# A Bayesian Approach to Imitation in Reinforcement Learning

**Bob Price**
**University of British Columbia**
**Vancouver, B.C., Canada V6T 1Z4**
***price@cs.ubc.ca***

**Craig Boutilier**
**University of Toronto**
**Toronto, ON, Canada M5S 3H5**
***cebly@cs.toronto.edu***

## Abstract

In multiagent environments, forms of social learning such as teaching and imitation have been shown to aid the transfer of knowledge from experts to learners in reinforcement learning (RL). We recast the problem of imitation in a Bayesian framework. Our *Bayesian imitation model* allows a learner to smoothly pool prior knowledge, data obtained through interaction with the environment, and information inferred from observations of expert agent behaviors. Our model integrates well with recent Bayesian exploration techniques, and can be readily generalized to new settings.

## 1 Introduction

Reinforcement learning is a flexible, yet computationally challenging paradigm. Recent results demonstrating that under certain assumptions the sample complexity of reinforcement learning is polynomial in the number of problem states [Kearns and Singh, 1998] are tempered by the sober fact that the number of states is generally exponential in the number of the attributes defining a learning problem. With recent interest in building interacting autonomous agents, reinforcement learning is increasingly applied to multiagent tasks, a development which only adds to the complexity of learning [Littman, 1994; Hu and Wellman, 1998]. In this paper, we examine multi-agent reinforcement learning under the assumption that other agents in the environment are not merely arbitrary actors, but actors "like me". That is, the other agents may have similar action capabilities and similar objectives. This assumption radically changes the optimal learning strategy. Information about other agents "like me" can give the learning agent additional information about its *own* capabilities and how these capabilities relate to its *own* objectives. A number of techniques have been developed to exploit this, including *imitation* [Demiris and Hayes, 1999; Mataric, 2002], *learning by watching* [Kuniyoshi *et al.,* 1994], *teaching or programming by demonstration* [Atkeson and Schaal, 1997] *behavioral cloning* [Sammut *et al.,* 1992], and *inverse reinforcement learning* [Ng and Russell, 2000].

Learning by observation of other agents has intuitive appeal; however, explicit communication about action capabilities between agents requires considerable infrastructure: a communication channel, a sufficiently expressive representation language, a transformation between possibly different agent bodies, and an incentive to communicate. In dynamic, competitive domains, such as web-based trading, it is unrealistic to expect all agents to be designed with compatible representations and altruistic intentions. Observation-based techniques, in which the learning agent observes only the *outward* behaviors of another agent, can reduce the need for explicit communication. Implicit communication through passive observations has been implemented as *implicit imitation* [Price and Boutilier, 1999; 2001]. In this model, the effects of other agents' action choices on the state of the environment can be observed, but the internal state of other agents and their action control signals are not observable. Independent exploration on the part of the observer is used to adapt knowledge implicit in observations of other agents to the learning agent's own needs. Unlike classic imitation models, the learner is not required to explicitly duplicate the behavior of other agents.

In this paper, we recast implicit imitation in a Bayesian framework. This new formulation offers several advantages over existing models. First it provides a more principled, and more elegant approach to the smooth pooling of information from the agent's prior beliefs, its own experience and the observations of other agents (e.g., it eliminates the need for certain ad hoc tuning parameters in current imitation models). Second, it integrates well with state-of-the-art exploration techniques, such as Bayesian exploration. Finally, the Bayesian imitation model can be extended readily to partially-observable domains, though the derivation and implementation are considerably more complex and are not reported here.

## 2 Background

We assume a reinforcement learning (RL) agent is learning to control a Markov decision processes (MDP) $\langle S, \mathcal{A}_o, R_o, D \rangle$, with finite state and action sets S, $A_o$, reward function $R_o$ $S \mapsto$ R, and dynamics *D.* The dynamics *D* refers to a set of transition distributions $\Pr(s, a, \cdot)$. The actions $\mathcal{A}_o$ and rewards $R_o$ are subscripted to distinguish them from those of other agents (see below). We assume throughout that the agent knows $R_o$ but not the dynamics *D* of the MDP (thus we adopt the "automatic programming" perspective), and has the objective of maximizing discounted reward over an infinite horizon. Any of a number of RL techniques can be used to learn an optimal policy $\pi : S \mapsto \mathcal{A}_o$. We focus here on *model-*

*based RL* methods. in which the observer maintains an estimated MDP $\langle \mathcal{S}, \mathcal{A}_o, \widehat{R}_o, \widehat{D} \rangle$, based on the set of experiences $\langle s, a, r, t \rangle$ obtained so far. At each stage (or at suitable intervals) this MDP can be solved exactly, or approximately using techniques such as prioritized sweeping [Moore and Atkeson, 1993]. Since $R_0$ is known, we focus on learning dynamics.

Bayesian methods in model-based RL allow agents to incorporate priors and explore optimally. In general, we employ a prior density *P* over possible dynamics D, and update it with each data point (s, a, *t).* Letting $H_{0)} = \langle s_0, s_1, \ldots, s_T \rangle$ denote the *(*current) *state history* of the observer, and $A_0 = \langle a_0, a_1, \ldots, a_{T-1} \rangle$ be the action history, we use the posterior $P(D|H_o, A_o)$ to update the action Q-values, which are used in turn to select actions. The formulation of Dearden *et al* 1999 renders this update tractable by assuming a convenient prior: *P* is the product of local independent densities for each transition distribution $\mathbf{Pr}(s, a, \cdot)$; and each density $P(D^{s,a})$ is a Dirichlet with parameters $\mathbf{n}^{s,a}$. To model $P(D^{s,a})$ we require one parameter $n^{s,a,s'}$ for each possible successor state $s'$. Update of a Dirichlet is straightforward: given prior $P(D^{s,a}; \mathbf{n}^{s,a})$ and data vector $\mathbf{c}^{s,a}$ (where $c_t^{s,a}$ is the number of observed transitions from *s* to *t* under a), the posterior is given by parameters $\mathbf{n}^{s,a} + \mathbf{c}^{s,a}$. Thus the posterior in Eq. 1 can be factored into posteriors over local families:

$$P(D^{s,a}|H_o^{s,a}) = \alpha \, \mathbf{Pr}(H_o^{s,a}|D^{s,a}) P(D^{s,a}) \qquad (1)$$

where $H_o^{s,a}$ is the subset of history composed of transitions from state *s* due to action a, and the updates themselves are simple Dirichlet parameter updates.

The Bayesian approach has several advantages over other approaches to model-based RL. First, it allows the natural incorporation of priors over transition and reward parameters. Second, approximations to optimal Bayesian exploration can take advantage of this approach, and the specific structural assumptions on the prior discussed above [Dearden *et al.*, 1999].

## 3 Bayesian Imitation

In multiagent settings, observations of other agents can be used in addition to prior beliefs and personal experience to improve an agent's model of its environment. These observations can have enormous impact when they provide information to an agent about parts of the state space it has not yet visited. The information can be used to bias exploration towards the most promising regions of state space and thereby reduce exploration costs and speed convergence dramatically.

The flexibility of the Bayesian formulation leads to an elegant and principled mechanism for incorporating these observations into the agent's model updates. Following Price and Boutilier 1999, we assume two agents, a knowledgeable *mentor 77*i and a naive *observer* o, acting simultaneously, but independently, in a fixed environment.[1] Like the observer, the mentor too is controlling an MDP $\langle \mathcal{S}, \mathcal{A}_m, R_m, D \rangle$ with the same underlying state space and dynamics (that is, for any action $a \in \mathcal{A}_o \cap \mathcal{A}_m$, the dynamics are identical). The assumption that the two agents have the same state space is not critical: more important is that there is some analogical mapping between the two [Nehaniv and Dautenhahn, 1998]. Wc assume full observability of the mentor's state space; but we do not assume the observer can identify the actions taken by the mentor—it simply observes state transitions.

[1] We assume that the agents are performing non-interacting tasks.

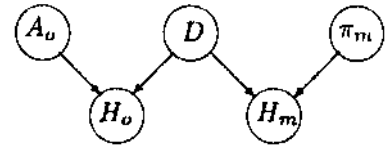


Figure 1: Dependencies among model and evidence sources

We make two additional assumptions regarding the mentor's dynamics: the mentor implements a stationary policy $\pi_m$, which induces a Markov chain $\mathbf{Pr}_m(s, s') = \mathrm{Pr}(s, \pi_m^s, s')$; and for each action $\pi_m^s$ taken by the mentor, there exists an action $a \in \mathcal{A}_o$ such that the distributions $\mathbf{Pr}(\cdot|s, a)$ and $\mathbf{Pr}(\cdot|s, \pi_m^s)$ are the same. This latter assumption is the *homogeneous action assumption* and implies that the observer can duplicate the mentor's policy.[2] As a consequence we can treat the dynamics *D* as the same for both agents. Note that we do not assume that the learner knows *a priori* which of its actions duplicates the mentor's (for any given state s), nor that the observer *wants* to duplicate this policy (as the agents may have different objectives).

Since the learner can observe the mentor's transitions (though not its actions directly), it can form estimates of the mentor's Markov chain, along with estimates of its own MDP (transition probabilities and reward function). In [Price and Boutilier, 1999], this estimate is used to augment the normal Bellman backup, treating the observed distribution Pr(s,.) as a model of an action available to the observer. Imitators using augmented backups based on their observations of a mentor often learn much more quickly, especially if the mentor's reward function or parts of its policy overlap with that of the observer. Techniques like interval estimation [Kaelbling, 1993] can be used to suppress augmented backups where their value has low "confidence."

In the Bayesian approach, the observer incorporates observations of the mentor directly into an *augmented model* of its environment. Let $H_m$ denote the history of mentor state transitions observed by the learner. As above, $H_0$ and $A_0$ represents the observer's state and action history respectively. Figure 1 illustrates the sources of information available to the imitator with which to constrain its beliefs about Z), and their probabilistic dependence. While the observer knows its own action history, $A_0$ it has no direct knowledge of the actions taken by the mentor: at best it may have (often weak) prior knowledge about the mentor's policy $\pi_m$. The learner's beliefs over *D* can then be updated w.r.t. the joint observations:

$$\begin{aligned} P(D|H_o, A_o, H_m) &= \alpha \, \mathrm{Pr}(H_o, H_m|D, A_o) P(D) \\ &= \alpha \, \mathrm{Pr}(H_o|D, A_o) \, \mathrm{Pr}(H_m|D) P(D). \end{aligned} \qquad (2)$$

[2] The homogeneous action assumption can be relaxed [Price and Boutilier, 2001]. Essentially, the observer hypothesizes that violations can be "repaired" using a local search for a short sequence of actions that roughly duplicates a short subsequence of the mentor's actions. If a repair cannot be found, the observer discards the mentor influence (at this point in state space).

We assume that the prior *P(D)* has the factored Dirichlet form described above. Without mentor influence, a learner can maintain its posterior in the same factored form, updating each component of the model $P(D^{s,a})$ independently. Unfortunately, complications arise due to the unobservability of the mentor's actions. We show, however, that the model update in Eq. 2 can still be factored into convenient terms.

We derive a factored update model for $P(D^{s,a})$ describing the dynamics at state *s* under action *a* by considering two cases. In case one, the mentor's unknown action $\pi_m^s$ could be different than the action *a.* In this case, the model factor $D^{s,a}$ would be independent of the mentor's history, and we can employ the standard Bayesian update Eq. 1 without regard for the mentor. In case two, the mentor action $\pi_m^s$ is in fact the same as the observer's action *a.* Then the mentor observations are relevant to the update of $P(D^{s,a})$:

$$\begin{aligned} &P(D^{s,a}|H_o^{s,a}, H_m^s, \pi_m^s = a) \\ &\quad = \alpha \Pr(H_o^{s,a}, H_m^s|D^{s,a}, \pi_m^s = a) P(D^{s,a}|\pi_m^s = a) \\ &\quad = \alpha \Pr(H_o^{s,a}|D^{s,a}) \Pr(H_m^s|D^{s,a}, \pi_m^s = a) P(D^{s,a}). \end{aligned}$$

Let $\mathbf{n}^{s,a}$ be the prior parameter vector for $P(D^{s,a})$, and $\mathbf{c}_o^{s,a}$ denote the counts of observer transitions from state *s* via action a, and $\mathbf{c}_m^s$ the counts of the mentor transitions from state *s.* The posterior *augmented model* factor density $P(D^{s,a}|H_o^{s,a}, H_m^s, \pi_m^s = a)$ is then a Dirichlet with parameters $\mathbf{n}^{s,a} + \mathbf{c}_o^{s,a} + \mathbf{c}_m^s$; that is, we simply update with the sum of the observer and mentor counts:

$$P(D^{s,a}|H_o^{s,a}, H_m^s, \pi_m(s) = a) = P(D^{s,a}; \mathbf{n}^{s,a} + \mathbf{c}_o^{s,a} + \mathbf{c}_m^s).$$

Since the observer does not know the mentor's action we compute the expectation w.r.t. these two cases:

$$\begin{aligned} &P(D^{s,a}|H_o^{s,a}, H_m^s) \\ &\quad = \Pr(\pi_m^s = a|H_o^{s,a}, H_m^s) P(D^{s,a}; \mathbf{n}^{s,a} + \mathbf{c}_o^{s,a} + \mathbf{c}_m^s) \\ &\qquad + \Pr(\pi_m^s \neq a|H_o^{s,a}, H_m^s) P(D^{s,a}; \mathbf{n}^{s,a} + \mathbf{c}_o^{s,a}). \end{aligned} \quad (3)$$

This allows a factored update of the usual conjugate form, but where the mentor counts $\mathbf{c}_m^s$ are distributed across all actions, weighted by the posterior probability that the mentor's policy chooses that action at state *s*.[3]

With a mechanism to calculate the posterior over the mentor's policy, Eq. 3 provides a *complete factored* update rule for incorporating evidence from observed mentors by a Bayesian model-based RL agent. To tackle this last problem—that of updating our beliefs about the mentor's policy—we have:

$$\begin{aligned} &\Pr(\pi_m|H_m, H_o) \\ &\quad = \alpha \Pr(H_m|\pi_m, H_o) \Pr(\pi_m|H_o) \\ &\quad = \alpha \Pr(\pi_m) \int_{D \in \mathcal{D}} \Pr(H_m|\pi_m, D) P(D|H_o). \end{aligned} \quad (4)$$

If we assume that the prior over the mentor's policy is factored in the same way as the prior over models—that is, we have independent distributions $\Pr(\pi_m^s)$ over $A_m$ for each *s*—this update can be factored as well, with history elements at state *s* being the only ones relevant to computing the posterior over $\pi_m(s)$. We still have the difficulty of evaluating the integral over models. Following Dearden *et al.* 1999, we tackle this by sampling models to estimate this quantity. Specifically, we sample models $\dot{D}^{s,a}$ from the factored Dirichlet $P(D^{s,a}|H_o^{s,a})$ over P.[4] Given a specific sample $\dot{D}^{s,a}$, with parameter vector $\dot{\mathbf{n}}^{s,a}$, and observed counts $\mathbf{c}_m^s$, the likelihood of $\dot{D}^{s,a}$ is:

$$\Pr(H_m^s|\pi_m, \dot{D}^{s,a}) = \prod_{t \in S} (n^{s,a,t})^{(c_m^{s,t})}. \quad (5)$$

We can combine the expression for expected model factor probability in Eq. 3 with our expression for mentor policy likelihood in Eq. 5 to obtain a tractable algorithm for updating the observer's beliefs about the dynamics model *D* based on its own experience, and observations of the mentor.[5].

A Bayesian imitator thus proceeds as follows. At each stage, it observes its own state transition and that of the mentor, using each to update its density over models as just described. Efficient methods are used to update the agent's value function. Using this updated value function, it selects a suitable action, executes it, and repeats the cycle.

Like any RL agent, an imitator requires a suitable exploration mechanism. In the *Bayesian exploration* model [Dearden *et al,* 1999], the uncertainty about the effects of actions is captured by a Dirichlet, and is used to estimate a distribution over possible Q-values for each state-action pair.[6] Notions such as value of information can then be used to approximate the optimal exploration policy. This method is computationally demanding, but total reward including reward captured during training is usually much better than that provided by heuristic techniques. Bayesian exploration also eliminates the parameter tuning required by methods like -greedy, and adapts locally and instantly to evidence. These facts makes it a good candidate to combine with imitation.

## 4 Experiments

In this section we attempt to empirically characterize the applicability and expected benefits of Bayesian imitation through several experiments. Using domains from the literature and two unique domains, we compare Bayesian imitation to non-Bayesian imitation [Price and Boutilier, 1999], and to several standard model-based RL (non-imitating) techniques, including Bayesian exploration, prioritized sweeping and complete Bellman backups. We also investigate how Bayesian exploration combines with imitation.

First, we describe the agents used in our experiments. The Oracle employs a fixed policy optimized for each domain,

[3]This assumes that *at least* one of the observer's actions is equivalent to the mentor's, but our model can be generalized to the heterogeneous case. An additional term is required to represent "none of the above".

[4]Sampling is efficient as only one local model needs to be resampled at any time step.

[5]Scaling techniques such as those used in HMM's may be required to prevent underflow in the term $(n^{s,a,t})^{(c_m^{s,t})}$ in Eq. 5.

[6]The Q-value distribution changes very little with each update and can be repaired efficiently using prioritized sweeping. In fact, the Bayesian learner is cheaper to run than a full Bellman backup over all states.

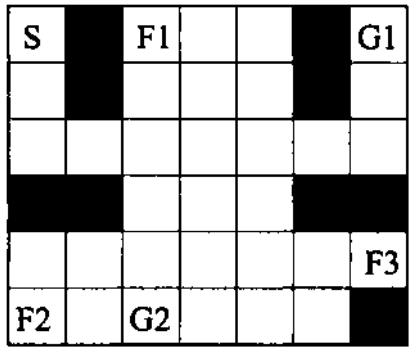


Figure 2: Flagworld Domain

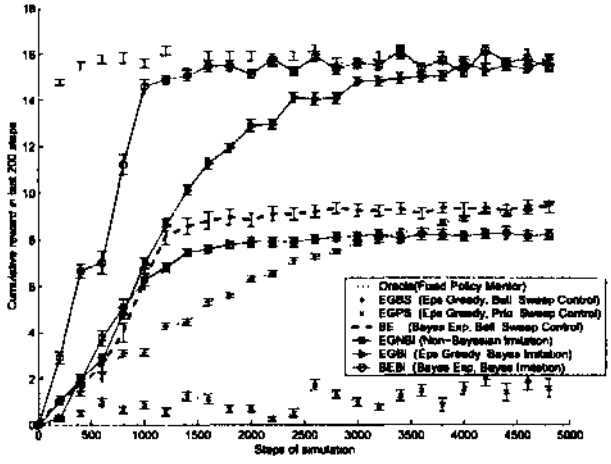


Figure 3: Flag world results (50 runs)

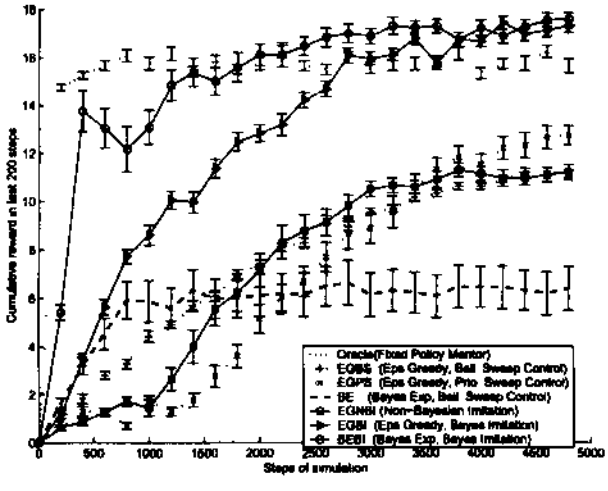


Figure 4: Flag World Moved Goal (50 runs)

providing both a baseline and a source of expert behavior for the observers. The EGBS agent combines $\epsilon$-greedy exploration (EG) with a full Bellman backup (i.e., sweep) at each time step. It provides an example of a generic model-based approach to learning. The EGPS agent is a model-based RL agent, using e-greedy (EG) exploration with prioritized sweeping (PS). EGPS use fewer backups, but applies them where they are predicted to do the most good. EGPS does not have a fixed backup policy, so it can propagate value *multiple* steps across the state space in situations where EGBS would not. The BE agent employs Bayesian exploration (BE) with prioritized sweeping for backups. BEBI combines Bayesian exploration (BE) with Bayesian imitation (Bi). EGBI combines c-greedy exploration (EG) with Bayesian imitation (BI). The EGNB1 agent combines e-greedy exploration with non-Bayesian imitation.

In each experiment, agents begin at the start state. The agents do not interact within the state space. When an agent achieves the goal, it is reset to the beginning. The other agents continue unaffected. Each agent has a fixed number of steps (which may be spread over varying numbers of runs) in each experiment. In each domain, agents are given locally uniform priors (i.e., every action has an equal probability of resulting in any of the local neighbouring states; e.g., in a grid world there are 8 neighbours). Imitators observe the expert oracle agent concurrently with their own exploration. Results are reported as the total reward collected in the last 200 steps. This sliding window integrates the rewards obtained by the agent making it easier to compare performance of various agents. During the first 200 steps, the integration window starts off empty causing the oracle's plot to jump from zero to optimal in the first 200 steps. The Bayesian agents use 5 sampled MDPs for estimating Q-value distributions and 10 samples for estimating the mentor policy from the Dirichlet distribution. Exploration rates for e-greedy agents were tuned for each experimental domain.

Our first test of the agents was on the "Loop" and "Chain" examples (designed to show the benefits of Bayesian exploration), taken from [Dearden *et al,* 1999]. In these experiments, the imitation agents performed more or less identically to the optimal oracle agent and no separation could be seen amongst the imitators.

Using the more challenging "FlagWorld" domain [Dearden *et al.,* 1999], we see meaningful differences in performance amongst the agents. In FlagWorld, shown in Figure 2, the agent starts at state *S* and searches for the goal state G1. The agent may pick up any of three flags by visiting states F1, *F2* and F3. Upon reaching the goal state, the agent receives 1 point for each flag collected. Each action (N,E,S,W) succeeds with probability 0.9 if the corresponding direction is clear, and with probability 0.1 moves the agent perpendicular to the desired direction. Figure 3 shows the reward collected in over the preceding 200 steps for each agent. The Oracle demonstrates optimal performance. The Bayesian imitator using Bayesian exploration (BEBI) achieves the quickest convergence to the optimal solution. The e-greedy Bayesian imitator (EGBI) is next, but is not able to exploit information locally as well as BEBI. The non-Bayesian imitator (EGNBI) does better than the unassisted agents early on but fails

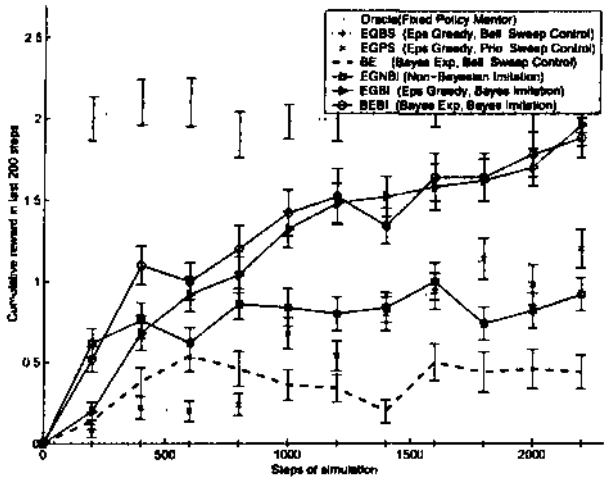


Figure 5: Tutoring domain results (50 runs)

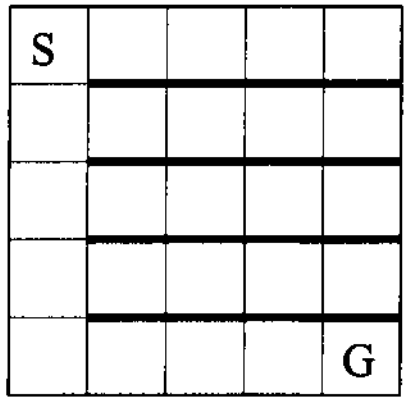


Figure 6: No-south domain

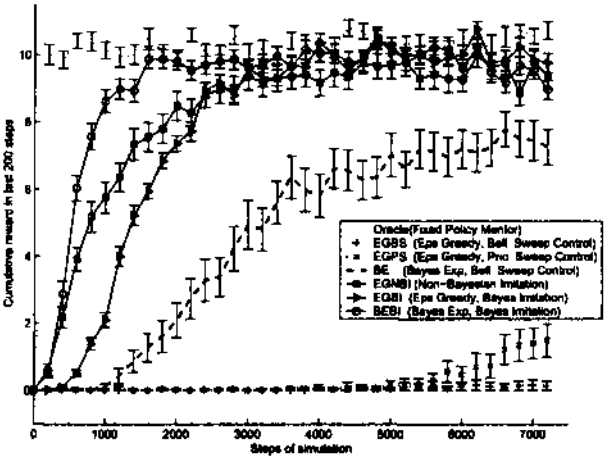


Figure 7: No South results (50 runs)

to find the optimal policy in this domain. A slower exploration rate decay would allow the agent to find the optimal policy, but would also hurt its early performance. The non-imitating Bayesian explorer fares poorly compared to the Bayesian imitators, but outperforms the remaining agents, as it exploits prior knowledge about the connectivity of the domain. The other agents show poor performance (though with high enough exploration rates they would converge eventually). We conclude that Bayesian imitation makes the best use of the information available to the agents, particularly when combined with Bayesian exploration.

We altered the Flag World domain so that the mentor and the learners had different objectives. The goal of the expert Oracle remained at location G1, while the learners had goal location G2 (Figure 2). Figure 4 shows that transfer due to imitation is qualitatively similar to the case with identical rewards. We see that imitation transfer is robust to modest differences in mentor and imitator objectives. This is readily explained by the fact that the mentor's policy provides model information over most states in the domain, which can be employed by the observer to achieve its own goals.

The *tutoring domain* requires agents to schedule the presentation of simple patterns to human learners in order to minimize training time. To simplify our experiments, we have the agents teach a simulated student. The student's performance is modeled by independent, discretely approximated, exponential forgetting curves for each concept. The agent's action will be its choice of concept to present. The agent receives a reward when the student's forgetting rate has been reduced below a predefined threshold for all concepts. Presenting a concept lowers its forgetting rate, leaving it unpresented increases its forgetting rate. Our model is too simple to serve as a realistic cognitive model of a student, but provides a qualitatively different problem to tackle. We note that the action space grows linearly with the number of concepts, and the state space exponentially.

The results presented in Figure 5 are based on the presentation of 5 concepts to a student. (EGBS has been left out as it is time-consuming and generally fares poorly.) We see that all of the imitators learn quickly, but with the Bayesian imitators BEBI and EGBI outperforming EGNBI (which converges to a suboptimal policy).[7] The generic Bayesian agent (BE) also chooses a suboptimal solution (which often occurs in BE agents if its priors prevent adequate exploration). Thus, we see that imitation mitigates one of the drawbacks of Bayesian exploration: mentor observations can be used to overcome misleading priors. We see also that Bayesian imitation can also be applied to practical problems with factored state and action spaces and non-geometric structure.

The next domain provides further insight into the combination of Bayesian imitation and Bayesian exploration. In this grid world (Figure 6), agents can move south only in the first column. In this domain, the optimal Oracle agent proceeds due south to the bottom corner and then east across to the goal. The Bayesian explorer (BE) chooses a path based on its prior beliefs that the space is completely connected. The agent can

Increasing exploration allows EGNBI to find the optimal policy, but further depresses short term performance.

easily be guided down one of the long "tubes" in this scenario, only to have to retrace it steps. The results in this domain, shown in Figure 7, clearly differentiate the early performance of the imitation agents (BEBI, EGB1 and EGNB1) from the Bayesian explorer (BE) and other independent learners. The initial value function constructed from the learner's prior beliefs about the connectivity in the grid world lead it to over-value many of the states that lead to a dead end. This results in a costly misdirection of exploration and poor performance. We see that the ability of the Bayesian imitator BEBI to adapt to the local quality of information allows it to exploit the additional information provided by the mentor more quickly than agents using generic exploration strategies like e-greedy. Again, mentor information is used to great effect to overcome misleading priors.

## 5 Conclusions

Bayesian imitation, like the non-Bayesian implementation of implicit imitation, accelerates reinforcement learning in the presence of other agents with relevant knowledge without requiring either explicit communication with or the cooperation of these other agents. The Bayesian formulation is built on an elegant pooling mechanism which optimally combines prior knowledge, model observations from the imitator's own experience and model observations derived from other agents. The combination of Bayesian imitation with Bayesian exploration eliminates parameter tuning and yields an agent that rapidly exploits mentor observations to reduce exploration and increase exploitation. In addition, imitation often overcomes one of the drawbacks of Bayesian exploration, the possibility of converging to a suboptimal policy due to misleading priors. Bayesian imitation can easily be extended to multiple mentors, and though we did not present the derivation here, it can also be extended to partially observable environments with known state spaces. Though the Bayesian formulation is difficult to implement directly, we have shown that reasonable approximations exist that result in tractable algorithms.

There are several very promising areas of future research that can benefit from the current formulation of Bayesian imitation. One obvious need is to extend the model to the heterogeneous action setting by incorporating the notions of feasibility testing and repair described in [Price and Boutilier, 2001]. We are particularly excited by the prospects of its generalization to richer environmental and interaction models. We have also derived one possible mechanism for using the Bayesian approach in domains with continuous attributes. We hope to extend this work to include methods for discovering correspondences between the state and action spaces of various agents. We also plan to introduce game-theoretic considerations into imitation so that agents can learn solutions to interacting tasks from experts and reason about both the reward-oriented aspects of their action choices as well as the information it reveals to others.

#### Acknowledgements

This research was supported by the Natural Sciences and Engineering Research Council and IRIS.

## References

[Atkcson and Schaal, 1997] C. G. Atkeson and S. Schaal. Robot learning from demonstration. In *Proc. Fourteenth Intl. Conf. on Machine Learning,* pp. 12-20, Nashville, TN, 1997.

[Dcarden *et aL,* 1999] R. Dearden, N. Friedman, and D. Andre. Model-based Bayesian exploration. In *Proc. Fifteenth Conf. on Uncertainty in Artificial Intelligence,* pp. 150-159, Stockholm, 1999.

[Demiris and Hayes, 1999] J. Demins and G. Hayes. Active and passive routes to imitation. In *Proc. AISB '99 Symposium on Imitation in Animals and Artifacts,* pp.81 -87, Edinburgh, 1999.

[Hu and Wellman, 1998] J. Hu and M. P. Wellman. Multiagent reinforcement learning: theoretical framework and an algorithm. In *Proc. Fijthteenth Intl. Conf. on Machine Learning,* pp.242 250, Madison, Wisconsin, 1998.

[Kaelbling, 1993] L. Pack Kaelbling. *Learning in Embedded Systems.* MIT Press, Cambridge,MA, 1993.

[Kearns and Singh, 1998] M. Kearns and S. Singh. Finite sample convergence rates for Q-learning and indirect algorithms. In *Eleventh Conf. on Neural Information Processing Systems,* pp.996 1002, Denver, Colorado, 1998.

[Kuniyoshi *et al,* 1994] Y. Kuniyoshi, M. Inaba, and H. Inoue. Learning by watching: Extracting reusable task knowledge from visual observation of human performance. *IEEE Transactions on Robotics and Automation,* 10(6):799 822, 1994.

[Littman, 1994] M. L. Littman. Markov games as a framework for multi-agent reinforcement learning. In *Proc. Eleventh Intl. Conf. on Machine Learning,* pp. 157-163, New Brunswick, NJ, 1994.

[Mataric, 2002] M.J. Mataric. Visuo-motor primitives as a basis for learning by imitation: Linking perception to action and biology to robotics. In *Imitation in Animals and Artifacts,* pp.392 422, Cambridge, MA, 2002. MIT Press.

[Moore and Atkcson, 1993] A. W. Moore and C. G. Atkeson. Prioritized sweeping: Reinforcement learning with less data and less realtime. *Machine Learning,* 13(1 ):103-30, 1993.

[Nehaniv and Dautenhahn, 1998] C. Nehaniv and K. Dautcnhahn. Mapping between dissimilar bodies: Affordanccs and the algebraic foundations of imitation. In *Proc. Seventh European Workshop on Learning Robots,* pp.64-72, Edinburgh, 1998.

[Ng and Russell, 2000] A. Y. Ng and S. Russell. Algorithms for inverse reinforcement learning. In *Proc. Seventeenth Intl. Conf on Machine Learning,* pp.663-670. Morgan Kaufmann, San Francisco, CA, 2000.

[Price and Boutilier, 1999] B. Price and C. Boutilier. Implicit imitation in multiagent reinforcement learning. In *Proc. Sixteenth Intl. Conf on Machine Learning,* pp.325-334, Bled, SI, 1999.

[Price and Boutilier, 2001] B. Price and C. Boutilier. Imitation and reinforcement learning in agents with heterogeneous actions. In *Proc. Fourteenth Canadian Conf on Artificial Intelligence,* pp. 111-120, Ottawa, 2001.

[Sammut et al., 1992] C. Sammut, S. Hurst, D. Kedzier, and D. Michie. Learning to fly. In *Proc. Ninth Intl. Conf. on Machine Learning,* pp.385 393, Aberdeen, UK, 1992.