Blending Autonomous and Apprenticeship Learning

Thomas J. Walsh and Daniel Hewlett and Clayton T. Morrison
Department of Computer Science
University of Arizona, Tucson, Arizona 85721
email: {twalsh,dhewlett,clayton}@cs.arizona.edu

I. Introduction

Apprenticeship learning, where teachers generally demonstrate behaviors for agents to follow, has been used to train agents to control complicated systems such as helicopters [1]. However, most work on this topic burdens the teacher with demonstrating even the simplest nuances of a task. This contrasts with *autonomous* reinforcement learning [5] where a number of domain classes are efficiently learnable by an actively exploring agent, although this class is provably smaller than those learnable with the help of a teacher (see [6]). Intuitively, this seems like a false choice; human teachers often use demonstration but also let the student explore parts of the domain on its own. Here we describe some of our recent theoretical and empirical results with a system that balances teacher demonstrations and autonomous exploration.

This work extends the apprenticeship learning protocol of [6] where a learning agent and teacher take turns running trajectories. We note this version of apprenticeship is fundamentally different from Inverse Reinforcement Learning (IRL) and imitation learning [3] because (1) our agents are allowed to enact better policies then their teachers and (2) our agents do not know the dynamics of the environment and do observe reward signals in teacher trajectories (the opposite is true in IRL). Our work modifies the previous efforts by (1) introducing a new architecture (called KWIK-MBP) based on a similar learning protocol from [4] that indicates areas where the agent should autonomously explore but also can make mistakes, and (2) introducing a communication of expected utility from the student to the teacher so the teacher can better determine when to give a trace.

II. KWIK, and MBP for Domain Learning

In model-based reinforcement learning (for an MDP $\langle S, A, T, R, \gamma \rangle$), recent advancements [2] have linked the efficient learnability of T and R in the KWIK ("Knows What It Knows") framework for supervised learning with PAC-MDP behavior. KWIK caps the number of times the agent will admit uncertainty in its predictions by forcing an agent to predict $\perp$ ("I don't know") in order to get a sample. The general result from [2] shows that if the transition and reward functions T and R of an MDP are KWIK learnable, then a PAC-MDP agent (taking at most a polynomial number of suboptimal steps with high probability) can be constructed for autonomous exploration. The mechanism for constructing such PAC-MDP agents is to use an *optimistic interpretation* of the model where it replaces any $\perp$ predictions from those learners with transitions to a trap state with reward R_{max}. While the class of functions that is KWIK learnable contains many interesting MDPs (including tabular and factored MDPs), it is ultimately limited as larger dynamics classes (such as those with conjunctions for pre-conditions) are not KWIK learnable.

However, in the apprenticeship setting, a larger class of models (including such pre-conditions) can be efficiently learned. In the protocol described in [6], an agent starts at state s_0 and is asked to take actions according to its current policy π_A, until a horizon H or a terminal state is reached. After each of these episodes, a teacher can demonstrate its own policy π_T starting from s_0. The learning agent is able to fully observe each transition and reward received both in its own trajectories as well as the teacher, which may be able to provide highly informative samples (such as those needed to learn conjunctive pre-conditions). In that work, the authors describe a measure of sample complexity called *PAC-MDP-Trace* (analogous to PAC-MDP from above) that measures (with probability $1-\delta$) the number of episodes where $V_{\pi_A}(s_0) < V_{\pi_T}(s_0) - \epsilon$, that is where the expected value of the agent's policy is significantly worse than the expected value of the teacher's policy (V_A and V_T for short) and connect it to a supervised framework called Mistake Bound Predictor (MBP). This mirrors the KWIK to PAC-MDP connection described earlier, except that the interpretation of the model is strict, and often *pessimistic*. Such interpretations would be catastrophic in the autonomous case, but are permissible in apprenticeship learning where teacher traces will make up for the missed data.

Notice if one considers a measure (as we do below) where the number of teacher traces is to be minimized, then MBP learning may overburden the teacher. In addition, PAC-MDP-Trace and MBP-Agent do not specify when a teacher can stop giving traces, but rather only count episodes when the agent's policy was worse than the teacher's.

III. Teaching by Demonstration with Mixed Interpretations

We now introduce a different criteria with the goal of minimizing teacher traces while not forcing the agent to explore exponentially long.

Definition 1. *A* Teacher Interaction *(TI) bound for a domain in apprenticeship learning bounds the number of episodes where the teacher provides a trace to an agent while ensuring the number of suboptimal steps by the agent between each teacher interaction (or after the last one) where* $V_A(s_0) < V_T(s_0) - \epsilon$ *is polynomial in the domain parameters with probability* $1-\delta$.

A good TI bound minimizes the teacher interactions, but only requires the suboptimal exploration steps to be polynomially bounded, not minimized. This reflects our judgement that teacher interactions are far more costly than autonomous agent steps, so as long as the latter are reasonably constrained, we should always seek to minimize the former.

Here, we propose a supervised-learning protocol that can separately quantify the number of changes made to a model through exploration and teacher demonstrations, based on the recent KWIK-MB protocol [4], which we extend slightly here for stochastic labels (KWIK-MBP).

Definition 2. *A hypothesis class $H : X \mapsto Y$ is said to be KWIK-MBP with accuracy parameters ϵ and δ under the following conditions. For each (adversarial) input x_t the learner must predict $y_t \in Y$ or $\perp$. With probability $(1-\delta)$, the number of $\perp$ predictions must be bounded by a polynomial K over $\langle |H|, \epsilon, \delta \rangle$ and the number of mistakes must be bounded by a polynomial M.*

KWIK-MB was originally designed for a situation where mistakes are more costly than $\perp$ predictions. So mistakes are minimized while $\perp$ predictions are only bounded. This is analogous to our own TI criteria so we now consider how a mix of optimism and pessimism and a KWIK-MBP learner would fare in the apprenticeship setting.

Our algorithm (KWIK-MBP-Agent) builds KWIK-MBP learners L_T and L_R for the transition and reward functions of an MDP. When planning with the subsequent model, the agent constructs a "mixed" interpretation, trusting the learner's predictions where mistakes might be made, but replacing all $\perp$ predictions from L_R with a reward of R_{max} and any $\perp$ predictions from $\hat{T}$ with transitions to the R_{max} trap state. This has the effect of drawing the agent to explore explicitly uncertain regions ($\perp$) and to either explore on its own or rely on the teacher for areas where a mistake might be made.

However, this change is not yet enough to guarantee a meaningful TI bound. As an example, suppose there was no communication in the algorithm just described and the teacher provided a trace whenever the student's previous policy was worse than the teacher's. Consider a domain where the pre-conditions of actions are governed by a disjunction over the n state factors. [4] showed that disjunctions can be learned using a combination of $M = n/3$ mistakes and $K = 3n/2 - 3M$ $\perp$ predictions. However, that learning algorithm defaults to predicting "true" and only learns from negative examples. Thus, in the apprenticeship setting, both mistaken predictions and $\perp$ predictions are being filled in optimistically and the agent should learn the pre-conditions purely through autonomous exploration. However, the teacher will provide many traces to the agent since it sees it performing suboptimally during its exploration. These traces, which may contain only positive examples, will be uninformative to L_T.

We eliminate this problem by providing a channel where the student communicates its expected utility U_A, which the teacher compares to its own utility to decide if it should give a trace. That is, a teacher will only show a trace to a pessimistic agent, but will "stand back" and let an over-confident student learn from its own mistakes.

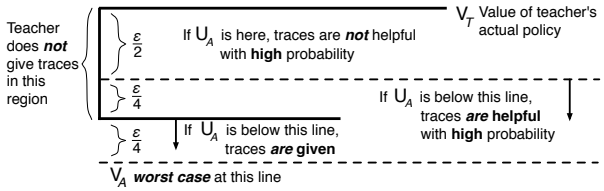

Fig. 1. An illustration of the cases for the main theorem.

Note this is different than comparing to the actual expected value of π_A in the real world (denoted V_A). A theorem for KWIK-MBP-Agent's TI bound appears below. The main argument from the proof is illustrated in Figure 1, where we show that if we force the student to (w.h.p.) learn an $\frac{\epsilon}{4}$-accurate value function we can guarantee traces below $V_T - \frac{\epsilon}{2}$ will be helpful to an agent, but because V_A nd U_A may be different (since the latter comes from the agent's model), the teacher does not give a trace unless $U_A < V_T - \frac{3}{4}\epsilon$, to make sure it will help. We bound the error on the true returns V_A by adding in an additional $\frac{\epsilon}{4}$ slack term. Because traces only come in when the student undervalues his performance, the number of traces is related only to the MBP portion of the KWIK-MBP bound.

Theorem 1 (main theorem)**.** *A KWIK-MBP-Agent that reports $U_A(s_0)$ before every episode will have a TI bound that is polynomial in $\frac{1}{\epsilon}$, $\frac{1}{\delta}$, and $\frac{1}{1-\gamma}$ and M, the latter of which is the number of mistakes (not counting $\perp$ predictions) made by L_T and L_R. A tighter bound on can be achieved by making M the number of pessimistic mistakes (those responsible for U_A being significantly less than V_A).*

Other changes to the protocol can also lead to efficient learning (for instance forcing the teacher to intervene only when the student must have finished learning and based on the observed values the student has collected).

IV. Experiments

Our first experiment is in a blocks world with dynamics based on *stochastic* STRIPS operators and a -1 step cost with a goal of stacking the blocks. The actions in this world are two versions of *pickup*(X, From) and two versions of *putDown*(X, To), with one version being "reliable", producing the expected result 80% of the time and otherwise doing nothing. The other version of each action has the probabilities reversed. The literals in the effects of the STRIPS operators (the Add and Delete lists) are given to the learning agents, but the *pre-conditions* and *probabilities* of the effects need to be learned. This is an interesting case because the effect probabilities can be learned autonomously while the conjunctive pre-conditions, require teacher input.

Figure 2, column **1**, shows KWIK, MBP, and KWIK-MBP agents as trained by a teacher who uses *un*reliable actions half the time. The KWIK learner never receives traces (since its confidence is always high), but spends exponential (in the

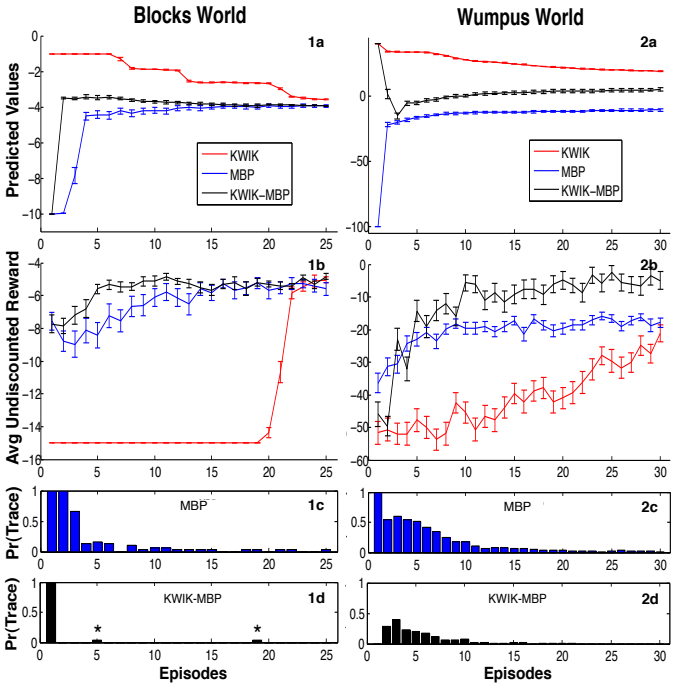

Fig. 2. A matrix of plots. Rows include: value predictions (i.e., $U_A(s_0)$, row **a**), average undiscounted reward sums (row **b**), and the proportion of trials where MBP and KWIK-MBP received teacher traces (**c** and **d**). The left column is Blocks World and the right a modified Wumpus World. Red corresponds to KWIK, blue to MBP, and black to KWIK-MBP.

number of literals) time exploring the potential pre-conditions of actions. In contrast, the MBP and KWIK-MBP agents use the first trace to learn the pre-conditions. The proportion of trials (out of 30) that the MBP and KWIK-MBP learners received teacher traces across episodes is shown in the bar graphs **1c** and **1d** of Fig. 2. The MBP learner continues to get traces for several episodes afterwards, using them to help learn the probabilities well after the pre-conditions are learned. This probability learning could be accomplished autonomously, but the MBP pessimistic value function prevents such exploration in this case. By contrast, KWIK-MBP receives 1 trace to learn the pre-conditions, and then explores the probabilities on its own. KWIK-MBP actually learns the probabilities faster than MBP because it targets areas it does not know about rather than relying on potentially redundant teacher samples. However, in rare cases KWIK-MBP receives additional traces; in fact there were two exceptions in the 30 trials, indicated by $*$'s at episodes 5 and 19 in **1d**. The reason for this is that sometimes the learner may be unlucky in the experience it gathers, constructing an inaccurate value estimate and the teacher then steps in and provides a trace.

The second domain we used is a variant of "Wumpus World" with 5 locations in a chain, an agent who can move, fire arrows (unlimited supply) or pick berries (also unlimited), and a wumpus moving randomly. The domain is represented by a Dynamic Bayes Net (DBN) based on the factors above and the reward is represented as a linear combination of the factor values (-5 for a live wumpus and $+2$ for picking a berry). The action effects are noisy, especially the probability of killing the wumpus, which depends on the exact (not just relative) locations of the agent, wumpus, and whether the wumpus is dead yet (3 parent factors in the DBN). While the reward function is KWIK learnable through linear regression [2] and though DBN CPTs with small parent sizes are also KWIK learnable, the high connectivity of this DBN makes autonomous exploration of all the parent configurations prohibitive. Instead, we constructed an "optimal hunting" teacher that finds the best combination of locations to shoot the wumpus from/at, but ignores the berries. We concentrate on the ability of our algorithm to explore and find a better policy than the teacher (i.e., learning to pick berries), but still staying close enough to the teacher's traces that it can hunt the wumpus effectively.

In plot **2a** we see the predicted values of the three learners, while the plot **2b** shows their performance. The KWIK learner starts with high U_A and gradually descends (in **2a**), but without traces spends most of its time exploring fruitlessly (very slowly inclining slope of **2b**). The MBP learner learns to hunt from the teacher and quickly achieves good behavior, but rarely learns to pick berries (only gaining experience on the reward of berries if it ends up in completely unknown state and picks berries at random many times). The KWIK-MBP learner starts with high U_A and explores the structure of just the reward function, discovering berries but not the proper location combinations for killing the wumpus. It's expected utility thus initially dips precipitously as it thinks all it can do is collect berries. Once this crosses the teacher's threshold, the teacher steps in with a number of traces showing the best way to hunt the wumpus—this is seen in plot **2d** with the small bump in the proportion of trials with traces, starting at episode 2 and declining roughly linearly until episode 10. The KWIK-MBP student is then able to fill in the CPTs with information from the teacher and reach an optimal policy that kills the wumpus and picks berries, avoiding both the over- and under-exploration of the KWIK and MBP agents—this increased overall performance is seen in plot **2b** as, between episodes 5 and 10 KWIK-MBP's average reward surpasses MBP.

References

[1] Pieter Abbeel and Andrew Y. Ng. Exploration and apprenticeship learning in reinforcement learning. In *ICML*, 2005.

[2] Lihong Li, Michael L. Littman, Thomas J. Walsh, and Alexander L. Strehl. Knows what it knows: A framework for self-aware learning. *Machine Learning*, 82(3):399–443, 2011.

[3] Nathan Ratliff, David Silver, and J. Bagnell. Learning to search: Functional gradient techniques for imitation learning. *Autonomous Robots*, 27:25–53, 2009.

[4] Amin Sayedi, Morteza Zadimoghaddam, and Avrim Blum. Trading off mistakes and don't-know predictions. In *NIPS*, 2010.

[5] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, March 1998. ISBN 0-262-19398-1.

[6] Thomas J. Walsh, Kaushik Subramanian, Michael L. Littman, and Carlos Diuk. Generalizing apprenticeship learning across hypothesis classes. In *ICML*, 2010.