The Influence of Reward on the Speed of Reinforcement Learning: An Analysis of Shaping

Adam Laud LAUD@UIUC.EDU
Gerald DeJong DEJONG@CS.UIUC.EDU
Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA

Abstract

Shaping can be an effective method for improving the learning rate in reinforcement systems. Previously, shaping has been heuristically motivated and implemented. We provide a formal structure with which to interpret the improvement afforded by shaping rewards. Central to our model is the idea of a reward horizon, which focuses exploration on an MDP's critical region, a subset of states with the property that any policy that performs well on the critical region also performs well on the MDP. We provide a simple algorithm and prove that its learning time is polynomial in the size of the critical region and, crucially, independent of the size of the MDP. This identifies low reward horizons with easy-to-learn MDPs. Shaping rewards, which encode our prior knowledge about the relative merits of decisions, can be seen as artificially reducing the MDP's natural reward horizon. We demonstrate empirically the effects of using shaping to reduce the reward horizon.

1. Introduction

Reinforcement Learning (RL) is a popular and effective technique for learning to act in stochastic domains with delayed feedback. Empirically, learning is often slow in the sense that many episodes may be required before a good policy emerges. To speed learning, researchers have explored the use of shaping rewards. In essence, shaping employs an augmented reward structure as a medium to convey prior knowledge to an otherwise-conventional RL system.

These artificial rewards can help to shepherd the reinforcement learner toward policies believed to be good or away from policies believed to be bad. Dorigo and Colombetti (1993) use shaping to convey suggestions from a trainer to a small mobile robot. Mataric (1994) uses successive levels of shaping to achieve better final performance during fixed training intervals of a robot foraging task. Randløv and Alstrøm (1998) show how a shaping reward function can be employed to learn how to drive a bicycle toward a goal. The shaping reward structure is often static in the sense that additional rewards are fixed prior to any domain observations. Laud and DeJong (2002) explore dynamic rewards, in which the additional rewards depend in part on characteristics derived from initial observations of the world. They apply dynamic shaping to bipedal walking.

The addition of shaping rewards changes the Markov decision process (MDP) to be solved. However, this transformation need not alter the optimal policy. Ng, Harada, and Russell (1999) show necessary and sufficient conditions for a policy to remain optimal under addition of a shaping reward structure. However, they, like other shaping investigators demonstrate accelerated learning empirically. The question of what sort of shaping policy will speed rather than delay convergence to the optimal policy is left largely unanalyzed. This is the focus of the present work.

A policy is a mapping or assignment of the highest utility action to each respective state. Thus, conceptually, policy acquisition in reinforcement learning can be viewed as solving a set of classification learning tasks but with several significant complications. A primary complication is that training feedback on a state classification is delayed and ambiguous. Executing an action is similar to obtaining a training example with partial feedback. This feedback is fully captured though a sequence of (possibly discounted) rewards extending on from the action decision. The reward sequences for successive action decisions overlap. The later that a reward appears in a sequence from an action, the greater the number of intervening action decisions, and the less certain we can be that this reward represents significant evidence about the original action decision.

It seems reasonable to consider that the learning rate of a reinforcement learner, like other learners, is significantly influenced by the quality of the feedback provided. Two factors degrade the quality of the feedback. First is the stochastic nature of the underlying rewards. Even for the same state-action-state triple, the observed reward may vary significantly. A higher-variance reward distribution makes any estimate of the average reward less confident. The second factor is the potential delay in rewards (Tesauro, 1992). As delay is increased, the number of

Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003), Washington DC, 2003.

reachable states and potentially relevant rewards grows exponentially. This translates into a much more complicated distribution of feedback sequences, and again a greater number of samples are required for any reliable characterization.

We introduce the notion of a *reward horizon* to summarize these difficulties in an MDP. The reward horizon is a measure of the number of decisions a learning agent must make before experiencing accurate feedback. We view shaping as reducing the reward horizon. Shaping provides artificial reward signals which embody more informative feedback than do the native rewards.

We also introduce a *critical region* of an MDP. These are the states with a significant probability of being visited by some near-optimal policy. Learning to behave within the critical region is sufficient for good behavior in general. The reward horizon defines a boundary to the search for a good decision and can therefore serve as a mechanism to approximate the MDP's critical region.

We provide two arguments for the importance of these ideas. Of primary significance is a first effort at formalizing these notions. It is difficult to analyze their effects on conventional reinforcement algorithms. But we show that with them one can construct a simple policy-learning algorithm. The experience complexity of this algorithm, while exponential in the reward horizon, is sub-quadratic in the size of the critical region and independent of the number of states in the MDP. Second, we provide a brief empirical investigation showing that a conventional reinforcement learning algorithm (Q-learning with ε-greedy exploration) can naturally benefit from reducing the reward horizon.

2. The Framework

We begin developing our reward-based perspective with a discussion of our formalization of the reinforcement learning task, and the essential definitions and notation for the rest of the paper. We will focus on a modified version of the traditional Markov decision process (MDP), the ideal critical region, and the reward horizon.

2.1 A Modified Approach

In addition to the standard description of a Markov decision process, we further characterize the process with a few new properties. We consider domains that are episodic, starting from one initial state. This choice models the repetition that is necessary for learning explicitly, rather than identifying cycles within the domain. Based on this property, we will be able to use the number of episodes until convergence as a clear metric for measuring the time complexity of a learning process. The choice of a single initial state is for simplicity; our results readily extend to a set of initial states. Another property of the learning task we choose to highlight is the task length.

Definition 2.1 The *task length*, L, of an MDP is the maximum number of actions that may be executed in any policy.

This length sets a limit on the size of all potential policies. Policies that do not reach terminal states during L actions are still considered, but will only be evaluated on the return of the first L actions. The task length allows us to distinguish policies in a way other than their expected return, and in a manner that is closely related to the learning complexity of the problem. Based on the idea of limited size episodes, we define an MDP as a collection of seven elements: the set of all states S, the set of all actions A, the set of reward distributions R, the set of transition probabilities T, the discount factor γ, the initial state s_0, and the task length L.

In this framework, the task for reinforcement learning is to compare all policies leaving s_0, that continue on in M for at most L steps, and to choose the one that will encounter the most reward on average. If we execute the resulting policy repeatedly, it will visit some subset of the state space. This set of states is the ideal critical region (CR) for the MDP.

Definition 2.2 The *ideal critical region*, CR, of an MDP, M, is the set of all states that may be reached by starting in the initial state of M, and executing the optimal policy.

The ideal critical region is another way of viewing convergence to the optimal policy. We have an optimal solution to the problem if we can (1) identify all states in the ideal critical region, and (2) make the optimal decision for each of these states. Notice that this region is small in many cases; its size is simply the task length in the deterministic case, and otherwise grows with the randomness imparted by the transition probabilities.

We focus on convergence by learning the CR in steps, looking at a local piece of the original MDP to optimize one state and to refine a working set that approximates the region. We call the local areas sub-MDPs, and they are determined by a start state anywhere in the original MDP, and a task length at most that of the original. Formally, the sub-MDP of an MDP $M = \{S, A, R, T, \gamma, s_0, L\}$ starting in state s with length $H \leq L$ is $m_H(s) = \{S, A, R, T, \gamma, s, H\}$.

2.2 The Reward Horizon

The central idea that enhances the process of learning is restricting the task length of the sub-MDPs to a minimal, yet still informative value. The reward horizon is the property of the reward structure that determines this bound. For any state in the CR, solving a sub-MDP centered on that state with task length equal to the reward horizon will yield the optimal first action for that state. Furthermore, after deciding on a reasonable chance of failure, we can find the solution to the sub-problem efficiently.

The reward horizon has two characteristics that guarantee its ability to put forth a useful sub-MDP task length. The first is that solving problems within the horizon will find the optimal first action. The second is that this solution process can be done efficiently. Inherently, the second characteristic implies that the distinction between the expected return of the optimal policy and the expected return of its closest competitor in any sub-MDP may be determined with reasonable experience in the domain. Let the mean and variance of the return of the optimal policy in the sub-MDP m be μ_* and σ_*^2, and likewise let the competitor policy have μ_c and σ_c^2. Then intuitively, in order to make it easy to select the optimal, we would like a large difference in means, $\mu_* - \mu_c$ and a small amount of variance of the two, $\sigma_*^2 + \sigma_c^2$. We define the variance-mean ratio (vmr) to mathematically formulate these ideas.

Definition 2.3 The *variance-mean ratio*, vmr, is

$$vmr = \frac{\sqrt{\sigma_*^2 + \sigma_c^2}}{\mu_* - \mu_c}$$

In order that the optimal solution is found with a reasonable amount of experience, it must be the case that the variance-mean ratio be small.

Based on these characteristics we define reward horizon with two necessary and sufficient conditions for the MDP.

Definition 2.4 A MDP M has a *reward horizon* of H if and only if:

(1) Solving any sub-MDPs of M centered on a state in the ideal critical region with task length H will determine the optimal first action in M, within a given chance of failure, δ.

(2) The maximum variance-mean ratio across all such sub-MDPs and across all competitor policies is bounded.

3. Using the Reward Horizon

We have argued that the reward horizon explains the success of shaping by describing a minimal search boundary. We now look to use knowledge of the reward horizon to derive the amount of work needed to converge to an approximately optimal policy. Based on this analysis, we will show that a simple algorithm can learn efficiently, even while disregarding parts of the state space.

Any given MDP will have a reward horizon as long as there is one optimal policy. The reward horizon may be as long as the task length, meaning that there is no locality to the rewards; all information is delayed until the end. On the other extreme, the reward horizon may be one, meaning that the feedback from a single transition is enough to determine the best decision. More commonly, we expect that shaping can transform a reward structure with a large horizon to one with a lesser value. This ability of shaping with prior knowledge is what motivates our research on the reward horizon.

We wish to investigate the effects of learning while exploiting the reward horizon. The approach we propose is to grow a set of known states that will eventually include enough states to know reliably that the policy we compute from these states is very close to the optimal. Marking a state known indicates that we have sufficient information to reliably decide on the best action for that state. We develop known states by solving the relevant sub-MDPs with length equal to the reward horizon. Growing the set is a forward-chaining operation; as early actions become fixed, later elements in the CR are visited more frequently and subsequently become known. Once enough states are known, we continue to follow the current policy until enough evidence is gathered to support termination with a good policy.

The next sections discuss in detail how to make a state known, how to learn a sufficient portion of the ideal critical region, and how to bound the total amount of work required to achieve termination. These ideas build the foundation for the algorithm in the next section.

3.1 Creating Known States

A known state is a state for which we believe we have found the optimal action with some level of confidence. In the case where we have a reward horizon H in an MDP M, a state s becomes known when we solve the sub-MDP $m_H(s)$. Once $m_H(s)$ is solved, by the definition of reward horizon, we have the optimal first action in M, which is the best action for s. Therefore, our approach will be to find enough members of the ideal critical region by solving each sub-MDPs based on the reward horizon.

We outline a simple, policy-based procedure for solving sub-MDPs. Namely, we will acquire a number of samples of every policy within the sub-MDP and choose the policy with the highest sample mean. Because the true mean of a policy is estimated from experience, it is not possible to know the best action with zero probability of error. We require a sampling method that will correctly choose the optimal policy within a reasonable chance of error. The following results derive such a method.

Theorem 3.1 *Consider two policies, π_1 and π_2. Let the mean and standard deviation of the return of the policies be μ_1, μ_2, σ_1, σ_2 respectively, with $\mu_1 > \mu_2$. Then choosing the higher sample mean over*

$$n \geq \left\lceil \left(\frac{\Phi^{-1}(1-\delta)\sqrt{\sigma_1^2 + \sigma_2^2}}{\mu_1 - \mu_2} \right)^2 \right\rceil$$

samples of each policy will result in choosing π_1 as the better policy with probability at least 1-δ. We use Φ to denote the cumulative distribution function for the standard normal distribution.

Proof. Let R_1 denote the random variable reporting the total discounted reward for one execution of π_1, and R_2 likewise for π_2. Let $D = R_1 - R_2$, with mean μ_D and standard deviation σ_D. Let $\overline{D}$ be the sample mean of n samples of D. We wish to ensure that n is large enough that the probability that the sample mean of R_1 is larger than that of R_2 is at least 1- δ. In other words,

$$\Pr(\overline{D} > 0) \geq 1 - \delta$$

$$\Pr\left(\frac{\overline{D} - \mu_D}{\sigma_D/\sqrt{n}} > \frac{-\mu_D}{\sigma_D/\sqrt{n}}\right) \geq 1 - \delta$$

$$\Phi\left(\frac{\mu_D}{\sigma_D/\sqrt{n}}\right) \geq 1 - \delta$$

Here we apply the central limit theorem in the case that R_1 and R_2 are not normal. Note that $\mu_D = \mu_1 - \mu_2$, and $\sigma_D = \sqrt{\sigma_1^2 + \sigma_2^2}$. Solving for n yields the stated result. □

We can easily generalize these results to the comparison of several policies.

Corollary 3.2 *Consider m policies, $\pi_1 \ldots \pi_m$. Let the mean and standard deviation of the return of the policies be $\mu_1 \ldots \mu_m$, and $\sigma_1 \ldots \sigma_m$ respectively, with μ_1 being the maximum. Then choosing the highest sample mean of*

$$n \geq \max_{i=2 \text{ to } m} \left\lceil \left(\frac{\Phi^{-1}(1 - \frac{\delta}{m-1}) \sqrt{\sigma_1^2 + \sigma_i^2}}{\mu_1 - \mu_i} \right)^2 \right\rceil$$

samples of each policy will result in choosing π_1 as the best policy with probability at least 1-δ.

Proof. We define $R_1 \ldots R_m$ as before.

$$p \equiv \Pr(\overline{R}_1 > \overline{R}_2 \text{ AND } \overline{R}_1 > \overline{R}_3 \ldots \text{AND } \overline{R}_1 > \overline{R}_m)$$
$$\tilde{p} \equiv \Pr(\overline{R}_1 < \overline{R}_2 \text{ OR } \overline{R}_1 < \overline{R}_3 \ldots \text{OR } \overline{R}_1 < \overline{R}_m)$$

We require $p \geq 1 - \delta$, or equivalently $\tilde{p} \leq \delta$. We can bound $\tilde{p}$ with the sum of the probability of the individual comparisons.

$$\hat{p} \equiv \sum_{i=2}^{m} \Pr(\overline{R}_1 < \overline{R}_i)$$

Then $\tilde{p} \leq \hat{p}$ since the events $\overline{R}_1 < \overline{R}_i$ are not mutually exclusive across i. For all i, let $\Pr(\overline{R}_1 < \overline{R}_i) \leq \frac{\delta}{m-1}$.

Then $\tilde{p} \leq \hat{p} \leq \delta$ which, by the previous theorem, requires n_i samples of each π_i, where

$$n_i = \left\lceil \left(\frac{\Phi^{-1}\left(1 - \frac{\delta}{m-1}\right) \sqrt{\sigma_1^2 + \sigma_i^2}}{\mu_1 - \mu_i} \right)^2 \right\rceil \qquad \text{and } n = \max n_i$$

□

Corollary 3.2 provides an effective sampling method, given we know the number of policies being compared, and the means and variances of each. In order to count the maximum number of policies within a given horizon, we must characterize the randomness of the domain. To this end, we introduce a parameter k, the maximum number of successor states for any state-action. The maximum number of policies is then $k^{H-1}|A|^H$, allowing for |A| decisions at the start state, and k|A| decisions branching from each step in the policy up to H. We will not assume specific knowledge of the means and variances of policy returns, but rather make use of the second part of the definition of reward horizon. This part guarantees a bound on the variance mean ratio, call it VMR. Substituting VMR and the maximum number of policies into the previous results, yields the following corollary.

Corollary 3.3 *A state s in an MDP M, with horizon H becomes known with δ chance of failure after sampling every policy in $m_H(s)$ evenly, with at most*

$$n_{known}^{\delta} = \left(k^{H-1} \mid A\mid^H\right) \left\lceil \left[\Phi^{-1}\left(1 - \frac{\delta}{k^{H-1} \mid A\mid^H - 1}\right) \cdot VMR \right]^2 \right\rceil$$

$$= O\left(\left(k^{H-1} \mid A\mid^H\right) \cdot \ln\left(\frac{k^{H-1} \mid A\mid^H - 1}{\delta} \right)^2 \cdot VMR^2 \right)$$

visits to the state s.

3.2 Learning the Critical Region

A major task of learning, under our framework, is identifying a sufficient portion of the ideal critical region such that we meet a high level of performance with a small chance of failure. We might think of learning the entire CR. In this case, we would achieve optimal returns with no chance of failure. The drawback is that we must learn every state, regardless of how improbable, or unrewarding. In fact, the difficulty in reaching unknown states in the CR plays a major role in the time it takes to learn. In order to reduce the number of unlikely states to learn, we choose to learn a subset of the critical region.

The idea of this approach is to learn a number of states in the CR, such that our chance of visiting only known states for an entire episode is high. Once this probability is high enough, we will experience many trials of the current policy which visit no unknown states. In other words, we have a policy which, although not optimal, executes the optimal actions for all of the most common paths. Each time an execution introduces no unknown states, we gain evidence to develop a confidence interval around the return of the current policy. If there is sufficient number of such observations, we can conclude that the current policy is very likely to be approximately optimal.

The following results outline the construction of the confidence interval, by defining a sufficient condition for its existence. We derive the number of episodes a given policy must visit only known states, in order that this policy be classified as approximately optimal with high degree of probability. We know that the return of a policy visiting only known states is high because it executes only

optimal actions within the specified chance of failure. However, we must limit the chances that this policy would, in the future, visit unknown states whose payoff is enough to warrant the current policy as not acceptable.

Theorem 3.4 *Let π be a policy for which the return has standard deviation σ. Then π has an expected return within ε of the optimal with probability 1-δ, if we observe*

$$n \geq \frac{\left[\Phi^{-1}(1-\frac{\delta}{2})\right]^2 \sigma^2}{\varepsilon^2}$$

sample episodes of π, and each episode performs only optimal actions.

Proof. Let R be the random variable for the return experienced during one episode while executing π. Applying the central limit theorem, we obtain the standard confidence interval based on n samples of R.

$$\overline{R} \pm \Phi^{-1}(1-\tfrac{\delta}{2})\frac{\sigma}{\sqrt{n}}$$

Setting the confidence interval width equal to ε and solving for n yields the stated results.

Because the samples obtained for R were found executing only optimal actions, $\overline{R}$ also reflects a sample mean for the optimal policy. Therefore, we obtain the desired confidence interval around $\overline{R}$ with n samples from π. The expected return of π must be within ε of the optimal 1- δ of the time. □

The idea of gaining confidence in the current policy is powerful because it gives purpose to every episode. That is, on any given episode, we either: (1) visit an unknown state, which refines the current policy, or (2) visit only known states, which builds confidence in the current policy. Once the confidence interval justifies the current policy as approximately optimal with high probability, we can end the learning process, even though some members of the CR are left unknown. However, Theorem 3.4 relies on the variance of the return of a policy, which is generally unknown. The following result places a bound on such variances.

Theorem 3.5 *Let any reward, r, from one transition in an MDP be bounded: $-r_{MAX} \leq r \leq r_{MAX}$, and let the maximum standard deviation of r be σ_{MAX}. Then the maximum variance for any policy in the MDP with task length L is*

$$\sigma \leq L\left(\sigma_{MAX}^2 + 4\,L\,r_{MAX}^2\right).$$

Proof. Let the random variable for the policies return be R, coming from the probability density function f with mean μ. The policy visits any path i, which occurs with probability p_i, has return drawn from f_i with mean μ_i and standard deviation σ_i. The variance of R is

$$\begin{aligned}\sigma^2 &= \int_{-\infty}^{\infty}(x-\mu)^2 f(x)\,dx \\ &= \int_{-\infty}^{\infty}(x-\mu)^2 \sum_i p_i \cdot f_i(x)\,dx \\ &= \sum_i p_i \left(\int_{-\infty}^{\infty} x^2 f_i(x)\,dx - 2\mu \int_{-\infty}^{\infty} x\, f_i(x)\,dx + \mu^2 \int_{-\infty}^{\infty} f_i(x)\,dx \right) \\ &= \sum_i p_i \left(\sigma_i^2 + \mu_i^2 - 2\mu_i\mu + \mu^2\right)\end{aligned}$$

Substituting $\mu = L\, r_{MAX}$, $\mu_i = -L\, r_{MAX}$, and $\sigma_i^2 = L\sigma_{MAX}^2$, we obtain the maximum value for the variance.

$$\begin{aligned}\sigma^2 &\leq \sum_i p_i\left(L\sigma_{MAX}^2 + 4L^2 r_{MAX}^2\right) \\ &\leq L\left(\sigma_{MAX}^2 + 4L\, r_{MAX}^2\right)\end{aligned}$$ □

Substituting this bound for the variance in Theorem 3.4, we develop a termination condition for the learning process.

Corollary 3.6 *Let π be a policy for which we have observed*

$$\begin{aligned}n_{terminal}^{\varepsilon,\delta} &\geq \frac{\left[\Phi^{-1}(1-\frac{\delta}{2})\right]^2 L\left(\sigma_{MAX}^2 + 4L\, r_{MAX}^2\right)}{\varepsilon^2} \\ &= O\left(\frac{1}{\varepsilon^2}\cdot \ln\left(\frac{2}{\delta}\right)^2 \cdot L\left(\sigma_{MAX}^2 + 4L\, r_{MAX}^2\right)\right)\end{aligned}$$

consecutive episodes visiting only known states. Then π has an expected return within ε of the optimal with probability 1-δ.

We would like to determine a subset of the critical region such that we can reach the termination condition in Corollary 3.6. However, this is possible for any nonempty subset that contains a complete path. To specify an approximate critical region, we need another parameter, α. This parameter denotes the probability that the next $n_{terminal}$ episodes will visit only known states – the chance of immediate termination with a near-optimal policy.

Definition 3.1 The *(approximate) critical region*, CR_α, is any subset of the critical region such that the probability of visiting only its elements is at least $\alpha^{1/n_{terminal}}$ on any given episode, executing a policy which knows the optimal action for every element.

The notion of α is an artificial one; we are not interested in specifying it, but rather we use it to illustrate the amount of approximation that will occur. Furthermore, it is a parameter that will need to be optimized to determine the minimum amount of time until convergence. For example, learning CR_1 requires a great deal of work to explore and optimize all the states in the critical region, and very little work to gain confidence in the resulting policy. On the other hand, learning $CR_{0.1}$ may be a much easier subset to optimize, but requires on average $10(n_{terminal})$ episodes to achieve termination. Rather than try to identify a good α prior to learning, we will continually add to the approximate critical region until termination. This idea will automatically determine an appropriate α. If several states in the critical region are very unlikely, we will tend toward a lower alpha. If most states are easily explored, α will tend to be close to 1.

Table 1. Parameters of Corollary 3.9 and bounds on their contributions to the total running time.

Parameter	Definition	DEPENDENCE
H	Reward horizon.	$\exp(H)\cdot H$
k	Maximum number of successor states for any state-action.	$k^{H-1}\ln(k)^3$
VMR	Maximum variance-mean ratio.	$VMR^2\ln(VMR)$
σ_{MAX}	Maximum standard deviation of reward on any transition.	σ_{MAX}^2
r_{MAX}	Maximum reward on any transition.	r_{MAX}^2
L	Task length of the MDP.	L^2
ε	Acceptable error in the expected return of the final policy.	$1/\varepsilon^2$
p	Probability of reaching least likely state in CR_α while learning.	$1/p$
α	Chance of immediate termination given CR_α is known.	$1/\alpha$
$\lvert CR_\alpha \rvert$	Size of the approximate critical region.	$\lvert CR_\alpha \rvert \ln \lvert CR_\alpha \rvert$
δ_1	Chance of failing to find optimal action for any sub-MDP.	$\ln(1/\delta_1)^2\cdot\ln[\ln(1/\delta_1)]$
δ_2	Chance of accepting a policy that is not within ε of the optimal.	$\ln(1/\delta_2)^2$
δ_3	Chance of failing to reach a necessary state to make it known.	$\ln(1/\delta_3)$
δ_4	Chance of not reaching termination within the given time.	$\ln(1/\delta_4)$

3.3 Reaching Convergence

The final part of our analysis is to count the number of episodes needed to achieve termination with a near optimal policy. Conceptually, the work that occurs for termination has two parts. The first is meeting the condition of Corollary 3.3 for a sufficient number of states to create the approximate critical region. Next, we must gain confidence in the policy created by optimizing the approximate critical region. This implies meeting the termination condition of Corollary 3.6.

Theorem 3.7 *Let CR_α be a subset of the critical region with size $|CR_\alpha|$ and let p denote the probability of reaching the least probable state within CR_α. Then, with probability 1-δ, every state in CR_α will be known in*

$$n_{CR_\alpha}^{\delta} = |CR_\alpha| \cdot F^{-1}\left(1-\frac{\delta}{2|CR_\alpha|}\right)$$

episodes. F is the cumulative distribution function for the negative binomial distribution with $n_{known}^{\delta/2}$ successes and probability of success p.

Proof. The number of episodes, t, it will take for the least probable state to become known is a negative binomial random variable with n_{known} number of successes, and probability of success equal to p. For t, a success is simply a visit to that state.

We divide our acceptable probability of failure, δ, into two parts. Half is the chance that solving a sub-MDP will not find the best action, and the other half is the chance that a given state is not visited enough. Because any state in the approximate critical region may fail the second part, this error is divided by the size of the region.

$t \sim Negative\ Binomial(n_{known}^{\delta/2}, p)$

And the least probable state will become known with $\delta/(2|CR_\alpha|)$ chance of error in

$$n = F^{-1}\left(1-\frac{\delta}{2|CR_\alpha|}\right)$$

episodes.

Since all other states are more probable, n represents an upper bound for any state in the approximate critical region. Thus, we may proceed for $|CR_\alpha|$ n episodes, which must make all the required states known with acceptable probability. □

Theorem 3.8 *Let CR_α be an approximate critical region in which all states are known, and the probability to experience $n_{terminal}$ successive episodes staying only within CR_α is α. Then, $n_{terminal}$ successive episodes staying only within CR_α will occur with probability 1-δ in*

$$n_{final}^{\delta} = n_{ter\min al}\cdot G^{-1}(1-\delta)$$

episodes. G is the cumulative distribution function for the geometric distribution with α probability of success.

Proof. Let one trial be a sequence of $n_{terminal}$ episodes. A trial is successful if it visits only known states. The number of trials for a success is a geometric random variable with chance of success α. We bound the number of trials with the acceptable chance of error, δ, and multiply by the length of a trial to obtain the stated result. □

We must apportion of the acceptable chance of failure to its various potential causes. We have discussed 4 areas of failure, and denote them as follows. Let δ_1 be the chance of error for not finding the correct action during the time given for a state to become known. This error must

further be shared with L other members within any given episode, so that the total chance of executing a non-optimal action during an episode is δ_1. Let δ_2 be the chance of termination without being within ε of optimal. Let δ_3 be the chance that any state in the approximate critical region is not made known because it was not reached enough times. This error must be shared between all members of the approximate critical region. Let δ_4 be the chance of not reaching termination within the given time, after the critical region is learned. We divide δ evenly to all four modes of failure.

With this notation, we consolidate our results into the main result of this section: the total number of episodes for the learning process. Table 1 explains all the parameters of Corollary 3.9 and the relevant terms from our previous results. Each parameter is given with its highest order term showing its contribution to the total running time.

Corollary 3.9 *The total number of episodes to learn a policy, by solving reward horizon bounded sub-MDPs, with expected return within ε of the optimal with probability 1-δ is bounded by*

$$|CR_\alpha| \cdot F^{-1}\left(1-\frac{\delta_3}{|CR_\alpha|}\right) + n^{\varepsilon,\delta_2}_{ter\min al} \cdot G^{-1}(1-\delta_4)$$

$$= O\left(|CR_\alpha| \cdot \frac{n^{\delta_1/L}_{known}}{p}\ln\left(\frac{n^{\delta_1/L}_{known}|CR_\alpha|}{\delta_3}\right) + \frac{n^{\varepsilon,\delta_2}_{ter\min al}}{\alpha}\ln\left(\frac{1}{\delta_4}\right)\right)$$

for any α. F is the cumulative distribution function for the negative binomial distribution with $n^{\delta_1/L}_{known}$ successes and probability of success p. G is the cumulative distribution function for the geometric distribution with α probability of success.

4. The Algorithm

In this section, we present a simple method that implements the reward horizon learning scheme according to the results in the previous section. The intent of this algorithm is to take advantage of the reward horizon for more efficient learning. In this way, we can illustrate the potential for shaping through a specific technique that exploits the reward horizon.

The behavior of HORIZON_LEARN is to undergo the process of making known states, until eventually, enough states are known such that we reach the termination requirement of Corollary 3.6. The method to achieve this is very simple – build up a policy π by solving sub-MDPs based on the reward horizon. At any time, π is either exploited, because the state is known, and we wish to explore other members of the critical region, or we execute any policy from a sub-MDP that is needed to make the state known. The number of times we must cycle through this procedure follows the Corollary 3.9.

HORIZON_LEARN (MDP M, Reward Horizon H)
  Initialize policy π randomly.
  Until enough successive episodes occur visiting only known states:
    Assign s as the current state of M.
    If s is terminal, reset to s_0.
    If s is known, execute π(s).
    Otherwise:
      Execute any policy in m_H(s) that still needs to be explored.
      If s becomes known:
        Select the action for s in the optimal policy of m_H(s), a.
        Set π(s) = a.
  Return π.

Figure 1. The HORIZON_LEARN algorithm.

HORIZON_LEARN is guaranteed to meet the requirements for both convergence, and approximate optimality with a polynomially-bounded amount of work given a fixed horizon. The reward horizon serves to guide exploration and focus it on the critical region. This process allows the algorithm to take advantage of any case where the critical region is smaller than the state space. The algorithm continues to learn and expand its knowledge of the critical region until enough confidence is gained that the policy it currently executes is good enough. This process will automatically terminate with some level of approximation for the critical region, α. Convergence occurs as proven in Corollary 3.9, which guarantees an approximately optimal policy with a reasonable amount of work.

5. Experimental Results

We have given a theory identifying the reward horizon as an integral parameter relating the reward structure to the speed of reinforcement learning. This relationship has been motivated by discussion, and proven for an algorithm that makes explicit use of the reward horizon. In this section, we explore the role of the reward horizon empirically, using standard Q-learning with an ε-greedy exploration strategy. This simple exploration strategy makes no direct use of the reward horizon, and is only loosely guided by rewards. The following experiment tests the claim that lower reward horizons correspond to faster learning in a more general setting.

Our experiment operates on a walking task for a bipedal mechanism simulation, as described in our earlier work on shaping (2002). The learning process attempts to maximize the distance traveled over a fixed time interval requiring approximately 30 actions per episode. Taking the best shaping function from our previous work, we vary the reward horizon by changing the delay between

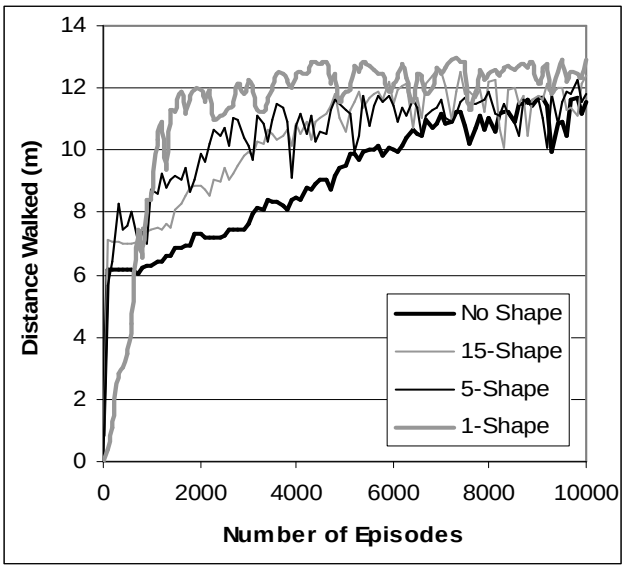

Figure 2. Average Learning Curves As the Reward Horizon is Reduced From No Shaping to Shaping Every Action

applying the average value of the shaping function over different intervals. The shaping function is accumulated over intervals of one, five, and fifteen, actions, and the average shaping reward is applied at the end of the interval. Because the shaping function is very successful, its feedback must be fairly accurate, and thus the interval size will correspond to the reward horizon of the problem.

Figure 2 shows the learning curves as the reward horizon is varied. Each curve is the average of ten experiments. Each shaping strategy is given with a number indicating the delay (interval size) during which no shaping signal is applied until the last interval. Thus 5-shape has four actions without the shaping signal, and the fifth action receives the average shaping signal for all five actions. As the shaping signal is made successively more immediate from the no shape case to the 1-shape case, we observe accelerated learning. Specifically, if we observe the first episode for which distance walked averages at least 11m, 1-shape meets this performance after 1400 episodes, 5-shape after 2700, 15-shape after 4500, and No Shape, after 7000. This data verifies that algorithms that explore based on reward feedback have the potential to be accelerated by reducing the reward horizon.

6. Conclusion

We have formulated an explanation of the potential of reward shaping to accelerate reinforcement learning with a reward-based analysis. The central parameter that characterizes the difficulty of a reward structure is the reward horizon. Given a reward horizon, we showed that a simple algorithm can learn an approximately optimal policy in polynomial time in all parameters except the reward horizon. This strong dependency upon the reward horizon demonstrates why shaping can be very powerful; reductions in the reward horizon result in large savings in the time it takes to learn.

Acknowledgements

We are indebted to the other members of the Explanation-Based Learning group at Illinois: Mike Cibulskis, Arkady Epshteyn, Valentin Moskovich, and Qiang Sun, and to the three anonymous reviewers for helpful suggestions. This material is based upon work supported by the Office of Naval Research under Award No. ONR N00014-01-1-0063. 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 Office of Naval Research.

References

Abramowitz and Stegun. (1972). *Handbook of Mathematical Functions*, 9th edition. Dover Publications, New York.

Dorigo, M., and Colombetti, M. (1993) *Robot Shaping: Developing Situated Agents through Learning.* Technical Report TR-92-040, International Computer Science Institute, Berkeley, CA.

Hogg and Tanis. (2001). *Probability and Statistical Inference*. Prentice-Hall, New Jersey.

Kearns and Singh. (1998) Near-optimal reinforcement learning in polynomial time. *Proceedings of the Fifteenth International Conference on Machine Learning* (pp. 260-268). Morgan Kaufmann, CA.

Laud and DeJong. (2002). Reinforcement Learning and Shaping: Encouraging Intended Behaviors. *Proceedings of the Nineteenth International Conference on Machine Learning* (pp. 355-362). Morgan Kaufmann, CA.

Mataric, M. J. (1994). Reward functions for accelerated learning. In Cohen, W. W. and Hirsh, H. (Eds.). *Proceedings of the Eleventh International Conference on Machine Learning*. Morgan Kaufmann, CA.

Ng, A., Harada, D., & Russell, S. (1999). Policy invariance under reward transformations: Theory and application to reward shaping. *Proceedings of the Sixteenth International Conference on Machine Learning*. Bled, Slovenia: Morgan Kaufmann.

Randløv, J., and Alstrøm, P. (1998). Learning to drive a bicycle using reinforcement learning and shaping. In Shavlik, J. (Ed.). *Proceedings of the Fifteenth International Conference on Machine Learning* (pp. 463-71). Morgan Kaufmann, CA.

Tesauro, G.J. (1992). Practical issues in temporal difference learning. *Machine Learning*, 8, 257-277.

Watkins, C.J.C.H. (1989). *Learning from Delayed Rewards*. Ph.D. thesis, Cambridge University.