Near Optimal On-Policy Control

Matthew Robards, Peter Sunehag *

Australian National University
NICTA

Abstract. We introduce two online gradient-based reinforcement learning algorithms with function approximation – one model based, and the other model free – for which we provide a regret analysis. Our regret analysis has the benefit that, unlike many other gradient based algorithm analyses for reinforcement learning with function approximation, it makes no probabilistic assumptions meaning that we need not assume a fixed behavior policy.

1 Introduction and Background

The ability to learn online is an important trait for reinforcement learning (RL) algorithms. Recently, there has been significant focus on using stochastic gradient descent to enable online reinforcement learning, [1], [5], [8], [9] with significant theoretical advances.

We will here introduce two new algorithms for reinforcement learning with function approximation; one of which can be understood as model-based reinforcement learning, and the other model-free. The methods in [1], [5], [8], [9] are all model free, and they are shown to converge under the assumption that the next state and reward is drawn from a steady-state distribution given the current state. This requires the agent to follow a fixed behavior policy. We wish to give theoretical analyses of our algorithms without placing restrictions on the behavior policy, and hence we look to an analysis with no probabilistic assumptions – regret bounds.

1.1 Related Theoretical Analyses

The classical methods SARSA(λ) and Q-estimation were introduced [7] in the tabular reinforcement learning setting and were heuristically extended to linear function approximation. These methods, however, are known to have convergence issues in this more general setting. Such problems have recently been addressed by a series of gradient descent methods proposed for temporal difference learning and optimal control [5], [8], [9]. Unlike the present work, however, these methods only come with guarantees in the very restricted case of having a fixed behavior policy.

* This author was supported by ARC grant DP0988049.

Bertsekas and Tsitsiklis ([2] proposition (4.8)) proved convergence of stochastic gradient descent algorithms under Markov sampling. These results obviously have positive implications for reinforcement learning, however they require a *steady state* assumption meaning that one must behave according to a fixed policy for it to be applicable. In this paper, our main theoretical analysis aims to give guarantees for the case of an unrestricted policy. Hence we move toward the regret analysis works of [3],[4] which gives guarantees in possibly adversarial situations.

1.2 Markov Decision Processes

We assume a (finite, countably infinite, or even continuous) Markov decision process (MDP) [6] given by the tuple $\{\mathcal{S}, \mathcal{A}, T, R, \gamma\}$. Here, we have states $s \in \mathcal{S}$, actions $a \in \mathcal{A}$, $T : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is a transition function with $T(s, a, s')$ defining the probability of transitioning from state s to s' after executing action a. $R : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ is a reward function where $r_t = R(s_t, a_t, s_{t+1})$ (with $R(s_t, a_t) = \mathbb{E}[R(s_t, a_t, \cdot)]$) is the (possibly stochastic) reward received for time t after observing the transition from state s_t to s_{t+1} on action a_t. Finally, $0 \leq \gamma < 1$ is a discount factor.

1.3 Outline Of Paper

We proceed by next introducing online convex optimization and citing the relevant results. We then introduce our model-free algorithm – residual gradient Q-estimation (RGQ) – in Section 3, and our model-based algorithm – model based Q-estimation (MBQ) – in Section 4. We provide regret bounds for our model learning in Section 5, and for the value learning of each algorithm in section 6. We then conclude in Section 7

2 Online Convex Optimization

In this section we introduce the online convex optimization framework. The work of [3] gives a logarithmic regret bound under strong convexity and twice continuously differentiability of the cost function, whereas [4] gives a square root regret bound with the looser assumption of a convex (possibly non-differentiable) cost function. We only present the logarithmic regret, but acknowledge the less restrictive result and weaker regret bound. Given a sequence of functions $\{C_t\}_{t=1,\ldots,m}$, with $C_t : \mathbb{R}^d \to \mathbb{R}$, we want a sequence $\{w_t\}_{t=1,\ldots,m}$ where $w_t \in \mathbb{R}^d$ and each w_t is chosen before observing C_t, which minimizes $\sum_{t=1}^{m} C_t(w_t)$.

We do this by making stochastic gradient descent updates of the form

$$w_{t+1} = \Pi\Big(w_t - \eta_t \nabla_{w_t} C_t(w_t)\Big) \tag{1}$$

where $\Pi : \mathbb{R}^d \to \{g | g \in \mathbb{R}^d, \|g\|_2 \leq D\}$ is a projection mapping its input to a ball of fixed diameter. Note that this projection is just a simple scaling of

the parameters which is only performed when their norm exceeds D. We now give some preliminary theoretical results which we will later use to show the properties of our algorithm with no policy restrictions. Assume the following:

Assumption 1

A.1 Each C_t is a twice continuously differentiable function.
A.2 Each C_t is H-strongly convex in f.

Note that the projection in Equation (1) ensures that $\|w\| < D$ for some D and hence, together with the assumption of twice continuously differentiability C_t has bounded gradients and hessians.

Theorem 1. *(Hazan et al. [3]) Assume we have a sequence $\{w_t\}_{t=1,\dots,m}$ produced by Equation* (1) *and a sequence of H-strongly convex cost functions C_t with bounded gradients and hessian. Let the step size η_t decay according to $\eta_t = \frac{1}{Ht}$, for some initial η. If Assumption 1 holds we get*

$$\sum_{j=t}^{m} C_t(w_t) \leq \min_{w^* \in \mathbb{R}^d, \|w^*\|_2 \leq D} \sum_{j=t}^{m} C_t(w^*) + O(\log(m)). \tag{2}$$

Remark 1. We note that we present logarithmic regret bounds for our algorithm with squared loss for ease of presentation. We can, however, broaden the applicability of our algorithm to a continuous (possibly piecewise continuous) convex loss function by appealing to the NORMA algorithm of [4] and loosening the regret bounds to $O(\sqrt{m})$.

3 Residual Gradient Q-Estimation

In this section we introduce our online model-free *residual gradient Q-estimation (RGQ)* algorithm with linear function approximation. That is, we represent our value function as $Q(s,a) = \langle \theta, \Phi(s,a) \rangle$, where $\theta, \Phi(s,a) \in \mathbb{R}^d$, and θ corresponds to the w in the previous section.

We evaluate the performance of a sequence of value function parameter estimates $\{\theta_t\}_{t=1\dots m}$ through

$$\sum_{t=1}^{m} C_t(\theta_t), \tag{3}$$

for a given sequence of functions $\{C_t\}_{t=1,\dots,m}$, where θ_t is chosen before time observing C_t. Here we have a sequence of regularized cost functions $\{C_t\}_{t=1,\dots,m}$ given by

$$C_t(\theta_t) = l\Big(Q_t(s_t,a_t) - \gamma Q_t(s_{t+1},a_{t+1}), r_t\Big) + \frac{\lambda}{2}\|\theta_t\|^2, \tag{4}$$

and $l : \mathbb{R} \times \mathbb{R} \to \mathbb{R}^+$ is some convex loss function. We then optimize our objective through gradient descent with the following update

$$\theta_{t+1} = \Pi\left[(1 - \eta_t\lambda)\theta_t - \eta_t \nabla_{\theta_t} l\Big(Q_t(s_t, a_t) - \gamma Q_t(s_{t+1}, a_{t+1}), r_t\Big)\right] \tag{5}$$

where one can easily plug various loss function into this update equation.

4 Model Based Q-Estimation

In this section we introduce our online model based algorithm. We begin by formulating the objective, and discussing the way in which we optimize linear function approximators.

4.1 The Objective

We aim to find a state-action value function Q^* such that

$$Q^*(\phi(s), a) \approx \mathbb{E}_{r|s,a}[r|s, a] + \gamma Q^*(\mathbb{E}_{s'|s,a}[\phi(s')|s, a], a'). \tag{6}$$

We note that the solution to this objective is the optimal solution in a constant delayed MDP (CDMDP) under certain conditions which are outlined in [10]. That is, we get equality in the above equation when Q^* is linear in $\phi(s)$ and the agent is restricted to choosing its next action before observing the next state. It should be clear, however, that the resulting solution is only necessarily useful in the case that the value function of the CDMDP is Lipschitz continuous.

We attempt to learn the value function by learning two models (one for the expected reward M^R, and one for the expected next states features M^T) and perform function approximation on the overall Q function. We measure the overall performance of a sequence of functions $\{Q_t\}_{t=1,\dots,m}$ through

$$\sum_{t=1}^{m} l\Big(Q_t\big(\phi(s_t), a_t\big), M_t^R(s_t, a_t) + \gamma Q_t\big(M_t^T(s_t, a_t), a_{t+1}\big)\Big) + \frac{\lambda}{2}\|Q_t\|^2. \tag{7}$$

where $l : \mathbb{R} \times \mathbb{R} \to \mathbb{R}^+$ is a convex loss function. Furthermore, $M^R(s, a)$ is the predictor of the rewards, and $M^T(s, a)$ is a predictor of the next state's features. We will save the details of the optimization of Q for later. Firstly, we must discuss our transition and reward models.

4.2 Optimizing The Approximators

The optimization problem involves firstly estimating the expectation of the next state's features, secondly estimating the expected reward and then, with this knowledge, optimizing the Q function.

Estimating the Next State Expectation We begin by discussing how to approximate the next state. For this, we introduce the function approximator $M^T : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^{d_\mathcal{S}}$, represented by $M^T(s,a) = W\Phi(s,a)$ where W is a parameter matrix of dimension $d_\mathcal{S} \times d$. We measure the performance of a sequence of $\{M_t^T\}_{t=1,\ldots,m}$ through

$$\sum_{t=1}^{m} \left[L\Big(M_t^T(s_t,a_t), \phi(s_{t+1}) \Big) + \lambda^T \left\| W_t \right\|_2^2 \right] \tag{8}$$

where λ^T is a regularization parameter associated with the transition model, $\|W\|_2^2 = \sum_{i,j} W_{i,j}^2$, and $L : \mathbb{R}^{d_\mathcal{S}} \times \mathbb{R}^{d_\mathcal{S}} \to \mathbb{R}$ is the summed component-wise loss. For example, the squared loss in this situation is given by $L(X,Y) = \|X - Y\|_2^2$. We perform gradient descent of the form

$$W_{t+1} = (1 - \eta_t^T \lambda^T) W_t - \eta_t^R \Big(M^T(s_t,a_t) - \phi(s_{t+1}) \Big) \Phi(s,a)^\top. \tag{9}$$

We also note here that when using a sparse feature vector Φ such as tile coding or RBF coding with finite radius RBFs, this operation is still linear in the number of features, assuming there is an upper bound on the number of non-zero features per state-action pair. That is, for example, if using tile coding the number of tilings must be bounded which we believe is only a minor restriction.

Estimating the Expected Reward To estimate the reward we use yet another function approximator. This is a function $M^R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ which is represented by $M^R(s,a) = \langle \psi, \Phi(s,a) \rangle$. We measure the performance of a sequence $\{M_t^R\}_{t=1,\ldots,m}$ through

$$\sum_{t=1}^{m} \left[l\Big(M_t^R(s_t,a_t) - r_t \Big) + \lambda^R \|\psi_t\|^2 \right]. \tag{10}$$

where λ^R is another regularization parameter. We perform a gradient descent of the form

$$\psi_{t+1} = (1 - \eta_t^R \lambda^R) \psi_t - \eta_t^R \Phi(s_t,a_t) \Big(M_t^R(s_t,a_t) - r_t \Big). \tag{11}$$

4.3 Optimizing The Value Function

We are now left with a very flexible choice of how to optimize Q. We generally opt to perform stochastic gradient descent updates online. Alternatively, however, one can perform batch updates asynchronously to the other two updates.

We also point out that, unlike other gradient descent methods for RL (such as GTD), we are not restricted to updates based on the samples drawn from the Markov process. We can simply draw samples s, a iid and together with M^T and M^R perform gradient based updates, since the next state need not be known.

This is a major advantage of our model-based algorithm over our model-free algorithm which requires either a sampled next state (meaning data cannot be sampled iid) or an enumerable state space in order to sum over all possible state transitions whereas we can generate artificial trajectories.

We restrict ourselves to a class of problems in which the agents next action can safely be chosen from the current state action pair. Action selection in such a *delayed* MDP [10] can be done optimally in the deterministic case, and near optimally in the case of a mildly stochastic MDP. Hence, we optimize for the problem given by the cost function C_t at time t, and this is given by

$$C_t(\theta_t) = l(Q_t(s_t, a_t) - \gamma Q_t(M_t^T(s_t, a_t), a_{t+1}), M_t^R(s_t, a_t)) + \frac{\lambda}{2}\|\theta_t\|^2, \quad (12)$$

by performing gradient descent steps through

$$\theta_{t+1} = \Pi\left(\theta_t - \eta_t \nabla_{\theta_t} C_t(\theta_t)\right) \quad (13)$$

where Π is a projection back onto a convex set.

5 Regret Analysis Of Our Models

We begin by analyzing the properties of our models

Assumption 2 *We place the following assumptions on the MDP.*

B.1 State-action feature space is finite diameter subset of $\mathbb{R}^d$*. ie* $\exists \delta : \forall x \in \mathcal{S} \times \mathcal{A},\ \Phi(x) \in \mathbb{R}^d,\ \|\Phi(x)\| \leq \delta$.

B.2 The reward signal is bounded. ie $\exists R_{\max} : \forall (s,a) \in \mathcal{S} \times \mathcal{A}, \quad |R(s,a)| < R_{\max}$.

B.3 The instantaneous regularized risk given by

$$C_t(\theta_t) = l(Q_t(s_t, a_t) - \gamma Q_t(M_t^T(s_t, a_t), a_{t+1}), M_t^R(s_t, a_t)) + \frac{\lambda}{2}\|\theta_t\|^2. \quad (14)$$

is strongly convex in Q.

If l is convex then the squared regularizer ensures a strongly convex cost function.

5.1 Transition Model

The following lemma tells us that, in the limit and on average, the transition prediction is no worse than the best transition model of the MDP in our class of function approximators.

Proposition 1. *Given an arbitrary sequence of data* $\{s_i, a_i, r_i\}_{i=1,\ldots,m}$ *and a sequence of functions*
$\{M_{t_i}^R\}_{t=1,\ldots,m}$ *for each* $i = 1, \ldots, d_{\mathcal{S}}$*, generated according to the update Equation*

(9). *Fix $\lambda^T > 0$, then under Assumptions (2) with a step size of $\eta_t^T = \frac{1}{Ht}$ (where H comes from the H-strong convexity), we get for all i*

$$\sum_{t=1}^{m} \left(M_{t,i}^T(s_t, a_t) - \phi_i(s_{t+1}) \right)^2 \leq \sum_{t=1}^{m} \left(T_i^*(s_t, a_t) - \phi_i(s_{t+1}) \right)^2 + O(\log(m)) \tag{15}$$

where T_i^* can be any transition function (including the best one) in our class of function approximators.

Proof. The result follows from Theorem 1. □

Given the above Proposition we now see that, under mild assumptions on Q, the error of our predicted transition model given our estimates Q_t is (on average and in the limit) still no worse than the error of the optimal transition function from the class.

Proposition 2. *Assume linear function approximation on Q of the form $Q_t(s, a) = \langle w, \Phi(s, a) \rangle$. Further assume that we have $\Phi(s, a) = A(a)\phi(s)$ for some function $A : \mathcal{A} \to \mathbb{R}^{d \times d_{\mathcal{S}}}$. Assume that there exists C_1, C_2 such that $\max_{x \in \mathbb{R}^{d_{\mathcal{S}}}} \frac{\|A(a)x\|_2}{\|x\|_2} < C_1, \ \forall a \in \mathcal{A}$. Finally assume that we have $d_{\mathcal{S}}$ function approximators $\{M_{t_i}^T\}_{i=1,\ldots,d_{\mathcal{S}}}$ which satisfy*

$$\sum_{t=1}^{m} \left(M_{t,i}^T(s_t, a_t) - \phi_i(s_{t+1}) \right)^2 \leq \sum_{t=1}^{m} \left(T_i^*(s_t, a_t) - \phi_i(s_{t+1}) \right)^2 + O(\log(m)) \tag{16}$$

for all i. It then holds that

$$\sum_{t=1}^{m} \left(Q_t(M_t^T(s_t, a_t), a_{t+1}) - Q_t(s_{t+1}, a_{t+1}) \right)^2 \leq K \sum_{t=1}^{m} \left\| T^*(s_t, a_t) - \phi(s_{t+1}) \right\|^2 + O(\log(m)) \tag{17}$$

for some K.

Proof. For any Q_t which fits our assumptions, we have

$$\begin{aligned} \left| Q_t(s, a_{t+1}) - Q_t(s', a_{t+1}) \right| &= \left| w_t^\top A(a_{t+1})(\phi(s) - \phi(s')) \right| \\ &\leq \|w_t^\top A(a_{t+1})\| \|\phi(s) - \phi(s')\|. \end{aligned} \tag{18}$$

Now, since $\|w_t\| \leq U \ \ \forall t$ (which is brought about by the projection of our gradient descent onto a finite diameter ball), we see that $\left| Q_t(s, a_{t+1}) - Q_t(s', a_{t+1}) \right| \leq$

$K\|\phi(s) - \phi(s')\|$, which implies that

$$\sum_{t=1}^{m}\Big(Q_t(M_t^T(s_t,a_t),a_{t+1}) - Q_t(s_{t+1},a_{t+1})\Big)^2 \leq \sum_{t=1}^{m} K\Big\|M_t^T(s_t,a_t) - \phi(s_{t+1})\Big\|^2$$
$$\leq K\sum_{t=1}^{m}\Big\|T^*(s_t,a_t) - \phi(s_{t+1})\Big\|^2 + O(\log(m)). \tag{19}$$

□

5.2 Reward Model

Here we give the regret bounds of our reward model.

Proposition 3. *Given an arbitrary sequence of data $\{s_i, a_i, r_i\}_{i=1,\ldots,m}$ and a sequence of functions $\{M_i^R\}_{i=1,\ldots,m}$ generated according to the update Equation (11) with $\lambda^R > 0$ and under Assumptions (2) with a step size of $\eta_t^R = \frac{1}{Ht}$ (where H comes from the H-strong convexity) we get*

$$\sum_{t=1}^{m}\Big(M_t^R(s_t,a_t) - r_t\Big)^2 \leq \sum_{t=1}^{m}\Big(r^*(s_t,a_t) - r_t\Big)^2 + O(\log(m)), \tag{20}$$

where $r^* = \arg\min_r \sum_{t=1}^{m}\Big(r(s_t,a_t) - r_t\Big)^2$.

Proof. This is a direct consequence of Theorem 1. □

6 Properties Of Our Value Function

6.1 Model Based Q-Estimation

We give two results for our value function with model based Q-estimation. The first (which is data independent) shows that for any sequence of state action pairs (which one can simply generate iid at random) the value function is only as bad as the model estimates. The second (which is data dependent) shows that the Bellman error is dominated only by the variance in $r_t|s_t,a_t$ and $\phi(s_{t+1})|s_t,a_t$. We then conclude with a series of corollaries which highlight the significance of these results in various special circumstances.

These results are presented for the squared loss, however as stated in Remark 1 they extend to more general loss functions, with the regret bound loosened to $O(\sqrt{m})$. The loss function must be of a form $l(x,z) = f(x-z) \leq \alpha(f(x)+f(z))$.

Theorem 2. *Assume we have a sequence of data $\{s_t, a_t, r_t, s_{t+1}\}_{t=1,\ldots,m}$ and a sequence of functions $\{Q_t\}_{t=1,\ldots,m}$ generated by equation (13) with a squared cost function and l_2 regularizer. Let the coefficient of regularization be $\lambda > 0$ and begin with $\|Q_1\| \leq U$ for some U. Assume we have a step size decaying*

according to $\eta_t = \frac{1}{Ht}$ *(where* H *comes from the* H*-strong convexity). Then, under Assumption 2, there exists* K *such that it holds for any* Q^* *in our class of function approximators and for any functions* r^*, T^* *(including the true models) that:*

$$\sum_{t=1}^{m}\left(Q_t(s_t,a_t) - r^*(s_t,a_t) - \gamma Q_t(T^*(s_t,a_t),a_{t+1})\right)^2 \leq$$
$$K\sum_{t=1}^{m}\left(\left(M_t^R(s_t,a_t) - r^*(s_t,a_t)\right)^2 + \left\|\gamma M_t^T(s_t,a_t) - \gamma T^*(s_t,a_t)\right\|^2\right.$$
$$\left. + \left(Q^*(s_t,a_t) - r^*(s_t,a_t) - \gamma Q^*(T^*(s_t,a_t),a_{t+1})\right)^2\right) + O(\log(m)) \quad (21)$$

Proof. Select any r^*, T^*. Using $(a+b+c)^2 \leq (3\max(a,b,c))^2 \leq 9a^2+9b^2+9c^2$

$$\sum_{t=1}^{m}\left(Q_t(s_t,a_t) - r^*(s_t,a_t) - \gamma Q_t(T^*(s_t,a_t),a_{t+1})\right)^2 = \sum_{t=1}^{m}\left(Q_t(s_t,a_t)\right.$$
$$- \gamma Q_t(M_t^T(s_t,a_t),a_{t+1}) - M_t^R(s_t,a_t) + M_t^R(s_t,a_t) - r^*(s_t,a_t)$$
$$\left. + \gamma Q_t(M_t^T(s_t,a_t),a_{t+1}) - \gamma Q_t(T^*(s_t,a_t),a_{t+1})\right)^2$$
$$\leq 9\sum_{t=1}^{m}\left(Q_t(s_t,a_t) - M_t^R(s_t,a_t) - \gamma Q_t(M_t^T(s_t,a_t),a_{t+1})\right)^2$$
$$+ 9\sum_{t=1}^{m}\left(M_t^R(s_t,a_t) - r^*(s_t,a_t)\right)^2 + 9\sum_{t=1}^{m}\left\|\gamma M_t^T(s_t,a_t) - \gamma T^*(s_t,a_t)\right\|^2$$
(applyingTheorem1)
$$\leq 9\sum_{t=1}^{m}\left(\left(Q^*(s_t,a_t) - M_t^R(s_t,a_t) - \gamma Q^*(M_t^T(s_t,a_t),a_{t+1})\right)^2\right.$$
$$\left. + \left(M_t^R(s_t,a_t) - r^*(s_t,a_t)\right)^2 + \left\|\gamma M_t^T(s_t,a_t) - \gamma T^*(s_t,a_t)\right\|^2\right) + O(\log(m))$$
$$\leq 9\sum_{t=1}^{m}\left(\left(M_t^R(s_t,a_t) - r^*(s_t,a_t)\right)^2 + \left(Q^*(s_t,a_t) - r^*(s_t,a_t)\right.\right.$$
$$- \gamma Q^*(T^*(s_t,a_t),a_{t+1}) - M_t^R(s_t,a_t) + r^*(s_t,a_t) - \gamma Q^*(M_t^T(s_t,a_t),a_{t+1})$$
$$\left.\left. + \gamma Q^*(T^*(s_t,a_t),a_{t+1})\right)^2 + \left\|\gamma M_t^T(s_t,a_t) - \gamma T^*(s_t,a_t)\right\|^2\right) + O(\log(m))$$
(again applying $(a+b+c)^2 \leq (3\max(a,b,c))^2 \leq 9a^2+9b^2+9c^2$)
(22)

$$\leq K \sum_{t=1}^{m} \left(\left(M_t^R(s_t, a_t) - r^*(s_t, a_t) \right)^2 + \left\| \gamma M_t^T(s_t, a_t) - \gamma T^*(s_t, a_t) \right\|^2 \right.$$
$$\left. + \left(Q^*(s_t, a_t) - r^*(s_t, a_t) - \gamma Q^*(T^*(s_t, a_t), a_{t+1}) \right)^2 \right) + O(\log(m)) \qquad (23)$$

□

Theorem 3. *Assume we have a sequence of data* $\{s_t, a_t, r_t, s_{t+1}\}_{t=1,\ldots,m}$ *and a sequence of functions* $\{Q_t\}_{t=1,\ldots,m}$ *generated by equation* (13) *with a squared cost function and* l_2 *regularizer. Let the coefficient of regularization be* $\lambda > 0$ *and begin with* $\|Q_1\| \leq U$ *for some* U*. Assume we have a step size decaying according to* $\eta_t = \frac{1}{Ht}$ *(where* H *comes from the* H*-strong convexity). Assume we have a fixed sequence* $\{M_t^R, M_t^T\}_{t=1,\ldots,m}$ *with a corresponding sequence of data* $\{s_t, a_t, r_t, s_{t+1}\}_{t=1,\ldots,m}$ *such that*

$$\sum_{t=1}^{m} \left(M_t^R(s_t, a_t) - r_t \right)^2 \leq \sum_{t=1}^{m} \left(r^*(s_t, a_t) - r_t \right)^2 + O(\log(m))),$$
$$\sum_{t=1}^{m} \left(Q^*(M_t^T(s_t, a_t), a_{t+1}) - Q^*(s_{t+1}, a_{t+1}) \right)^2 \leq \sum_{t=1}^{m} \left\| T^*(s_t, a_t) - \phi(s_{t+1}) \right\|^2$$
$$+ O(\log(m)). \qquad (24)$$

for all r^*, T^* *in our class of function approximators, where* $T^* = [T_1^*, \ldots, T_{d_{\mathcal{S}}}^*]^\top$*. Then it holds under Assumption 2, when following the greedy policy, that for any* Q^* *in our class of function approximators*

$$\sum_{t=1}^{m} \left(Q_t(s_t, a_t) - r_t - \gamma Q_t(s_{t+1}, a_{t+1}) \right)^2 \leq K \sum_{t=1}^{m} \left(\left(r_t - r^*(s_t, a_t) \right)^2 \right.$$
$$\left. + \left\| \gamma \phi(s_{t+1}) - \gamma T^*(s_t, a_t) \right\|^2 + \left(Q^*(s_t, a_t) - r_t - \gamma Q^*(s_{t+1}, a_{t+1}) \right)^2 \right)$$
$$+ O(\log(m)) \qquad (25)$$

Proof. The proof of this theorem is very similar to that of Theorem 2 and is hence omitted for brevity.

Corollary 1. To Theorem 2: *Suppose there are* r^* *and* T^* *in our class of function approximators such that*

$$r^*(s_t, a_t) = \mathbb{E}[r_t | s_t, a_t], \qquad T^*(s_t, a_t) = \mathbb{E}[\phi(s_{t+1} | s_t, a_t] \qquad (26)$$

then we get

$$\sum_{t=1}^{m}\left(Q_t(s_t,a_t)-\mathbb{E}[r_t|s_t,a_t]-\gamma Q_t(\mathbb{E}[\phi(s_{t+1})|s_t,a_t],a_{t+1})\right)^2 \leq$$
$$K\sum_{t=1}^{m}\left(\left(M_t^R(s_t,a_t)-\mathbb{E}[r_t|s_t,a_t]\right)^2+\gamma^2\left\|M_t^T(s_t,a_t)-\mathbb{E}[s_{t+1}|s_t,a_t]\right\|^2\right.$$
$$\left.+\left(Q^*(s_t,a_t)-\mathbb{E}[r_t|s_t,a_t]-\gamma Q^*(\mathbb{E}[\phi(s_{t+1})|s_t,a_t],a_{t+1})\right)^2\right)+O(\log(m)) \tag{27}$$

Corollary 2. To Theorem 3: *If we assume that the expected reward and next state given a current state action pair lies in our class of function approximators then we get*

$$\frac{1}{m}\sum_{t=1}^{m}\left(Q_t(s_t,a_t)-r_t-\gamma Q_t(s_{t+1},a_{t+1})\right)^2 \leq$$
$$\frac{1}{m}\sum_{t=1}^{m}\left(r_t-\mathbb{E}[r_t|s_t,a_t]\right)^2+\frac{1}{m}\sum_{t=1}^{m}\left(Q^*(s_t,a_t)-r_t-\gamma Q^*(s_{t+1},a_{t+1})\right)^2$$
$$+\frac{\gamma^2}{m}\sum_{t=1}^{m}\left\|\phi(s_{t+1})-\mathbb{E}[\phi(s_{t+1})|s_t,a_t]\right\|^2+O\left(\frac{\log(m)}{m}\right). \tag{28}$$

Corollary 3. Optimality in deterministic environments: *If we further assume a deterministic world, and that the true value function lies in the same class of function approximators as our predicted value function, we see that*

$$\frac{1}{m}\sum_{t=1}^{m}\left(Q_t(s_t,a_t)-r_t-\gamma Q_t(s_{t+1},a_{t+1})\right)^2$$
$$\leq\frac{1}{m}\sum_{t=1}^{m}\left(Q^*(s_t,a_t)-r_t-\gamma Q^*(s_{t+1},a_{t+1})\right)^2+O\left(\frac{log(m)}{m}\right). \tag{29}$$

The above results tell us: (a) when the expected reward and next state's features lie in our class of model function approximators the the average regret of Q is proportional to the noise in the environment (Corollary 2); (b) in deterministic environments, the average regret of our algorithm goes to zero (Corollary 3); (c) the regret is proportional to the error of our models' approximation (Corollary 1).

6.2 Residual Gradient *Q*-Estimation

Here we present the simple regret bound for our RGQ-Estimation algorithm.

Theorem 4. *Assume a sequence of cost functions* $\{C_t\}_{t=1,\dots,m}$ *with*

$$C_t(\theta_t) = l(Q_t(s_t, a_t) - \gamma Q_t(s_{t+1}, a_{t+1}), r_t) + \frac{\lambda}{2}\|\theta_t\|^2$$

is generated for the sequence of data $\{s_t, a_t, r_t, s_{t+1}, a_{t+1}\}_{t=1,\dots,m}$ *and value functions* $\{Q_t\}_{t=1,\dots,m}$ *are generated by update Equation* (5) *with a twice continuously differentiable convex loss function* l*. Then it holds under Assumption 2 that*

$$\sum_{t=1}^{m} C_t(\theta_t) \leq \sum_{t=1}^{m} C_t(\theta^*_{\text{batch}}) + O(\log m) \tag{30}$$

Proof. This follows directly from Theorem 1.

7 Conclusions

We provide two new reinforcement learning algorithms – one model free and the other model based. We provide regret bounds for the models of MBQ, and for the value functions of both algorithms. We see that in deterministic environments the algorithms each achieve an logarithmic regret, and that in mildly stochastic environments they are still on average as good as the best in their class.

References

1. Baird, L., Moore, A.: Gradient descent for general reinforcement learning. In: In Advances in Neural Information Processing Systems 11. pp. 968–974. MIT Press (1998)
2. Bertsekas, D., Tsitsiklis, J.: Neuro-Dynamic Programming. Athena Scientific (1996)
3. Hazan, E., Kalai, A., Kale, S., Agarwal, A.: Logarithmic regret algorithms for online convex optimization. In: In 19th COLT. pp. 499–513 (2006)
4. Kivinen, J., Smola, A.J., Williamson, R.C.: Online learning with kernels. In: Dietterich, T.G., Becker, S., Ghahramani, Z. (eds.) NIPS. pp. 785–792. MIT Press (2001)
5. Maei, H., Szepesvri, C., Bhatnagar, S., Sutton, R.: Toward off-policy learning control with function approximation. In: In Proceedings of the 27th International Conference on Machine Learning (2010)
6. Puterman, M.L.: Markov Decision Processes: Discrete Stochastic Dynamic Programming. Wiley, New York (1994)
7. Sutton, R., Barto, A.: Reinforcement Learning. The MIT Press (1998)
8. Sutton, R., Maei, H., Precup, D., Bhatnagar, S., Silver, D., Szepesvri, C., Wiewiora, E.: Fast gradient-descent methods for temporal-difference learning with linear function approximation. In: In Proceedings of the 26th International Conference on Machine Learning (2009)
9. Sutton, R., Szepesvári, C., Maei, H.: A convergent o(n) temporal-difference algorithm for off-policy learning with linear function approximation. In: NIPS. pp. 1609–1616. MIT Press (2008)
10. Walsh, T., Nouri, A., Li, L., Littman, M.: Learning and planning in environments with delayed feedback. Autonomous Agents and Multi-Agent Systems 18, 83–105 (2009), http://dx.doi.org/10.1007/s10458-008-9056-7, 10.1007/s10458-008-9056-7