Learning Non-Myopically from Human-Generated Reward

W. Bradley Knox
Massachusetts Institute of Technology
Media Lab
Cambridge, Massachusetts
bradknox@mit.edu

Peter Stone
University of Texas at Austin
Department of Computer Science
Austin, Texas
pstone@cs.utexas.edu

ABSTRACT

Recent research has demonstrated that human-generated reward signals can be effectively used to train agents to perform a range of reinforcement learning tasks. Such tasks are either episodic—i.e., conducted in unconnected episodes of activity that often end in either goal or failure states—or continuing—i.e., indefinitely ongoing. Another point of difference is whether the learning agent highly discounts the value of future reward—a myopic agent—or conversely values future reward appreciably. In recent work, we found that previous approaches to learning from human reward all used myopic valuation [7]. This study additionally provided evidence for the desirability of myopic valuation in task domains that are both goal-based and episodic.

In this paper, we conduct three user studies that examine critical assumptions of our previous research: task episodicity, optimal behavior with respect to a Markov Decision Process, and lack of a failure state in the goal-based task. In the first experiment, we show that converting a simple episodic task to non-episodic (i.e., continuing) task resolves some theoretical issues present in episodic tasks with generally positive reward and—relatedly—enables highly successful learning with non-myopic valuation in multiple user studies. The primary learning algorithm in this paper, which we call "VI-TAMER", is *the first algorithm to successfully learn non-myopically from human-generated reward*; we also empirically show that such non-myopic valuation facilitates higher-level understanding of the task. Anticipating the complexity of real-world problems, we perform two subsequent user studies—one with a failure state added—that compare (1) learning when states are updated asynchronously with local bias—i.e., states quickly reachable from the agent's current state are updated more often than other states—to (2) learning with the fully synchronous sweeps across each state in the VI-TAMER algorithm. With these locally biased updates, we find that *the general positivity of human reward creates problems even for continuing tasks*, revealing a distinct research challenge for future work.

IUI'13, March 19–22, 2013, Santa Monica, CA, USA.

Author Keywords

reinforcement learning; modeling and prediction of user behavior; end-user programming; human-agent interaction; interactive machine learning; human teachers

ACM Classification Keywords

H.1.2 User/Machine Systems: Miscellaneous

INTRODUCTION

The constructs of reward and punishment form the foundation of psychological models that have provided powerful insights into the behavior of humans and other animals. Reward and punishment are frequently received in a social context, from another social agent. In recent years, this form of communication and its machine-learning analog—reinforcement learning—have been adapted to permit teaching of artificial agents by their human users [4, 14, 6, 13, 11, 10]. In this form of teaching—which we call interactive shaping—a user observes an agent's behavior while generating *human reward* instances through varying interfaces (e.g., keyboard, mouse, or verbal feedback); each instance is received by the learning agent as a time-stamped numeric value and used to inform future behavioral choices. Here the trainer considers his or her reward to encompass colloquial concepts like "reward" and "punishment", "approval" and "disapproval", or something similar.[1]

Interactive shaping enables people—without programming skills or complicated instruction—to specify desired behavior and to share task knowledge when correct behavior is already indirectly specified (e.g., by a pre-coded reward function). Further, in contrast to the complementary approach of learning from demonstration [1], learning from human reward employs a simple task-independent interface, exhibits learned behavior *during* teaching, and, we speculate, requires less task expertise and places less cognitive load on the trainer.

Concepts and definitions

Interactive shaping can be framed as having two different objectives. The *task objective* represents what the trainer is attempting to teach and determines an agent's *task performance*. For instance, when training an agent to play Tetris [6], the task objective might be maximizing the number of lines cleared per game or stacking as many blocks to the far right as

[1]The term "punishment" is used here only in psychological and colloquial contexts. In artificial intelligence, the term "reward" includes both positively and negatively valued feedback.

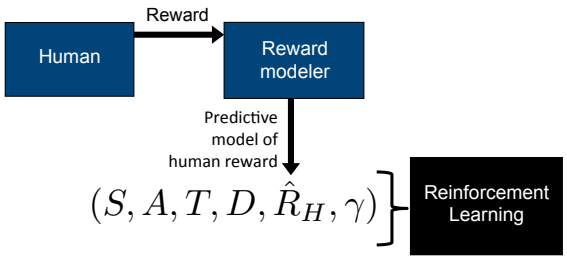


Figure 1. An illustration of the agent's learning algorithm, where inputs *to* the human are not represented. Human reward instances change the predictive reward model $\hat{R}_H$, which is used as the reward function in an MDP. A reinforcement learning agent learns from the most recent MDP.

possible; the trainer decides the task objective unless it is pre-specified and communicated to the trainer, as we do for experimental purposes. The agent has no access to the trainer's task objective; setting aside the possibility of pre-programmed a priori knowledge (without loss of generality), we observe that the agent instead only experiences its own state-action history and the reward instances given by the trainer. Thus, the learning agent needs its own objective, which we define with respect to the human-generated reward it receives. Following this approach, any solution to the interactive shaping problem includes two parts: (1) a *reward-based objective* for the agent and (2) a learning algorithm that effectively improves on this objective with experience.

We focus on the space of reward-based objectives used in reinforcement learning (RL) for Markov Decision Processes (MDPs) [12]. MDPs are denoted as $\{S, A, T, R, \gamma, D\}$.[2] RL algorithms seek to learn policies ($\pi : S \rightarrow A$) for an MDP that improve its discounted sum long-term reward—i.e., its *return*—from each state, where return is expressed as $Q^\pi(s, \pi(s))$ and defined in terms of reward as $Q^\pi(s, a) = \sum_{t=0}^{\infty} E_\pi[\gamma^t R(s_t, a_t)]$ (with $0^0 = 1$). We refer to return-maximizing policies as *MDP-optimal*; on the other hand, policies that maximize the task objective are called *task optimal*.

For the experiments described in this paper, a Markovian model of human reward, $\hat{R}_H$, is learned from human reward instances. This model completes an MDP specification for the agent to learn in, $\{S, A, T, \hat{R}_H, \gamma, D\}$ (Figure 1). Thus, the output of $\hat{R}_H(s, a)$ for an agent's current state s and action a is the actual reward experienced by the learning agent. In this research, we seek to find reward-based objectives such that algorithms that perform well on the reward-based objective *with reward functions modeled on human trainers' reward* also perform well on the task objective. If we were concerned only with optimal behavior (we are not), this goal could be restated as finding reward-based objectives such that MDP-optimal behavior is also task optimal.

[2]Here, S and A are the sets of possible states and actions; T is a function describing the probability of transitioning from one state to another given a specific action, such that $T(s_t, a_t, s_{t+1}) = P(s_{t+1}|s_t, a_t)$; R is a reward function, $R : S \times A \rightarrow \Re$, with a state and an action as inputs; $\gamma \in [0, 1]$ is a discount factor, controlling how much expected future reward is valued; and D is the distribution of start states for each learning episode.

An important aspect of reward-based objectives is temporal discounting, which is controlled by the γ parameter of MDPs. As shown in the expression of return above, when $\gamma = 0$ the objective is fully myopic. A fully myopic agent only values the reward from its immediate state and action; expected future reward is not valued. At the other extreme, an agent with a $\gamma = 1$ objective values near-term reward equally to reward infinitely far in the future. When $\gamma \in (0, 1)$, future reward is discounted at an exponential rate, making near-term reward more valuable than long-term reward. Higher γ values result in a lower discount rate.

An additional dimension of MDPs is *episodicity*: whether the task is episodic or continuing. In an episodic task, the agent can reach one or more episode-terminating states, which are called *"absorbing states"* in the RL literature [12]. Upon reaching an absorbing state, the learning episode ends, a new episode starts with state chosen independently of the reached absorbing state, and the agent experiences reward that is not attributable to behavior during the previous episode. Absorbing states often either represent success or failure at the task, constituting *goal states* or *failure states*; we call tasks with goal states "goal-based". In contrast to an episodic task, a continuing task is ongoing, wherein all reward is attributable to all past behavior, if discounting permits.

As we describe more specifically later in this section, we explore the impact on task performance of the agent's reward-based objective along four dimensions: (1) the discount factor, (2) whether a task is episodic or continuing, (3) whether the agent acts approximately MDP-optimal or is less effective in maximizing its return, and (4) the effect of having a failure state as well as the goal state. In addition to understanding the effects of these four dimensions on the agent's task performance, this paper concerns two questions: How can an agent learn from human reward at low discount rates (i.e., non-myopically, with high discount factors)? What benefits are conferred by low discount rates?

Background

A critical precursor to this work is our investigation of the impact of discounting in goal-based, episodic tasks [7]. Investigating the six previous projects that we know to have involved learning from positively and negatively valued human-generated reward [4, 14, 13, 11, 9, 10] (including by email with corresponding authors), we identified a curious trend: all such projects have been much more myopic—i.e., using high discount rates—than is usual in RL. We hypothesized that a cause of this pattern is the general positivity of human reward. More specifically, if the previously observed positive bias in human reward creates at least one "positive circuit"—i.e., repeatable sequences of behavior that net positive total reward—then an MDP-optimal agent in an episodic, goal-based task *will avoid the goal* if acting without discounting, since the absorbing state of the goal prevents accrual of further reward. To investigate this hypothesis, we previously conducted a user study of agent training in a simple grid-world task (shown later in Figure 4), varying the experimental conditions only by the discount factor in the agent's reward-based objective. Experimental results showed (1) that higher

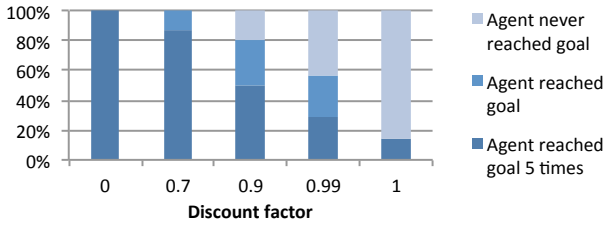


Figure 2. Success rates by discount factor for our prior experiment.

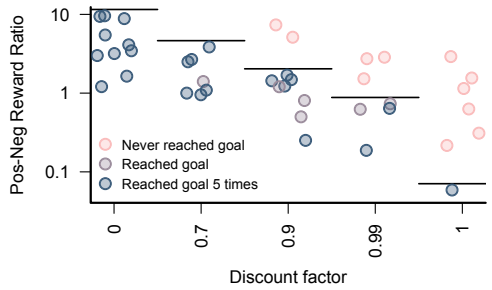


Figure 3. From our prior experiment, the ratio of cumulative positive reward to cumulative negative reward given by each trainer, separated by discount factor condition and task performance. X-axis jitter was added for readability. Within each condition, a horizontal line was placed above the mark for the highest ratio at which a subject trained the agent to reach the goal at least once.

discount rates (i.e., lower γs) led to better performance, (2) that lower discount rates result in more negative reward overall (fittingly, since increased negativity helps address the positive circuits problem), and (3) that successful trainers at lower discount rates give more negative reward than do unsuccessful ones. Additionally, 66.7% of trainers across all conditions created at least one positive circuit, meaning that those agents would never greedily reach the goal if discounting at $\gamma = 1$. The results of this experiment are reproduced in Figures 2 and 3.

Preview of experiments

Using adaptations of the same task, experimental design, and agent algorithm—which we dub VI-TAMER—this paper includes **three user studies** that examine three important assumptions of our prior work:

1. The task is episodic.
2. The agent can act approximately MDP-optimally with respect to the agent's current predictive model of human reward.
3. The task has a goal state but lacks failure states.

The task and agent algorithm are described in detail in the *Continuing-task experiment* section.

Repeating our prior experiment in a continuing version of the same task—in what we call the **continuing-task experiment**—we make the following four observations. First, we find for discount factors $\gamma < 1$ that task performance of MDP-optimal policies during training is generally high and independent of discounting; in contrast, such policies do not perform well at high γs when the task is episodic. Second, strong correlations observed in their episodic-task experiment disappear. Third, in this investigation, the $\gamma = 0.99$ condition with the VI-TAMER algorithm is the *first known instance of successful learning from human-generated reward at a low discount rate* (i.e., a high gamma with relatively long time steps). Fourth, in two additional tests using the training data from this continuing-task experiment, we find evidence for the theoretically based conjecture that low discount rates facilitate the communication of higher-level task information—e.g., the location of the goal rather than the exact sequence of steps to reach it. Such higher-level information enables learning that is more robust to environmental changes, better guides action in unexperienced states, and leads the agent to learn policies that surpass those known to the trainer.

In our second experiment—the **local-bias experiment**—we examine task performance under low discount rates *when the agent cannot generally act approximately MDP-optimally*. This experiment is motivated by an anticipation of converting VI-TAMER to more complex tasks, where MDP-optimal behavior is intractable. Specifically, we focus on an agent that updates the values of state-action pairs with a *local bias*, meaning that pairs reachable within a few steps of the agent's current state receive more updates than those further away. Such locally biased updates are a common characteristic among RL algorithms. With such local bias, we find that performance decreases, apparently because of further problems created by the positivity of human reward.

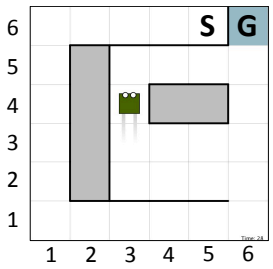


Figure 4. The baseline grid world task. To display the agent's actions and state transitions to the trainer, (1) wheel tracks are drawn from the agent's previously occupied cell, and (2) the simulated robot's eyes point in the direction of the last action. The start and goal cells are labeled 'S' and 'G' respectively.

Our third experiment—the **failure-state experiment**—boosts the generality of conclusions from the previous two experiments (and our prior experiments [7]) by adding a failure state to the goal-based task and then repeating the $\gamma = 0.99$ conditions from these previous experiments. The results from this failure-state experiment follow patterns observed in the previous experiments, though sometimes with less significance (likely because of smaller effect sizes).

In the following section, we describe baselines for the agent algorithm, the task, and the experimental design. In each of the subsequent three sections, one of the three experiments is described along with an analysis of its results. We then conclude the paper.

BASELINES FOR THE TASK, AGENT, AND EXPERIMENT

Here we describe baseline versions of the task, the agent, and the experiment, which are equivalent to those for the grid-world experiment from our prior work [7]. Unless otherwise specified, this baseline set is used in each of this paper's experiments. Given space requirements, we keep these descriptions at a high level. Full details can be found at `http://www.cs.utexas.edu/users/bradknox/papers/13iui/`.

The task

The task is a grid world with 30 states, shown in Figure 4. At each step, the agent acts once by moving up, down, left, or right, and attempted movement through a wall results in no movement during the step. Task performance metrics are based on the time steps taken to reach the goal. The agent always starts a learning episode in the state labeled "**S**" in Figure 4. The shortest path from the start state requires 19 actions. Each time step lasts approximately 800 ms.

In the episodic version of this task, the goal state is absorbing. In the continuing version, upon transitioning into the goal, the agent instead experiences a transition to the start state. Consequently, reward received in one "episode" can be attributed to state-action pairs in the previous "episode" (and farther in the past). Though reaching the goal in the continuing version does not mark the end of an episode, *we continue to use the word "episode" to refer to the periods of time that are divided by the attainment of the goal*. Another valid perspective for the reader is to assume the task is fundamentally episodic and that the continuing version is simply tricking the agent to make it experience the task as continuing.

The agent

In all experiments a model of human reward, $\hat{R}_H$, is learned through the TAMER framework [6], and the output of this model provides reward for the agent within an MDP specified as $\{S, A, T, \hat{R}_H, \gamma, D\}$. Figure 1 illustrates this scenario. During training, human reward signals form labels for learning samples that have state-action pairs as features; a regression algorithm continuously updates $\hat{R}_H$ with new features. For experiments in this paper, the TAMER module represents $\hat{R}_H$ as a linear model of Gaussian radial basis functions (RBFs) and updates the model by incremental gradient descent, as in our prior work [7]. One RBF is centered on each cell of the grid world, effectively creating a pseudo-tabular representation that generalizes slightly between nearby cells.

During training for all experiments, human reward was communicated via the '/' and 'z' keys on the keyboard, which respectively mapped to 1 and -1. This mapping to 1 and -1, though not infallible, is an intuitive choice that is similar to that of related works that explain their exact mappings [14, 13, 10, 11]. Additionally, this interface allows richer feedback than it superficially appears to for two reasons. First, reward signals are asynchronous to actions, so the rate of reward signaling determines intensity. Second, to account for delays in giving feedback, the causal attribution of each reward is distributed across multiple recent time steps by TAMER's credit assignment module [5], further adding variety to the label values of samples for training $\hat{R}_H$.

The agent seeks to improve its return—i.e., its reward-based objective—with respect to the current $\hat{R}_H$, $\sum_{t=0}^{\infty} E[\gamma^t \hat{R}_H(s_t, \pi(s_t))]$, but we empirically evaluate the agent by task performance metrics that are not reward-based (see the list of statistical tests in description of the baseline experiment below). The reinforcement learning algorithm used by the agent is value iteration [12] with greedy action selection. However, unlike traditional value iteration (in which the algorithm iterates until state values converge), here one update sweep occurs over all of the states every 20 ms, creating approximately 40 sweeps per step. The agent's value function is initialized to zero once, at the start of training only, for reasons discussed in the analysis of the second experiment. We call this value iteration algorithm with TAMER-based modeling of human reward "VI-TAMER".

Because the agent in this experiment learns from a frequently changing reward function, behaving optimally with respect to the current reward function is difficult. For the simple task we have chosen, value iteration creates approximately MDP-optimal behavior with small lag in responding to changes to the reward function, a lag of a few time steps or less. Thus, *we can be confident that observed differences between experimental conditions can be attributed to the reward-based objective, not deficiencies in maximizing that objective.*

The experiment

All experiments were conducted through subjects' web browsers via Amazon Mechanical Turk. Subjects were randomly assigned to an experimental condition. They were prepared with video instructions and a period of controlling the agent followed by a practice training session. During these instructions, subjects are told to give "reward and punishment" to the green "Kermitbot" to "teach the robot to find the water as fast as possible." Trainers were left to determine their own reward strategies, possibly including rewarding every time the agent acts as they would have acted or rewarding highly when the agent reaches the goal.

The actual training session stopped after the agent reached the goal 10 times (i.e., 10 episodes) or after 450 steps, whichever came first (unless otherwise specified). This stopping time is the only difference between these baseline components and our prior grid-world experiment, in which trainers were stopped after the first of 5 episodes or 300 time steps [7].

A training sample is removed from analysis if it fulfills any of the following conditions: the sample was created from the second or later time that a specific worker acted as a subject, the log file is incomplete, the user mistyped his or her condition-specifying user ID such that the condition was incorrectly specified, or the user gave less than 2 instances of feedback per 100 time steps, which we consider to be non-compliance with our experimental instructions.

Because we repeat many of the same statistical tests throughout this paper, we define and name our more common tests here:

- *Fisher Success* - a Fisher's Tests comparing outcomes of reaching the goal all N times or not by condition, where N is specified as a threshold
- *MWU Episodes Finished* - a Mann Whitney U test where the dependent variable is the number of episodes completed before training was stopped
- *MWU Time To Goal* - a Mann Whitney U test where the dependent variable is how many steps occurred before the agent reached the goal for *the first time*

- *Spearman Success* - a Spearman correlation test of the ratios of positive to negative reward and success within a specific condition

CONTINUING-TASK EXPERIMENT

In the experiment described in this section, we investigate the impact of the discount rate when a task is continuing. For episodic tasks, we previously argued that a positive reward bias among human trainers combined with high discount factors can lead to infinite behavioral circuits—created by what we call "positive circuits"—and consequently minimal task performance [7]. For episodic tasks, we found that myopic discounting (low γs) avoids this problem; the *Introduction* describes these prior results in more detail. However, positive circuits may only cause severe problems in episodic tasks, since an agent reaching an absorbing goal state is effectively penalized; it is exiting a world rich with positive reward. We examine another strategy to remove this penalty: formulate the task as continuing, making the goal a gateway to more opportunity to accrue reward. Unlike for episodic MDPs, the optimal policy of a continuing MDP is unaffected by adding a constant value to all reward; thus, the positivity of human reward should not present the same problem for MDP-optimal policies in continuing tasks.

Experiment and analysis of results

This experiment uses the baseline agent, experimental design, and continuing task, repeating our prior experiment almost exactly, only changing the task to be continuing as described in *The task*. For consistency with the previous experiment, we analyze data from the first 5 episodes that occur before the 301st time step and also retain the $\gamma = 1$ condition, even though such discounting is generally inappropriate for continuing tasks. 25 subjects were run per condition, and one subject in the $\gamma = 0.9$ condition was replaced by another subject for not following instructions (a practice followed only in this experiment). After filtering, for γs of 0, 0.7, 0.9, 0.99, and 1, there were respectively 20, 21, 20, 23, and 23 subjects. Figures 5 and 6 show results (presented analogously to Figures 2 and 3).

In comparison to our prior episodic-task experiment [7], the results in the continuing-task version are markedly different. As shown in Figure 5, the task success rate at $\gamma = 1$ is lower than at other conditions, which we expect given that this discounting is generally avoided for continuing tasks to make rewards in the finite future meaningful. The other discount factors create similar task performance, with $\gamma = 0.99$ achieving the highest mean rate of full success. Fisher-Success tests with a 5-episode threshold find a marginally significant difference between $\gamma = 0.99$ and $\gamma = 1$ conditions ($p = 0.0574$); no other pairwise comparisons between conditions are significantly different.[3]

[3]Note that episodicity cannot affect a $\gamma = 0$ agent, making this condition identical to the $\gamma = 0$ condition of our prior experiment. The difference in success rate at $\gamma = 0$ in the two grid world experiments is likely because of either randomness—their difference is insignificant by a Fisher-Success test with a 5-episode threshold ($p = 0.2890$)—or this experiment, run at a different time, sampled from a lower-performing population.

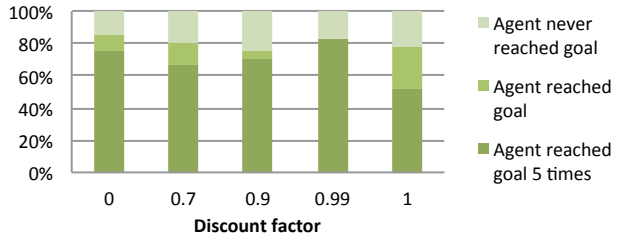


Figure 5. Success rates for the continuing-task experiment by discount factor.

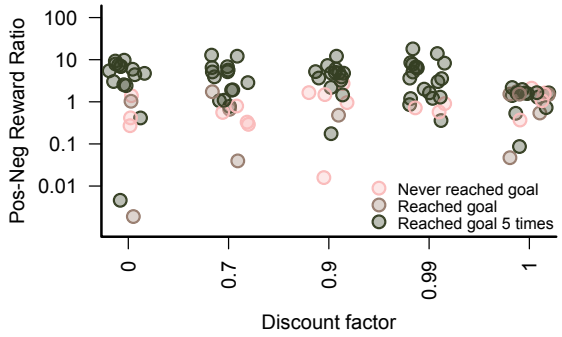


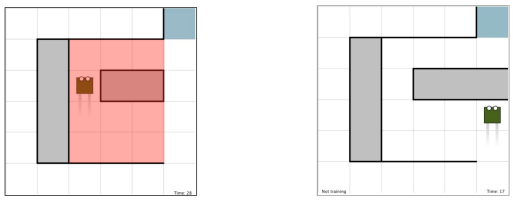
Figure 6. For the continuing-task experiment, ratio of cumulative positive reward to cumulative negative reward given by each trainer (with x-axis jitter).

Patterns exhibited by the ratios of cumulative positive reward to cumulative negative reward among trainers (as shown in Figure 6) also differ from the episodic experiment. Specifically, there is no significant correlation between the ratios of fully successful trainers and discount factor when $\gamma = 1$ is excluded (Spearman coefficient $\rho = -0.0564$, $p = 0.6628$), though the correlation is significant with $\gamma = 1$ included ($\rho = -0.3233$, $p = 0.0050$). Further, the relationship between reward positivity and task performance is closer to the intuitive expectation that high-performance agents receive more positively-biased reward: ratios and success categories (no, partial, and full success) are significantly correlated in the $\gamma \leq 0.9$ conditions (Spearman's coefficient $\rho > 0.595$, $p < 0.006$ for all $\gamma \leq 0.9$) and $\rho > 0$ as well for the other conditions, though the correlation is insignificant.

*For this task and this approximately MDP-optimal RL algorithm (*VI-TAMER*), converting the task to continuing does indeed appear to remove the adverse impact of reward positivity at high discount factors, overcoming the positive circuits problem.* However, based only on the roughly equivalent task performance for all $\gamma < 1$ conditions, the choice of which discounting to use is unclear. In the next subsection, we investigate whether higher-level task information was communicated by the trainer, making learning more robust to changes to the environment or more general at certain γ values.

Benefits of non-myopic learning

At non-myopic discount rates (i.e., γs near 1), reward can communicate a desired policy, the goals of the task, or a mix of the two. Using the full training data from this experiment (up to 10 episodes or 450 time steps), we now investigate whether the trained agents do learn more than a policy. Since $\gamma = 1$ is generally inappropriate for continuing tasks, we ex-



(a) (b)

Figure 7. Illustrations of the two tests of the benefits of non-myopic learning, testing agent performance after training. (a) Starting from (highlighted) states off the optimal path. (b) Blocking the optimal path.

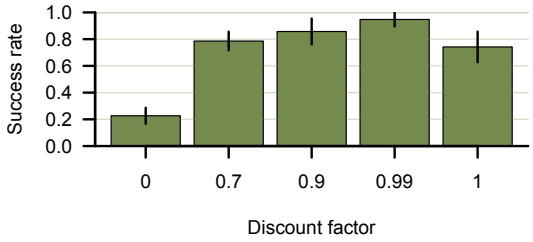


Figure 8. Mean rate of successfully trained agents reaching the goal in less than 100 time steps *from the 10 states off of the optimal path***. Standard error bars are calculated using a single agent's success rate as one sample.**

pect $\gamma = 0.99$ to yield the best results. We restrict our analysis to those agents successfully trained to get to the goal 5 times in less than 300 steps. Thus, we effectively ask, *given that an agent learns a good (and usually optimal) policy from human reward, what else does the agent learn?*

We first test the learned policy from 10 states that are not along the optimal path from the start state, which are highlighted in Figure 7a. These states may have never been experienced by the agent during training, in which case $\hat{R}_H$ is built without any samples from the state. Simple policy generalization from nearby, optimal-path states would help in only some of these 10 states, so the ability of the agent to get to the goal reflects whether the agent has learned some information about the task goals. Agents that had learned their policies at higher γs were more often able to get to the goal in less than 100 time steps (Figure 8). 18 of 19 successfully trained agents in the $\gamma = 0.99$ condition reached the goal from every tested state. We note though that different discount factors might lead to different levels of experience in these tested states, providing a confounding factor.

In the second test, an obstacle is placed in the state two cells below the goal (Figure 7b), blocking the optimal path, and we then determine whether the agent can still reach the goal in less than 100 time steps. Thus, we test the effects of changing the task-optimal policy but keeping constant the task goal: get to the goal state as quickly as possible. For the two state-action pairs that previously transitioned into the newly blocked state, the agent's reward function is modified to output 0 to reflect the agent's lack of knowledge about how the trainer would reward these transitions. In the $\gamma = 0.99$ condition, 9 of 19 successfully trained agents reached the goal. One agent from each of the $\gamma = 0.9$ and $\gamma = 1$ conditions also reached the goal (of 14 and 12 total, respectively); no agents with $\gamma < 0.9$ did.

These analyses support the conjecture that agents taught with higher discount factors learn about the task goals themselves, making the agents generally more robust to changes in the environment and more able to act appropriately in previously unexperienced states. That agents may learn task goals raises the tantalizing prospect that, under the right circumstances, an agent receiving reward from a human trainer could learn a policy that is far superior to that envisioned by the trainer. These benefits of non-myopic learning underscore the importance of creating algorithms for complex, real-world tasks that can learn non-myopically. As the next two sections confirm, achieving this goal is non-trivial.

LOCAL-BIAS EXPERIMENT

In considerably more complex domains than the 30-state grid-world task used in our continuing-task experiment, agents will likely be unable to perform value iteration with iterating sweeps over the entire state; even ignoring the possibility of continuous states or actions, some tasks simply contain too many states to quickly and repeatedly perform temporal difference updates on all states. In anticipation of scaling the high-γ, continuing-task approach found successful in the previous experiment, we implemented a version of value iteration that learns asynchronously, which we call aVI-TAMER. Instead of updating each state once and then repeating as in VI-TAMER,[4] aVI-TAMER updates state-action pairs through the Monte Carlo tree search strategy Upper Confidence Trees (UCT). UCT-based search has been successful in tasks with especially large state spaces [8], originally in games like Go [2] but also in more general reinforcement learning tasks [3].

aVI-TAMER is mostly identical to VI-TAMER: the human reward model $\hat{R}_H$ is learned as before, using TAMER; a tabular action-value function is maintained; and the agent acts greedily with respect to that function. Unlike VI-TAMER, aVI-TAMER's "planning" consists of repeatedly considering different possible 40-step trajectories from its current state. Transitions from these trajectories provide updates for value iteration.[5] Planning trajectories are chosen by UCT [8], where the search tree is reset at the start of each time step to respect the corresponding change to the reward function $\hat{R}_H$ at that step, which generally makes past search results inaccurate. The confidence value for UCT is 1.

For this aVI-TAMER algorithm, the number of updates to each state's value differs considerably among states; in contrast, between sweep iterations in our value iteration implementation, all states have been updated an equal number of times. Instead of a balanced distribution of updates, state transitions that can quickly be reached from the current state receive many more temporal difference updates than transitions

[4] VI-TAMER is not synchronous in the strictest sense—where the entire sweep across state updates with the same value function—but we find this term "asynchronous" useful for distinguishing these two approaches.

[5] Using experienced transitions for action-value-updating value iteration is only valid for deterministic policies and transitions, as we have here. Also, note that the update mechanism is not itself Monte Carlo.

from less "local" states. For complex tasks in general, this bias towards local updating appears desirable, since an effective learning agent will likely visit regions of the state space that are worth understanding more often than areas that can be ignored during learning. Additionally, this local updating bias occurs in a large fraction of common RL algorithms (e.g., Sarsa(λ), Q-learning, and Monte Carlo tree search algorithms). We chose aVI-TAMER as a representative of this class of algorithms because it learns much more quickly than most other locally-biased algorithms. However, we recognize that there may be unforeseen benefits from the worse MDP-based performance of these other algorithms, similar to aVI-TAMER in the following section's failure-state experiment outperforming VI-TAMER in the episodic framing of the task.

This experiment departs from the baseline set of agent, task, and experiment specifications only by the inclusion of aVI-TAMER as the algorithm for two conditions. Since this investigation is focused on the effect of locally-biased updates on a high-γ algorithm, all three conditions calculate return with $\gamma = 0.99$. We are primarily interested in two conditions: VI-cont and aVI-cont, which respectively denote VI-TAMER and aVI-TAMER acting in a continuing version of the task. Note that this VI-cont condition is identical to the $\gamma = 0.99$ condition of the continuing-task experiment of the previous section; it is rerun here to account for the differing population that subjects will be drawn from. As a third condition called aVI-epis, we added aVI-TAMER in an episodic version to see what gains are retained by making the task continuing when updates are locally biased. The results of this experiment are shown in Figures 9 and 10. All results concern the full duration of training unless otherwise specified . 26 subjects were run per condition, resulting in the following number of samples by condition after filtering: VI-cont, 22; aVI-cont, 19; aVI-epis, 18.

We observe that the results for the VI-cont condition are similar to that of the equivalent $\gamma = 0.99$ condition in the continuing-task experiment (shown in Figures 5 and 9). The performance is insignificantly higher in this experiment by a Fisher-Success test with a 10-episode threshold ($p = 0.1085$). *This experiment further supports the assertion that the* VI-TAMER *algorithm successfully learns from human-generated reward at high discount factors in the continuing task.*

Effect of local bias in the continuing task

Comparing the two continuing conditions of this experiment—VI-cont and aVI-cont—*locally biased updates result in worse performance than* VI-TAMER*'s uniform updates* (Figure 9). This difference is highly significant by the 10-episode Fisher-Success test ($p = 0.0001$) and by a MWU-Episodes-Finished test ($p = 0.0016$). We also consider how many steps it took the agent to reach the goal *the first time*: a MWU-Time-To-Goal test is also highly significant ($mean_{VI-cont} = 93.45$; $mean_{aVI-cont} = 272.11$; $median_{VI-cont} = 70$; $median_{aVI-cont} = 250$; $p < 0.0001$), indicating that the change to locally biased updates slows early learning.

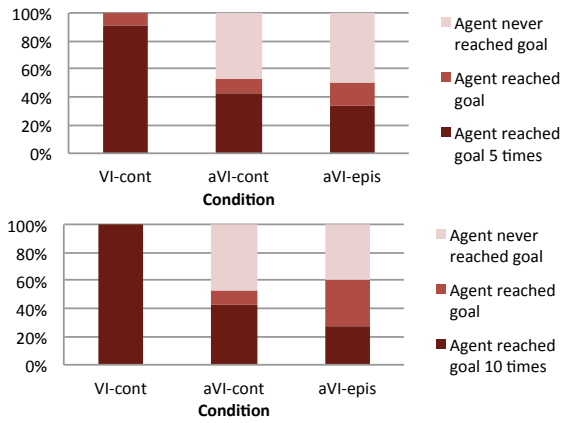


Figure 9. Success rates by condition for the local-bias experiment. Through their bias towards updating "local" states, the aVI-TAMER conditions create behavior that is farther from MDP-optimal for the current reward function than is behavior learned by VI-TAMER. The top plot shows success with the stopping points used for Figures 5 and 6, the first of 5 episodes or 300 time steps. The lower plot displays success with this experiment's stopping points, the first of 10 episodes or 450 time steps.

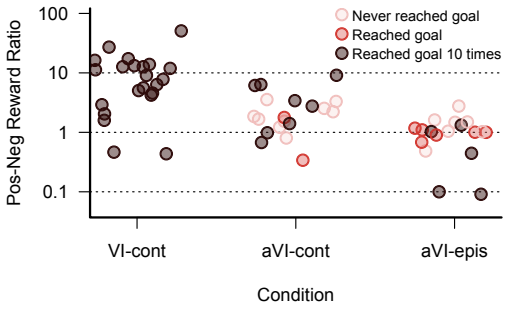


Figure 10. For the local-bias experiment, ratio of cumulative positive reward to cumulative negative reward given by each trainer (with x-axis jitter).

VI-TAMER effectively performs 4800 temporal difference updates per time step (40 sweeps $\times$ 30 states $\times$ 4 actions per state with deterministic transitions), compared to medians of 589 for the aVI-cont group and 1004 the aVI-epis group. Though aVI-TAMER's updates—dependent on the subject's computer—were less frequent, we doubt this difference is a meaningful factor in the results; a four-fold increase in aVI-TAMER's update speed would add less than one level of depth to its exhaustive search tree (which is extended by greedy rollouts to reach the full trajectory depth of 40).

Other than the number of updates per step, the only difference between aVI-TAMER and VI-TAMER in the continuing conditions is which state-action values are updated. We suspect that the locally-biased character of aVI-TAMER's updates is interacting with the positivity of human reward to lower the aVI-TAMER's algorithm's performance. Specifically, local bias causes nearby states to receive more updates, and the positivity of reward—with an action-value function initialized to 0, as in all experiments of this paper—makes states that are updated more often appear more desirable, consequently strengthening the local bias even further. In early learning, the aVI-TAMER agent will not learn the true values of states along its MDP-optimal path if it does not get near

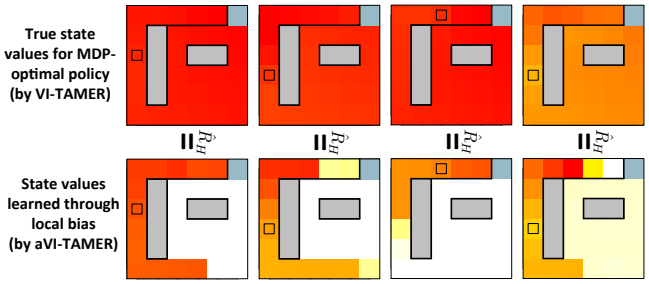


Figure 11. Heat maps showing value functions learned by VI-TAMER and aVI-TAMER, where both maps in a column was created through the same experience trajectory. Both learn from a training log from the VI-TAMER condition, containing a sequence of state-action pairs and time-stamped human reward signals. Training is stopped during the *first* step at which the reward function specifies the task-optimal path to the goal. 4 logs were chosen from the 22 of the VI-cont condition at varying time-to-first-goal values. The VI-TAMER algorithm used to create these heat maps performs 2000 update sweeps per step, increasing from the 40 per step in the experiment to approximate the MDP-optimal value function with further accuracy. The aVI-TAMER algorithm experiences the same trajectory, learning from 1000 planning trajectories per step. Each heat map shows state values after learning of $\hat{R}_H$ is stopped and the corresponding algorithm performs one step worth of updates. The deepest red corresponds to highest value of both value functions in the column, and white squares represent the lowest values, which is 0 or less in all four columns. The location of the agent at the trajectory's stopping point is shown by an agent-sized square.

those states, and policies that bring the agent back to previously experienced—and thus highly valued—states will be followed. The value-function heat maps in Figure 11 support this explanation of the performance differences; states that are far from experienced states often have not been updated even once and retain their original state values of 0.

One apparent solution to this problem of overly local exploration—optimistically initializing the action-value function—is not an option in complex domains for two reasons. First, optimism leads to thorough exploration of the state-action space. Such exhaustive exploration during training would frustrate, exhaust, and confuse the trainer, since such exploration would include much behavior that goes against the trainer's feedback, making the agent appear unresponsive and unable to learn for a considerable period. Thorough exploration would also sacrifice the fast learning that is one of the chief appeals of interactive shaping. Planning-only exploration from optimistic initialization, where the agent's actual actions are greedy, might be possible, but it would be greatly complicated by the following second reason that optimistic initialization is problematic: the reward function $\hat{R}_H$ is constantly changing during training. If the agent reinitializes its action-value function optimistically each time step, it forfeits all knowledge gained during previous time steps about action values under similar $\hat{R}_H$s, knowledge that should be critical to learning quickly to perform well with the new $\hat{R}_H$. On the other hand, if the agent only initializes at the beginning of training, then it will not explore with new $\hat{R}_H$s, largely removing the impact of optimistic initialization. Another option, optimistically initializing $\hat{R}_H$ directly, also generally creates exhaustive exploration.

Effect of episodicity for locally biased learning

Given that the locally biased updates of aVI-TAMER worsen performance compared to the approximately MDP-optimal performance of VI-TAMER, we now ask whether the choice of episodic or continuing formulations still has an appreciable effect in the context of aVI-TAMER's locally biased updates. Comparing these two conditions by the same three statistical tests applied above to compare the VI-cont and aVI-cont conditions, none are significant ($all\ p > 0.49$) and the corresponding means, medians, and proportions for the tests are quite similar across conditions.

The ratios of positive to negative reward in Figure 10 are negatively and significantly correlated with success for aVI-epis (Spearman-Success test, $p = 0.035$; $\rho = -0.50$) and uncorrelated for aVI-cont (Spearman-Success test, $p = 0.45$; $\rho = 0.19$), following the pattern observed from the continuing-task experiment for VI-TAMER. We also look at how often an agent *relapses* into sub-optimal task performance after an episode that is completed in minimal time. As mentioned previously, uniformly raising the value of all rewards by a constant value does not affect the MDP-optimal policy for a continuing task but often will for an episodic task. Consequently, we suspect that trainers will generally give more positive reward after their agent acts optimally (though their reward is not necessarily uniformly higher), which may cause more problems for the episodic aVI-TAMER condition. In the aVI-cont condition, of the 11 agents that finish an episode in minimal time during the first five episodes, 36.6% subsequently relapse into non-optimal behavior before reaching the 10-episode or 450-time-step endpoint. In the episodic condition, 77.7% of the 9 such agents do. This difference is marginally significant in a Fisher's test ($p = 0.0923$).

In overall performance, aVI-TAMER-guided updates do not clearly benefit from making the task continuing. However, we do observe that success in the continuing version still appears unrelated to reward positivity. Additionally, the relapse rate is lower in the continuing task. We suspect that these strengths of the continuing version are balanced against one advantage of the episodic version, that there is a simple but counterintuitive method for teaching the agent to act task-optimal: giving only negative reward. This hypothesis informs the following experiment.

FAILURE-STATE EXPERIMENT

In this experiment, the task is further manipulated to have a failure state that is closer to the start state than the goal state, as shown in Figure 12. In the episodic version of this task, both failure and goal states are absorbing state; in the continuing version, transitions to either create a transition to the start state. When this modified task is episodic, giving only negative reward will create MDP-optimal policies that fail by repeatedly looping through the failure state. In designing this task, we hope to represent the large class of tasks for which failure can be achieved more quickly than the goal (e.g., driving to a destination without crashing). In this task, we predicted that the continuing version would outperform the episodic version by a greater margin. This experiment tests both this prediction and the generality of the results from

the other experiments, which thus far have been exclusively demonstrated in a goal-only task.

We use the same algorithms and discount rate ($\gamma = 0.99$) as in the previous experiment, adding as a fourth condition the VI-TAMER algorithm with the episodic version of the task, making this a full 2x2 experiment. We refer to the new condition as VI-epis. Except for an additional instruction to make the agent avoid the failure state—a "black pit that takes [the agent] back to the start"—the experiment was conducted identically as the baseline experiment until the agent reaches the goal a 10th time. If a trainer reaches that point, instead of stopping the experiment we allow the user to continue training until all 450 time steps have passed. We made this adjustment to add resolution among the most successful trainers (e.g., between trainers who would get the agent to the goal 11 times or 18 times). 20 subjects were run per condition; after filtering the data (see *The experiment* for details), the number of subjects were as follows: 15 for VI-cont, 16 for VI-epis, 14 for aVI-cont, and 14 for aVI-epis.

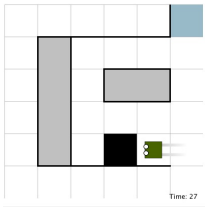


Figure 12. The task with a failure state added.

Results from this experiment are described by Figures 14 and 15 and the table of statistical test results in Figure 13. Generally speaking, we observe the same performance patterns as in the experiments with the goal-only task, though these patterns are less pronounced. VI-cont performs best in all comparisons, significantly so in 5 of 6. For both VI-TAMER and aVI-TAMER algorithms, performance is better for the continuing version of the task, except for aVI-TAMER's mean rank of time to goal from the MWU-Time-To-Goal test, where aVI-epis is insignificantly better than aVI-cont. Interestingly, the new comparison between VI-epis and aVI-epis reveals that the agent performs better in the algorithm with locally biased updates, aVI-epis. We suspect that this result arises because the sub-optimality of aVI-TAMER makes it less likely to find existing positive circuits, which could prevent the agent with a $\gamma = 0.99$ reward-based objective from going to the goal. In other words, failing to achieve an undesirable objective can be better than achieving it. This result raises the surprising prospect that the combination of an agent objective that poorly aligns MDP-optimal and task-optimal behavior combined with an agent that poorly maximizes that objective might produce better results than can be achieved by any considerably sub-optimal agent attempting to maximize a perfectly aligned objective.

We also examine reward positivity (Figure 15). For both VI-TAMER and aVI-TAMER, reward was more positive in the continuing version of the task, which fits observations from the two experiments on the goal-only task. In VI-epis, there is a marginally significant negative correlation between success and reward positivity by the Spearman-success test ($\rho = -0.4266$, $p = 0.0994$); this correlation is insignificant for the other 3 conditions. Overall, we see that episodicity has a smaller effect on reward positivity in this failure-state task than in the goal-only task. We suspect that this observation is connected to a previously described effect of adding the failure state: that the simple strategy of always giving negative reward, without regard for the state or action, no longer creates MDP-optimal behavior that is also task-optimal.

From analyzing these results, we believe that adding the failure state affected the ease of training in both positive and negative ways. As an alternate absorbing state to the goal, the failure state generally forces trainers to give more discriminating reward (e.g., the arbitrarily all-negative strategy for the episodic version becomes unsuccessful). In comparison to the goal-only task, on the other hand, the aVI-TAMER algorithm preformed better overall in both the continuing and episodic versions of the failure-task; this increase might be due in part to the failure state being used as an intermediate "goal" that the learner makes updates for, goes to, and then gets experience and reward for those states near it, which then help it go to the real goal. Because of these multiple factors that likely affect performance (as well as randomness and different subject populations at different experiment times), we hesitate to draw strong conclusions from comparisons *across* experiments. However, we can say that this experiment does not reveal an increased performance difference between the aVI-epis and aVI-cont conditions, as we had predicted.

Nonetheless, the results from this experiment give additional empirical support for the generality of the patterns that have been identified previously in this paper, showing that they appear when the task contains both desirable and undesirable absorbing state. Most important among these patterns are that (1) performance is better for a continuing formulation of the task—especially when the agent acts approximately MDP-optimally as with the VI-TAMER algorithm—and that (2) the choice of the best algorithm for complex tasks, where MDP-optimal behavior is generally intractable, is a challenging direction for future work.

CONCLUSION

Any solution to the problem of interactive shaping—i.e., learning sequential tasks from human-generated reward—requires the definition of a reward-based objective and an agent algorithm. Building on previous work [7], this paper examines the relationship between reward discounting rates, episodicity, reward positivity, acting approximately MDP-optimally or not, and task performance. These relationships are examined thoroughly in a goal-only task and are examined further for non-myopic discounting only in a task with both goal and failure states. The table in Figure 16 summarizes our findings.

We see four main contributions in this paper:

1. empirically finding that for approximately MDP-optimal agents, converting the otherwise episodic grid-world task to a continuing task (a) enables successful training at non-myopic discount rates, (b) removes negative correlations between reward positivity and discount factor values, and (c) removes negative correlations between reward positivity and task performance within non-myopic conditions;

Test	VI-epis: Fisher Success	VI-epis: MWU Eps. Finished	VI-epis: MWU Time to Goal	aVI-cont: Fisher Success	aVI-cont: MWU Eps. Finished	aVI-cont: MWU Time to Goal	aVI-epis: Fisher Success	aVI-epis: MWU Eps. Finished	aVI-epis: MWU Time to Goal
VI-cont	**p<0.0001, VI-cont**	**p=0.0022, VI-cont**	**p=0.0128, VI-cont**	p=0.2635, VI-cont	**p=0.0574, VI-cont**	**p=0.0257, VI-cont**			
VI-epis							**p=0.0365, aVI-epis**	**p=0.0615, aVI-epis**	p=0.5222, aVI-epis
aVI-cont							p=0.4401, aVI-cont	p=0.5823, aVI-cont	p=0.4593, aVI-epis

Figure 13. Statistical tests for comparisons of interest. Each cell contains the p-value for the corresponding test as well as the name of the condition that performed better on the metric, which could be highest proportion of success, highest mean ranking of episodes finished, or lowest mean ranking of time to first goal. (The Mann-Whitney U test converts samples to rankings.) Cells with p-values below 0.1 are emboldened.

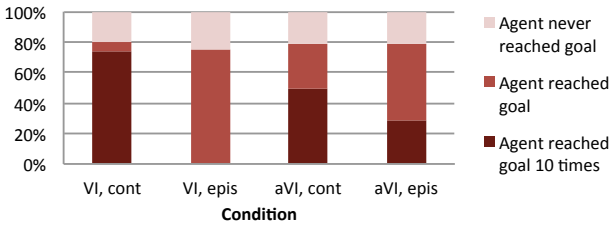


Figure 14. Success rates by condition for the failure-state experiment, which investigates the effects of locally-biased updates (the aVI-TAMER conditions) and episodicity when there is both a goal state and a failure state.

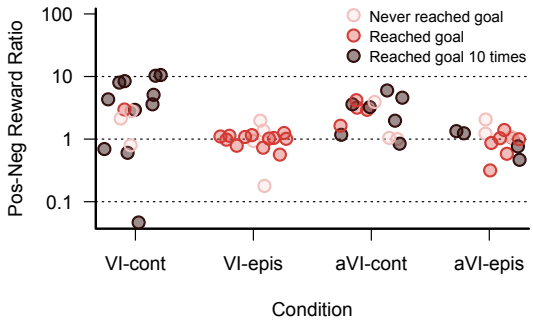


Figure 15. For the same experiment, the ratio of cumulative positive reward to cumulative negative reward given by each trainer (with x-axis jitter).

2. achieving the first known instance of consistently successful training of a non-myopic agent by live, human-generated reward signals;

3. demonstrating that successfully trained agents with non-myopic objectives learn higher-level task information, making them more robust to changes in their environments and better able to act from states in which they lack experience;

4. and showing that when the agent's MDP-based performance is worsened—as it must be for complex tasks—by the common practice of locally biased learning, task performance worsens significantly in continuing tasks.

This paper represents a step forward in the effort to create effective algorithms for learning from human reward. We note,

	Approximately MDP-optimal (VI-TAMER)	Locally biased learning (UCT-driven aVI-TAMER)
Episodic	• Effective with **few** trainers • Success **negatively correlated** with reward positivity in both tasks	• Effective with **some** trainers • Success **negatively correlated** with reward positivity in goal-only task
Continuing	• Effective with **almost all** trainers • Success **independent** of reward positivity in both tasks	• Effective with **some** trainers • Success **independent** of reward positivity in both tasks

Figure 16. Qualitative summary of this paper's experimental conclusions on non-myopic learning.

however, that more analysis is required to establish the generality of our observations. Changing the reward-giving interface, the mapping of reward cues (e.g. keys) to scalar values, the instructions to trainers, our algorithmic choices, and the task to be learned—though all carefully chosen to avoid overt bias—might create qualitatively different results.

From this research, we believe that the greatest future contributions of learning from human reward will come from non-myopic objectives and will likely be in continuing tasks. However, we expect that naively designed agents with biases towards local updates—agents often well-suited for complex tasks—will ineffectively learn from human reward even in continuing tasks; the problems of reward positivity extend beyond episodic tasks. Identifying algorithms that learn non-myopically from human-generated reward in complex domains—where approximately MDP-optimal behavior will likely be impossible—remains a critical research question.

Acknowledgments

This work has taken place in the Personal Robots Group at the MIT Media Lab and in the Learning Agents Research Group (LARG) at UT Austin. LARG research is supported in part by NSF (IIS-0917122), ONR (N00014-09-1-0658), and the FHWA (DTFH61-07-H-00030). We thank George Konidaris and Rich Sutton for fruitful discussions on discounting human reward.

REFERENCES

1. Argall, B., Chernova, S., Veloso, M., and Browning, B. A survey of robot learning from demonstration. *Robotics and Autonomous Systems 57*, 5 (2009), 469–483.
2. Gelly, S., and Silver, D. Achieving master level play in 9× 9 computer go. In *Proceedings of AAAI* (2008), 1537–1540.
3. Hester, T., and Stone, P. Learning and using models. In *Reinforcement Learning: State of the Art*, M. Wiering and M. van Otterlo, Eds. Springer Verlag, Berlin, Germany, 2011.
4. Isbell, C., Kearns, M., Singh, S., Shelton, C., Stone, P., and Kormann, D. Cobot in LambdaMOO: An Adaptive Social Statistics Agent. *Proceedings of The 5th Annual International Conference on Autonomous Agents and Multiagent Systems (AAMAS)* (2006).
5. Knox, W. B. *Learning from Human-Generated Reward*. PhD thesis, Department of Computer Science, The University of Texas at Austin, August 2012.
6. Knox, W. B., and Stone, P. Interactively shaping agents via human reinforcement: The TAMER framework. In *The 5th International Conference on Knowledge Capture* (September 2009).
7. Knox, W. B., and Stone, P. Reinforcement learning from human reward: Discounting in episodic tasks. In *21st IEEE International Symposium on Robot and Human Interactive Communication (Ro-Man)* (September 2012).
8. Kocsis, L., and Szepesvári, C. Bandit based monte-carlo planning. *Machine Learning: ECML 2006* (2006), 282–293.
9. León, A., Morales, E., Altamirano, L., and Ruiz, J. Teaching a robot to perform task through imitation and on-line feedback. *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications* (2011), 549–556.
10. Pilarski, P., Dawson, M., Degris, T., Fahimi, F., Carey, J., and Sutton, R. Online human training of a myoelectric prosthesis controller via actor-critic reinforcement learning. In *IEEE International Conference on Rehabilitation Robotics (ICORR)*, IEEE (2011), 1–7.
11. Suay, H., and Chernova, S. Effect of human guidance and state space size on interactive reinforcement learning. In *20th IEEE International Symposium on Robot and Human Interactive Communication (Ro-Man)* (2011), 1–6.
12. Sutton, R., and Barto, A. *Reinforcement Learning: An Introduction*. MIT Press, 1998.
13. Tenorio-Gonzalez, A., Morales, E., and Villaseñor-Pineda, L. Dynamic reward shaping: training a robot by voice. *Advances in Artificial Intelligence–IBERAMIA* (2010), 483–492.
14. Thomaz, A., and Breazeal, C. Teachable robots: Understanding human teaching behavior to build more effective robot learners. *Artificial Intelligence 172*, 6-7 (2008), 716–737.