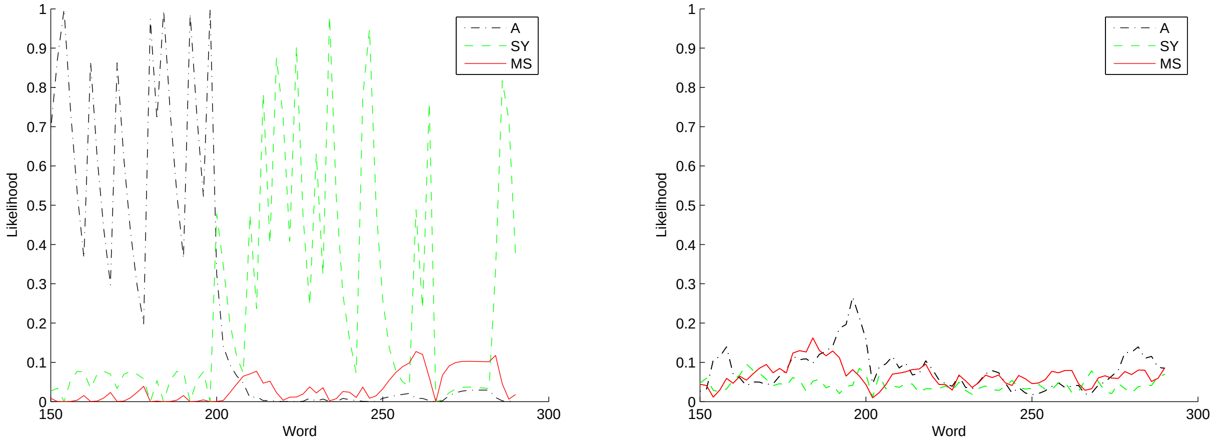

Figure 4: Topic Tracking with 10 features. K-means (on the left) vs. random clustering (on the right)

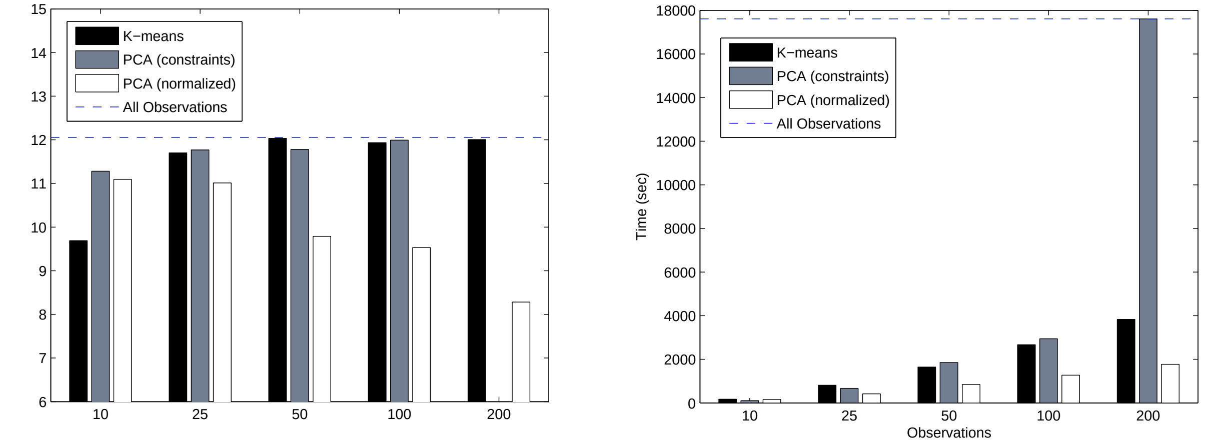

Figure 5: Results for Dialogue POMDP: Expected Reward (on the left) and Planning time (on the right)

Planning

To construct a POMDP using the dialogue data, we assume there is a computer aiding the wizard by presenting information based on the user's requests. The goal of the agent is to present the correct piece or pieces of information. As explained before, the state space is defined as a single or pair of topics (not all pairs of topics occur in the data therefore we only have 25 states). We assume there is one relevant piece of information per topic (a map, a list of hotels and addresses, a list of restaurants and their prices). Thus the action space is the same size as the state space and there is essentially one correct action per state.

If the agent displays the correct information, there is a small reward, otherwise, it incurs a small penalty. The state transitions learned directly from the data (as used in the previous set of results on topic tracking) are assumed to the be transition probabilities for the "correct action". Transition probabilities for "incorrect action" assume that the state stays the same with high probability, and transitions to any other state with a small (uniform) probability. This adds some noise to the system. As with tracking, the observation probabilities reflect the probability of a word occurring in a state and can be determined by counting the original data.

Once the POMDP is constructed, the feature reduction was applied. Policies were then generated based on these reduced POMDPs using the PBVI algorithm (Pineau, Gordon, & Thrun 2003a). Figure 5 (left) shows the average reward based on the number of features and the type of reduction used. This average reward is calculated by 1000 simulated runs using the reduced observation-space policy, but assuming that belief tracking is done using full observation POMDP. The results show that we can consistently learn high quality policies with K-means or PCA clustered observations even with with very few features. The planning time as seen in Figure 5 (right) can be significantly reduced by using fewer features at little cost to the average reward.

The time required to perform the compression is worth discussing. For the K-means and normalized PCA transform, the time required to reduce the number of features is negligible compared to the planning time only requiring 5-6 seconds in a worst case. Determining the reduced observations using the PCA with constraints requires the use of a quadratic program solver which can be prohibitively slow. In our experiments, determining the solution for 50 or fewer features was very fast, but 100 observations required a significant amount of time and 200 features was unsolvable due to memory constraints.

Conclusions

The work presented in this paper gives the first indication that approximate feature reduction techniques can be applied to accelerate planning in POMDPs with rich observation spaces. We found that in general, there is enough information captured in the observation matrix itself to properly reduce the observation space, and by operating directly on this matrix, we can compress features automatically without injecting domain-dependent information. The ideas have been validated on a real-world dialogue management problem.

We found that the simple EM-type clustering worked well in this complex dialogue domain. This is encouraging because the clustering algorithm is simple to implement, fast to compute, and generates intuitive compressed representations. We also found the constrained-based PCA performed better than the normalized PCA, with performance competitive to the EM-clustering. 450 words is relatively small in a domain which typically encountered ten-thousand word dictionaries. PCA's advantage may become more noticeable as the data set and observation space becomes larger.

This work, while preliminary, provides promising evidence that POMDP planning is feasible in domains with rich input spaces. We are encouraged with the results on the dialogue domain, and plan to investigate more sophisticated data reduction techniques in the future.

Acknowledgements

The use of the SACTI corpus was by kind permission of Jason Williams and Karl Weilhammer of the Cambridge University speech group, and funding from the EU Talk Project

References

Hoey, J., and Poupart, P. 2005. Solving POMDPs with continuous or large discrete observation spaces. In *IJCAI*, 1332–1338.

Madani, O.; Hanks, S.; and Condon, A. 1999. On the undecidability of probabilistic planning and inifinite-horizon partially observable markov decision problems. In *AAAI*.

Pineau, J.; Gordon, G.; and Thrun, S. 2003a. Point-based value iteration: An anytime algorithm for POMDPs. In *IJCAI*.

Pineau, J.; Gordon, G.; and Thrun, S. 2003b. Policy-contingent abstraction for robust robot control. In *UAI*, 477–484.

Pineau, J. 2004. *Tractable Planning Under Uncertainty: Exploiting Structure*. Ph.D. Dissertation, Carnegie Mellon University, Pittsburgh, PA.

Poupart, P., and Boutilier, C. 2003. Value-directed compression of POMDPs. In *NIPS*, volume 15.

Roy, N.; Gordon, G.; and Thrun, S. 2005. Finding approximate POMDP solutions through belief compression. *JAIR* 23:1–40.

Roy, N.; Pineau, J.; and Thrun, S. 2000. Spoken dialogue management using probabilistic reasoning. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL2000)*.

Singh, S.; Litman, D.; Kearns, M.; and Walker, M. 2002. Optimizing dialogue management with reinforcement learning: Experiments with the NJFun system. *JAIR* 16:105–133.

Sutton, R. S., and Barto, A. G. 1998. *Reinforcement Learning: An Introduction*. MIT Press.

Theocharous, G.; Rohanimanesh, K.; and Mahadevan, S. 2001. Learning hierarchical partially observable Markov decision process models for robot navigation. In *ICRA*, 511–516.

Williams, J. D., and Young, S. 2004. Characterizing task-oriented dialog using a simulated asr channel. In *International Conference on Speech and Language Processing (ICSLP)*.

Williams, J. D.; Poupart, P.; and Young, S. 2005. Partially observable markov decision processes with continuous observations for dialogue management. In *SigDial Workshop on Discourse and Dialogue*.