

Towards Finite-Sample Convergence of Direct Reinforcement Learning

Shiau Hong Lim and Gerald DeJong

Dept. of Computer Science, University of Illinois, Urbana-Champaign
{shonglim, dejong}@cs.uiuc.edu

Abstract. While direct, model-free reinforcement learning often performs better than model-based approaches in practice, only the latter have yet supported theoretical guarantees for finite-sample convergence. A major difficulty in analyzing the direct approach in an online setting is the absence of a definitive exploration strategy. We extend the notion of admissibility to direct reinforcement learning and show that standard Q-learning with optimistic initial values and constant learning rate is admissible. The notion justifies the use of a greedy strategy that we believe performs very well in practice and holds theoretical significance in deriving finite-sample convergence for direct reinforcement learning. We present empirical evidence that supports our idea.

1 Introduction

Indirect (or model-based) reinforcement learning (RL) derives a policy from an estimation of state utilities and transition probabilities. It has been shown that indirect RL can provide near-optimal performance in polynomial time, e.g. E^3 (Kearns and Singh, 1998), R-max (Brahman and Tennenholtz, 2002). That is, given a Markov decision problem (MDP), a good solution can be found efficiently.

Direct (or model-free) reinforcement learners, on the other hand, estimate the utility of performing an action in a state. Thus, they learn a policy directly. But there is no corresponding formal guarantee that direct RL can produce near-optimal solutions to MDPs. However, direct methods, such as Q (Watkins, 1989) or SARSA (Rummery and Niranjan, 1994) often outperform their indirect counterparts.

While the theoretical analyses provide an important formal assurance, they appear to contribute little in the way of insights as to how to improve actual reinforcement learning algorithms. Furthermore, the particular algorithms that possess the formal guarantees (E^3 and R-max), can perform quite poorly in practice. Figure 1 shows the number of learning examples needed for convergence to a good policy as a function of problem difficulty. Here we employ a simple family of Markov decision problems we call Task 1. It will be described in section 4.

Note that the ordinate axis is logarithmic, indicating that the performance of E^3 is many orders of magnitude worse than Q. Furthermore, this Q explores purely randomly. Since Q is an off-policy learner, it acquires the optimal policy even though every action is chosen with equal probability during learning.

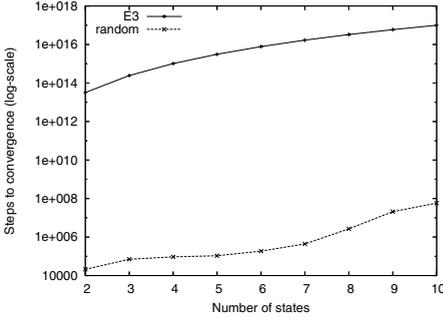


Fig. 1. Task 1

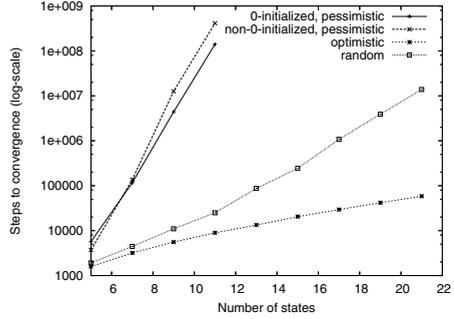


Fig. 2. Task 2 - Different initial Q-values

Clearly, Task 1 represents only a narrow family of MDPs but the results suggest that a theoretical analysis of direct RL may shed light on this discrepancy and may point toward even more efficient learning algorithms.

Experience suggests that there is an interesting structure differentiating direct reinforcement learners. Figure 2 shows the behavior of three Q learners that differ only in initialization. Following the principle of optimism under uncertainty (Brahman and Tennenholtz, 2002, Sutton and Barto, 1998, Koenig and Summons, 1996) one learner is initialized to optimistic random Q values, one to pessimistic random Q values, and one to all zeros. Since all the rewards are positive, this is also a pessimistic initialization. All 3 cases employ a constant learning rate ($\alpha = 0.05$) with ϵ -greedy exploration ($\epsilon = 0.1$). For comparison we also include as a fourth condition, the purely random Q learner of Fig. 1. All are exercised on a simple but more challenging problem, Task 2, which is also described in section 4. Figure 2 illustrates that optimistic initial Q-values can result in an exponential improvement in learning rate, suggesting that any attempt to derive a polynomial convergence bound without accounting for this distinction will likely be futile.

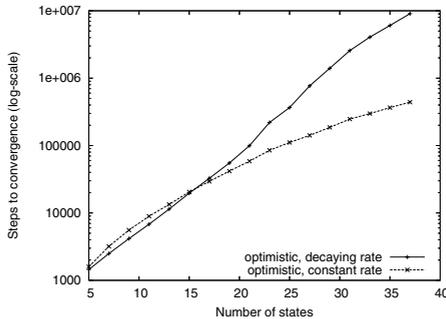


Fig. 3. Task 2 - Different learning rates

Another tension between practical and theoretical understanding concerns the learning rate, α . Asymptotic convergence of reinforcement learning dictates that $\sum \frac{1}{\alpha_i}$ must diverge but $\sum (\frac{1}{\alpha_i})^2$ must converge uniformly (Bertsekas and Tsitsiklis, 1996). But Sutton and Barto (1998) note that often a constant learning rate performs much better than a decaying one. Figure 3 compares two versions of the successful optimistic Q learner of Fig. 2. One employs the constant rate of 0.05 while the other employs a decaying rate of $\frac{1}{i^p}$ for the i^{th} update of a state-action pair. The results shown are based on $p = 0.5001$, which is nearly the slowest possible decay that fulfills the stochastic approximation requirement for asymptotic convergence. A larger p will result in slower convergence. The results clearly favor the constant rate, at least on Task 2.

We believe there is a theoretical justification for the observed good behavior of direct reinforcement learning under optimistic initial Q-values and a small constant learning rate. The motivation of this paper is to share our theoretical findings and empirically-guided intuitions concerning such a tight polynomial convergence bound. Central is an adaptation to stochastic domains of Koenig and Simmons (1996) notion of “admissible” Q-values and Q-updates. We discuss the implications of this notion and explain the empirical observations above, as well as the results from more challenging problems.

2 Previous Work

Koenig and Simmons (1996) have shown that online Q-learning acquires an optimal policy in polynomial time if the initial Q-values are optimistic, but only for deterministic environments. The notion of “admissible” Q-values and Q-updates (similar to admissibility in A* search) plays a major role in their results. As in A*, admissibility ensures that a greedy algorithm does not miss the true optimal solution. The notion unfortunately does not directly extend to the stochastic domains. Since Q-values are defined as expected values and are estimated through sampling, any action may (temporarily) result in a Q-value that is lower than its true average and therefore risks being permanently missed by an algorithm that chooses actions greedily. This problem is called “sticking” by Kaelbling (1990) due to the fact that the algorithm may not be able to abandon a sub-optimal solution. The workaround for the problem is usually to allow some randomness in selecting actions instead of being purely greedy. It is, however, unclear how to correctly balance the tradeoff between exploration and greedy exploitation. Clearly, a purely random action selection is not desirable; a random walk may require an exponential number of steps, on average, to reach a particular state.

Even when following the optimal policy in the stochastic case, the expected number of steps to reach a particular state (e.g. the goal state) may be exponential in the size of the state space. This makes it impossible to solve a problem in polynomial time (with respect to the number of states) unless some form of relaxation is introduced. It has been shown for the model-based reinforcement learning algorithm, that a PAC learning setting allows polynomial

time convergence to a near-optimal solution with high probability. In E^3 for example, this problem is addressed by defining the “ ϵ -return mixing time” for a policy (with respect to the average return), and limiting attention to the class of policies with bounded mixing times. Alternatively, when discounted return is used with bounded rewards, one only needs to consider the states that are close (given the discount rate) to the initial state and reachable with significant probability. Other states simply cannot sufficiently influence the expected utility of decisions. These suggest that a similar setting should be applicable to the direct algorithms.

A critical advantage in analyzing a model-based algorithm is that the exploration can be separated from the actual learning updates. Due to strong statistical guarantees for independent samples, an accurate model can be built with a reasonable amount of sampling, and an appropriate exploration strategy that either exploits or explores (Kearns and Singh, 1998) can be derived from any partial model. In the direct, online case, such as the standard Q-learning algorithm, the coupling between noise from sampling and the policy improvement itself, frustrate assessing the benefit of exploration based upon Q-values alone. This also suggests that if certain properties of the estimated Q-values (such as admissibility) can be preserved, one might have a more easily analyzed strategy for exploration.

In (Kearns and Singh, 1999), a parallel sampling model is employed, and a direct algorithm (phased Q-learning) based on this model can achieve polynomial-time convergence, in terms of number of calls to the parallel sampling procedure. This procedure can in turn be simulated by a stationary (possibly stochastic) policy π that defines an ergodic Markov process in the MDP. The overall complexity depends on the mixing time of π to reach its stationary distribution. In practice, however, such policy might not exist, and even if it does, its mixing time might be prohibitively large with exponential dependency on the number of states and actions (the policy could simply be a random walk). We encounter this in our Task 2.

Even-Dar and Mansour (2003) analyzed the convergence rate for Q-learning with respect to different choice of the learning rate, α . They show that a linearly decaying rate ($\alpha = 1/k$ for the k^{th} update) has an exponential dependence on $\frac{1}{1-\gamma}$. This eliminates the possibility of polynomial-time convergence for direct Q-learning based on linearly decaying α . For a polynomial learning rate ($\alpha = 1/k^w$ where $0.5 < w < 1$), the convergence rate depends largely on what was called the “covering time”, which is a bound on the number of learning steps within which every state will be visited. Again, the “covering time” depends on the exploration strategy, and may not be polynomially bounded.

3 Admissibility of Q-Learning

We have seen that the major difficulty in analyzing direct, online reinforcement learning comes mostly from the fact that different exploration strategy may result in very different performance. Analysis that decouples the

exploration strategy from the learning algorithm, while useful theoretically, hides the problem that we face in practice. We have also seen that “optimism under uncertainty”, which is a useful notion in AI, plays significant roles in the design and verification of many algorithms. It can be realized, in the case of reinforcement learning, by using optimistic initial values and exploring greedily, as long as the probability of getting stuck is low. This motivates the definition of a slightly relaxed notion of admissibility and we show that Q-learning with optimistic initial Q-values and a constant learning rate is admissible.

We consider standard Q-learning on an MDP with a finite set of states S and a finite set of actions A . We define the optimal policy π^* with respect to a discounted return with discount factor $0 \leq \gamma < 1$ where the optimal action for a state s is given by $\pi^*(s)$. The Q-value of a state-action pair (s, a) at an unspecified point during learning is denoted by $Q(s, a)$. The Q-value of a state-action pair (s, a) after k updates (of that particular pair) is denoted by $Q_k(s, a)$. The optimal Q-value of a state-action pair (i.e. with respect to the optimal policy) is denoted by $Q^*(s, a)$, and the optimal value (utility) of a state s is given by $V^*(s) = \max_a Q^*(s, a)$. We assume discrete time step $t \in \{0, 1, 2, \dots\}$, where at each time step, exactly one state-action pair will be executed and updated with Q-update:

$$Q^{t+1}(s, a) = (1 - \alpha_t)Q^t(s, a) + \alpha_t(r_t + \gamma V^t(s'))$$

where $Q^t(s, a)$ denotes the Q-value of (s, a) at time t , α_t denotes the learning rate used at time t , r_t denotes the reward received after executing the action at time t , s' denotes the next state visited, and $V^t(s') = \max_{a'} Q^t(s', a')$. We assume that the magnitude of any reward is bounded by $R_{max} > 0$, and therefore the magnitude of any discounted return is bounded by $V_{max} = \frac{R_{max}}{1-\gamma}$.

We define the notion of admissibility for direct, online Q-learning as follows:

Definition 1. *A Q-learning algorithm is admissible if after every Q-update, with probability at least $1 - \delta$,*

$$Q(s, a) \geq Q^*(s, a) - \epsilon, \quad \forall s \in S, \forall a \in A .$$

We would like to establish the fact that Q-learning with optimistic initial Q-values and a constant learning rate is admissible. The following lemma will help.

Lemma 1. *For any state-action pair (s, a) , after k updates with constant learning rate α , with probability at least $1 - \delta$,*

$$Q_k(s, a) - Q^*(s, a) \geq -\beta + (1 - \alpha)^k \Delta_0(s, a) + \gamma \alpha \sum_{i=1}^k (1 - \alpha)^{k-i} \Delta^{t(i)}(s'_i)$$

and

$$|Q_k(s, a) - Q^*(s, a)| \leq \beta + (1 - \alpha)^k |\Delta_0(s, a)| + \gamma \alpha \sum_{i=1}^k (1 - \alpha)^{k-i} |\Delta^{t(i)}(s'_i)|$$

where $\Delta_0(s, a) = Q_0(s, a) - Q^*(s, a)$, s'_i is the next state visited after the i^{th} update, $t(i)$ is the time step during the i^{th} update, $\Delta^{t(i)}(s'_i) = V^{t(i)}(s'_i) - V^*(s'_i)$, and

$$\beta = V_{max} \sqrt{\frac{\alpha}{2(2-\alpha)} \ln \frac{2}{\delta}} .$$

Proof. For a particular state s and action a , the Q-value after the k^{th} update is given by

$$Q_k(s, a) = (1 - \alpha)^k Q_0(s, a) + \alpha \sum_{i=1}^k (1 - \alpha)^{k-i} \left(r_{t(i)} + \gamma V^{t(i)}(s'_i) \right)$$

and therefore

$$\begin{aligned} Q_k(s, a) &= (1 - \alpha)^k \left(Q^*(s, a) + \Delta_0(s, a) \right) + \\ &\quad \alpha \sum_{i=1}^k (1 - \alpha)^{k-i} \left(r_{t(i)} + \gamma V^*(s'_i) + \gamma \Delta^{t(i)}(s'_i) \right) \\ &= (1 - \alpha)^k Q^*(s, a) + \alpha \sum_{i=1}^k (1 - \alpha)^{k-i} \hat{Q}_i(s, a) \\ &\quad + (1 - \alpha)^k \Delta_0(s, a) + \gamma \alpha \sum_{i=1}^k (1 - \alpha)^{k-i} \Delta^{t(i)}(s'_i) \end{aligned}$$

where $\hat{Q}_i(s, a)$ is an unbiased sample of $Q^*(s, a)$. Since $|\hat{Q}_i(s, a)| \leq V_{max}$ and each sample is weighted by $\alpha(1 - \alpha)^{k-i}$, we define

$$\bar{Q}_k(s, a) = \alpha \sum_{i=1}^k (1 - \alpha)^{k-i} \hat{Q}_i(s, a)$$

as sum of k random variables with bounded values. Then

$$E(\bar{Q}_k(s, a)) = \alpha \sum_{i=1}^k (1 - \alpha)^{k-i} Q^*(s, a) = (1 - (1 - \alpha)^k) Q^*(s, a) .$$

By Hoeffding's inequality,

$$Pr(|\bar{Q}_k(s, a) - E(\bar{Q}_k(s, a))| \geq \beta) \leq 2e^{-2\beta^2 / \sum_{i=1}^k \alpha^2 (1-\alpha)^{2(k-i)} V_{max}^2} \leq 2e^{-\frac{2\beta^2(2-\alpha)}{V_{max}^2 \alpha}} .$$

Then, with probability at least $1 - \delta$, $|\bar{Q}_k(s, a) - E(\bar{Q}_k(s, a))| \leq \beta$. Since

$$\begin{aligned} Q_k(s, a) - Q^*(s, a) &= \bar{Q}_k(s, a) - E(\bar{Q}_k(s, a)) \\ &\quad + (1 - \alpha)^k \Delta_0(s, a) + \gamma \alpha \sum_{i=1}^k (1 - \alpha)^{k-i} \Delta^{t(i)}(s'_i) \end{aligned}$$

the Lemma follows. □

Due to the constant learning rate, we cannot expect the error in any estimated Q-value to become arbitrarily small since there is always a “window” of the most recent updates with significant weight that potentially increase the error in the Q-value. Given any rate α , there is always potential error with the magnitude of β introduced with each update. In fact, if all Q-values are initialized to the true optimal Q-values of the optimal policy and a greedy exploration is used, errors will be introduced within each Q-value in the order of β due to the inherent stochasticity of the problem. Lemma 1 suggests that we can bound this error by choosing a small enough α . This leads to the following result.

Proposition 1. *If the Q-value for every state-action pair is initialized such that*

$$\Delta_0(s, a) = Q_0(s, a) - Q^*(s, a) \geq 0, \quad \forall s \in S \forall a \in A$$

and with a constant learning rate

$$\alpha \leq 2 \left(\frac{\epsilon^2(1-\gamma)^2}{V_{max}^2 \ln \frac{2}{\delta}} \right)$$

then the standard Q-learning algorithm is admissible.

Proof. We will show that for all $t \geq 0$, s and a ,

$$Q^t(s, a) \geq Q^*(s, a) - \epsilon$$

by strong induction on t . The statement holds trivially when $t = 0$ (before any updates) since all Q-values are initialized optimistically. Assume that it holds for $0 \leq t \leq T$. Let the update at time step T be at state s and for action a . Assume that this is the k^{th} update for (s, a) . By Lemma 1, with probability at least $1 - \delta$,

$$Q_k(s, a) - Q^*(s, a) \geq -\beta + (1 - \alpha)^k \Delta_0(s, a) + \gamma \alpha \sum_{i=1}^k (1 - \alpha)^{k-i} \Delta^{t(i)}(s'_i) .$$

By the definition of β (in Lemma 1), $\alpha \leq 2 \left(\frac{\epsilon^2(1-\gamma)^2}{V_{max}^2 \ln \frac{2}{\delta}} \right)$ implies that $\frac{\beta}{1-\gamma} < \epsilon$. Let $b_i = \operatorname{argmax}_{a'} Q^{t(i)}(s'_i, a')$. By the induction hypothesis,

$$\Delta^{t(i)}(s'_i) = Q^{t(i)}(s'_i, b_i) - V^*(s'_i) \geq Q^{t(i)}(s'_i, b) - Q^*(s'_i, b) \geq -\epsilon$$

and therefore

$$Q_k(s, a) - Q^*(s, a) \geq -\epsilon(1 - \gamma) - \gamma \alpha \sum_{i=1}^k (1 - \alpha)^{k-i} \epsilon \geq -\epsilon . \quad \square$$

Proposition 1 essentially provides a guideline to select a learning rate small enough such that the (estimated) Q-values are never too far below the true optimal Q-value:

$$\alpha = 2 \left(\frac{\epsilon^2(1-\gamma)^2}{V_{max}^2 \ln \frac{2}{\delta}} \right)$$

Note that the definition of admissibility only requires that the probability of error after each update is bounded by δ . This means that the probability of error increases with the number of updates. This is partially due to the fact that we use a constant learning rate. However, note that the cost of having higher confidence (logarithmic dependency) is much lower than having higher accuracy (quadratic dependency). This means that as long as the total number of updates needed is a polynomial (in terms of all other parameters), the effect on α is relatively small.

4 Experiments

We have established the admissibility of Q-learning with optimistic initial values and a constant learning rate. The natural consequence is that a greedy exploration strategy would seem to be the most efficient, since it always focuses on the most promising region of the state space. We support this by first analyzing the experiment results for both Task 1 and Task 2.

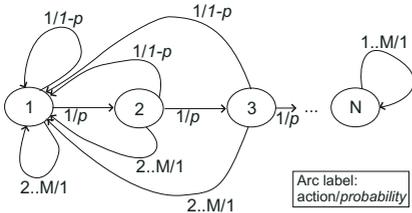


Fig. 4. Task 1

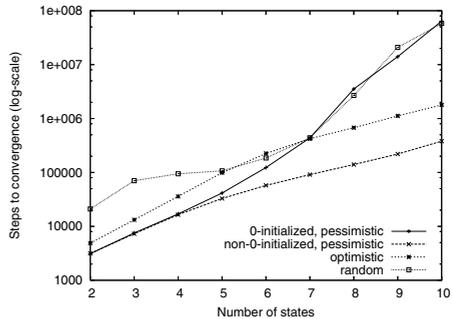


Fig. 5. Task 1 Results

Figure 4 shows task 1, which is an MDP with N states ($N \geq 2$) and M actions. State 1 is a fixed, initial state, and state N is an absorbing, goal state. Executing action 1 in any non-goal state s results in state $s + 1$ with probability p and in state 1 with probability $1 - p$. Executing actions $2 \leq a \leq M$ in any non-goal state results in state 1 with probability 1. All transitions give zero rewards except the transition to the goal state, where a reward $R = 10$ will be received.

The learning results for direct Q-learning with various setups are shown in Fig. 5. Note that the y-axis uses logarithmic scale. Each graph is the average of 10 independent learning instances. For each instance, we evaluate the resulting policy after every episode (of 200 steps each) and stop when more than 95% of the last 500 episodes end with the optimal policy. We omit error bars in all the results for both Task 1 and Task 2 since the observed variances are barely visible under the logarithmic scale.

Task 1 is a problem where the average number of steps to reach the goal is exponential in the number of states (in the order of $(\frac{1}{p})^{N-1}$), even with the optimal policy. As mentioned in section 2, under the PAC model, we allow a small amount of error, and the cost (in terms of learning time) to reduce the error will be considered reasonable as long as it is a polynomial in $\frac{1}{\epsilon}$. This is best illustrated by an example. Consider task 1, where the optimal Q-value for the initial state is given by:

$$Q^*(1, 1) = \frac{Rp(1 - p\gamma)(p\gamma)^{N-2}}{1 - \gamma + \gamma(1 - p)(\gamma p)^{N-1}} .$$

The optimal policy for task 1 requires that action 1 be chosen in every non-goal state. Any deviation from the optimal policy will result in a policy π with expected return $Q^\pi(1, \pi(1)) = 0$. This also implies that if $Q^\pi(1, \pi(1)) - Q^*(1, 1) < \epsilon$, then every policy satisfies the requirement and therefore no learning is needed. We therefore assume that $Q^\pi(1, \pi(1)) - Q^*(1, 1) < \epsilon$, which implies:

$$\frac{Rp(1 - p\gamma)(p\gamma)^{N-2}}{1 - \gamma + \gamma(1 - p)(\gamma p)^{N-1}} \geq \epsilon \quad \Rightarrow \quad \left(\frac{R}{1 - \gamma}\right)\frac{1}{\epsilon} \geq \left(\frac{1}{p}\right)^{N-1} .$$

Regardless of the value of p , whenever ϵ is small enough such that finding the optimal policy is necessary, the average number of steps to reach the goal state is in the order of $\mathcal{O}\left\{\left(\frac{R}{1-\gamma}\right)\frac{1}{\epsilon}\right\}$. This renders the problem tractable in terms of the amount of learning needed with respect to the acceptable error.

We see that in task 1, the pessimistic case with non-zero initial Q-values actually performs better than the optimistic case, but they both have roughly the same rate of growth in learning time as the size of the problem grows. The difference in performance can be accounted by the fact that the optimistic case starts with a much higher Q-value estimate for most of the states and needs more learning updates to approach the true Q-values. On the other hand, the pessimistic case with zero-initialized values has a significantly higher rate of growth, and actually performs no better than the purely random strategy (especially when N is large). Since task 1 naturally requires an exponential number of learning steps (with respect to N), one might ask whether the error that we discussed above applies. It only applies whenever the optimal policy is executed. Since in the case of zero-initialized Q-values, the initial phase of learning is essentially a random walk (the Q-values remain unchanged) and in Task 1, it takes an exponential number of trials (in M) to realize a sequence of actions that corresponds to the optimal policy, this cost is in addition to the fact that the average number of steps to reach the goal through the optimal policy is exponential in N.

Task 1 does not reveal the problem with using pessimistic (but non-zero) initial Q-values since there is only one goal state and one optimal policy. Figure 6 shows Task 2, whose learning results are shown in Fig. 2 (Section 1). In this MDP, there are two absorbing states (N and -N). The states that lead to state N behaves exactly like those in Task 1, that is, progress is made only when action 1 is executed. On the other hand, progress is made toward state -N when any action except action 1 is executed. In our actual experiment, we use $p = 0.8$

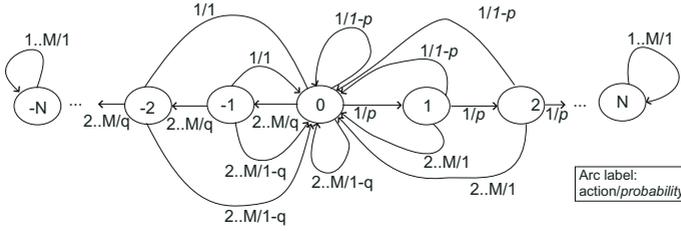


Fig. 6. Task 2

and $q = 0.99$. The reward for reaching state N is 10 while the reward for reaching state $-N$ is 1. This makes state N harder to reach, but more desirable in terms of the actual return (for small N). We use ϵ -greedy exploration with $\epsilon = 0.1$ for both pessimistic cases. Both cases converge to the wrong goal very quickly, and get “stuck” on this policy. It takes an extremely large number of trials to escape from this situation, relying on the randomness (ϵ) in the exploration strategy. Figure 2 shows that a purely random exploration actually performs better in this case since sticking cannot happen. Task 2 shows a clear performance advantage for the strategy with optimistic initial values.

4.1 More Challenging Problems

We further illustrate the advantage of greedy exploration with optimistic initial Q-values with more realistic problems. We use the acrobot swing-up problem as described in Sutton and Barto, (1998). We first run the acrobot using the original configuration, then we add friction and noise to the system to increase its difficulty. We run 10 independent instances of both optimistic and pessimistic strategies on both experiments. The results are shown in Figs. 7 and 8. We observe that when the difficulty of the problem increases (in the sense that less random action sequences can reach the goal) the difference in performance becomes more significant as well.

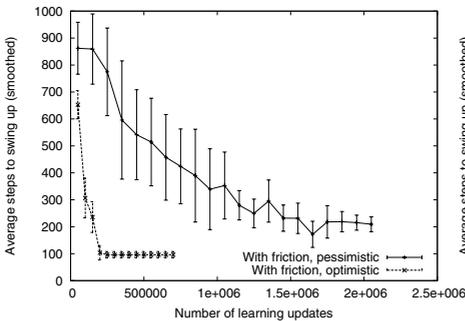


Fig. 7. Acrobot (No friction)

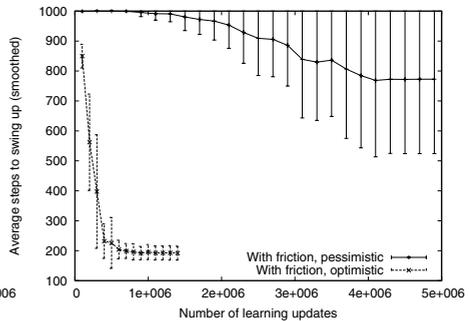


Fig. 8. Acrobot (With friction)

We repeat the experiment on a 2-dimensional mountain-car problem, which is similar to the mountain-car problem described in (Sutton, 1996), but extended to 2 dimensions (for the navigation). We use a steepness factor to control the difficulty of the problem. The results are shown in Figs. 9 and 10. We observe the same performance pattern as in the acrobot problem when we increase the steepness of the mountain. It is conceivable that good action sequences become rarer as problems become harder. We believe that greedy exploration with optimistic initial Q-values results in a rather “uniform” but efficient search in the policy space. This explains the relatively minor increase in learning time as the problem becomes more difficult.

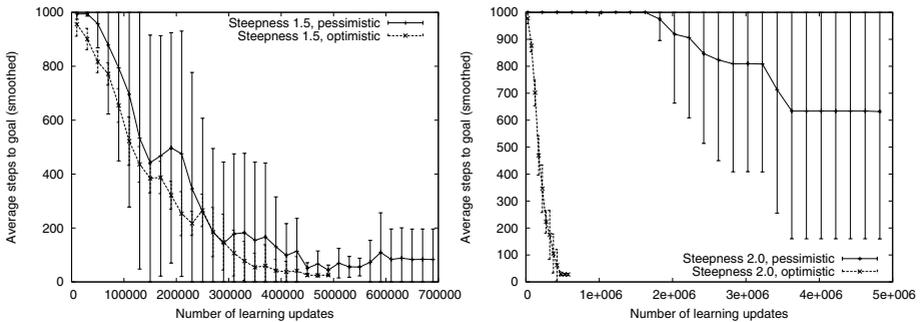


Fig. 9. 2D Mountain Car (steepness 1.5) **Fig. 10.** 2D Mountain Car (steepness 2.0)

5 Conclusion

The admissibility of Q-learning justifies the use of a greedy policy with constant learning rate. We have observed empirically that this strategy outperforms the others in simple and challenging problems. We believe that the empirical evidence of the effectiveness of this strategy makes it worthy of further attention in search of a more theoretical understanding of direct reinforcement learning. We believe that the same strategy also applies to more general learning algorithms that require exploration. We speculate that the final piece of puzzle needed to obtain a polynomial convergence bound for direct reinforcement learning is to establish the fact that the errors introduced in the initial Q values vanish at a reasonably high rate. We argue but cannot yet prove that this is achievable (at least with a constant learning rate) due to the contraction property of value iteration, which underlies the Q-learning algorithm.

Acknowledgements

This material is based upon work supported in part by the Information Processing Technology Office of the Defense Advanced Research Projects Agency

under award HR0011-05-1-0040 and in part by the National Science Foundation under Award NSF IIS 04-13161. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the Defense Advanced Research Projects Agency or the National Science Foundation.

References

- Bertsekas, D.P., Tsitsiklis, J.N. (1996): *Neuro-Dynamic Programming*. Athena Scientific, Belmont, MA.
- Brafman, R.I., Tenenbholz, M. (2002): R-max, A General Polynomial Time Algorithm for Near-Optimal Reinforcement Learning. *Journal of Machine Learning Research*, 3, pp. 213–231.
- Even-Dar, E., Mansour, Y. (2003): Learning rates for Q-Learning. *Journal of Machine Learning Research*, 5, pp. 1–25.
- Kaelbling, L. (1990): *Learning in Embedded Systems*. PhD thesis, Computer Science Department, Stanford University.
- Kearns, M., Singh, S. (1998): Near-Optimal Reinforcement Learning in Polynomial Time. *Proc. of 15th ICML*, pp. 260–268, Morgan Kaufman.
- Kearns, M., Singh, S. (1999): Finite-Sample Rates of Convergence for Q-Learning and Indirect Methods. *Advances in Neural Information Processing Systems 11*, The MIT Press, pp. 996–1002.
- Koenig, S., Simmons, R.G. (1996): The Effect of Representation and Knowledge on Goal-Directed Exploration with Reinforcement Learning Algorithms. *Machine Learning*, 22 (1/3), pp. 227–250.
- Rummery, G. A., Niranjan, M. (1994): *On-line Q-learning using connectionist systems*. Tech. Report CUED/F-INFENG/TR 166, Cambridge University Engineering Dept.
- Sutton, R. (1996): *Generalization in Reinforcement Learning: Successful Examples Using Sparse Coarse Coding*. *Advances in Neural Information Processing Systems 8* pp. 1038–1044, MIT Press.
- Sutton, R., Barto, A. (1998): *Reinforcement Learning*. MIT Press, Cambridge, MA.
- Watkins, C.J.C.H. (1989): *Learning from Delayed Rewards*. PhD thesis, Cambridge, England.