A Bayesian Approach to Imitation in Reinforcement Learning

Harshith Srinivas
Paderborn University
harshith@campus.uni-paderborn.de

April 5th, 2022

1 Abstract

The Reinforcement learning problem is decomposed into the following inferences: Firstly, estimating the parameters of a model for the underlying process; Secondly, determining behavior that maximizes return under the estimated model sampled hypothesis. This allows the Bayesian model [15] to always converge towards optimal policy for a stationary process with discrete states. Imitation Learning [3] refers to a problem with mimicking human behavior for a given task. An agent (a learning machine) is trained to perform a task from demonstrations by learning a mapping between observations and actions, with minimal expert knowledge of the tasks.

In a multi-agent systems environment, forms of social learning such as teaching and imitation have been showing how to aid the transfer of knowledge from experts to learners in reinforcement learning (RL). In this paper, the authors have re-visited the problem of the Bayesian Imitation model showing how a learner makes use of prior pool knowledge, data obtained from interaction, and a final set of observations provided by expert agents. Also, the Authors have outlined how their model integrates with the latest Bayesian Exploration techniques and the generalized new settings.

2 Introduction

With an increase in interest in developing interacting autonomous agents, the Reinforcement Learning paradigm is extensively applied to multi-agent tasks because of the flexibility it provides. However[5], shows that the sample complexity of reinforcement learning is polynomial in the number of problem states are tempered by the sober fact that the number of states is generally exponential in the number of the attributes defining a learning problem. Leading to an increase in the complexity of learning in further developments.In this [7] paper, the authors examine multi-agent reinforcement learning by assuming that there are other agents in the environment ('Like-me'), who has similar action capabilities and similar objectives. This 'Like-me' assumption is directly proportional to optimal learning strategy. (that is this assumption provides the learner with extra information about its own capabilities and how they relate to its own objectives.

Several techniques have been developed to exploit this, including imitation [12] , learning by watching [6], teaching or programming by demonstrating [1], behavioral cloning [14], and inverse reinforcement learning [10].

Learning by using any of these approaches on multi-agents has an intuitive appeal. But accordingly, in explicit communication, we need a platform infrastructure comprised of: a communication channel, appropriate expressive language representation, change in actions between possible different agents, and finally an incentive to perform communication.

In dynamic system domains (like web-based trading), it is unrealistic to expect all the agents in the domains to be designed with compatible representations and unselfish 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 was implemented as implicit imitation by [11]. Here, the learner can see the effect of agents' action choices but at the same time, the internal state of other agents and their actions 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, the author summarises Implicit Imitation in a Bayesian Framework which offers several advantages over the existing systems. Like it provides a more principled and keenly intellective approach to the smooth pooling of information from the agent's prior notions, its own experience, and the optical observations of other agents. Withal, It integrates well with state-of-art techniques. Conclusively, the Bayesian Imitation model can be applied to partially-overt domains.

3 Background

Authors in this paper assume Reinforcement Learning (RL) agents to learn Markov decision processes (MDP)

$\langle \mathcal{S}, \mathcal{A}_o, R_o, D \rangle$ where $\mathcal{S}, \mathcal{A}_o$, are finite and action set, reward function $R_o : \mathcal{S} \mapsto \mathbf{R}$, and dynamics D referring to transition distributions $Pr(s, a, \cdot)$ .The actions $\mathcal{A}_o$ and rewards R_o are obtained from other agents. Also, automatic programming is adopted with the belief that R_o is known but not the dynamics D of the MDP and has the motivation to maximize reward ratio over infinite range. Any of the Reinforcement Learning available techniques can be used to learn optimal policy $\pi : \mathcal{S} \mapsto \mathcal{A}_o$ wherein author focus on Model-based RL methods,in which the observer maintains an estimated MDP $\left\langle \mathcal{S}, \mathcal{A}_o, \widehat{R_o}, \widehat{D} \right\rangle$, obtained based on previous experiences $\langle s, a, r, t \rangle$. At suitable intervals, we can solve this MDP using approximation techniques like prioritized sweeping [8]. Since R_o is known, authors focus on learning dynamics.

Bayesian methods in model-based Reinforcement Learning methods generally allow prior density P over possible dynamics D, and update it with each data point $\langle s, a, t \rangle$. Letting $H_o = \langle s_0, s_1, \ldots, s_T \rangle$ denote the (current) state history of the observer, and $A_o = \langle a_0, a_1, \ldots, a_{T-1} \rangle$ as action history. Author make use of the posterior $P(D \mid H_o, A_o)$ to update the action Q-values, which are used in turn to select actions. The [2] renders this update tractable by assuming a convenient prior: P is the product of local independent densities for each transition distribution $Pr(s, a, \cdot)$; 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:

$$P(D^{s,a} \mid H_o^{s,a}) = \alpha Pr(H_o^{s,a} \mid D^{s,a}) P(D^{s,a})$$

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.

Authors in this paper assume Reinforcement Learning (RL) agents to learn Markov decision processes (MDP) $\langle \mathcal{S}, \mathcal{A}_o, R_o, D \rangle$ where $\mathcal{S}, \mathcal{A}_o$, are finite and action set, reward function $R_o : \mathcal{S} \mapsto \mathbf{R}$, and dynamics D referring to transition distributions $Pr(s, a, \cdot)$ .The actions $\mathcal{A}_o$ and rewards R_o are obtained from other agents. Also, automatic programming is adopted with the belief that R_o is known but not the dynamics D of the MDP and has the motivation to maximize reward ratio over infinite range. Any of the Reinforcement Learning available techniques can be used to learn optimal policy $\pi : \mathcal{S} \mapsto \mathcal{A}_o$ wherein author focus on Model-based RL methods,in which the observer maintains an estimated MDP $\left\langle \mathcal{S}, \mathcal{A}_o, \widehat{R_o}, \widehat{D} \right\rangle$, obtained based on previous experiences $\langle s, a, r, t \rangle$. At suitable intervals, we can solve this MDP using approximation techniques like prioritized sweeping [Moore and Atkeson, 1993]. Since R_o is known, authors focus on learning dynamics.

Bayesian methods in model-based Reinforcement Learning methods generally allow prior density P over possible dynamics D, and update it with each data point $\langle s, a, t \rangle$. Letting $H_o = \langle s_0, s_1, \ldots, s_T \rangle$ denote the (current) state history of the observer, and $A_o = \langle a_0, a_1, \ldots, a_{T-1} \rangle$ as action history. Author make use of the posterior $P(D \mid H_o, A_o)$ to update the action Q-values, which are used in turn to select actions. The————— 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 $Pr(s, a, \cdot)$; 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:

$$P(D^{s,a} \mid H_o^{s,a}) = \alpha Pr(H_o^{s,a} \mid D^{s,a}) P(D^{s,a})$$

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.

4 Bayesian Imitation

In a multi-agent system, extra addition of observations with prior beliefs and experiences can enhance the model environment as this addition provides information about the unvisited state. This additional information can be used to handle bias and also improve model complexity (cost and speed).

Based on Price and Boutilier 1999, the authors in this paper assume two agents, a knowledgeable mentor m and a naïve observer O, acting simultaneously, but independently, in a fixed (non-interactive) environment.
Similarly to observer, even mentor too has $MDP \langle \mathcal{S}, \mathcal{A}_m, R_m, D \rangle$ the same underlying state space and identical dynamics.
The assumption that the two agents have same state space is not critical: more important is that there is some analogical mapping between the two [9]. We 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.

Author also makes two additional assumptions, First the mentor implements a stationary policy π_m, which induces a Markov chain $Pr_m(s, s') = Pr(s, \pi_m^s, s')$; Second, for each action π_m^s taken by the mentor, there exists an action $a \in A_o$ such that the distributions $Pr(\cdot \mid s, a)$ and $Pr(\cdot \mid s, \pi_m^s)$ are the same. This later assumption is the homogeneous action [12] and also and implies that the observer can duplicate the mentor's policy. At this stage, we

also know that Dynamics D is same for both the agents. Also author confirms 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).

In [11], the estimate (obtained by learner observing mentor's transitions and reward functions) is used to augment the normal Bellman backup, treating the observed distribution $Pr(s, \cdot)$ as a model of an action available to the observer enabling quick learning especially if the mentor's reward function or components of its policy overlap with that of the observer.Techniques like interval estimation[4] can be used to suppress augmented backups where their value has low "confidence."

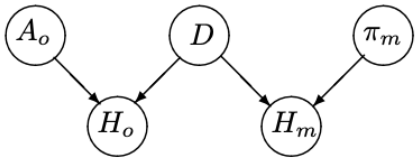

Figure 1: Dependencies among model and evidence sources [13]

In the Bayesian approaches, mentor observations are directly incorporated with these augmented models. Let H_m denote mentor history As mentioned above, H_o and A_o 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 D, and their probabilistic dependence. Also author notes that observer has no direct knowledge of mentor actions: but may have prior knowledge about mentor policy π_m.The learner's beliefs over D can then be updated w.r.t. the joint observations:

$$\begin{aligned} & P\left(D \mid H_o, A_o, H_m\right) \\ & = \alpha Pr\left(H_o, H_m \mid D, A_o\right) P(D) \\ & = \alpha Pr\left(H_o \mid D, A_o\right) Pr\left(H_m \mid D\right) P(D) \end{aligned}$$

At this point, we know that $P(D)$ has the factored Dirichlet form described above. Additionally, Learner can maintain its posterior form by updating each component of the model $P\left(D^{s,a}\right)$ independently (in the same factored version).At this point, we came to know that $P(D)$ has the factored Dirichlet form described above. Additionally, Learner can maintain its posterior form by updating each component of the model $P\left(D^{s,a}\right)$ independently (in the same factored version). The author shows that the model update in Eq. 2 can still be factored into convenient terms, even though there is a complication due to the unobservability of the mentor's actions.

Author here, derives factor model $P\left(D^{s,a}\right)$ under dynamic state s, action a by considering two cases. 1) The mentor's unknown action π_m^s could be different than the action a and apply the standard Bayesian update Eq. 1 without consideration of the mentor. 2) Mentor observations are appropriate to the update $P\left(D^{s,a}\right)$ (because mentor action π_m^s is similar to the observer's action a).

$$\begin{aligned} & P\left(D^{s,a} \mid H_o^{s,a}, H_m^s, \pi_m^s = a\right) \\ = \quad & \alpha Pr\left(H_o^{s,a}, H_m^s \mid D^{s,a}, \pi_m^s = a\right) P\left(D^{s,a} \mid \pi_m^s = a\right) \\ = \quad & \alpha Pr\left(H_o^{s,a} \mid D^{s,a}\right) Pr\left(H_m^s \mid D^{s,a}, \pi_m^s = a\right) P\left(D^{s,a}\right) (Eq.1) \end{aligned}$$

Let $\mathbf{n}^{s,a}$ be the prior parameter vector for $P\left(D^{s,a}\right)$, and $\mathbf{c}_o^{s,a}$ be 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 $P\left(D^{s,a} \mid H_o^{s,a}, H_m^s, \pi_m^s = a\right)$ is then a Dirichlet with parameters $\mathbf{n}^{s,a} + \mathbf{c}_o^{s,a} + \mathbf{c}_m^s$; that is, author updated with the sum of the observer and mentor counts as:

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

And as the observer does not know the mentor's action author computes the expectation in regard to these two (above mentioned) cases as following :

$$\begin{aligned} & P\left(D^{s,a} \mid H_o^{s,a}, H_m^s\right) \\ = & Pr\left(\pi_m^s = a \mid H_o^{s,a}, H_m^s\right) P\left(D^{s,a}; \mathbf{n}^{s,a} + \mathbf{c}_o^{s,a} + \mathbf{c}_m^s\right) \\ & + Pr\left(\pi_m^s \neq a \mid H_o^{s,a}, H_m^s\right) P\left(D^{s,a}; \mathbf{n}^{s,a} + \mathbf{c}_o^{s,a}\right) \end{aligned}$$

Where the mentor counts $\mathbf{c}_m^s$ are distributed across all actions, weighted by the posterior probability that the mentor's policy selects that at least one of the observer's actions is similar to the mentor's actions, but our model in this paper is generalized to the heterogeneous case with an additional term is required to represent "none of the above"

Now to calculate the posterior over the mentor's policy, Eq. 3 delivers a complete factored update rule for integrating evidence from observed mentors by a Bayesian model-based RL agent. To embark on this last problem-that of revamping our beliefs about the mentor's policy-we have following Eq. 4:

$$\begin{aligned} & Pr\left(\pi_m \mid H_m, H_o\right) \\ & = \alpha Pr\left(H_m \mid \pi_m, H_o\right) Pr\left(\pi_m \mid H_o\right) \\ & = \alpha Pr\left(\pi_m\right) \int_{D \in \mathcal{D}} Pr\left(H_m \mid \pi_m, D\right) P\left(D \mid H_o\right) \end{aligned}$$

Now authors consider that the prior over the mentor's policy is factored in the same way as the prior over models (that is we have to factor the update on independent distributions $Pr\left(\pi_m^s\right)$ over $\mathcal{A}_m$ for each $s-$. With history elements at the state s being the only ones pertinent to computing the posterior over $\pi_m(s)$ integrals over the models has to be explored. Following [2], we fine-tune this by sampling models $\dot{D}^{s,a}$ from the factored Dirichlet $P\left(D^{s,a} \mid H_o^{s,a}\right)$

over $\mathcal{D}$.[4] Given a categorical sample $\dot{D}^{s,a}$, with parameter vector $\mathbf{n}^{s,a}$, and observed counts $\mathbf{c}_m^s$, now the likelihood of $\dot{D}^{s,a}$ is:

$$Pr\left(H_m^s \mid \pi_m, \dot{D}^{s,a}\right) = \prod_{t \in \mathcal{S}} \left(n^{s,a,t}\right)^{\left(c_m^{s,t}\right)}$$

In next step, authors have combined expected model factor probability in the Eq. 3 with the above derived likelihood equation (Eq. 5) to obtain a traceable algorithm for updating the observer's beliefs about the dynamics model D based on its personal experience and observations of the mentor.

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

Like any other Reinforcement Learning agent, Imitator requires exploration techniques. In the (current) Bayesian exploration model [2], the uncertainty about the effects of actions are captured by a Dirichlet and is used to estimate a distribution over possible Q-values for each state-action pair.

This method of belief in observations to approximate optimal policy is highly in demand than those provided by heuristic approaches. Bayesian exploration also eliminates the parameter tuning required by methods like ε-greedy, and adapts locally and instantly to evidence. These facts make it a good candidate to combine with imitation.

5 Experiments

In this section, the author focuses on the expected benefits of Bayesian Imitation through several experiments utilizing domains from literature and two unique domains to compare Bayesian imitation to non-Bayesian imitation [11] and several standard model-predicated RL (non-imitating) techniques, including Bayesian exploration, prioritized sweeping and consummate Bellman backups. We additionally investigate how Bayesian exploration cumulates with imitation.

The author begins by describing the agents applied in the experiment setting. The Oracle employs a fixed policy optimized for each domain, presenting both a baseline and a supply of expert behavior for the observers. The EGBS agent combines $epsilon$-greedy exploration (EG) with a complete Bellman backup (i.e., sweep) at every time step. The EGPS agent is a version-primarily based $mathrmRL$ agent, the usage of $epsilon$-greedy (EG) exploration with prioritized sweeping (ps). The BE agent employs Bayesian exploration (BE) with prioritized sweeping for backups. BEBI combines Bayesian exploration (BE) with Bayesian imitation (BI). EGBI combines $epsilon$-greedy exploration (EG) with Bayesian imitation (BI). The EGNBI agent combines $epsilon$-greedy exploration with non-Bayesian imitation.

At the start of every experiment, agents don't interact with others at the beginning state. When the agent accomplishes the goal, it's reset to the start, whereas other agents are unaffected and continue learning. In each domain, agents have locally uniform priors (example: 8 neighbors in grid world) with a designated number of steps. Imitators observe the expert oracle agent concurrently with their (own) exploration. Results are reported with the total reward collected within the last 200 steps. This window integrates the rewards obtained by the agent making it easier to check the performance of other agents. During the first 200 steps, the combination window starts empty causing the oracle's plot to leap from zero to optimal within the first 200 steps. The Bayesian agents use 5 sampled MDPs for estimating Q-value distributions and 10 samples for estimating the mentoring policy from the Dirichlet distribution. Exploration rates for ϵ-greedy agents were tuned for each experimental domain.

The author performs the first test of agents on the "Loop" and "Chain" examples 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.

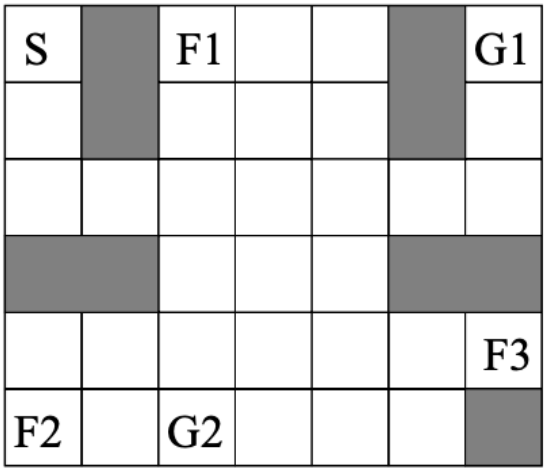

Figure 2: Flagworld Domain [13].

The second test was on the "FlagWorld" domain [2], leading to meaningful performance among agents. In Figure 2, we will see the agent starts at state S and is trying to find the goal state $G1$. The agent can then choose any of three flags by visiting states $F1$, $F2$, and $F3$. Upon reaching the goal state, the agent receives 1 point for every flag

collected. Each action (N, E, S, W) succeeds with probability 0.9 with the corresponding direction is evident, and with probability 0.1 moves the agent perpendicular to the specified direction. In figure 3, the author shows how reward is

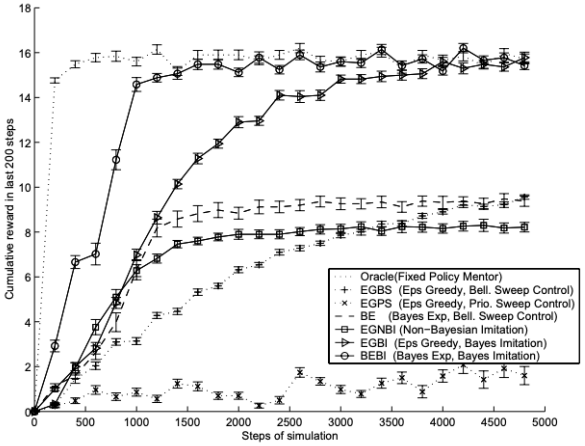

Figure 3: Flag world results (50 runs) [13].

collected over the preceding 200 steps for each agent with optimal performance demonstration. The Bayesian imitator using Bayesian exploration (BEBI) achieves the quickest convergence to the optimal solution. The ϵ-greedy Bayesian imitator (EGBI) is next though it 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 find the optimal policy (in domain). We can also see that non-imitating Bayesian Explorer has poor performance than Bayesian Imitators but outperforming the remaining agents, as they exploit prior knowledge about the connectivity of the domain. By this, the author conclude that Bayesian imitation makes the best use of the information available to the agents, particularly when combined with Bayesian exploration.

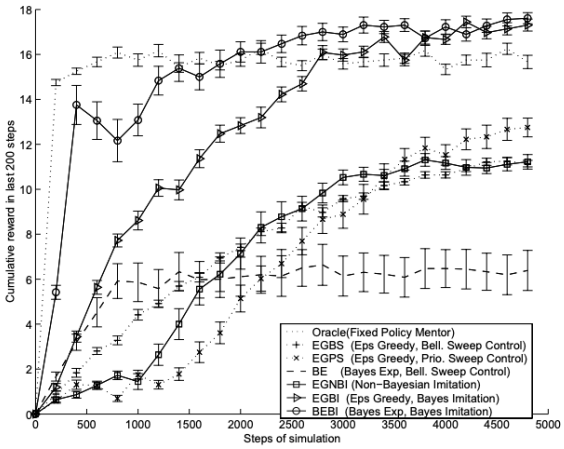

Figure 4: Flag World Moved Goal (50 runs) [13].

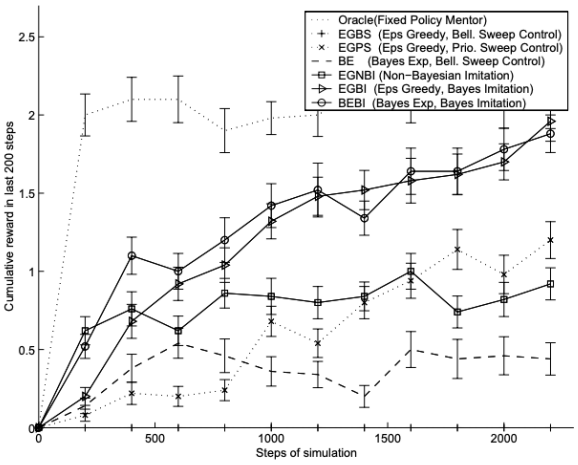

Figure 5: Tutoring Domain Results (50 runs) [13].

Author updated the "FlagWorld" domain (with different objectives for both mentors and learners) goal of the expert Oracle remained at location G1, while the learners had goal location G2 (Figure 2).
In Figure 4, we can see that imitation is similar to the case with identical rewards and how it is utilized 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. For simplification, the Author considers a simulated student. The student's performance is modeled by independent, discretely approximated, exponential forgetting curves for each concept. The agent receives a reward when the student's forgetting rate has been reduced below a predefined threshold for all concepts. The author even mentions that model is too simple to serve as a realistic cognitive model for 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 and we can see that all of the imitators learn quickly, but with the Bayesian imitators BEBI and EGBI outperforming EGNBI converging to a sub-optimum policy (that is increasing exploration allows EGNBI to find the optimal policy, but further depresses short term performance).The Generic Bayesian agent BE also chooses sub-optimal policy.Thus, we see that imitation mitigates one of the drawbacks of Bayesian exploration: mentor observations can be used to overcome misleading priors.

In the next domain, we see 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, and the Bayesian explorer (BE) chooses a path based on its

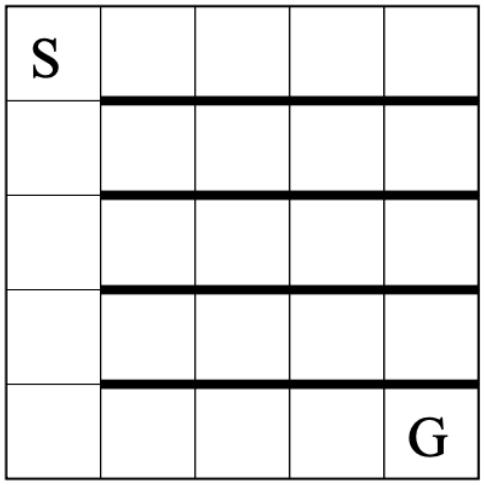

Figure 6: No-South Domain) [13].

prior beliefs that the space is completely connected. The result of this domain is illustrated in Figure 7, where we can differentiate the early performance of the imitation agents (BEBI, EGBI, and EGNBI) from the Bayesian explorer (BE) and other independent learners.

The dead-end lead during the initial value function

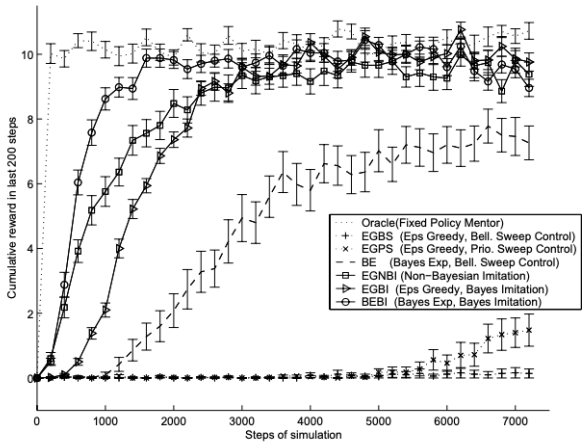

Figure 7: No South results (50 runs) [13].

implies costly misdirection of exploration and poor performance. We finally 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 ϵ-greedy. Again here, mentor information is used to great effect to overcome misleading priors.

6 Conclusion

We noticed how Bayesian imitation, accelerates reinforcement learning in the presence of other agents with appropriate understanding without requiring either explicit conversation with or the cooperation of those different marketers. The author demonstrates how the Bayesian system is developed primarily based on a pooling mechanism (through combining prior knowledge, model observations from the imitator's personal experience, and model observations derived from other agents). On aggregating Bayesian imitation with Bayesian exploration removes parameter tuning and gives you an agent that swiftly exploits mentor observations to reduce exploration and increase exploitation.

Further, we also saw, how Bayesian exploration overwhelms drawbacks, the opportunity of converging to a suboptimal policy because of deceptive priors. Bayesian imitation can be extended to more than one mentor, it could additionally be extended to partially observable environments with recognized nation spaces. Even though the Bayesian formula is difficult to put into effect at once, we've got to know how reasonable approximations can be achieved through traceable algorithms.

References

[1] C. G. Atkeson and S. Schaal. Robot learning from demonstration. In *Machine Learning: Proceedings of the Fourteenth International Conference (ICML '97)*, pages 12–20, Nashville, TN, July 8-12, 1997, 1997. Morgan Kaufmann. clmc.

[2] Richard Dearden, Nir Friedman, and David Andre. Model based bayesian exploration. In *UAI*, 1999.

[3] Ahmed Hussein, Mohamed Medhat Gaber, Eyad Elyan, and Chrisina Jayne. Imitation learning: A survey of learning methods. *ACM Comput. Surv.*, 50(2), apr 2017.

[4] Leslie Pack Kaelbling. *Learning in Embedded Systems*. The MIT Press, 05 1993.

[5] Michael Kearns and Satinder Singh. Finite-sample convergence rates for q-learning and indirect algorithms. In M. Kearns, S. Solla, and D. Cohn, editors, *Advances in Neural Information Processing Systems*, volume 11. MIT Press, 1998.

[6] 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.

[7] Michael L. Littman. Markov games as a framework for multi-agent reinforcement learning. In *In Proceedings of the Eleventh International Conference on Machine Learning*, pages 157–163. Morgan Kaufmann, 1994.

[8] Andrew Moore and C. G. Atkeson. Prioritized sweeping: Reinforcement learning with less data and less real time. *Machine Learning*, 13(1):103 – 130, October 1993.

[9] Chrystopher Nehaniv and Kerstin Dautenhahn. Mapping between dissimilar bodies: Affordances and the algebraic foundations of imitation, 1998.

[10] Andrew Y. Ng and Stuart Russell. Algorithms for inverse reinforcement learning. In *in Proc. 17th International Conf. on Machine Learning*, pages 663–670. Morgan Kaufmann, 2000.

[11] Bob Price and Craig Boutilier. Implicit imitation in multiagent reinforcement learning. In *Proceedings of the Sixteenth International Conference on Machine Learning*, pages 325–334. Morgan Kaufmann, 1999.

[12] Bob Price and Craig Boutilier. A bayesian approach to imitation in reinforcement learning. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, IJCAI'03, page 712–717, San Francisco, CA, USA, 2003. Morgan Kaufmann Publishers Inc.

[13] Bob Price and Craig Boutilier. A bayesian approach to imitation in reinforcement learning. In *IJCAI*, pages 712–720, 2003.

[14] Claude Sammut, Scott Hurst, Dana Kedzier, and Donald Michie. Learning to fly. In *In Proceedings of the Ninth International Conference on Machine Learning*, pages 385–393. Morgan Kaufmann, 1992.

[15] Malcolm Strens. A bayesian framework for reinforcement learning. In *In Proceedings of the Seventeenth International Conference on Machine Learning*, pages 943–950. ICML, 2000.