

Exploration in POMDP belief space and its impact on value iteration approximation

Masoumeh T. Izadi¹, Doina Precup¹

Abstract. Decision making under uncertainty is among the most challenging tasks in the artificial intelligence. Although solution methods to this class of problems are intractable in general, some promising approximation methods have been proposed recently. In particular, point-based planning algorithms for solving partially observable Markov decision processes (POMDPs) have demonstrated that a good approximation of the value function can be obtained by interpolating between the values of a selected set of points. The agent must make a choice as to how to sample these points. Ideally, we need to sample in order to build an accurate approximation in less time. In this paper, we relate this problem to the exploration-exploitation tradeoff in the space of POMDP reachable beliefs. Furthermore, we show that there exists an influential control parameter for this tradeoff. As a result, we provide a controllable tighter bound for the point-based value iteration (PBVI) approximation [4] based on knowledge about the domain. We study two criteria designed to improve point-based value iteration algorithms when selecting candidate points. The first is based on reachability analysis from the given initial belief state. The second criterion is based on the degree of stochasticity of the problem domain and the topological structure of possible beliefs experienced by the agent. We present an empirical evaluation illustrating the effect of these criteria on the performance of point-based value iteration.

1 Introduction

Partially Observable Markov Decision Processes (POMDPs) provide a standard framework for studying decision making under uncertainty. In a POMDP, the state of the system in which the decisions take place is never fully observed. Only observations that depend probabilistically on the hidden state are available. POMDPs have gained a lot of attention in the AI and operations research community and several planning algorithms have been developed. However, the best exact algorithms for POMDPs can be very inefficient in terms of both space and time requirements. Therefore a huge research effort has been devoted to developing approximation techniques in this field.

Most planning algorithms attempt to estimate values for belief states, i.e. probability distributions over the hidden states of the system. Recent research has been devoted to algorithms that take advantage of the fact that for most POMDP problems, a large part of the belief space is never experienced by the agent. Such approaches, which are known as point-based methods, consider only a finite set of belief points and compute values for the different actions only for these points. The generalization over the entire simplex is done based

on the assumption that “nearby” points (in terms of the L1 norm) will have close values. This assumption is based on the fact that the optimal value function is a piecewise linear and convex function over the continuous belief space. Point-based value iteration methods have been used very successfully in solving problems which are orders of magnitude larger than classical POMDP problems. This algorithm performs point-based updates on a small set $B = \{b_0, b_1, \dots, b_m\}$ of reachable points. The error of the approximation is proved to be bounded and it can be decreased by expanding the set of beliefs. However, value improvement depends to a large extent on which belief points are added to this set. Hence, the choice of belief points is a crucial problem in point-based value iteration, especially when dealing with large problems, and has been discussed by several authors. Spaan and Vlassis [8] explored the use of a large set of randomly generated reachable points. Pineau et al. discussed several heuristics for sampling reachable belief states. Smith and Simmons [5] designed a heuristic search value iteration algorithm which maintains an upper and lower bound on the value function to guide the search for good beliefs.

In this paper we address the issue of dynamically generating, in an efficient way, a good ordering of beliefs that should be considered. We explore the point-based value iteration algorithm in combination with a number of belief point selection heuristics. First, we make some corrections to the reachability metric proposed by Smith and Simmons [6] which were discovered via private communications with the authors. This metric is designed to give more priority of being selected to points that are reachable in the near future. The intuition is that in discounted reward problems, belief points that are only reachable in many time steps do not play an important part in the computation of the value function approximation and we can ignore them. We compare this metric to the 1-norm distance metric previously suggested by Pineau et al [4] and study the applicability of this metric to the point-based value iteration algorithm.

Our study also points out the fundamental exploration versus exploitation dilemma that appears throughout decision theory, in the context of sampling the reachable beliefs for which to do value backups. We propose and investigate a new strategy for point selection in PBVI. The main idea is to give priority to beliefs that are reachable in closer future while still considering the distance of the candidate point to the current set B . This way, we avoid the overwhelming complexity of considering all reachable belief states in a breadth-first manner, but at the same time we try to pick points that can provide a better approximation. This is motivated by the observation that the complexity of the optimal value function can be inferred up to some extent from the difference between the number of belief states being backed up, $|B_i|$, and the number of alpha vectors representing the current approximate value function, $|\Gamma_i|$. Whenever this difference

¹ McGill University Montreal, Quebec, Canada email: mtabae@cs.mcgill.ca, dprecup@cs.mcgill.ca

is large, a lot of points share the same optimal policy. Therefore, we can sample points in a more sparsely, imposing a larger threshold on their distance to the current set of beliefs. A small (or zero) difference between $|B_i|$ and $|\Gamma_i|$ indicates that for a more accurate approximation we need to sample more densely from the space of reachable beliefs, because different beliefs have different optimal policies.

We provide empirical results comparing these two approaches on a set of standard POMDPs. The results suggest that the second method is the winning approach.

2 Background on POMDPs

Formally, a POMDP is defined by the following components: a finite set of hidden states S ; a finite set of actions A ; a finite set of observations Z ; a transition function $T : S \times A \times S \rightarrow [0, 1]$, such that $T(s, a, s')$ is the probability that the agent will end up in state s' after taking action a in state s ; an observation function $O : A \times S \times Z \rightarrow [0, 1]$, such that $O(a, s', z)$ gives the probability that the agent receives observation z after taking action a and getting to state s' ; an initial belief state b_0 , which is a probability distribution over the set of hidden states S ; and a reward function $R : S \times A \times S \rightarrow \mathbb{R}$, such that $R(s, a, s')$ is the immediate reward received when the agent takes action a in hidden state s and ends up in state s' . Additionally, there can be a discount factor, $\gamma \in (0, 1)$, which is used to weigh less rewards received farther into the future.

The goal of planning in a POMDP environment is to find a way of choosing actions, or policy π , which maximizes the expected sum of future rewards

$$V^\pi(b) = E \left[\sum_{t=0}^T \gamma^t r_{t+1} | b, \pi \right] \quad (1)$$

where T is the number of time steps in an episode (typically assumed finite) and r_{t+1} denotes the reward received at time step $t + 1$. The agent in a POMDP does not have knowledge of the hidden states, it only perceives the world through noisy observations as defined by the observation function O . Hence, the agent must keep a complete history of its actions and observations, or a sufficient statistic of this history, in order to act optimally. The sufficient statistic in a POMDP is the belief state b , which is a vector of length $|S|$ specifying a probability distribution over hidden states. The elements of this vector, $b(i)$, specify the conditional probability of the agent being in state s_i , given the initial belief b_0 and the history (sequence of actions and observations) experienced so far.

After taking action a and receiving observation z , the agent updates its belief state using Bayes' Rule:

$$b'_{b a z}(s') = P(s' | b, a, z) = \frac{O(a, s', z) \sum_{s \in S} b(s) T(s, a, s')}{P(z | a, b)} \quad (2)$$

where denominator is a normalizing constant and is given by the sum of the numerator over all values of $s' \in S$:

$$P(z | a, b) = \sum_{s \in S} b(s) \sum_{s' \in S} T(s, a, s') O(a, s', z)$$

We can transform a POMDP into a "belief state MDP" [1]. Under this transformation, the belief state b becomes the (continuous) state of the MDP. The actions of the belief MDP are the same as in the original POMDP, but the transition and reward functions are transformed appropriately, yielding the following form of Bellman opti-

mality equation for computing the optimal value function, V^* :

$$V^*(b) = \max_{a \in A} \sum_{z \in Z} P(z | a, b) * \left[\sum_{s \in S} b(s) \left(\sum_{s'} b'_{b a z}(s') R(s, a, s') \right) + \gamma V^*(b'_{b a z}) \right]$$

where $b'_{b a z}$ is the unique belief state computed based on b , a and z , as in equation (2). As in MDPs, the optimal policy that the agent is trying to learn is greedy with respect to this optimal value function. The problem here is that there is an infinite number of belief states b , so solving this equation exactly is challenging.

Exact solution methods for POMDPs take advantage of the fact that value functions for belief MDPs are piecewise-linear and convex, and thus can be represented using a finite number of hyperplanes in the space of beliefs [7]. Value iteration updates can be performed directly on these hyperplanes. Unfortunately, exact value iteration is intractable for most POMDP problems with more than a few states, because the number of hyperplanes defining the value function can grow exponentially with each step. For any fixed horizon n , the value function can be represented using a set of α -vectors. The value function is the upper bound over all the α -vectors: $V_n(b) = \max_{\alpha} \sum_{s \in S} \alpha(s) b(s)$. Given V_{n-1} , V_n can be obtained using the following backup operator:

$$V_n(b) = \max_{a \in A} \sum_{z \in Z} P(z | a, b) \sum_{s \in S} \left(\sum_{s' \in S} b(s) b'_{b a z}(s') R(s, a, s') \right) + \gamma \max_{\alpha_{n-1}} \sum_{s' \in S} b'_{b a z}(s') \alpha_{n-1}(s')$$

where α_{n-1} are the α -vectors used to represent V_{n-1} .

Exact value iteration algorithms, e.g. [7, 1, 9] perform this backup by manipulating directly the α -vectors, using set projection and pruning operations. Although many α -vectors can usually be pruned without affecting the values, this approach is still prohibitive for large tasks. Approximate methods attempt instead to approximate the value function in some way. These solution methods usually rely on maintaining hyperplanes only for a subset of the belief simplex. Different methods use different heuristics in order to define which belief points are of interest, e.g., [3, 4, 5].

3 Point Based Value Iteration

The computational inefficiency of exact value updates leads to the exploration of various approximation methods that can provide good control solutions with less computational effort. Point-based value iteration (PBVI) is based on the idea of maintaining values and α -vectors for a selected set of belief points. This approach is designed based on the intuition that much of the belief simplex will not be reachable in general.

The algorithm starts with a set of beliefs and computes α -vectors only for these beliefs. The belief set B can then be expanded, in order to cover more of the belief space. New α -vectors can then be computed for the new belief set, and the algorithm continues.

The update function with a fixed set of belief points B can be expressed as an operator H on the space of value functions, such that $V_{i+1} = H V_i$, as defined above. In order to show the convergence of such algorithms, we need to show that H is a contraction mapping, and that each estimate V_i is an upper bound on the optimal value

function. If both of these conditions hold, the algorithm will converge to a fixed point solution, $\bar{V}^* \geq V^*$.

Given a fixed belief set B , the error over multiple updates in point-based value updates in the i th value function estimate (horizon i), is bounded by:

$$\|V_t^* - V_t^B\| \leq \frac{(R_{max} - R_{min})}{1 - \gamma^2} \epsilon_B \quad (3)$$

where ϵ_B depends on the maximum L_1 distance from any reachable belief to B :

$$\epsilon_B = \max_{b' \in \bar{\Delta}} \min_{b \in B} \|b - b'\|_1 \quad (4)$$

where $\bar{\Delta}$ is the set of all reachable beliefs.

4 Belief Point Selection

In the PBVI algorithm, the selection of the belief points that will be used to represent the value function is done in an anytime fashion, with the goal of covering as densely as possible the set of reachable beliefs. The belief set B is initialized with just the initial belief state, b_0 . Then, the space of reachable beliefs is sampled by a forward simulation, taking one action and receiving one observation. Different heuristics for sampling points have been proposed in [4], but the Stochastic Simulation by Explorative Action heuristic (SSEA) is considered to perform best in general. In this approach, all possible actions at a given belief state in B are considered. One observation is sampled for each action and the new belief states are computed using (2). Then, the algorithm greedily picks the belief state that is farthest away from B , in the sense of the L_1 distance. Hence, the number of points in B at most doubles at each iteration, because at most one extra belief point is added for each existing belief. This heuristic is motivated by (4) and attempts to greedily reduce ϵ_B as quickly as possible. The authors also discuss other approaches for picking belief states, such as picking beliefs randomly (similarly to grid-based methods), or sampling reachable beliefs by using either random or greedy actions. In all of these cases (with the exception of random sampling), the space of reachable beliefs is covered gradually, as the algorithm progresses. This is due to the fact that at each point, the candidate beliefs that are considered are reachable in one time step from the current beliefs. Moreover, because PBVI is greedy in picking the next belief state to add, it can potentially overlook, at least for a few iterations, belief states that are important from the point of view of estimating the value function. Spaan and Vlassis [8] propose a different way of choosing belief points in the PERSEUS algorithm, which is aimed at addressing this problem. They sample a large set of reachable beliefs B during a random walk, but then only update a subset of B , which is sufficient to improve the value function estimate overall. The core belief selection heuristic proposed by Izadi et al. [2] tries to start a point-based value iteration with a set of beliefs which span the whole simplex of reachable beliefs. Although this will help the algorithm converge to a good approximation in only a few iterations, finding this set of desired beliefs is computationally demanding, and the approach is problematic for large problems. Heuristic search value iteration (HSVI) [5] keeps the value function bounded between an upper bound and a lower bound, an approach aimed at ensuring good performance for the candidate control policies. The algorithm was improved in [6], by designing a tighter bound and much smaller size controllers, which make HSVI better in terms of planning time and values achieved.

4.1 Selecting Belief States Based on Reachability

In order to reason about the space of reachable beliefs, one can consider the initial belief vector, b_0 , and all possible one-step sequences of actions and observations following it. Equation (2) then defines the set of all beliefs reachable in one step. By considering all one-step action-observation sequences that can occur from these beliefs, we can obtain all beliefs reachable from b_0 in two steps, and so on. This will produce a tree rooted at the initial belief state b_0 . The space of reachable beliefs consists of all the nodes of this tree (which is infinite in general, but cut to a finite depth in finite-horizon tasks). The discounted reachability ρ is a mapping from the space of reachable beliefs Δ to the real numbers, defined as: $\rho(b) = \gamma^L$ where L is the length of the shortest sequence of transitions from the initial belief state b_0 to b . This definition implies that

$$\rho(b'_{b_{az}}) \geq \gamma \rho(b) \quad (5)$$

because either $b'_{b_{az}}$ is obtained in one step from b (in which case we have equality), or if not, it may be obtained along a shorter path. Based on the definition of discounted reachability, Smith and Simmons [6] define a generalized sample spacing measure δ_P ($0 \leq P < 1$). Their argument is that they want to give more weight to beliefs that are reachable in the near future, because their values influence the value function estimates more. To do this, they divide the L_1 norm of the beliefs by $(\rho(b))^P$. However, this division actually has an opposite effect, emphasizing more beliefs that are in the distant future. The problem was discovered via private communications with the authors. To correct this problem, the sample spacing measure should be:

$$\delta_P(B) = \max_{b \in \bar{\Delta}} \min_{b' \in B} \|b - b'\|_1 [\rho(b)]^P \quad (6)$$

where P is a parameter in $[0, 1)$. Note that if $P = 0$, we obtain exactly the heuristic described in [4], and given here in equation (4). However, the theoretical development for that case has to be different than the one we will give here. In this case, an equal weight is given to beliefs that are at equal distance to the current set of belief points, regardless how easy or hard they are to reach. The theoretical results and arguments stated in [6] hold with the above definition of $\delta_P(B)$. For clarity, we present these results in here.

First we need to show that the update operator on the selected set of points by the above metric is a contraction mapping. To do this, consider the following weighted norm:

$$\|V - \bar{V}\|_\xi = \max_b |V(b) - \bar{V}(b)| \xi(b) \quad (7)$$

In other words, this is like a max norm but the elements are weighted by weights ξ .

We can show that the following theoretical results hold for this norm. Note that the proofs are very similar to the ones in [4] but with the new metric plugged in; therefore we do not include them here due to the space limit.

Theorem 1 *The exact Bellman update is a contraction mapping under the norm (7) with contraction factor γ^{1-P} .*

As a side note we need to mention that equation (10) in [6] and the contraction factor as stated there should be fixed as well.

The next theorem bounds the error of a policy based on an approximate value function \hat{V} .

Theorem 2 *The expected error introduced by a policy $\hat{\pi}$ induced by an approximate value function \hat{V} , starting at the initial belief b_0 is bounded by:*

$$\frac{2\gamma^{1-p}}{1-\gamma^{1-p}} \|V^* - \hat{V}\|_{\rho^p}$$

Theorem 3 *Let H_B be the update operator applied using only beliefs from set B . Then the error induced by a single application of H_B instead of the true operator H is bounded in ρ^p norm as:*

$$\|HV - H_B V\|_{\rho^p} \leq \frac{(R_{max} - R_{min})\delta_P(B)}{1 - \gamma^{1-p}}$$

Theorem 4 *The error $\|V_t - V_t^B\|$ at any update step t is at most:*

$$\frac{(R_{max} - R_{min})\delta_P(B)}{(1 - \gamma^{1-p})^2}$$

Algorithm 1 Average-norm Belief Expansion (Initial belief set B)

```

for all  $b \in B$  do
  for all  $a \in A$  do
    Sample the current state  $s$  from  $b$ 
    Sample the next state  $s'$  from  $T(s, a, \cdot)$ 
    Sample the next observation  $z$  from  $O(a, s', \cdot)$ 
    Compute the next belief  $b'_{b_{az}}$  reachable from  $b$ 
  end for
   $b^* = \arg \max_{b'_{b_{az}}} \min_{b'' \in B} \|b'_{b_{az}} - b''\|_1 [\rho(b'_{b_{az}})]^p$ 
   $B = B \cup \{b^*\}$ 
end for
return  $B$ 

```

Algorithm 1 presents an approach for expanding the belief set using the reachability heuristic. Note that instead of looking at all reachable beliefs, we just sample, for each belief in the current set, one possible successor. Of course, this algorithm could be changed to take more samples, or to look farther into the future. However, farther lookahead is less likely to matter, because of the weighting used in the heuristic. The drawback of this approach is that after a few cycles the strategy will sample points from a rather restricted set of points and that could lead to smaller and smaller improvements. For instance, for domains with deterministic observations and transitions, this approach gives very similar results to the stochastic simulation with explorative actions (SSEA) heuristic [4], because of the narrow distribution of reachable beliefs. Considering that PBVI converges to a good approximation in just a few expansions, and that the factor γ is usually between 0.75 to 0.99, the effect of $\delta_P(B)$ is not much different, in Algorithm 1, compared to the effect of $\epsilon(B)$ in SSEA-PBVI.

4.2 Topological Distance Selection Criterion

A more extreme alternative to the reachability heuristics is to include *all* the beliefs that are reachable in the near future in the set of belief points. Theoretically, this should provide the best approximation in terms of the weighted norm that we are considering. But in many problems this is not feasible, because the size of the fringe of the belief tree grows exponentially. Besides, not all the beliefs reachable in one time step are good candidates to improve the value function estimate, and including all of these beliefs in the point set B results in many unnecessary value updates or backups. Theoretically, a subset of these points which are farther away from the current set of

beliefs (in the L_1 norm sense) has more potential to provide an improvement to the current approximation, due to the basic intuition that nearby points have nearby values. In order to get only potentially useful points, we consider adding all candidate belief points b' that are reachable for some possible action $a \in A$ from some belief $b \in B$, but only if b' is farther from B than a threshold parameter $e \in [0, 2]$:

$$d(b', B) = \min_{b \in B} \|b - b'\|_1 > e \quad (8)$$

Algorithm 2 summarizes this idea. Note that in this algorithm, the size of B can be more than doubled at every iteration. However, because this approach is expected to add better beliefs at each step, we would expect it to obtain a good approximation in a smaller number of expansions. But we want to ensure that the number of beliefs is also small enough to enable efficient value function backups. Therefore the threshold e should be selected based on two parameters: 1) the stochasticity of the domain (i.e. degree of stochasticity in action transitions and observation emissions), which indicates how dense or sparse the space of reachable beliefs is expected to be; 2) the expected complexity of the value function that is captured in part by the size of current optimal policy. The intuition for the first criterion is that if there is a lot of stochasticity, the optimal value function is likely to be smoother. As a result, fewer beliefs should be sampled for a good approximation and the parameter e can be set higher. The intuition for the second criterion is that if the current optimal policy is much smaller than the current optimal value function, then a coarse approximation of the value function is likely to be sufficient, and again e can be set higher. While the amount of stochasticity can be determined having the model of the world, it is not obvious how to measure the second criterion before computing the optimal value function. We tried different values of e in the empirical study of this approach, in order to evaluate this aspect in more detail.

Algorithm 2 Distance Based Belief Expansion (Initial belief set B , threshold e)

```

for all  $b \in B$  do
  for all  $a \in A$  do
    Sample current state  $s$  from  $b$ 
    Sample next state  $s'$  from  $T(s, a, \cdot)$ 
    for all  $z \in O$  do
      Compute the next belief  $b'_{b_{az}}$  reachable from  $b$ 
      if  $d(b'_{b_{az}}, B) > e$  then  $B = B \cup \{b'_{b_{az}}\}$ 
    end for
  end for
end for
return  $B$ 

```

It should be noted that the PBVI-SSEA heuristic is the most conservative version of the distance-based expansion and it can easily overlook critical points in each expansion by considering only the farthest point from current B . The breath first heuristic for selecting all one-step away points from B can be considered as a special case of the above algorithm in which $e = 0$.

The error bound on PBVI equation (4) is quite loose, and is not usually useful in practice. As we add more points to B , naturally the approximation error is expected to decrease. However, there is still no theoretical guarantee that this happens. Although the point selection method in Algorithm 2 still does not make the decrease in error monotonic from one expansion to the next, it attempts to keep the value ϵ_B as low as e at all times. However, there is no theoretical guarantee on this approach either.

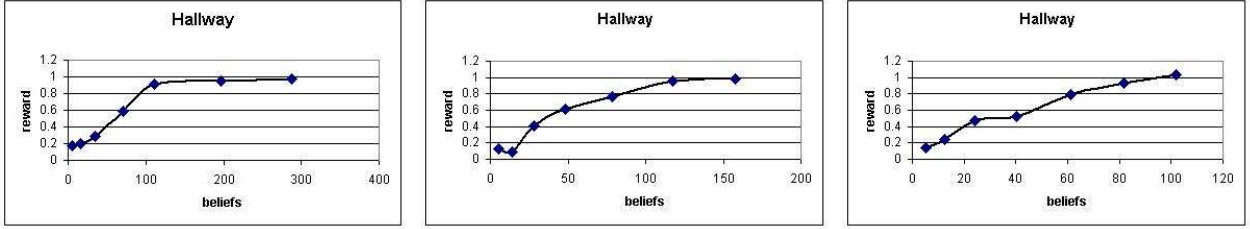


Figure 1. Policy performance for belief points selected by Algorithm 2 with different values of distance threshold in hallway domain.

5 Experiments

In order to compare the performance of the belief selection methods discussed in the previous section, we selected a few standard domains previously used in the literature: Hallway, Hallway2 and RockSample.

We performed 5 iterative expansions of B using Algorithms 1 and 2 and generated the approximate value function based on the resulting point set B . In each domain, we ran 250 trajectories starting from a fixed given initial belief following the approximately optimal policy generated by each method. We measured the discounted sum of the rewards obtained on these trajectories. It should be noted that the rewards here are computed based on following belief states in the trajectory rather than the usual way of following hidden states, therefore, the results might be slightly different from what has been previously reported in the literature. Table 1 shows this measure averaged over 10 independent runs, for the Hallway, Hallway2 and RockSample problems. We present average performance and standard deviation over these runs. The first row in this table shows the standard PBVI algorithm in which Stochastic Simulation with Explorative Action has been used for belief set expansion. For the second algorithm, the parameter P is chosen such that the resulting value function fits best the true value function according to the average-norm heuristic. The second row reports these results with $P = 0.99$; however, we experimented with many different settings of P and all results are very similar. The third, fourth, and fifth rows contain results for the distance-based heuristic with the values of the controlling distance threshold e equal to 0.5, 0.7, 0.9 respectively. In our experiments with Algorithm 2, we do not actually loop through all the observations; instead, the observations are still sampled.

The complexity of the optimal value function can be measured by the number of α -vectors used to represent it. PBVI keeps at most one α -vector for each belief state in the set B . In the domains Hallway and Hallway2, there are 5 possible actions and a high level of stochasticity. In the RockSample domain, actions are deterministic, and there is significantly less noise in the observations.

Table 1. The performance of various point-selection strategies on some POMDP benchmarks

Domain	Hallway	Hallway2	RockSample[4,4]
Method			
PBVI-SSEA	0.51 ± 0.03	0.35 ± 0.03	17.78 ± 1.08
Reachability	0.52 ± 0.03	0.37 ± 0.04	19.36 ± 2.50
$e = 0.5$	0.91 ± 0.09	0.52 ± 0.06	18.33 ± 3.13
$e = 0.7$	0.67 ± 0.10	0.44 ± 0.07	14.63 ± 1.92
$e = 0.9$	0.79 ± 0.05	0.35 ± 0.10	14.76 ± 3.29

◇ PBVI-SSEA □ Reachability metric △ distance threshold

As can be seen, on the Hallway and Hallway2 domains, the topological distance criterion is always better than both SSEA and the reachability criterion, at all levels of e , and the difference is significant. On the RockSample domain, the performance of all three algorithms is very close, at the best parameters level. It is worth noting that RockSample is a deterministic domain.

The set of α -vectors Γ for the Hallway and Hallway2 problems is always roughly equal to the number of beliefs b . This would indicate that the optimal policy is large and we should sample more belief points. Indeed, as seen in the table above, the best quality results are obtained with a smaller value of e . However, in this case, the algorithm can generate too many samples, which can slow down the progression of the algorithm. In order to further illustrate the impact of the threshold parameter e on the number of belief points and on the quality of the obtained solution, we also plot the number of beliefs vs the total reward obtained for different values of e , by running 7 iterative expansions of the belief set. The results are presented in Figure 1 for Hallway and Figure 2 for Hallway2. As can be seen, after 7 expansions, the value $e = 0.9$ provides performance that is very close to that of the other methods, while building a significantly smaller representation. The difference in quality that can be observed after 5 iterations disappears after 7 iterations. One can hypothesize that there is a certain number of points that is needed for a good representation, and that the other points added for lower values of e are not necessary.

In the RockSample domain, only a small proportion of belief points offer distinct alpha vectors. This is mainly due to the fact that the almost deterministic dynamics makes it possible to generalize the approximate optimal policy well enough on the whole space based on a small number of points. This would suggest that higher values of e would work better. However, the results still show large values of e bring worse than small values, after 5 iterations. An interesting observation is that as we change the threshold e to higher values, the difference between $|B|$ and $|\Gamma|$ is decreased. This can be interpreted as showing that more of the points that are added with this parameter are actually useful. A closer examination of the number of beliefs and solution quality for different values of e is provided in Figure 3. It is worth noting that the quality of the solution does not improve uniformly by adding beliefs. We conjecture that this behavior is due to the deterministic nature of the problem, and the fact that we use the currently greedy policy to gather data. Hence, the visited beliefs may not cover the space well enough. We conjecture that a different way of exploring the belief space would help attract better points in B .

Overall, the average-norm heuristic is preferable to the 1-norm, as it uses roughly the same number of belief points but provides a slightly better quality solution. We also tried considering all beliefs which are only one step away from our current set. The number of

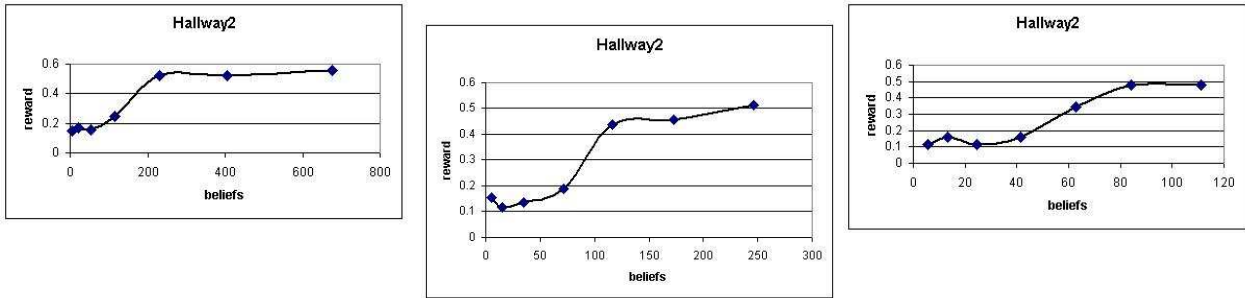


Figure 2. Policy performance for belief points selected by Algorithm 2 with different values of $\epsilon = 0.5$ (left), 0.7 (middle) and 0.9 (right) in Hallway2 domain.

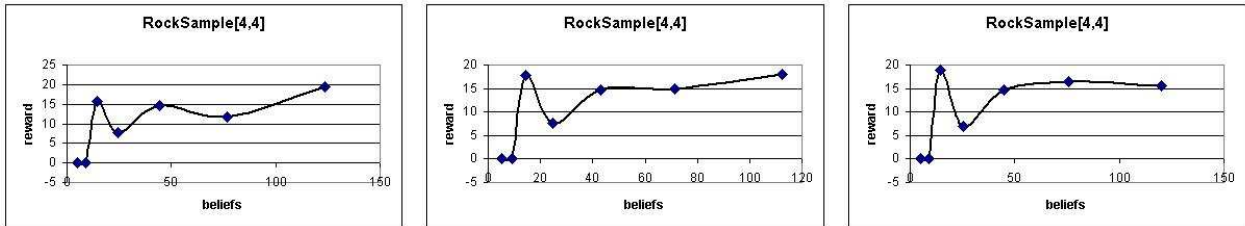


Figure 3. Policy performance for belief points selected by Algorithm 2 with different values of distance threshold in RockSample[4,4] domain.

backed up belief points is considerably larger in this case, so in principle this can allow a better approximation, although in the examples we studied the control quality does not improve much.

In general the effect of the topological distance approach seems to be quite promising. This method obtains a higher quality solution with comparably smaller number of points. However, it should be noted that the choice of the right threshold is quite important to achieve the beneficial results. Our results confirm that the exploration-exploitation trade-off can impact significantly the quality of the solutions obtained.

6 Conclusions and Future Work

The set of points selected for value iteration in point-based methods is very important for the quality of the computed approximate plan. In this paper, we introduced and evaluated two point selection criteria for point-based POMDP approximation methods. First, we studied the reachability metric as an alternative to 1-norm distance between belief states. This approach gives a higher priority to beliefs in the immediate future. Second, we proposed to further restrict the beliefs considered using the distance to the current set of beliefs. This approach seems to have an advantage in terms of the number of beliefs recruited and the quality of the solution obtained. In future, we intend to test this approach on a wider class of problems. Combining the topological distance with the value estimates, in order to provide a better selection of the points that reduce estimation error is another interesting direction for future study.

Acknowledgments

This research was supported in part by funding from NSERC and CFI. The authors wish to thank Joelle Pineau for very helpful discussions.

REFERENCES

- [1] A. R. Cassandra, M. L. Littman, and L. P. Kaelbling. A simple, fast, exact methods for partially observable Markov decision processes. In *Proceedings of Uncertainty in Artificial Intelligence (UAI 97)*, pages 54-61, 1997
- [2] M. T. Izadi, A. Rajwade, and D. Precup. Using core beliefs for point-based value iteration. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI05)*, pages 1751-1753, 2005.
- [3] M. Hauskrecht. Value-function approximations for Partially Observable Markov Decision Processes. *Journal of Artificial Intelligence Research*, vol.13, pages 33-94, 2000.
- [4] J. Pineau, G. Gordon, and S. Thrun. Point-based value iteration: An anytime algorithms for POMDPs. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI03)*, pages 1025-1032, 2003.
- [5] T. Smith, and R. Simmons. Heuristic search value iteration for POMDPs. In *Proceedings of Uncertainty in Artificial Intelligence (UAI'04)* pages 520-527, 2004.
- [6] T. Smith, and R. Simmons. Point-based POMDP Algorithm: Improved Analysis and Implementation. In *Proceedings of Uncertainty in Artificial Intelligence (UAI'05)*, 2005.
- [7] E.J. Sondik. The optimal control of Partially Observable Markov Processes. *Ph.D. thesis, Stanford University*, 1971.
- [8] M.T.J. Spaan, and N. Vlassis. Perseus: Randomized point-based value iteration for POMDPs. *Journal of Artificial Intelligence Research*, pages 195-220, 2005.
- [9] N.L. Zhang, and W. Zhang. Speeding up the convergence of value iteration in partially observable Markov decision processes. *Journal of Artificial Intelligence Research*, vol.14, pages 29-51, 2001.