Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence

Covering Number as a Complexity Measure for POMDP Planning and Learning

Zongzhang Zhang
School of Computer Science
University of Sci. and Tech. of China
Hefei, Anhui 230027 China
zzz@mail.ustc.edu.cn

Michael Littman
Department of Computer Science
Rutgers University
Piscataway, NJ 08854 USA
mlittman@cs.rutgers.edu

Xiaoping Chen
School of Computer Science
University of Sci. and Tech. of China
Hefei, Anhui 230027 China
xpchen@ustc.edu.cn

Abstract

Finding a meaningful way of characterizing the difficulty of partially observable Markov decision processes (POMDPs) is a core theoretical problem in POMDP research. State-space size is often used as a proxy for POMDP difficulty, but it is a weak metric at best. Existing work has shown that the covering number for the reachable belief space, which is a set of belief points that are reachable from the initial belief point, has interesting links with the complexity of POMDP planning, theoretically. In this paper, we present empirical evidence that the covering number for the reachable belief space (or just "*covering number*", for brevity) is a far better complexity measure than the state-space size for both planning and learning POMDPs on several small-scale benchmark problems. We connect the covering number to the complexity of learning POMDPs by proposing a provably convergent learning algorithm for POMDPs without reset given knowledge of the covering number.

Introduction

Defining a good complexity measure for capturing the difficulty of partially observable Markov decision processes (POMDPs) is a challenging and significant research topic in AI research. It is well known that the intractability of a POMDP originates from the *curse of dimensionality* and the *curse of history* (Pineau, Gordon, and Thrun 2006; Silver and Veness 2010; Lim, Hsu, and Lee 2011). In a planning problem with n states, computation takes place in an n-dimensional belief space. The number of distinct histories grows exponentially as the planning horizon increases. The state-space size and the number of distinct histories are often used as measures for describing the difficulty in POMDP planning in terms of the curses of dimensionality and history, respectively.

In the past decade, point-based value-iteration algorithms have made impressive progress (Smith and Simmons 2005; Pineau, Gordon, and Thrun 2006; Kurniawati, Hsu, and Lee 2008). The success of these algorithms tells us that the curse of history plays a much more important role in affecting POMDP value iteration than the curse of dimensionality. Thus, it is reasonable to believe that the number of possible histories should be a far better predictor of the difficulty of solving POMDPs. However, compared to the size of the state space, characterizing the number of histories is much less straightforward. The number of possible histories is infinite even for the smallest possible POMDP problems. We will argue that the notion of the *covering number for the reachable belief space* (Hsu, Lee, and Rong 2007), briefly "*covering number*", is a viable measure for characterizing the number of possible histories. In POMDPs, each history can be mapped to a belief point—a probability distribution over states. Thus, all possible histories can be represented as a set of belief points, called the *reachable belief space* $\mathcal{R}(b_0)$ in this paper. Intuitively, the covering number for $\mathcal{R}(b_0)$ is the minimum number of balls with a given radius $\delta \geq 0$ so that all reachable belief points lie in some ball in the set (Hsu, Lee, and Rong 2007). Since we can cover the whole state space using just a finite number of balls, the covering number is always finite in all POMDPs for $\delta > 0$.

The finiteness of the covering number does not imply its value can be computed efficiently. We are not aware of any published results providing an algorithm for computing the covering number. In this paper, we provide two simple algorithms for estimation. Experimentally, our estimated covering numbers are far better than other complexity measures, such as state-space size, in predicting the difficulty of both POMDP *planning* and *learning* on several well-known test problems. This data also leads to an observation that benchmark problems that have been proven hard for planning also appear to be hard for learning.

The theoretical connection between planning difficulty and the covering number is already established by Hsu, Lee, and Rong (2007). Here, we make an initial foray into connecting POMDP learning to covering numbers. We present a provably convergent POMDP learning algorithm without reset (meaning systems do not allow the learning algorithm to transition to a fixed initial configuration on demand) using the insight from the covering number. The learning algorithm is composed of an AND/OR graph representation and an existing structure-learning algorithm.

Preliminaries

POMDPs provide a powerful mathematical model for sequential decision making in partially observable stochastic domains. A discrete and discounted POMDP model can be formally defined by a tuple $(S, A, Z, T, \Omega, R, \gamma)$. In the tuple, S, A and Z are the finite and discrete state space, action space and observation space, respectively, $T(s, a, s') : S \times A \times S \to [0, 1]$ is the state-transition function $(P(s'|s, a))$, $\Omega(a, s', z) : A \times S \times Z \to [0, 1]$ is the observation function $(P(z|a, s'))$, $R(s, a) : S \times A \to \mathbb{R}$ is the reward function, and $\gamma \in (0, 1)$ is the discount factor. Because the system's current state is not fully observable, the agent has to rely on the complete history of the past actions and observations to select a desirable current action. A *belief point* (or *belief*, briefly) b is a sufficient statistic for the history of actions and observations—it is a discrete probability distribution over the state space whose element $b(s)$ gives the probability that the system's state is s. The reachable belief space $\mathcal{R}(b_0)$, as mentioned before, is a set of belief points that are reachable from the *initial belief point* b_0 under arbitrary sequences of actions and (non-zero probability) observations. It can be represented as an AND/OR tree rooted at b_0. When the agent takes action a at belief point b and receives observation z, it will arrive at a new belief point $b^{a,z}$:

$$b^{a,z}(s') = \frac{1}{\eta}\Omega(a, s', z) \sum_{s \in S} T(s, a, s')b(s), \qquad (1)$$

where η is a normalizing constant (Kaelbling, Littman, and Cassandra 1998). The constant is the probability of receiving observation z after the agent takes action a at belief point b, which can be specified as:

$$P(z|b, a) = \sum_{s' \in S} \Omega(a, s', z) \sum_{s \in S} T(s, a, s')b(s). \qquad (2)$$

The definition of the covering number for a set of points follows.

Definition 1. *Given a set of points B in an L^p metric space X, a δ-region with the center point b_c, denoted $c(b_c, \delta)$ or sometimes c for short, is a subspace in X that satisfies $||b - b_c|| \leq \delta$ for all b in $c(b_c, \delta)$. $\hat{C}_B(\delta)$, a set of δ-regions, is a δ-cover for B satisfying the condition that all points b in B are included in at least one δ-region in $\hat{C}_B(\delta)$. The δ-cover for B with the smallest number of δ-regions is denoted by $C_B(\delta)$. The δ-covering number for B, denoted by $|C_B(\delta)|$, is the size of the smallest δ-cover for B.*

In this paper, we measure the distance between belief points in an L^1 metric space $\mathcal{B}$: for $b_1, b_2 \in \mathcal{B}$, $||b_1 - b_2|| = \sum_{s \in S} |b_1(s) - b_2(s)|$. Since all belief points b in $\mathcal{B}$ satisfy $\sum_{s \in S} b(s) = 1$ and $b(s) \geq 0$, we have $\max_{b_1, b_2 \in \mathcal{B}} ||b_1 - b_2|| = 2$. In a two- (or three-) dimensional space, a δ-region, or ball, represents a square (or regular octahedron). Following Definition 1, we denote $C_{\mathcal{R}}(\delta)$ as the smallest δ-cover for $\mathcal{R}(b_0)$.

The notion of the covering number of the state space appeared previously in a reinforcement-learning paper (Kakade, Kearns, and Langford 2003). It refers to the number of neighborhoods, finite but much less than $|S|$, required for accurate local modeling in Markov decision processes (MDPs) with very large or infinite state spaces. Hsu, Lee, and Rong (2007) extended the concept from MDPs to POMDPs, and revealed that an approximately optimal POMDP solution can be computed in planning time polynomial in $|C_{\mathcal{R}}(\delta)|$, the covering number for $\mathcal{R}(b_0)$.

Algorithm 1: Breadth-First Search (BFS)

```
Input: $expandSet = \{b_0\}$, $\mathcal{R}^{depth}(b_0) = \{b_0\}$, $depth$.
Output: $\mathcal{R}^{depth}(b_0)$.
while $expandSet$ is not empty do
    $expandSet = expandSet \setminus \{b\}$;
    for $a \in A$ do
        for $z \in Z$ do
            if $b^{a,z} \notin \mathcal{R}^{depth}(b_0)$ and $d_b \leq depth$ then
                $\mathcal{R}^{depth}(b_0) = \mathcal{R}^{depth}(b_0) \bigcup \{b^{a,z}\}$;
            if $b^{a,z} \notin \mathcal{R}^{depth}(b_0)$ and $d_b < depth$ then
                $expandSet = expandSet \bigcup \{b^{a,z}\}$;
```

Approximation Algorithms for Computing the Covering Number

In this section, we describe two approaches for estimating $|C_{\mathcal{R}}(\delta)|$, the δ-covering number for the reachable belief space $\mathcal{R}(b_0)$. They are only different in how they collect a finite subset of $\mathcal{R}(b_0)$. Both methods have their advantages and disadvantages.

Our approaches to approximation start with the *breadth-first search (BFS)* algorithm and a *revised breadth-first search (R-BFS)* algorithm to obtain a finite subset of $\mathcal{R}(b_0)$. Then, we use the *complete-link clustering* method (Manning, Raghavan, and Schütze 2008) on this subset to estimate the covering number. Note that exact computation of $|C_{\mathcal{R}}(\delta)|$ is NP-hard even when $\mathcal{R}(b_0)$ is a finite set (Hochbaum 1996).

Breadth-First Search

The BFS algorithm outputs $\mathcal{R}^{depth}(b_0)$, which is a set of distinct beliefs that are reachable from b_0 under arbitrary sequences of no more than $depth$ action-observation steps (see Algorithm 1).

Now, we reveal the relationship between the size of $\mathcal{R}(b_0)$ and the size of $\mathcal{R}^{depth}(b_0)$ in terms of weight. We give each belief b in the AND/OR tree a weight:

$$weight(b) = \gamma^{d_b} P(b|b_0), \qquad (3)$$

where d_b represents the number of action-observation steps ($depth$) from b_0 to b, and $P(b|b_0)$ denotes the likelihood of arriving at b from b_0 by following random actions. For example, if b is reachable from b_0 by following $a_1 z_1 a_2 z_2 ... a_n z_n$ and going through $b_1, b_2, ..., b_{n-1}$, then $P(b|b_0) = \frac{1}{|A|^n} P(z_1|b_0, a_1) P(z_2|b_1, a_2) ... P(z_n|b_{n-1}, a_n)$. It is reasonable to weight the belief points this way because belief points that are likely to be reached from b_0 should be more important than those rarely encountered belief points. Thus, the total weight of all possible belief points in

Algorithm 2: Revised Breadth-First Search (R-BFS)

```
Input: $expandSet = \{b_0\}, \epsilon(> 0), \mathcal{R}_\epsilon(b_0) = \{b_0\}$.
Output: $\mathcal{R}_\epsilon(b_0)$.
while $expandSet$ is not empty do
    $expandSet = expandSet \setminus \{b\}$;
    for $a \in A$ do
        for $z \in Z$ do
            if there does not exist a belief $\hat{b}$ in $\mathcal{R}_\epsilon(b_0)$
            that satisfies $||\hat{b} - b^{a,z}|| \leq \epsilon$ then
                $\mathcal{R}_\epsilon(b_0) = \mathcal{R}_\epsilon(b_0) \bigcup \{b^{a,z}\}$;
                $expandSet = expandSet \bigcup \{b^{a,z}\}$;
```

Algorithm 3: Complete-Link Clustering

```
Input: $B$,$\delta$.
Output: $|\hat{C}_B(\delta)|$.
for $b_i \in B$ do
    $c_i = \{b_i\}$;
$cover = \{c_1, c_2, ..., c_{|B|}\}$;
while $d$(cover) $\leq 2\delta$ do
    $c_i, c_j = \text{argmin}_{c_i \neq c_j \in cover}\, d(c_i, c_j)$;
    $cover = cover \bigcup \{c_i \bigcup c_j\} \setminus \{\{c_i\}, \{c_j\}\}$;
$\hat{C}_B(\delta) = cover$;
```

the AND/OR tree rooted at b_0, denoted by *weight*($\mathcal{R}(b_0)$), is $\sum_{b \in \mathcal{R}(b_0)} weight(b) = \sum_{b \in \mathcal{R}(b_0)} \gamma^{d_b} P(b|b_0) = \sum_{depth=0}^{\infty} \gamma^{depth} = \frac{1}{1-\gamma}$, and the total weight of all belief points in $\mathcal{R}^{depth}(b_0)$, denoted by *weight*($\mathcal{R}^{depth}(b_0)$), is $\frac{1-\gamma^{depth+1}}{1-\gamma}$. So, at least $1 - \gamma^{depth+1}$ of all possible belief points in $\mathcal{R}(b_0)$ in terms of weights have been included in our collected set $\mathcal{R}^{depth}(b_0)$.

Revised Breadth-First Search

The R-BFS algorithm provides another way of obtaining a finite subset of $\mathcal{R}(b_0)$, called $\mathcal{R}_\epsilon(b_0)$ in this paper (see Algorithm 2). The $\mathcal{R}_\epsilon(b_0)$ that the algorithm returns is always finite because each new element added to the set must be ϵ away from existing points and the volume of the belief simplex is finite. Proposition 1 provides a mathematical link between $|C_{\mathcal{R}}(\delta)|$ and $|C_{\mathcal{R}_\epsilon}(\delta)|$ under a contraction assumption. Please see Appendix for proofs.

Proposition 1. *Assume that* $||b_1^{a,z} - b_2^{a,z}|| = \eta ||b_1 - b_2||$, *where* $0 \leq \eta < 1$, *for all actions a, observations z, and beliefs* b_1 *and* b_2 *in the reachable belief space* $\mathcal{R}(b_0)$. *Let* $\epsilon, \delta > 0$. *Then,* $|C_{\mathcal{R}}(\delta + \frac{\epsilon}{1-\eta})| \leq |C_{\mathcal{R}_\epsilon}(\delta)| \leq |C_{\mathcal{R}}(\delta)|$.

The proof of Proposition 1 is based on Lemma 1, which says that all beliefs in $\mathcal{R}(b_0)$ can be included in a set of $\frac{\epsilon}{1-\eta}$-regions with size $|\mathcal{R}_\epsilon(b_0)|$ under the contraction assumption.

Lemma 1. *If* $||b_1^{a,z} - b_2^{a,z}|| = \eta ||b_1 - b_2||$, *where* $0 \leq \eta < 1$, *for all actions a, observations z, and belief points* b_1 *and* b_2 *in* $\mathcal{R}(b_0)$. *Then,* $\mathcal{R}(b_0) \subseteq \bigcup_{b \in \mathcal{R}_\epsilon(b_0)} c(b, \frac{\epsilon}{1-\eta})$.

An advantage of the BFS algorithm is that the difference between *weight*($\mathcal{R}^{depth}(b_0)$) and *weight*($\mathcal{R}(b_0)$) can be estimated without any assumption. Its disadvantage is the heavy burdens that it leaves to the complete-link clustering method. For example, assume $\gamma = 0.95$, and we want to get 95% of all belief points in $\mathcal{R}(b_0)$ in terms of weights, we need to set $depth = \lceil \log_{0.95} 0.05 \rceil - 1 = 58$. It means that the BFS algorithm needs to output $\frac{(|A||Z|)^{59}-1}{|A||Z|-1}$ belief points, which then need to be clustered.

A benefit of the R-BFS algorithm is that $|C_{\mathcal{R}_\epsilon}(\delta)|$ is usually bigger than $|C_{\mathcal{R}^{depth}}(\delta)|$, and therefore closer to $|C_{\mathcal{R}}(\delta)|$, when $|\mathcal{R}_\epsilon(b_0)| = |\mathcal{R}^{depth}(b_0)|$. A weakness is that the contraction assumption in Proposition 1 is not always satisfied in general, although it can be always satisfied in an "expected" sense (Even-Dar, Kakade, and Mansour 2005).

Complete-Link Clustering

Algorithm 3 shows how we use the complete-link clustering method to estimate $|C_B(\delta)|$. Here, the set of points B can be $\mathcal{R}^{depth}(b_0)$ or $\mathcal{R}_\epsilon(b_0)$. We define a *cover* as a set of δ-regions. Each δ-region, denoted c_i, consists of a set of belief points. In the beginning, only one belief point is included in each δ-region. Then, the algorithm checks the possibility of merging two δ-regions into one based on the following distance definition. We define the distance between regions in *cover* as: $d(cover) = \min_{c_i \neq c_j \in cover} d(c_i, c_j)$, where $d(c_i, c_j) = \max_{b_1 \in c_i, b_2 \in c_j} ||b_1 - b_2||$. Thus, the distance between two δ-regions is the distance between two remotest belief points in the two δ-regions. These distances are then used to merge δ-regions in the complete-link clustering algorithm, whose overall time complexity is $\mathcal{O}(|B|^2 \log |B|)$ (Manning, Raghavan, and Schütze 2008). The δ-cover that the complete-link clustering discovers, $\hat{C}_B(\delta)$, may not be the smallest δ-cover for B. However, $|\hat{C}_B(\delta)|$ it returns is often quite close to $|C_B(\delta)|$ empirically (Xu and Wunsch 2005).

Note that complete-link clustering is not absolutely necessary to cluster the output of R-BFS. We can use the output of R-BFS by setting $\epsilon = 2\delta$ to estimate the covering number directly. We will see estimated covering numbers of medium POMDPs based on this insight in experiments.

The Covering Number and the Complexity of POMDP Learning

In this section, we extend the covering number from POMDP planning to POMDP learning. We finally present the main result: a learning algorithm for POMDPs without reset with *provable convergence* given the covering number. It suggests that accurate predictions of observations can be made after a polynomial number of inaccurate predictions that grow *exponentially* with the covering number.

We start with Lemma 2, which says belief points in the same δ-region have similar observation probabilities. Lemma 2 is used in our main result to treat all belief points in a δ-region c as the region's central point b_c. Such an approximation method brings no more than δ prediction errors in predicting observation probabilities.

Lemma 2. *For any two belief points b and b', if $||b-b'|| \leq \delta$, then $|P(z|b,a) - P(z|b',a)| \leq \delta$.*

To our knowledge, a major difficulty in learning POMDPs is to approximately reflect all infinite possible mappings between successive belief points and δ-regions, in spite of the lack of its discussion in prior work. Here, we suggest using a finite set of AND/OR graphs with $|C_{\mathcal{R}}(\delta)|$ nodes to handle the challenge. It provides a novel way of changing the problem of learning POMDPs into a much easier problem of learning MDPs with $|C_{\mathcal{R}}(\delta)|$ states. In each AND/OR graph, each node represents a δ-region, OR is over actions and AND is over observations. There are $|C_{\mathcal{R}}(\delta)|^{|A||Z|}$ AND/OR graphs with $|C_{\mathcal{R}}(\delta)|$ nodes. A δ-accurate AND/OR graph defines the right action-observation edges that connect any two δ-regions. "Right" here means that for all reachable belief points b in a given δ-region, the δ-accurate graph predicts which δ-regions the successive beliefs b^{t_n}, where $t_n = a_1z_1a_2z_2...a_nz_n$, belong to with accurate probability. Such a δ-accurate graph definitely exists in the graph set, as stated in Lemma 3:

Lemma 3. *For two arbitrary belief points in the reachable belief space $\mathcal{R}(b_0)$, if they are in a δ-region, after the same random action-observation pairs, they end up in the same successive δ-regions with probability almost one.*

Once the δ-region that b_0 belongs to and the δ-accurate graph are known, the POMDP-learning problem reduces to a $|C_{\mathcal{R}}(\delta)|$-state MDP learning problem. The adaptive k-meteorologists algorithm (Diuk, Li, and Leffler 2009), an existing structure-learning algorithm, can be used to determine the δ-accurate graph among graphs in the graph set.

Using Lemmas 2 and 3, we can prove Proposition 2, which can be considered an extension of Theorem 1 in Hsu, Lee, and Rong (2007) in POMDP learning.

Proposition 2. *Given $0 < 2\delta \leq \varepsilon \leq 1$ and $|C_{\mathcal{R}}(\delta)|$, the δ-covering number for the reachable belief space $\mathcal{R}(b_0)$, let $a_1z_1a_2z_2...a_Lz_L$ be a sequence generated by choosing L actions uniformly at random. Let b_{t+1} be the belief point arrived by taking action a_{t+1} and receiving observation z_{t+1} at belief point b_t. Then, we can build a learning algorithm for POMDPs without reset that guarantees $\frac{1}{L}\sum_{t=0}^{L-1}[P(z_{t+1}|b_t,a_{t+1}) - \hat{P}(z_{t+1}|b_t,a_{t+1})]^2 \leq \varepsilon$ with probability at least $1-\alpha$ with an upper bound of $L = \mathcal{O}(\frac{k}{\varepsilon^3}\ln\frac{k}{\alpha})$, where $k = |C_{\mathcal{R}}(\delta)|^{|A||Z|+1}$.*

The significance of Proposition 2 is that it provides a novel way of building a *provably convergent* POMDP-learning algorithm using the insight from covering numbers.

Experimental Results

In this section, we provide empirical evidence that the covering number is a good predictor of performance for both planning and learning POMDPs. We estimated the covering numbers on the suite of small-scale POMDP problems listed in the first column of Table 1 taken from Cassandra's POMDP website[1]. We chose these seven benchmark problems only because their average one-step learning errors were available from James and Singh (2004). The Bridge Repair problem in James and Singh (2004) was not included because its γ was 1 and we only discussed POMDPs with $\gamma \in (0,1)$ because our theoretical result on BFS only applied when γ was less than 1.

Columns 2–4 list the sizes of the state spaces, action spaces and observation spaces in all test POMDPs, respectively. Columns 5 and 6 list the covering numbers for these POMDP problems as approximated by the BFS and R-BFS algorithms, respectively[2]. We set $|\mathcal{R}^{depth}(b_0)|$, the size of the collected belief set $\mathcal{R}^{depth}(b_0)$, as 1000 for all tested problems in the BFS algorithm, used $\epsilon = 0.04$ as the input of the R-BFS algorithm, and set $\delta = 0.2$ in the complete-link clustering method. We think setting the same $|\mathcal{R}^{depth}(b_0)|$ is fairer than setting the same $depth$ on test problems in collecting representative subsets of $\mathcal{R}(b_0)$.

Column 7 in Table 1 reports the difficulty of solving each of these POMDPs via the Witness algorithm (Littman 1994), a well-known POMDP planning algorithm. The difficulty level ranges from "easy" to "hard" according to whether a POMDP's planning time is located in $[0, 10)$, $[10, 1000)$ and $[1000, +\infty)$ seconds. We computed them using the pomdp-solve-5.3 software package, also taken from Cassandra's web-page, on an AMD dual core processor 3600+ 2.00GHz with 2GB memory. Most results in Column 7 can also be found in the work of Littman, Cassandra, and Kaelbling (1995). As shown in Column 7, the Shuttle and 4x3 Maze problems are the two hardest planning problems. Note that they are also the problems with the two largest covering numbers (Columns 5 and 6). Furthermore, the four "easy" planning problems are the four with the smallest covering numbers (Column 6). In contrast, the problems with the two biggest state-space sizes are the Cheese Maze and 4x3 Maze problems. *State-space sizes do not tell us the Cheese Maze problem is easier to solve than the 4x3 Maze problem, but (estimated) covering numbers work well in distinguishing them.*

Columns 8 and 9 in Table 1 show the prediction error after learning the POMDPs using myopic learning, also known as gradient-based learning (Singh et al. 2003), and reset learning algorithms (James and Singh 2004). These two algorithms were designed based on the predictive state representation or PSR (Littman, Sutton, and Singh 2002), which is an alternative representation to POMDPs intended to be better suited for learning. The columns report $\frac{1}{L}\sum_{t=0}^{L-1}[P(z_{t+1}|b_t,a_{t+1}) - \hat{P}(z_{t+1}|b_t,a_{t+1})]^2$, the average one-step prediction error on a single or spliced action-observation sequence with given length $L = 10^7$ (James and Singh 2004). Here, we see the Shuttle and 4x3 Maze

[1] http://www.pomdp.org

[2] Besides the above two algorithms, another approach to generating representative subsets of the reachable belief points is a random path algorithm that samples belief points from a long stochastic action-observation trajectory started from b_0. Such a method is promising for large POMDPs. Empirical results from this algorithm on test problems are available on request.

1	2	3	4	5	6	7	8	9	10	11
Problems	$\|S\|$	$\|A\|$	$\|Z\|$	$\|\hat{C}_{\mathcal{R}^{depth}}(0.2)\|$	$\|\hat{C}_{\mathcal{R}_{0.04}}(0.2)\|$	Witness	Reset Learning	Myopic Learning	$\|S\|\|A\|(\|S\|$ +$\|Z\|$-2)	$\|Q\|(\|Q\|+1)$ $\|A\|\|Z\|+\|Q\|$
Tiger	2	3	2	**3**	**3**	easy	3.5e-7	4.3e-6	12	38
Paint	4	4	2	**23**	**22**	easy	2.7e-7	1.0e-5	64	50
Float-reset	5	2	2	**8**	**7**	easy	3.7e-8	1.0e-4	50	125
Cheese Maze	11	4	7	**16**	**16**	easy	3.8e-6	3.7e-4	704	3707
Network	7	4	2	**22**	**29**	medium	3.2e-6	8.3e-4	196	455
Shuttle	8	3	5	**39**	**42**	hard	2.2e-5	2.7e-2	264	847
4×3 Maze	11	4	6	**96**	**146**	hard	6.4e-5	6.6e-2	660	2650

Table 1: Summary table of empirical results. See text for details.

Measures	Reset Learning	Myopic Learning
$\|S\|$	0.6237	0.5580
$\|S\|\|A\|(\|S\|+\|Z\|-2)$	0.6159	0.5672
$\|Q\|(\|Q\|+1)\|A\|\|Z\|+\|Q\|$	0.4744	0.4220
$\|\hat{C}_{\mathcal{R}^{depth}}(0.2)\|$	**0.9782**	**0.9702**
$\|\hat{C}_{\mathcal{R}_{0.04}}(0.2)\|$	**0.9811**	**0.9659**

Table 2: Linear correlation coefficients between different measures and two learning algorithms.

problems are again the two hardest problems to learn. *This result provides some additional empirical support for the idea that the difficulty of POMDPs (for planning and learning) is an inherent property of the POMDP and is well captured by its covering number.*

Now, *we further argue that the covering number is a more appropriate measure of complexity for POMDP learning than several natural alternatives.* Besides state-space size, we consider two reasonable measures: the number of independent parameters in its POMDP representation and the number of independent parameters in its PSR representation, see Columns 10 and 11 in Table 1. The parameters in a POMDP representation consist of the elements in the inherent state-transition and observation matrices ($T(s, a, s')$ and $\Omega(a, s', z)$). Since the number of independent elements in T is $|S||A|(|S| - 1)$ and the number of independent elements in Ω is $|S||A|(|Z|-1)$, the independent parameter number in POMDPs is $|S||A|(|S| + |Z| - 2)$. In the PSR representation, Q is a set of core tests, which can be considered as a substitute for S, leading to a total number of independent parameters of $|Q|(|Q| + 1)|A||Z| + |Q|$. The numbers of core tests in the test problems were listed by James and Singh (2004). As shown in Table 2, *the linear correlation coefficients between estimated covering numbers and errors in learning algorithms are* $0.96 \sim 0.98$ *(see Rows 5 and 6), which indicate a much better correlation between learning errors and covering numbers than with the other three proposed learning complexity measures.*

These strong linear correlation coefficients do not just provide empirical support that the covering number of a POMDP is a good complexity measure of its learning, but also leave the following intriguing puzzle:

Open Problem. Why do current PSR learning algorithms appear to be *subconsciously* influenced by the covering number in spite of the lack of an explicit connection to this concept in the algorithms themselves?

One possible first step to address this problem might be to study the relationship between the covering number for $\mathcal{R}(b_0)$ and the covering number for the set of core prediction vectors $P(Q|h) = [P(q_1|h), \ldots, P(q_{|Q|}|h)]$ on all possible histories h.

Finally, we would like to estimate covering numbers of medium POMDPs (Smith and Simmons 2005; Pineau, Gordon, and Thrun 2006; Smith 2007). To make the calculation tractable, we have obtained an extremely coarse approximation using R-BFS with $\epsilon = 2\delta = 1.0$. Although the results, shown in Table 3, are useless in terms of formal error bounds, we can still find interesting phenomena. The covering number suggests that the RockSample[4,4] problem should be more easily solved than the Tiger-Grid, Hallway, and Hallway2 problems, although its state-space size is the biggest among them. Indeed, the HSVI2 algorithm needed less than 10 seconds to converge on the RockSample[4,4] problem, but needed more than 1000 seconds on the Hallway2 problem. Please see Figure 3 in Smith and Simmons (2005) for details. We hope the preliminary results in Table 3 will help spur additional research on these medium-scale POMDPs.

Conclusion

This paper proposes that the covering number is an appropriate measure of difficulty for both POMDP planning and learning. The notion of the covering number (implicitly) worked as an important driver in the development of point-based value iteration algorithms in the past decade. We believe there are opportunities for creating new efficient POMDP learning algorithms by taking advantage of covering numbers directly.

There are two major contributions in this work. First, we presented two algorithms for estimating the covering number and discussed their advantages and disadvantages. Experiments showed that our estimated covering number was far better than state-space size and number of parameters in predicting planning time and learning

Problems	$\lvert\hat{C}_{\mathcal{R}_{1.0}}(0.5)\rvert$	$\lvert S\rvert$	Problems	$\lvert\hat{C}_{\mathcal{R}_{1.0}}(0.5)\rvert$	$\lvert S\rvert$	Problems	$\lvert\hat{C}_{\mathcal{R}_{1.0}}(0.5)\rvert$	$\lvert S\rvert$
Tiger-Grid	**213**	36	Hallway	**607**	61	Hallway2	**1747**	93
TagAvoid	**527**	870	RockSample[4,4]	**17**	256	LifeSurvey1	**2931**	7001

Table 3: Estimated covering numbers and state-space sizes on a suite of medium-scale POMDPs.

accuracy on a suite of POMDP problems. Second, we proposed a POMDP learning algorithm with convergence guarantee using the covering-number concept.

Acknowledgments

Special thanks to anonymous reviewers for their helpful feedback and suggestions. This work was supported in part by the China Scholarship Council, the National Science Foundations of China under Grant Nos. 60745002, 61175057, 61105039, the National High-Tech Research and Development Plan of China under Grant No. 2008AA01Z150, and NSF Award 0713148.

Appendix

Proof of Lemma 1. Since $\mathcal{R}_\epsilon(b_0) \subseteq \bigcup_{b\in\mathcal{R}_\epsilon(b_0)} c(b, \frac{\epsilon}{1-\eta})$, we only need to prove that any $\bar{b} \in \mathcal{R}(b_0) \setminus \mathcal{R}_\epsilon(b_0)$ satisfies $\bar{b} \in \bigcup_{b\in\mathcal{R}_\epsilon(b_0)} c(b, \frac{\epsilon}{1-\eta})$. When we search backward from $\bar{b}$ along the AND/OR tree of reachable belief points with the root belief point b_0, an ancestor belief point b (or $b = \bar{b}$) can always be found that was not included in $\mathcal{R}_\epsilon(b_0)$ because some belief point b_1 in $\mathcal{R}_\epsilon(b_0)$ satisfied $||b - b_1|| \leq \epsilon$. Assume $\bar{b}$ was the result of transforming b by following the action-observation sequence $a_1z_1a_2z_2...a_nz_n$. Similarly, assume $\bar{b}_1$ resulted from transforming b_1 by following the same action-observation step. By chaining applications of the contraction assumption, it follows that $||\bar{b} - \bar{b}_1|| \leq \eta^n\epsilon$. Search forward from b_1 along $a_1z_1a_2z_2...a_nz_n$ until some child belief point $b_1^{a_1z_1...a_iz_i}$ is reached—where $i \geq 1$—that is not included in $\mathcal{R}_\epsilon(b_0)$. If no such child belief point exists, then $\bar{b}_1 \in \mathcal{R}_\epsilon(b_0)$ and $\bar{b} \in c(\bar{b}_1, \eta^n\epsilon) \subseteq \bigcup_{b\in\mathcal{R}_\epsilon(b_0)} c(b, \frac{\epsilon}{1-\eta})$. If such a child belief point does exist, we can find the belief point b_2 that caused $b_1^{a_1z_1...a_iz_i}$ to be excluded from $\mathcal{R}_\epsilon(b_0)$ ($||b_1^{a_1z_1...a_iz_i} - b_2|| \leq \epsilon$). Assume $\bar{b}_2$ is the result of transforming b_2 by following the action-observation sequence $a_{i+1}z_{i+1}...a_nz_n$. By similar reasoning, we have $||\bar{b}_1 - \bar{b}_2|| \leq \eta^{n-i}\epsilon$. Repeating this procedure $k \leq n$ times, we have $\bar{b}_k \in \mathcal{R}_\epsilon(b_0)$. Since $||\bar{b} - \bar{b}_1|| \leq \eta^{n_1}\epsilon, ||\bar{b}_1 - \bar{b}_2|| \leq \eta^{n_2}\epsilon, \ldots, ||\bar{b}_{k-1} - \bar{b}_k|| \leq \eta^{n_k}\epsilon$, where $0 \leq n_k < ... < n_2 < n_1 = n$, we have $||\bar{b} - \bar{b}_k|| \leq ||\bar{b} - \bar{b}_1|| + ||\bar{b}_1 - \bar{b}_2|| + ... + ||\bar{b}_{k-1} - \bar{b}_k|| \leq (\eta^{n_1} + \eta^{n_2} + ... + \eta^{n_k})\epsilon \leq \frac{1-\eta^{n+1}}{1-\eta}\epsilon \leq \frac{\epsilon}{1-\eta}$. Thus, we have $\bar{b} \in c(\bar{b}_k, \frac{\epsilon}{1-\eta}) \subseteq \bigcup_{b\in\mathcal{R}_\epsilon(b_0)} c(b, \frac{\epsilon}{1-\eta})$. □

Proof of Proposition 1. We have $|C_{\mathcal{R}_\epsilon}(\delta)| \leq |C_{\mathcal{R}}(\delta)|$ since $\mathcal{R}_\epsilon(b_0)$ is a subset of $\mathcal{R}(b_0)$. By Lemma 1, we use a set of $\frac{\epsilon}{1-\eta}$-regions $\{c(b, \frac{\epsilon}{1-\eta})\}$, where $b \in \mathcal{R}_\epsilon(b_0)$, to cover all beliefs in $\mathcal{R}(b_0)$. Thus, $\mathcal{R}_\epsilon(b_0)$ consists of all centers of these $\frac{\epsilon}{1-\eta}$-regions. We can use $C_{\mathcal{R}_\epsilon}(\delta)$ to cover all centers of $\frac{\epsilon}{1-\eta}$-regions. Then, we extend each δ-region in $C_{\mathcal{R}_\epsilon}(\delta)$ into a $(\delta + \frac{\epsilon}{1-\eta})$-region with the same center so as to cover all $\frac{\epsilon}{1-\eta}$-regions. Thus, we get a $(\delta + \frac{\epsilon}{1-\eta})$-cover for $\mathcal{R}(b_0)$ with size $|C_{\mathcal{R}_\epsilon}(\delta)|$. So, $|C_{\mathcal{R}}(\delta + \frac{\epsilon}{1-\eta})| \leq |C_{\mathcal{R}_\epsilon}(\delta)|$. □

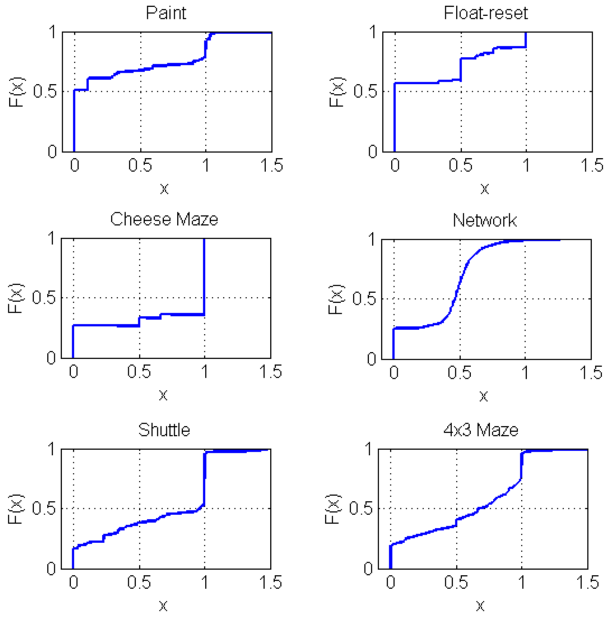

Figure 1: Empirical cumulative distribution functions $F(x)$, where $x = \frac{||b_1^{a,z} - b_2^{a,z}||}{||b_1 - b_2||}$, on tested problems.

Proof of Lemma 2. We assume $b = b' + \zeta$. Then, we have $\sum_{s\in S} |\zeta(s)| \leq ||b - b'|| \leq \delta$. Following Equation 2, we have $P(z|b, a) = P(z|b' + \zeta, a) = P(z|b', a) + \sum_{s\in S} \zeta(s) \sum_{s'\in S} T(s, a, s')\Omega(a, s', z)$. Since $\sum_{s'\in S} T(s, a, s') = 1$ and $\Omega(a, s', z) \leq 1$ for all $s' \in S$, we have $\sum_{s'\in S} T(s, a, s')\Omega(a, s', z) \leq 1$. So, $|P(z|b, a) - P(z|b', a)| \leq |\sum_{s\in S} \zeta(s)| \leq \sum_{s\in S} |\zeta(s)| \leq \delta$. □

Proof of Lemma 3. Let $\mathbf{E}_{x_1\sim X_1,\ldots,x_n\sim X_n} f(x_1, \ldots, x_n)$ denote $\frac{1}{\prod_{i=1}^n |X_i|} \sum_{x_1\in X_1} \cdots \sum_{x_n\in X_n} f(x_1, \ldots, x_n)$. The contraction property that beliefs in the reachable belief space conform to in the expected sense can be formalized as $\mathbf{E}_{b_1\sim\mathcal{R}(b_0),b_2\sim\mathcal{R}(b_0),a\sim A}[||b_1^{a,z} - b_2^{a,z}||] = \eta\mathbf{E}_{b_1\sim\mathcal{R}(b_0),b_2\sim\mathcal{R}(b_0)}[||b_1 - b_2||]$, where $0 \leq \eta < 1$. Thus, we have $\mathbf{E}_{b_1\sim\mathcal{R}(b_0),b_2\sim\mathcal{R}(b_0),t_n\sim T_n}[\frac{||b_1^{t_n} - b_2^{t_n}||}{||b_1 - b_2||}] = \eta^n$, where t_n is an n-step action-observation sequence and T_n is the set of all possible n-step action-observation sequences. Then, we have $P(\frac{||b_1^{t_n} - b_2^{t_n}||}{||b_1 - b_2||} \leq 1) = 1 - \eta^n$, and therefore, $\lim_{n\to+\infty} P(\frac{||b_1^{t_n} - b_2^{t_n}||}{||b_1 - b_2||} \leq 1) = 1$.

Now, we use empirical results on tested problems to further support this lemma (see Figure 1). For each problem, we first randomly generated a 10^5-step action-observation trajectory starting at b_0, and put all successive belief points into a belief set. Then, we randomly selected two different b_1 and b_2 from the set, randomly chose an action a from A, and obtained z from Z based on b_1's observation probability $P(z|b_1, a)$ 10^4 times. (We threw out b_1 and b_2 whenever $P(z|b_2, a) = 0$.) For each pair, we computed $\frac{||b_1^{a,z} - b_2^{a,z}||}{||b_1 - b_2||}$. The curves in Figure 1 represent the empirical cumulative distribution functions $F(x)$, where $x = \frac{||b_1^{a,z} - b_2^{a,z}||}{||b_1 - b_2||}$, on tested problems. As they show, all $F(1)$s are very close to one. Furthermore, $P(\frac{||b_1^{t_n} - b_2^{t_n}||}{||b_1 - b_2||} \leq 1)$ is almost one even when n is a small positive integer. □

Proof of Proposition 2. The POMDP learning algorithm first builds $k = |C_{\mathcal{R}}(\delta)|^{|A||Z|+1}$ AND/OR graphs along with the option of where b_0 belongs. By Lemma 2, all belief points b in a δ-region c satisfy $|P(z|b, a) - P(z|b_c, a)| \leq \delta$. By Lemma 3, the graph describing the δ-accurate mapping between successive belief points and δ-regions exists in these AND/OR graphs. The algorithm uses the adaptive k-meteorologists algorithm to estimate observation probabilities for each graph simultaneously and only makes predictions when there has been enough data to estimate observation probabilities precisely (Diuk, Li, and Leffler 2009). By replacing the "I don't know" predictions with arbitrary outputs, the algorithm is guaranteed to make δ-accurate (and therefore $\frac{\varepsilon}{2}$-accurate) predictions with probability at least $1 - \alpha$ for all but a small number of them. Using the upper sample-complexity bound in the adaptive k-meteorologists algorithm, we have an upper bound of the number of erroneous predictions, and therefore accumulated prediction error can be shown to be at most $(\frac{\varepsilon}{2})^2 L + \mathcal{O}(\frac{k}{(\varepsilon/2)^2} \ln \frac{k}{\alpha} + \sum_{i=1}^{k} \mathcal{O}(\frac{1}{(\varepsilon/16)^2} \ln \frac{k+1}{\alpha})) = (\frac{\varepsilon}{2})^2 L + \mathcal{O}(\frac{k}{\varepsilon^2} \ln \frac{k}{\alpha})$, where the first term is used as a loose total error upper bound of the $\frac{\epsilon}{2}$-accurate predictions, and the second term originates from Theorem 1 in Diuk, Li, and Leffler (2009). When $L = \mathcal{O}(\frac{k}{\varepsilon^3} \ln \frac{k}{\alpha})$, we have $\frac{(\frac{\varepsilon}{2})^2 L + \mathcal{O}(\frac{k}{\varepsilon^2} \ln \frac{k}{\alpha})}{L} = \mathcal{O}(\varepsilon)$, and therefore, can guarantee that the average one-step prediction error is no more than ε. □

References

Diuk, C.; Li, L.; and Leffler, B. R. 2009. The adaptive k-meteorologists problem and its application to structure learning and feature selection in reinforcement learning. In *Proceedings of International Conference on Machine Learning (ICML-2009)*, 249–256.

Even-Dar, E.; Kakade, S. M.; and Mansour, Y. 2005. Reinforcement learning in POMDPs without resets. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI-2005)*, 690–695.

Hochbaum, D. S. 1996. Approximating covering and packing problems: Set cover, vertex cover, independent set, and related problems. In *Approximation algorithms for NP-hard problems*, 94–143.

Hsu, D.; Lee, W. S.; and Rong, N. 2007. What makes some POMDP problems easy to approximate. In *Proceedings of Advances in Neural Information Processing Systems (NIPS-2007)*.

James, M. R., and Singh, S. 2004. Learning and discovery of predictive state representations in dynamical systems with reset. In *Proceedings of International Conference on Machine Learning (ICML-2004)*, 417–424.

Kaelbling, L. P.; Littman, M. L.; and Cassandra, A. R. 1998. Planning and acting in partially observable stochastic domains. *Artificial Intelligence* 101(1-2):99–134.

Kakade, S.; Kearns, M.; and Langford, J. 2003. Exploration in metric state spaces. In *Proceedings of International Conference on Machine Learning (ICML-2003)*, 206–312.

Kurniawati, H.; Hsu, D.; and Lee, W. S. 2008. SARSOP: Efficient point-based POMDP planning by approximating optimally reachable belief spaces. In *Proceedings of Robotics: Science and Systems*.

Lim, Z. W.; Hsu, D.; and Lee, W. S. 2011. Monte-Carlo planning in large POMDPs. In *Proceedings of Advances in Neural Information Processing Systems (NIPS-2011)*.

Littman, M. L.; Cassandra, A. R.; and Kaelbling, L. P. 1995. Learning policies for partially observable environments: Scaling up. In *Proceedings of International Conference on Machine Learning (ICML-1995)*, 362–370.

Littman, M. L.; Sutton, R. S.; and Singh, S. 2002. Predictive representations of state. In *Proceedings of Advances in Neural Information Processing Systems (NIPS-2002)*, 1555–1561.

Littman, M. L. 1994. The witness algorithm: Solving partially observable Markov decision processes. In *Technical Report CS-94-40, Brown University, Department of Computer Science, Providence, RI*.

Manning, C. D.; Raghavan, P.; and Schütze, H. 2008. Hierarchical clustering. In *Introduction to Information Retrieval*, 382–388. Cambridge University Press.

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

Silver, D., and Veness, J. 2010. Monte-Carlo planning in large POMDPs. In *Proceedings of Advances in Neural Information Processing Systems (NIPS-2010)*, 2164–2172.

Singh, S.; Littman, M. L.; Jong, N. K.; Pardoe, D.; and Stone, P. 2003. Learning predictive state representations. In *Proceedings of International Conference on Machine Learning (ICML-2003)*, 712–719.

Smith, T., and Simmons, R. 2005. Point-based POMDP algorithms: Improved analysis and implementation. In *Proceedings of Conference on Uncertainty in Artificial Intelligence (UAI-2005)*, 542–547.

Smith, T. 2007. *Probabilistic planning for robotic exploration*. Ph.D. Dissertation, The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA.

Xu, R., and Wunsch, D. 2005. Survey of clustering algorithms. *IEEE Transactions on Neural Networks* 16(3):645–678.