A Variance Analysis for POMDP Policy Evaluation

Mahdi Milani Fard and **Joelle Pineau**
School of Computer Science
McGill University, Montreal, Canada

Peng Sun
Fuqua School of Business
Duke University, Durham, USA

Abstract

Partially Observable Markov Decision Processes have been studied widely as a model for decision making under uncertainty, and a number of methods have been developed to find the solutions for such processes. Such studies often involve calculation of the value function of a specific policy, given a model of the transition and observation probabilities, and the reward. These models can be learned using labeled samples of on-policy trajectories. However, when using empirical models, some bias and variance terms are introduced into the value function as a result of imperfect models. In this paper, we propose a method for estimating the bias and variance of the value function in terms of the statistics of the empirical transition and observation model. Such error terms can be used to meaningfully compare the value of different policies. This is an important result for sequential decision-making, since it will allow us to provide more formal guarantees about the quality of the policies we implement. To evaluate the precision of the proposed method, we provide supporting experiments on problems from the field of robotics and medical decision making.

Introduction

It is common in the context of Markov Decision Processes (MDPs) to calculate the value function of a specific policy, based on some transition and reward model. When the model is not known *a priori*, one can compute an empirical model from some sample on-policy trajectories using a basic frequentist approach and then use this model (along with Bellman's equation) to calculate the value function of the target policy. Using imperfect models however will introduce some error in the estimated value function. As a general practice with learning methods, we might want to know how good this estimate of the value function is, given the error in the estimated models. This can be expressed in terms of bias and variance of the calculated value function.

The variability of the value function may have two different sources. One is the stochastic nature of MDPs (internal variance), and the other is the error due to the use of the imperfect empirical model instead of the true model (parametric variance). Internal variance and its reduction have been studied in several works (Greensmith, Bartlett, & Baxter 2004). Here we are mostly interested in finding the latter.

Mannor *et al.* (2004) showed that when the empirical model is reasonably close to the true model, we can use a second order approximation to calculate these terms in the value function of an MDP. In this paper we extend these ideas to the context of Partially Observable Markov Decision Processes (POMDPs) and derive similar expressions for the bias and variance terms.

This is an important result for the deployment of autonomous decision-making systems in real-world domains since it is well-known that POMDPs are a more realistic model of decision-making than MDPs (because they allow partial state observability). It is worth noting that approximation methods for POMDPs have made large leaps in recent years; and while these approaches consistently assume a perfect model of the domain, in real-world applications, these models must often be estimated from data. The method outlined in this paper can be used to assess when we have gathered sufficient data to have a good estimate of the value function. The method can also be used to assess whether we can confidently select one policy over another. Finally, the method can be used to define classes of equivalent policies. These are useful considerations when developing expert systems, especially for critical applications such as human-robot interaction and medical decision-making.

Background

In this section we define the model and notation that will be used in the following sections.

Partially Observable Markov Decision Process

We consider finite-state, finite-action, discounted reward POMDP (Sondik 1971; Cassandra, Kaelbling, & Littman 1994):

- S: finite set of states
- A: finite set of actions
- Z: finite set of observations
- R_a: $|S|$ dimensional matrix of rewards when selecting action a
- T_a: $|S|\times|S|$ dimensional matrix of transition probabilities when selecting action a

Copyright © 2008, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

- O_a: $|S| \times |Z|$ dimensional matrix of observation probabilities when selecting action a
- γ: discount factor

It is well known that the value function of the optimal policy of a POMDP in the finite horizon is a convex piecewise linear function of the belief state (Sondik 1971). It is often convenient to use a finite-horizon approximation in the infinite horizon case. Thus, we work only with piecewise linear value functions.

Finite State Controller and Value Function

Sondik (1971) points out that an optimal policy for a finite-horizon POMDP can be represented as an acyclic finite-state controller in which each of the machine states represents a linear piece (or the corresponding *alpha vector*) in the piecewise linear value function. The state of the controller is based on the observation history and the action of the agent will only be based on the state of the controller. For deterministic policies, each machine state i issues an action $a(i)$ and then the controller transitions to a new machine state according to the received observation. This finite-state controller is usually represented as a *policy graph*. An example of a policy graph for a POMP dialog manager is shown in Fig 2.

Cassandra, Kaelbling, & Littman (1994) state that dynamic programming algorithms for infinite-horizon POMDPs, such as value iteration, sometimes converge to an optimal piecewise value function that is equivalent to a cyclic finite-state controller. In the case that the optimal value function is not piecewise linear, it is still possible to find an approximate or suboptimal finite-state controller (Poupart & Boutilier 2003).

Given a finite-state controller for a policy, we can extract the value function of the POMDP using a linear system of equations. To extract the ith linear piece of the POMDP value function, we calculate the value of each POMDP state over that linear piece. For each machine state i (corresponding to a linear piece), and each POMDP state s, the value of s over the ith linear piece is:

$$\begin{aligned} v^i(s) &= r(s, a(i)) + \\ &\quad \gamma \sum_{s',z} T_{a(i)}(s, s') O_{a(i)}(s', z) v^{l(i,z)}(s'), \end{aligned}$$

where $r(s, a)$ is the immediate reward and $l(i, z)$ is the next machine state from state i and given observation z (Hansen 1998). We can rewrite the above system of equations in matrix form using the following definitions:

- K: finite set of machine states in the policy graph
- v^k for $k \in K$: $|S|$ dimensional vector of coefficients representing a linear piece in the value function
- V: $|S||K|$ dimensional vector, vertical concatenation of v^k's representing the POMDP value function
- $a(k)$ for $k \in K$: the action associated with machine state k according to the fixed policy
- $r^k = R_{a(k)}$ for $k \in K$: $|S|$ dimensional vector of coefficients representing a linear piece in the piecewise linear immediate reward function
- R: $|S||K|$ dimensional vector, concatenation of r^k's
- T: $|S||K| \times |S||K|$ dimensional block diagonal matrix of $|K| \times |K|$ blocks, with $T_{a(k)}$ as the kth diagonal sub-matrix
- O: $|S||K| \times |Z||S||K|$ dimensional block diagonal matrix of $|K| \times |K|$ blocks. Each diagonal block is a $|S| \times |Z||S|$ block diagonal sub-matrix of $|S| \times |S|$ sub-blocks. Each sub-block is therefore a $|Z|$ dimensional row vector. The kth block, sth sub-block contains the sth row in the $O_{a(k)}$.
- Π: $|Z||S||K| \times |S||K|$ dimensional block matrix of $|K| \times |K|$ blocks. Each block $\Pi_{k_1 k_2}$ is itself a $|Z||S| \times |S|$ block diagonal sub-matrix of $|S| \times |S|$ sub-blocks. Each sub-block is therefore a $|Z|$ dimensional vector. For all s, the zth component of the sth diagonal block of the (k_1, k_2) sub-matrix, $[(\Pi_{k_1 k_2})_s]_z$, is equal to 1 if k_2 is the succeeding index of the machine state when the machine state is k_1 and the observation is z, and 0 otherwise. This matrix represents the transition function $l(i, z)$ of the finite-state controller which are the arcs in the policy graph.

We can write the system of equations representing the value of a policy π in the following matrix form:

$$V^\pi = R + \gamma T O \Pi^\pi V. \tag{1}$$

leading to:

$$V^\pi = (I - \gamma T O \Pi^\pi)^{-1} R. \tag{2}$$

The above equation can be used to calculate the value function of a given policy, if the models for T, O and R are known. This equation is at the core of most policy iteration algorithms for POMDPs (Hansen 1997; 1998), including one of the most recent highly successful approximation method (Ji *et al.* 2007). Thus having confidence intervals over the calculated value function might be of great use in such algorithms.

Model Error

Given a POMDP (as defined in the previous section), a fixed policy and a set of *labeled* on-policy trajectories, one can use a frequentist approach to calculate the models for T, O and R. The assumption of having training data with known labeled states is a strong assumption and in many POMDP domains may not be plausible. However, it is still more practical than the assumption of having exact true models of T, O and R. In the case where EM-type algorithms are used to label the data (Koenig & Simmons 1996), the derivation of the estimates with the above assumption is not exactly correct, but might still provide a useful guide to compare competing policies.

Here we focus on the case in which the model for immediate reward is known, while T and O are estimated from data. The method can be further extended to the case where rewards are also estimated from data.

If action a is used N_i^a times in state s_i, from which there were N_{ij}^a transitions to s_j, we can write down the empirical transition probability from s_i to s_j given action a as:

$$\hat{T}_a(i, j) = \frac{N_{ij}^a}{N_i^a}. \tag{3}$$

A similar method can be used with the observation model. If there were M_i^a transitions to s_i after action a, and z_j was observed in M_{ij}^a of them, the empirical model of observation probabilities would be:

$$\hat{O}_a(i,j) = \frac{M_{ij}^a}{M_i^a}. \quad (4)$$

From these empirical models we can create the T and O models as defined in the previous section.

As our training data has a finite number of samples, and therefore these empirical models are likely to be imperfect, containing error terms $\tilde{T}$ and $\tilde{O}$. We therefore have:

$$\hat{T} = T + \tilde{T}, \quad \hat{O} = O + \tilde{O}. \quad (5)$$

As we used a simple frequentist approach to calculate the empirical models, we can assume independence of errors in the following manner: Different rows in $\hat{T}_a$ and $\hat{O}_a$ are independent from each other, and each row is drawn from a multinomial distribution. Considering statistical properties of the multinomial distribution, we know that the expected errors are zero and independent:

$$\mathbb{E}[\tilde{T}] = \mathbb{E}[\tilde{O}] = \mathbb{E}[\tilde{T}\tilde{O}] = 0. \quad (6)$$

We can write the covariance of the i'th row of $\hat{T}_a$ (denoted $\hat{T}_a^{(i)}$) as:

$$cov(T_a^{(i)}) = \frac{1}{N_i^a}\left(diag(\hat{T}_a^{(i)}) - (\hat{T}_a^{(i)})^T\hat{T}_a^{(i)}\right), \quad (7)$$

where $diag(\hat{T}_a^{(i)})$ is a diagonal matrix with $\hat{T}_a^{(i)}$ along the diagonal. Similarly for $\hat{O}_a^{(i)}$ we have:

$$cov(O_a^{(i)}) = \frac{1}{M_i^a}\left(diag(\hat{O}_a^{(i)}) - (\hat{O}_a^{(i)})^T\hat{O}_a^{(i)}\right). \quad (8)$$

Using the above derivations and the definition of T and O matrices from the previous section, it is straight-forward to calculate the four dimensional covariance matrices of $\tilde{T}$ and $\tilde{O}$ in terms of $cov(T_a^{(i)})$ and $cov(O_a^{(i)})$. With $\tilde{T}$ and $\tilde{O}$ being zero mean variables, the covariance matrices will be:

$$cov(T(i,j), T(k,l)) = \mathbb{E}[\tilde{T}(i,j)\tilde{T}(k,l)], \quad (9)$$

$$cov(O(i,j), O(k,l)) = \mathbb{E}[\tilde{O}(i,j)\tilde{O}(k,l)]. \quad (10)$$

These terms capture the variance in the empirical models. The interesting question that arises is how these errors in the empirical models impact our estimate of the value function.

Calculation of Bias and Variance

If we use the empirical models instead of the true models to calculate the value of a given policy π, we will have:

$$\hat{V}^\pi = (I - \gamma\hat{T}\hat{O}\Pi^\pi)^{-1}R, \quad (11)$$

To simplify the notation, we will drop the π superscript in the later derivations. The above expression can be rewritten as:

$$\hat{V} = (I - \gamma(T + \tilde{T})(O + \tilde{O})\Pi)^{-1}R. \quad (12)$$

Now using Taylor expansion and matrix manipulation (Mannor *et al.* 2007), we can re-write the above expression as:

$$\hat{V} = \sum_{k=0}^{\infty}\gamma^k f_k(\tilde{T}, \tilde{O})R, \quad (13)$$

where

$$X = (I - \gamma TO\Pi)^{-1}, \quad (14)$$

$$f_k(\tilde{T}, \tilde{O}) = (X(\tilde{T}O\Pi + T\tilde{O}\Pi + \tilde{T}\tilde{O}\Pi))^k X. \quad (15)$$

We will use the above derivation to approximate the expectation of the calculated value function:

$$\mathbb{E}[\hat{V}] = \mathbb{E}[\sum_{k=0}^{\infty}\gamma^k f_k(\tilde{T}, \tilde{O})R]. \quad (16)$$

Because the exact expression of the above equation cannot be further simplified, we consider a second order approximation instead. The expectation of the value function then becomes:

$$\mathbb{E}[\hat{V}] = XR + \gamma\mathbb{E}[f_1]R + \gamma^2\mathbb{E}[f_2]R. \quad (17)$$

As $\tilde{O}$ and $\tilde{T}$ are zero mean and independent, $\mathbb{E}[f_1(\tilde{T}, \tilde{O})]$ will be 0. By substituting X, the above expression becomes:

$$\mathbb{E}[\hat{V}] = V + \gamma^2\mathbb{E}[f_2(\tilde{T}, \tilde{O})]R, \quad (18)$$

which shows that the calculated value function is expected to have some non-zero bias term.

Using a similar approximation, we can write down the second moment of value function as:

$$\begin{aligned}\mathbb{E}[\hat{V}\hat{V}^T] &= VV^T + \gamma^2(\mathbb{E}[f_1RR^Tf_1^T]) \quad (19)\\ &\quad +\gamma^2(\mathbb{E}[f_0RR^Tf_2^T]) + \gamma^2(\mathbb{E}[f_2RR^Tf_0^T]).\end{aligned}$$

The covariance matrix will therefore be:

$$\mathbb{E}[\hat{V}\hat{V}^T] - \mathbb{E}[\hat{V}]\mathbb{E}[\hat{V}]^T = \gamma^2(\mathbb{E}[f_1RR^Tf_1^T]). \quad (20)$$

Substituting f_1 with the definition we get:

$$\begin{aligned}cov(\hat{V}) = &\ \gamma^2X\mathbb{E}[\tilde{T}O\Pi VV^T\Pi^TO^T\tilde{T}^T]X^T.\\ &+\gamma^2XT\mathbb{E}[\tilde{O}\Pi VV^T\Pi^T\tilde{O}^T]T^TX^T \quad (21)\end{aligned}$$

We will approximately calculate the above expression by substituting the true models with our empirical models (which is a standard classical approach).

We also require the following lemma:

Lemma 1. *Let Q be an $n \times n$ dimensional matrix:*

$$Q = AXA^T, \quad (22)$$

where A is an $n \times m$ matrix of zero mean random variables and X is a constant matrix of $m \times m$ dimensions. The $ijth$ entry of $\mathbb{E}[Q]$ is equal to:

$$\begin{aligned}E[\sum_{k,l}A_{ik}X_{kl}A_{lj}^T] &= \sum_{k,l}X_{kl}E[A_{ik}A_{jl}]\\ &= \sum_{k,l}X_{kl}cov(A_{ik}, A_{jl}), \quad (23)\end{aligned}$$

which is only dependent on four dimensional covariance of the matrix A.

By applying Lemma 1 to Eqn 21, we can calculate the covariance of the calculated value function using the covari-

ance of $\tilde{T}$ and $\tilde{O}$ defined in the previous section.

In summary, we propose a second order approximation to estimate the expected error in the value function, in terms of the expected error in the empirical models. Using similar calculations, we can also calculate the bias as defined by Eqn 18 (the derivation will appear in a longer version of this paper; in most cases this term is much smaller than the variance).

Experiment and Results

The purpose of this section is two-fold. First, we wish to evaluate the approximations used when deriving our estimate of the variance in the value function. Second we wish to illustrate how the method can be used to compare different policies for a given task.

POMDP dialog manager

We begin by evaluating the method on synthetic data from a human-robot dialog task. The use of POMDP-based dialog managers is well-established (Doshi & Roy 2007; Williams & Young 2006). However, it is often not easy to get training data in human-robot interaction domains. With small training sets, error terms tend to be important. Estimates of the error variance will therefore be helpful to evaluate and compare policies.

Here we focus on a small simulated problem which requires evaluating dialog policies for the purpose of acquiring motion goals from the user. We presume a human operator is instructing an assistive robot to move to one of two locations (e.g. bedroom or bathroom). While the human intent (i.e. the state) is one of these goals, the observation received by the robot (presumably through a speech recognizer) might be incorrect. The robot has the option to ask again to ensure the goal was understood correctly. Note however that the human may change his/her intent (the state) with a small probability. Fig 1 shows a model of the described situation.

In the generative model (used to provide the training data), we assume the probability of a wrong observation is 0.15 and the human might change goals with probability 0.05. If the robot acts as requested, it gets a reward of 10; otherwise it gets a -40 penalty. There is a small penalty of -1 when asking for clarification. We assume $\gamma = 0.95$.

Fig 2 shows a policy graph for the described POMDP dialog manager. This policy graph corresponds to the policy where the robot keeps asking the human until it receives an observation twice more than the other one.

We ran the following experiment: given the fixed policy of Fig 2 and a fixed number n, we draw on-policy labeled

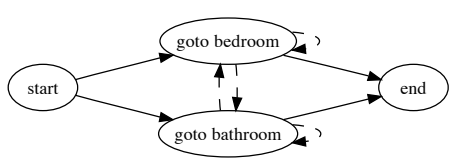

Figure 1: Example of a dialog POMDP - Dashed lines refer to taking action "*ask*"

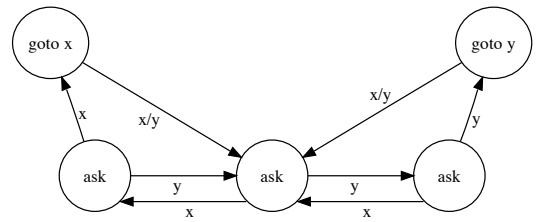

Figure 2: Policy graph for the POMDP dialog manager

trajectories that on the whole contain n transitions. We use these to calculate the empirical models (Eqns 3 and 4), and use Eqn 2 to calculate the value function. Then we use Eqn 21 to calculate the covariance and standard deviation of the value function at the initial belief point ($b_0 = [0.5; 0.5]$).

Let $V(b_0)$ be the expected value at the initial belief state b_0, and let $\alpha = [\alpha_1; \alpha_2]$ be the vector of coefficients describing the corresponding linear piece in the piecewise linear value function. We have $V(b_0) = \mathbb{E}[\alpha \cdot b] = (\alpha_1 + \alpha_2)/2$ and thus the variance of $V(b_0)$ can be calculated as:

$$var(V(b_0)) = \frac{var(\alpha_1) + var(\alpha_1) + 2cov(\alpha_1, \alpha_2)}{4}. \quad (24)$$

Fixing the size of the training set, we run the above experiment 1000 times. In each time, we calculate the empirical value of the initial belief state ($\hat{V}(b_0)$), and estimate its variance using Eqn 24. We then calculate the percentage of cases in which the estimated value ($\hat{V}(b_0)$) lies within 1 and 2 estimated standard deviations of the true value ($V(b_0)$). Assuming that the error between the calculated and true value has a Gaussian distribution (this was confirmed by plotting the histogram of error terms), these values should be 68% and 95% respectively. Fig 3 confirms that the variance estimation we propose satisfies this criteria. The result holds for a variety of sample set sizes (from n=1000 to n=5000).

To investigate how these variance estimates can be useful to compare competing policies, we calculate the variance of the value function for two other policies on this dialog problem (we presume these dialog policies are provided by an

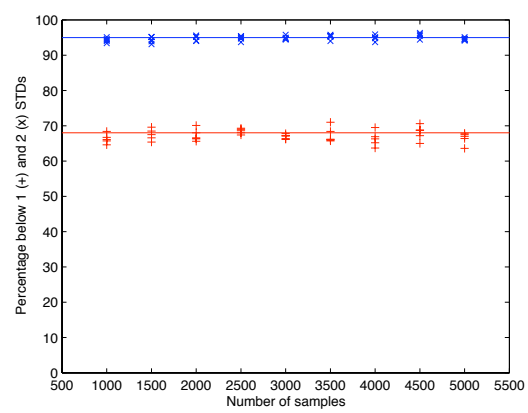

Figure 3: Percentage of the cases in which $\hat{V}(b_0)$ lies within 1 (+) and 2 ($\times$) approximately calculated standard deviations from $V(b_0)$ - the dialog problem

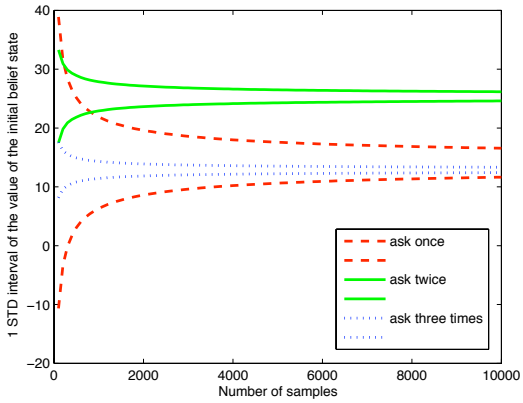

Figure 4: 1 standard deviation interval for the calculated value of the initial belief state for different policies on the dialog problem

expert, though they could be acquired from a policy iteration algorithm such as Ji *et al.* (2007)). One policy is to ask for the goal only once, and then act according to that single observation. The other policy is to keep asking until the robot observes one of the goals three times more than the other one, and then act accordingly. Fig 4 shows the 1 standard deviation interval for the calculated value of the initial belief state as a function of the number of samples, for each of our three policies (including the one shown in Fig 2). Given larger sample sizes, the policy in Fig 2 becomes a clear favorite, whereas the other two are not significantly different from each other. This illustrates how our estimates can be used practically to assess the difference between policies using more information than simply their expected value (as is usually standard in the POMDP literature).

Medical Domain

We now evaluate the accuracy of our approximation in a medical decision-making task involving real data. The data was collected as part of a large (4000+ patients) multi-step randomized clinical trial, designed to investigate the comparative effectiveness of different treatments provided sequentially for patients suffering from depression (Fava *et al.* 2003). The POMDP framework offers a powerful model for optimizing treatment strategies from such data. However given the sensitive nature of the application, as well as the cost involved in collecting such data, estimates of the potential error are highly useful.

The dataset provided includes a large number of measured outcomes, which will be the focus of future investigations. For the current experiment, we focus on a numerical score called the Quick Inventory of Depressive Symptomatology (QIDS), which roughly indicates the level of depression. This score was collected throughout the study in two different ways: a self-report version (QIDS-SR) was collected using an automated phone system; a clinical version (QIDS-C) was also collected by a qualified clinician. For the purposes of our experiment, we presume the QIDS-C score completely describes the patient's *state*, and the QIDS-SR score is a noisy *observation* of the state. To make the

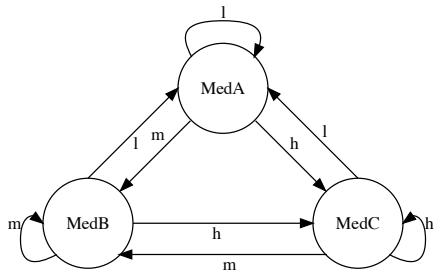

Figure 5: The policy graph for the STAR*D problem

problem tractable with small training data, we discretize the score (which usually ranges from 0 to 27) uniformly according to quantiles into 2 states and 3 observations. The dataset includes information about 4 steps of treatments. We focus on policies which only differ in terms of treatment options in the second step of the sequence (other treatment steps are held constant). There are seven treatment options at that step. A reward of 1 is given if the patient achieves remission (at any step); a reward of 0 is given otherwise.

Although this a relatively small POMDP domain, it is nonetheless an interesting validation for our estimate, since it uses real data, and highlights the type of problem where these estimates are particularly crucial.

We focus on estimating the variance in the value estimate for the policy shown in Fig 5. This policy includes only three treatments: medication A is given to patients with low QIDS-SR scores, medication B is given to patients with medium QIDS-SR scores, and medication C is given to patients with high QIDS-SR scores. Since we do not know the *exact* value of this policy (over an infinitely large data set), we use a bootstrapping estimate. This means we take all the samples in our dataset which are consistent with this policy, and presume that they define the true model and true value function. Now to investigate the accuracy of our variance estimate, we subsample this data set, estimate the corresponding parameters, and calculate the value function using Eqn 2.

To summarize the value function into a single value (de-

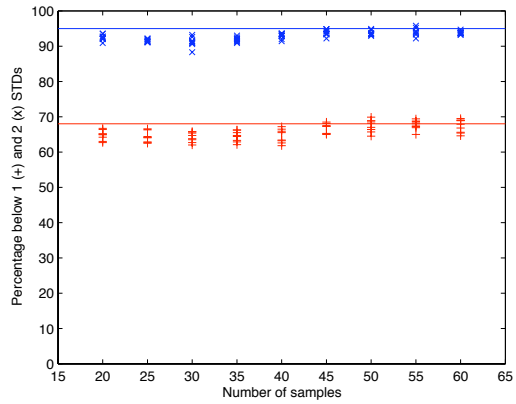

Figure 6: Percentage of cases in which $\hat{V}(B)$ lies within 1 (+) and 2 ($\times$) approximately calculated standard deviations from $V(B)$ - the STAR*D problem

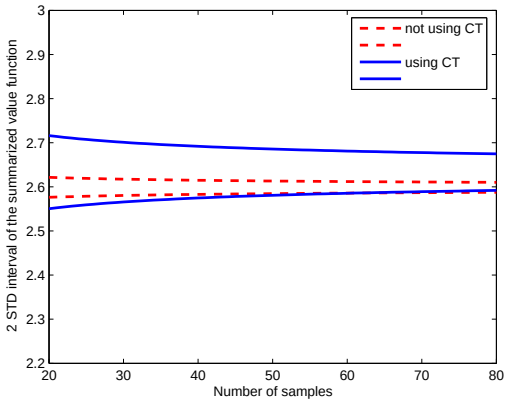

Figure 7: 2 standard deviation interval for the calculated value of the summarized belief state for different policies on the STAR*D problem

noted by $V(B)$), we simply take the average over the 3 linear pieces in the value function. The variance of $\hat{V}(B)$ will therefore be the average of the elements of the covariance matrix we calculated for the value function. To check the quality of the estimates, we calculate the percentage of cases in which the calculated value lies within 1 and 2 standard deviations from the true value. If the error term in the value function has a normal distribution these percentages should again be 68 and 95. Fig 6 shows the mentioned percentages as a function of the number of samples. Here again, the variance estimates are close to what is observed empirically.

Finally, we conducted an experiment to compare policies with different choice of medications in the policy graph of Fig 5. During the STAR*D experiment, patients mostly preferred not to use a certain treatment (CT:Cognitive Therapy). To study the effect of this preference, we compared two policies only one of which uses CT. As shown in Fig 7, the CT-based policy has a slightly better expected value and much higher variation. Using the result of this analysis, one might prefer the non CT-based policy for two reasons: Even with high empirical values, we have small evidence to support the CT-based policy. Moreover, CT is not preferred by most patients. Such method can be applied in similar cases for comparison between an empirically optimal policy and medically preferred ones.

Discussion

Most of the literature on sequential decision-making focuses strictly on the problem of making the *best* possible decision. This paper argues that it is sometimes important to take into account the error in our value function, when comparing alternative policies. In particular, we show that when we use imperfect empirical models generated from sample data (instead of the true model), some bias and variance terms are introduced in the value function of a POMDP. We also present a method to approximately calculate these errors in terms of the statistics of the empirical models.

Such information can be highly valuable when comparing different action selection strategies. During policy search, for instance, one could make use of these error terms to search for policies that have high expected value and low expected variance. Furthermore, in some domains (including human-robot interaction and medical treatment design), where there is an extensive tradition of using hand-crafted policies to select actions, the method we present would be useful to compare hand-crafted policies with the best policy selected by an automated planning method.

The method we presented can be further extended to work in cases where the reward model is also unknown and is approximated by sampling. However, the derived equations are more cumbersome as we need to take into account the potential correlations between reward and transition models.

Acknowledgment

Funding for this work was provided by the National Institutes of Health (grant R21 DA019800) and the NSERC Discovery Grant program.

References

Cassandra, A. R.; Kaelbling, L. P.; and Littman, M. L. 1994. Acting optimally in partially observable stochastic domains. In *Proceedings of AAAI*.

Doshi, F., and Roy, N. 2007. Efficient model learning for dialog management. In *Proceeding of HRI*.

Fava, M.; Rush, A.; Trivedi, M.; Nierenberg, A.; Thase, M.; Sackeim, H.; Quitkin, F.; Wisniewski, S.; Lavori, P.; Rosenbaum, J.; and Kupfer, D. 2003. Background and rationale for the sequenced treatment alternatives to relieve depression (STAR*D) study. *Psychiatr Clin North Am* 26(2):457–94.

Greensmith, E.; Bartlett, P. L.; and Baxter, J. 2004. Variance reduction techniques for gradient estimates in reinforcement learning. *J. Mach. Learn. Res.* 5:1471–1530.

Hansen, E. A. 1997. An improved policy iteration algorithm for partially observable MDPs. In *Proceedings of NIPS*.

Hansen, E. A. 1998. Solving POMDPs by searching in policy space. In *Proceedings of UAI*.

Ji, S.; Parr, R.; Li, H.; Liao, X.; and Carin, L. 2007. Point-based policy iteration. In *Proceedings of AAAI*.

Koenig, S., and Simmons, R. 1996. Unsupervised learning of probabilistic models for robot navigation. In *Proceedings of ICRA*.

Mannor, S.; Simester, D.; Sun, P.; and Tsitsiklis, J. N. 2004. Bias and variance in value function estimation. In *Proceedings of ICML*.

Mannor, S.; Simester, D.; Sun, P.; and Tsitsiklis, J. N. 2007. Bias and variance approximation in value function estimates. *Manage. Sci.* 53(2):308–322.

Poupart, P., and Boutilier, C. 2003. Bounded finite state controllers. In *Proceedings of NIPS*, volume 16.

Sondik, E. J. 1971. *The optimal control of partially observable Markov processes*. Ph.D. Dissertation, Stanford.

Williams, J. D., and Young, S. 2006. Partially observable markov decision processes for spoken dialog systems. *Computer Speech and Language* 21(2).