Proceedings of Machine Learning Research vol 145:1099–1136, 2021 2nd Annual Conference on Mathematical and Scientific Machine Learning

Borrowing From the Future: Addressing Double Sampling in Model-free Control

Yuhua Zhu YUHUAZHU@STANFORD.EDU
Department of Mathematics, Stanford University, Stanford, CA 94305

Zachary Izzo ZIZZO@STANFORD.EDU
Department of Mathematics, Stanford University, Stanford, CA 94305

Lexing Ying LEXING@STANFORD.EDU
Department of Mathematics and ICME, Stanford University, Stanford, CA 94305

Editors: Joan Bruna, Jan S Hesthaven, Lenka Zdeborova

Abstract

In model-free reinforcement learning, the temporal difference method is an important algorithm but might become unstable when combined with nonlinear function approximations. Bellman residual minimization with stochastic gradient descent (SGD) is stable but suffers from the double sampling problem: given the current state, two independent samples for the next state are required, but often only one sample is available. Recently, the borrowing-from-the-future (BFF) algorithm was introduced in (Zhu et al., 2020) to address this issue for policy evaluation. The main idea is to borrow extra randomness from the future to approximately re-sample the next state when the underlying dynamics of the problem are sufficiently smooth. This paper extends the BFF algorithm to action-value function based model-free control. We prove that BFF is close to unbiased SGD when the underlying dynamics vary slowly with respect to actions. We confirm our theoretical findings with numerical simulations.

1. Introduction

Background The goal of reinforcement learning (RL) is to find an optimal policy which maximizes the return of a Markov decision process (MDP) (Sutton & Barto, 2018). One of the most common ways of finding an optimal policy is to treat it as the fixed point of the Bellman operator. Researchers have developed efficient iterative methods such as temporal difference (TD) (Sutton, 1988), Q-learning (Watkins, 1989), and SARSA (Rummery & Niranjan, 1994) based on the contraction property of the Bellman operator.

Nonlinear function approximations have recently received a great deal of attention in RL. This follows the successful application of deep neural networks (DNNs) to Atari games (Mnih. et al., 2013, 2015), as well as in Alpha Go and Alpha Zero (Silver et al., 2016, 2017). However, when nonlinear approximations such as neural networks are used, the contraction property of the Bellman operator may no longer hold. This can in turn result in unstable training of the network. Many variants and modifications have been proposed to stabilize training. For example, DQN (Mnih. et al., 2015) and A3C (Mnih. et al., 2016) stabilize Q-learning by using a slowly changing target network and replaying over past experiences or using parallel agents for exploration; double DQN reduces instability by using two separate Q value estimators, one for choosing the action and the other for evaluating the action's quality (van Hasselt et al. , 2015).

Another way to stabilize RL with a nonlinear approximation is to formulate it as a minimization problem. This approach is known as Bellman residual minimization (BRM) (Baird, 1995). However, applying stochastic gradient descent (SGD) to BRM directly suffers from the so-called double sampling problem: at a given state, two independent samples for the next state are required in order to perform unbiased SGD. Such a requirement is often hard to fulfill in a model-free setting, especially for problems with a continuous state space.

Contributions In this paper, we revisit BRM for Q-value prediction and control problems in the model-free RL setting. The main assumption is that the underlying dynamics of the MDP can be written as a discretized stochastic differential equation with a small step size. Note that knowledge of the dynamics is not required to implement the algorithm. We extend the borrowing-from-the-future (BFF) algorithm of (Zhu et al., 2020) to action-value based RL. The key idea is to borrow extra randomness from the future by leveraging the smoothness of the underlying RL problem. We prove that for both Q-value predition and control problems, when the underlying dynamics change slowly with respect to actions, the training trajectory of the proposed algorithm is statistically close to the training trajectory of unbiased SGD. The difference between the two algorithms will first decay exponentially and eventually stabilize at an error of $O(\epsilon\delta_*)$, where δ_* is the smallest Bellman residual that unbiased SGD can achieve and ϵ is the size of the time step in the underlying SDE discretization.

Related work Our work is inspired by (Zhu et al., 2020), which we extend in four important, nontrivial ways. First, (Zhu et al., 2020) introduced BFF and proved an approximation theorem only for policy evaluation with the state value function, while this paper generalizes to state-action value function Q-evaluation and finds an optimal policy. Second, we generalize the one-step BFF algorithm to n-step BFF. (See Remark 1.) Third, we observe that BFF-loss from Zhu et al. does not generalize to Q-evaluation explain the reason for this phenomenon. (See Remark 3 for details.) Last but not least, we give a much sharper error bound for the difference between BFF and unbiased SGD. (See Remark 5.)

In (Wang et al., 2017, 2016), the stochastic compositional gradient method (SCGD), a two step-scale algorithm, is proposed to address the double sampling problem. However, it is not clear how to apply SCGD to BRM with a continuous state space.

Another way to avoid the double sampling problem in BRM is to consider the primal-dual (PD) formulation of the minimization problem and view the solution as a saddle point of a minimax problem. Such methods include GTD and its variants (Sutton, 2008; Sutton et al., 2009; Bhatnagar et al., 2009; Mahadevan et al., 2011; Liu et al., 2015), and policy gradient methods (Dai et al., 2018; Wang , 2020). However, when a nonlinear function approximation is used, the minimax is no longer taken over a convex-concave function. This renders the PD formulation significantly more difficult than solving the minimization problem directly, and our numerical experiments demonstrate the instability of the PD method. (See Section 4 for details.)

2. Models and key ideas

2.1. Continuous state space

Working in the model-free RL setting, we consider a discrete-time MDP with a compact continuous state space $\mathbb{S} \subset \mathbb{R}^{d_s}$. The action space $\mathbb{A} \subset \mathbb{R}^{d_a}$ can be a compact continuous set or a finite discrete

set. The transition kernel of the MDP

$$P^a(s, s') = \mathbb{P}\left(s_{m+1} = s' | s_m = s, a_m = a\right). \tag{1}$$

denotes the likelihood of transferring from the current state $s_m = s$ under the current action $a_m = a$ to the next state $s_{m+1} = s'$. The immediate reward function $r(s', s, a)$ specifies the reward if one takes action a at state s and ends up at state s'. The immediate reward can also be random, in which case $r(s', s, a)$ represents the expected reward. A policy $\pi(a|s)$ gives the probability of taking action a at state s, i.e., $\mathbb{P}\{\text{take action } a \text{ at state } s\} = \pi(a|s)$. For a continuous state space, it is often convenient to rewrite the underlying transition in terms of the states:

$$s_{m+1} = s_m + \mu(s_m, a_m)\epsilon + \sqrt{\epsilon}\sigma Z_m. \tag{2}$$

where Z_m is a mean-zero noise. This form is particularly relevant when the MDP arises as a discretization of an underlying stochastic differential equation (SDE), with ϵ as its discretized time step. We note that Z_m need not be i.i.d. Gaussian. Our theoretical and numerical results can be extended to any independent mean-zero noise with the same variance at each time step. In addition, the diffusion term σ can depend on state and action as well. We provide the proof for this case in Appendix B. Since the extension is trivial and does not affect the theoretical and numerical results, we stick to a constant diffusion term in the main paper for simplicity. We note that this form is quite general: it encompasses both deterministic linear differential equations (e.g. (Bradtke , 1993; Doya., 2000)) as well as cases with a nonlinear drift $\mu(s, a)$ and stochastic transitions, e.g. (Pedersen , 2017). Throughout the paper, we consider the case where for each state, the variation of the underlying drift $\mu(s, a)$ is a priori bounded in the action space, and for each action, the drift is smooth in the state space. Additionally, we assume the immediate reward $r(s', s, a)$ is continuous in $s', s \in \mathbb{S}$ for each action.

Given a trajectory $\{s_m, a_m\}_{m\geq 0}$, the main object under study is the action-state pair value function $Q(s, a)$. There are two types of problems: Q-evaluation and Q-control. Q-evaluation refers to the prediction of the value function $Q^\pi(s, a)$ when the policy π is given, while Q-control refers to finding the optimal policy π_* through the maximization of $Q^\pi(s, a)$ over all possible policies. For the Q-evaluation problem the state space and action space can be continuous or discrete, while for the Q-control problem we mainly consider the case of a finite discrete action space.

Q-evaluation Given a fixed policy π, the value function $Q^\pi(s, a)$ represents the expected return if one takes action a at state s and follows π thereafter, i.e.,

$$Q^\pi(s, a) = \mathbb{E}\left[\sum_{t\geq 0} \gamma^t r(s_{m+t+1}, s_{m+t}, a_{m+t}) \middle| s_m = s, a_m = a\right],$$

where $\gamma \in (0, 1)$ is a discount factor. The value function Q^π satisfies the Bellman equation (Sutton & Barto, 2018) $Q^\pi(s, a) = \mathbb{T}^\pi Q^\pi(s, a)$, with the Bellman operator $\mathbb{T}^\pi$ defined as

$$\mathbb{T}^\pi Q^\pi(s, a) \equiv \mathbb{E}[r(s_{m+1}, s_m, a_m) + \gamma Q^\pi(s_{m+1}, a_{m+1}) | (s_m, a_m) = (s, a)], \tag{3}$$

where the expectation is taken over (s_{m+1}, a_{m+1}) when the policy π is applied. In the nonlinear approximation setting, one seeks a solution to (3) from a family of functions $Q^\pi(s, a; \theta)$ parameterized by $\theta \in \Omega \subseteq \mathbb{R}^{d_\theta}$. For example, the function approximation family could be the set of all NNs

of a given architecture, and θ specifies the network weights. One way to find $Q^\pi(s,a;\theta)$ is to solve the following *Bellman residual minimization (BRM) problem*:

$$\min_{\theta\in\mathbb{R}^{d_\theta}} \mathop{\mathbb{E}}_{(s,a)\sim\rho(s,a)} \delta^2(s,a;\theta) \tag{4}$$

where $\rho(s,a)$ is a distribution over $\mathbb{S}\times\mathbb{A}$ and

$$\delta(s,a;\theta) \equiv |\mathbb{T}^\pi Q^\pi(s,a;\theta) - Q^\pi(s,a;\theta)| \tag{5}$$

is the absolute value of the Bellman residual. Note that the expectation in (4) can be taken with respect to different distributions ρ. For on-policy learning, it is often the stationary distribution of the Markov chain. When $\mathbb{S}$ and $\mathbb{A}$ are discrete, it is also reasonable to choose a uniform distribution over $\mathbb{S}\times\mathbb{A}$. Our experience shows that doing so often accelerates the rate of convergence compared to the stationary measure.

One approach for solving the Bellman minimization problem (4) is to directly apply SGD. The unbiased gradient estimate to the loss function is

$$F_m(\theta) = j(s_m,a_m,s_{m+1};\theta)\nabla_\theta j(s_m,a_m,s'_{m+1};\theta), \tag{6}$$

where (s_m,a_m) is a sample from the distribution $\rho(s,a)$ and

$$j(s_m,a_m,s_{m+1};\theta) = r(s_{m+1},s_m,a_m) + \gamma\int Q^\pi(s_{m+1},a;\theta)\pi(a|s_{m+1})da \ - Q^\pi(s_m,a_m;\theta) \tag{7}$$

is an unbiased estimate for the Bellman residual $\mathbb{T}^\pi Q^\pi(s_m,a_m) - Q^\pi(s_m,a_m)$. If the immediate reward is random, then the first term becomes a sample of the reward. The second term is an expectation over the action, which can be replaced by a sample from the policy, i.e. $Q(s_{m+1},a_{m+1})$ for $a_{m+1}\sim\pi(a|s_{m+1})$. These modifications do not alter the theoretical results in the paper. Here s_{m+1} is the next state in the trajectory, while s'_{m+1} is an independent sample for the next state according to the transition process. However, in model-free RL, as the underlying dynamics are unknown, another independent sample s'_{m+1} of the next state is unavailable. Therefore, this unbiased SGD, referred to as *uncorrelated sampling* (US), is impractical. Even if one can store the whole trajectory, it is impossible to revisit a certain state multiple times when the state space is either continuous or discrete but of high dimension. This is the so-called *double sampling problem*. One potential solution, called *sample-cloning* (SC), simply uses s_{m+1} as a surrogate for s'_{m+1}, i.e. $s'_{m+1} = s_{m+1}$. However, sample-cloning is not an unbiased algorithm for BRM, and its bias grows rapidly with the conditional variance of s_{m+1} on s_m as proved in Lemmas D.1 and D.2.

To address the double sampling problem, (Zhu et al., 2020) introduced the borrowing from the future (BFF) algorithm. The main idea of the BFF algorithm is to borrow the future difference $\Delta s_{m+1} = s_{m+2} - s_{m+1}$ and approximate the second sample s'_{m+1} with $s_m + \Delta s_{m+1}$. During SGD, the parameter θ is updated based on the following estimate of the unbiased gradient:

$$\hat{F}_m(\theta) = j(s_m,a_m,s_{m+1};\theta)\nabla_\theta j(s_m,a_m,s_m+\Delta s_{m+1};\theta), \tag{8}$$

where j is a sample of the Bellman residual defined in (7). When the difference between Δs_m and Δs_{m+1} is small, the new s'_{m+1} is statistically close to the distribution of the true next state. Among the two versions (gradient based and loss function based) introduced in (Zhu et al., 2020), we adopt

Algorithm 1 BFF for Q-evaluation

Require: η: Learning rate
Require: $Q^\pi(s;\theta) \in \mathbb{R}^{|\mathbb{A}|}$ or $Q^\pi(s,a;\theta) \in \mathbb{R}$: Nonlinear approximation of Q^π parameterized by θ
Require: $j(s,a,s';\theta) := r(s',s,a) + \gamma \int Q^\pi(s',a;\theta)\pi(a|s')da - Q^\pi(s,a;\theta)$
Require: θ_0: Initial parameter vector
Require: s_0: Initial state

1: Sample a_0 from $\pi(\cdot|s_0)$
2: Transition to state s_1 from state s_0 and action a_0
3: $m \leftarrow 0$
4: **while** θ_m not converged **do**
5: Sample a_{m+1} from $\pi(\cdot|s_{m+1})$
6: Transition to state s_{m+2} from state s_{m+1} and action a_{m+1}
7: $s'_{m+1} \leftarrow s_m + (s_{m+2} - s_{m+1})$
8: $\hat{F}_m \leftarrow j(s_m, a_m, s_{m+1}; \theta_m)\nabla_\theta j(s_m, a_m, s'_{m+1}; \theta_m)$
9: $\theta_{m+1} \leftarrow \theta_m - \eta \hat{F}_m$
10: $m \leftarrow m+1$
11: **end while**

the gradient version, detailed in Algorithm 1. We comment on why the loss version is less accurate in Remark 3.

We prove in Lemma 2 that whether $\hat{F}$ is a good approximation of the unbiased estimate F mainly depends on three factors: 1) the variation of the drift $\mu(s,a)$ over the action space; 2) the size of the discretized time step ϵ; 3) the size of the discount factor γ. The smaller these three elements are, the closer BFF is to US.

Remark 1 *In Algorithm 1, only one future step is used for generating a new sample of s_{m+1}. In order to reduce the variance of the BFF gradient, it is useful to consider replacing the future step by a weighted average of multiple future steps. The estimate of the gradient then takes the form*

$$\hat{F}^n_m(\theta) = j(s_m, a_m, s_{m+1}; \theta) \sum_{i=1}^{n} \alpha_i \nabla_\theta j(s_m, a_m, s_m + \Delta s_{m+i}; \theta) \tag{9}$$

with $\sum_i \alpha_i = 1$. This comes at the cost of potentially increasing the estimate's bias.

The only change in implementation as a result of this remark is that line 4 in Algorithm 1 is replaced by the formula in equation (9).

Q-control The BFF algorithm mentioned above can be extended easily to Q-control, i.e., finding the value function Q^* of the optimal policy π_*. Q^* satisfies the Bellman equation $Q^*(s,a) = \mathbb{T}^{\pi_*}Q^*(s,a)$, where $\mathbb{T}^{\pi_*}$ is the optimal Bellman operator,

$$\mathbb{T}^{\pi_*}Q^*(s,a) = \mathbb{E}\left[r(s_{m+1}, s_m, a_m)] + \gamma \max_{a'} Q^*(s_{m+1}, a'; \theta) \middle| (s_m, a_m) = (s,a) \right], \tag{10}$$

where the expectation is taken over s_{m+1} when the optimal policy π_* is applied. The BRM problem is the same as (4) but with the absolute value of the Bellman residual $\delta(s,a;\theta)$ given by

$$\delta(s,a;\theta) = \left| \mathbb{T}^{\pi_*}Q^*(s,a) - Q^*(s,a) \right|. \tag{11}$$

Rather than generating a trajectory offline with a fixed policy, we instead generate a training trajectory online using an ϵ-greedy policy. The algorithm for this case is identical to Algorithm 1, but with j replaced by $j(s_m, a_m, s_{m+1}; \theta) = r(s_{m+1}, s_m, a_m) + \gamma \max_a Q^*(s_{m+1}, a; \theta) - Q^*(s_m, a_m; \theta)$. This is summarized in Algorithm 2.

Algorithm 2 BFF for Q-control

Require: η: learning rate
Require: $Q^*(s;\theta) \in \mathbb{R}^{|\mathbb{A}|}$: nonlinear function approximation of Q^* parameterized by θ
Require: $j(s_m, a_m, s_{m+1}; \theta) = r(s_{m+1}, s_m, a_m) + \gamma \max_a Q^*(s_{m+1}, a; \theta) - Q^*(s_m, a_m; \theta)$
Require: θ_0: Initial parameter vector
Require: s_0: Initial state

```
Sample a_0 with an ε-greedy policy from Q*(s_0; θ_0)
Transition to state s_1 from state s_0 and action a_0
m ← 0
while θ_m not converged do
    Sample a_{m+1} with an ε-greedy policy from Q*(s_{m+1}; θ_m)
    Transition to state s_{m+2} from state s_{m+1} and action a_{m+1}
    s'_{m+1} ← s_m + (s_{m+2} − s_{m+1})
    F̂_m ← j(s_m, a_m, s_{m+1}; θ_m) ∇_θ j(s_m, a_m, s'_{m+1}; θ_m)
    θ_{m+1} ← θ_m − η F̂_m
    m ← m + 1
end while
```

Why BFF works We prove in Lemmas D.1 and D.2 that the difference between the SC and US gradients is $O(\epsilon)$, while the difference between the BFF and US gradients is $O(\mathbb{E}[\delta\epsilon])$ (see Lemma 2). Although both differences are $O(\epsilon)$, BFF depends on the Bellman residual δ while SC does not. As the algorithm proceeds, δ approaches 0, causing the difference between BFF and US to further decrease. On the other hand, the difference between SC and unbiased SGD is always $O(\epsilon)$ as proved in Theorem D.3. This is the high-level reason why BFF outperforms SC. (See Section 4 for numerical comparisons.)

2.2. Discrete state space

The proposed method also works for discrete states in certain settings. A typical example is when the state space $\mathbb{S}$ is a discretization of an underlying continuous state space $\Omega_{\mathbb{S}}$[1] and the transition matrix is smooth in state s and action a. When the state space is discrete, one can view $Q \in \mathbb{R}^{|\mathbb{S}|\times|\mathbb{A}|}$ as a matrix. In this tabular form, one can directly use the previous function approximation framework by letting $Q^\pi(s, a; \theta) = \Phi(s, a) \cdot \theta$, where $\Phi(s_i, a_j) \in \mathbb{R}^{|\mathbb{S}|\times|\mathbb{A}|}$ is the matrix with (i, j)-th entry equal to 1 and all other entries equal to 0. Here $\theta \in \mathbb{R}^{|\mathbb{S}|\times|\mathbb{A}|}$ and we take the dot product of the flattened matrices.

Equivalently, one can also derive the BFF algorithm directly by computing an unbiased estimate for the gradient of the Bellman residual with respect to Q. We will denote the unbiased gradient at time m by the vector $F_m \in \mathbb{R}^{|\mathbb{S}||\mathbb{A}|}$, where F_m is indexed by state, action pairs. We first initialize

1. More specifically, a Riemannian manifold endowed with a connection structure

F_m to the 0 vector. Then, set

$$F_m(s_m, a_m) = -j(s_m, a_m, s_{m+1}).$$

Finally, for each $a \in \mathbb{A}$, set

$$F_m(s'_{m+1}, a) = \pi(a|s'_{m+1})\gamma j(s_m, a_m, s_{m+1}).$$

Here s'_{m+1} is an independent sample of the next step in the trajectory given s_m and a_m and $j(s_m, a_m, s_{m+1}) = r(s_{m+1}, s_m, a_m) + \gamma \sum_a Q^\pi(s_{m+1}, a)\pi(a|s) - Q^\pi(s_m, a_m)$. By replacing the independent sample s'_{m+1} with the BFF approximation $s_m + \Delta s_m$, we obtain a BFF algorithm for the tabular case, summarized in Algorithm 3.

Algorithm 3 BFF for Q-evaluation (tabular case)

Require: η: Learning rate
Require: $Q^\pi \in \mathbb{R}^{|\mathbb{S}|\times|\mathbb{A}|}$: matrix of $Q^\pi(s, a)$ values
Require: $j(s_m, a_m, s_{m+1}) = r(s_{m+1}, s_m, a_m) + \gamma \sum_a Q^\pi(s_{m+1}, a)\pi(a|s) - Q^\pi(s_m, a_m)$
Require: s_0: Initial state

```
Sample a_0 from π(·|s_0)
Transition to state s_1 from state s_0 and action a_0
m ← 0
while Q^π not converged do
    Sample a_{m+1} from π(·|s_{m+1})
    Transition to state s_{m+2} from state s_{m+1} and action a_{m+1}
    s'_{m+1} ← s_m + (s_{m+2} − s_{m+1})
    F̂_m ← 0 ∈ ℝ^{|S|×|A|}
    F̂_m(s_m, a_m) ← −j(s_m, a_m, s_{m+1})
    for a ∈ A do
        F̂_m(s'_{m+1}, a) ← π(a|s'_{m+1}) j(s_m, a_m, s_{m+1})
    end for
    Q^π ← Q^π − η F̂_m
    m ← m + 1
end while
```

The BFF algorithm for Q-control in the tabular case can be derived similarly. As in equation (9), one can use multiple future steps to reduce the variance of the gradient in the tabular case as well. Since we demonstrated this process for Q-evaluation in a continuous state space, we will demonstrate it for Q-control in this example to showcase its applicability for both policy evaluation and control. Specializing equation (9) for the tabular case is slightly more computationally involved than for the general case. As such, we provide the nBFF algorithm for tabular control in Algorithm 4 below. The single-step BFF algorithm for the tabular case (Algorithm 5) can be found in the appendix.

Algorithm 4 BFF for Q-control (Multiple-future-step, tabular case)

Require: η: Learning rate
Require: $Q^* \in \mathbb{R}^{|\mathbb{S}|\times|\mathbb{A}|}$: matrix of $Q^*(s,a)$ values
Require: $\{\alpha_k\}_{k=1}^n$ as in equation (9)
Require: $j(s_m, a_m, s_{m+1}) = r(s_{m+1}, s_m, a_m) + \gamma \max_a Q^*(s_{m+1}, a) - Q^*(s_m, a_m)$
Require: s_0: Initial state

for $i = 0, \ldots, n-1$ **do**
  Sample a_i with an ϵ-greedy policy from $Q^*(s_i; \theta_0)$
  Transition to state s_{i+1} from state s_i and action a_i
end for
$m \leftarrow 0$
while Q^* not converged **do**
  Sample a_{m+n} with an ϵ-greedy policy from $Q^*(s_{m+n}; \theta_m)$
  Transition to state s_{m+n+1} from state s_{m+n} and action a_{m+n}
  $\hat{F}_m \leftarrow 0 \in \mathbb{R}^{|\mathbb{S}|\times|\mathbb{A}|}$
  for $k = 1 \ldots, n$ **do**
    $s'_{m+k} \leftarrow s_{m+k-1} + (s_{m+k+1} - s_{m+k})$
    $\hat{F}_m(s_m, a_m) \leftarrow \hat{F}_m(s_m, a_m) - j(s_m, a_m, s_{m+1})$
    $a^*_{m+k} \leftarrow \arg\max_a Q^*(s'_{m+k}, a)$
    $\hat{F}_m(s'_{m+k}, a^*_{m+k}) \leftarrow \hat{F}_m(s'_{m+k}, a^*_{m+k}) + \alpha_k j(s_m, a_m, s_{m+1})$
  end for
  $Q^* \leftarrow Q^* - \eta \hat{F}_m$
  $m \leftarrow m + 1$
end while

3. Theoretical results

This section states the main theoretical results which bound the difference between BFF and US on a continuous state space. Recall that the one-step transition is governed by the state dynamics

$$s_{m+1} = s_m + \mu(s_m, a_m)\epsilon + \sigma\sqrt{\epsilon} Z_m, \tag{12}$$

where $\mu(s,a)$ is the drift, Z_m is assumed to be normal $N(0, I_{d_s \times d_s})$, and σ is the diffusion coefficient. It is convenient to introduce $\Delta s_m := s_{m+1} - s_m = \mu(s_m, a)\epsilon + \sigma\sqrt{\epsilon} Z_m$. For a discrete action space $\mathbb{A}$, the drift term $\{\mu(s,a)\}_{\alpha\in\mathbb{A}}$ is a family of continuous functions, while for a continuous action space, $\mu(s,a)$ is a continuous function in both state and action.

3.1. Error bound at each step

The following lemma bounds the difference between BFF and US at each step. That is, assuming the current parameters θ are the same, Lemma 2 bounds the expected difference between BFF and US for Q-evaluation and Q-control after one step. See Lemmas A.1 and A.2 for an extension of Lemma 2 to the difference of the variances. The complete proof of Lemma 2 is also give in Appendix A.

Lemma 2 (short version) *Suppose that $Q^\pi(s,a;\theta)$ and $\max_{a\in\mathbb{A}} Q^*(s,a;\theta)$ are Lipschitz continuous in θ, and that $\partial_s \nabla_\theta Q^\pi(s,a;\theta)$ and $\partial_s \nabla_\theta \max_{a\in\mathbb{A}} Q^*(s,a;\theta)$ are continuous in the state and*

action space. Then the difference between the BFF gradient $\hat{F}$ and the unbiased gradient F satisfies

$$\mathbb{E}[\hat{F}_m(\theta)] - \mathbb{E}[F_m(\theta)] = \mathbb{E}\left[\delta(s_m, a_m; \theta)\left(C(s_m;\theta)\epsilon + o(\epsilon)\right)\right] = O(\mathbb{E}[\delta\epsilon]),$$

where δ is the absolute value of the Bellman residual defined in (5), (11) for Q-evaluation and Q-control respectively. For Q-evaluation, $C(s_m;\theta)$ is defined as

$$C(s_m;\theta) = \gamma\left(\partial_s \mathbb{E}_{a\sim\pi(a|s_m)}[\nabla_\theta Q^\pi(s_m, a; \theta)]\right) C_2(s_m), \tag{13}$$

and for Q-control, $C(s_m;\theta)$ is defined as

$$C(s_m;\theta) = \gamma\left(\partial_s \nabla_\theta \max_{a'\in\mathbb{A}} Q^*(s, a; \theta)\right) C_2(s_m) \tag{14}$$

with $C_2(s_m)$ an upper bound for the variation of the drift in the action space: $|\mu(s_m, a_{m+1}) - \mu(s_m, a_m)| \leq C_2(s_m)$.

Note that the common factor $C_2(s)$ affects the magnitude of the difference between BFF and US. That is to say, when the drift changes more slowly with respect to the action, the difference is smaller and BFF performs better. The sizes of γ and ϵ play an important role in determining the statistical difference between BFF and US as well.

Remark 3 *(Zhu et al., 2020) proposed two versions of BFF for policy evaluation. One applies to the gradient (the same as the approach we take in this paper). The other approach (BFF-loss) applies the same idea to the loss function by minimizing a biased Bellman residual given by*

$$\mathbb{E}[\mathbb{E}[j(s_m, a_m, s_{m+1}; \theta) j(s_m, a_m, s_m + \Delta s_{m+1}; \theta)|s_m, a_m]]$$

with j an unbiased estimate of the Bellman residual defined in (7). However, this method does not extend to Q-evaluation. The reason is the following. The gradient of BFF-loss for Q-evaluation is

$$\frac{1}{2}\mathbb{E}[\mathbb{E}[j(s_{m+1})\nabla_\theta j(s_m + \Delta s_{m+1})|s_m, a_m] + \mathbb{E}[\nabla_\theta j(s_{m+1}) j(s_m + \Delta s_{m+1})|s_m, a_m]].$$

We omit the first two variables s_m, a_m in $j(s_m, a_m, s_{m+1})$ to emphasize that the main difference between BFF and US lies in the next state. From the proof of Lemma A.1/(A.6), given (s_m, a_m), one has $\nabla_\theta j(s_m + \Delta s_{m+1}) - \nabla_\theta j(s_{m+1}) = O(\epsilon)$. Similarly, we has $j(s_m + \Delta s_{m+1}) - j(s_{m+1}) = O(\epsilon)$. It follows that the difference between the BFF-loss gradient and the true loss gradient is

$$\text{(BFF-loss gradient)} - \mathbb{E}[\mathbb{E}[j(s_{m+1})|s_m]\nabla_\theta \mathbb{E}[j(s_{m+1})|s_m]] = O(\delta\epsilon + \nabla_\theta \delta\epsilon),$$

where $\delta = \mathbb{E}[|j(s_{m+1})||s_m, a_m]$ is the absolute value of the Bellman residual defined in (5). Since $\nabla\delta$ does not necessarily decrease as the algorithm proceeds (for example, if $\delta^2 = \theta^2$, i.e., $\delta = \theta$, then $\nabla\delta = 1$ is a constant), the bias for BFF-loss in Q-evaluation is still dominated by $O(\epsilon)$, which is similar to the behavior of SC. On the other hand, Lemma 3.1 shows that the difference for BFF-gradient is $O(\delta\epsilon)$, which will keep decreasing as δ decreases. Hence, BFF-gradient performs better.

The reason for this contrast with (Zhu et al., 2020) is that the state-action value function evaluation in this paper is based on a conditional expectation on state and action, while the state value function evaluation in (Zhu et al., 2020) is based on a conditional expectation on state alone. When conditioning only on the state, we have $j(s_m + \Delta s_{m+1}) - j(s_{m+1}) = O(\epsilon^2)$. This means that in the setting of (Zhu et al., 2020), BFF-loss has a difference of only $O(\epsilon^2)$ from US, which in turn makes its performance superior to the $O(\epsilon)$ bias incurred by SC.

3.2. Differences of density evolutions

This subsection compares the probability density functions (p.d.f.) for the parameters over the course of the complete BFF and US algorithms. To simplify the analysis, the p.d.f.s of the two algorithms are modeled with the p.d.f.s of the continuous stochastic processes. The updates of the parameter θ_k by SGD follows

$$\begin{aligned} \text{US:} \quad & \theta_{k+1} = \theta_k - \eta F_m(\theta_k) \\ \text{BFF:} \quad & \theta_{k+1} = \theta_k - \eta \hat{F}_m(\theta_k) \end{aligned}$$

where $F_k, \hat{F}_k$ are the estimates of the loss function's gradient defined in (6) and (8) for US and BFF algorithm respectively. The above updates can be viewed as a discretization of a function in time $\Theta_t \equiv \Theta(t)$. It is shown in (Li et al., 2017; Hu et al., 2017) that when the learning rate η is small, the dynamics of SGD can be approximated by a continuous time SDE

$$\begin{aligned} \text{US:} \quad & d\Theta_t = -\mathbb{E}[F_m(\Theta_t)]dt + \sqrt{\eta}\mathbb{V}[F_m(\Theta_t)]dB_t \\ \text{BFF:} \quad & d\Theta_t = -\mathbb{E}[\hat{F}_m(\Theta_t)]dt + \sqrt{\eta}\mathbb{V}[\hat{F}_m(\Theta_t)]dB_t \end{aligned} \tag{15}$$

for $\Theta_{t=k\eta} \approx \theta_k$ with an error of $O(\sqrt{\eta})$, where $\mathbb{E}$ and $\mathbb{V}$ are expectation and variance taken over $\rho(s,a)$, the distribution defined in the loss function (4). Here $\mathbb{E}[F_m(\Theta_t)]$ denotes the true gradient of the population loss function used in US, and $\mathbb{E}[\hat{F}_m(\Theta_t)]$ denotes the biased gradient of the population loss used in BFF. For simplicity, we assume $\mathbb{V}[F_m] \equiv \xi$ is constant. Let $p(t,\theta)$ and $\hat{p}(t,\theta)$ be the p.d.f.s of the parameter θ at step $k = t/\eta$ for US and BFF, respectively. These p.d.f.s satisfy the following two equations (Pavliotis, 2014):

$$\text{US:} \quad \partial_t p = \nabla_\theta \cdot \left[\mathbb{E}[F_m]p + \frac{\eta}{2}\nabla_\theta \cdot (\mathbb{V}[F_m]p)\right]; \tag{16}$$

$$\text{BFF:} \quad \partial_t \hat{p} = \nabla_\theta \cdot \left[\mathbb{E}[\hat{F}_m]\hat{p} + \frac{\eta}{2}\nabla_\theta \cdot \left(\mathbb{V}[\hat{F}_m]\hat{p}\right)\right]. \tag{17}$$

In addition, we assume $\theta \in \Omega$ with Ω compact. As a result, we need a reflective boundary condition for the PDEs (16), (17), i.e.

$$\left(\mathbb{E}[F_m]p + \frac{\eta}{2}\nabla\left(\mathbb{V}[F_m]p\right)\right) \cdot \boldsymbol{n}\Big|_{\partial\Omega} = 0, \quad \left(\mathbb{E}[\hat{F}_m]\hat{p} + \frac{\eta}{2}\nabla\left(\mathbb{V}[\hat{F}_m]\hat{p}\right)\right) \cdot \boldsymbol{n}\Big|_{\partial\Omega} = 0.$$

It is not clear whether the compactness assumption can be removed; we leave this question for future study. In practice, the BFF algorithm still works for unconstrained domains.

We define

$$\hat{d}(t,\theta) = p - \hat{p}$$

to be difference of the p.d.f.s for US and BFF. The goal in this section is to analyze how $\left\|\hat{d}\right\|_*^2$ evolves in time for some specific norm $\|\cdot\|_*$.

We introduce the following weighted norm to measure the difference between the p.d.f.s:

$$\left\|\hat{d}\right\|_*^2 := \int \hat{d}^2 \frac{1}{p^\infty} d\theta$$

where p^∞ is the steady distribution of US. p^∞ is obtained by setting the RHS of (16) to be zero. Since $\mathbb{E}[F_m] = \nabla_\theta \mathbb{E}_{(s,a)\sim\rho}[\delta^2(s,a;\theta)]$, where $\rho(s,a)$ is defined in the loss function (4) and $\delta(s,a;\theta)$

is the absolute Bellman residual defined in (5) for Q-evaluation and in (11) for Q-control, it is easy to check that the steady distribution of US is

$$p^{\infty} = \frac{1}{Z} e^{-\frac{2}{\eta\xi}\mathbb{E}[\delta^2]}, \tag{18}$$

where $Z = \int e^{-\frac{2}{\eta\xi}\mathbb{E}[\delta^2]} d\theta$ is a normalizing constant. We have reduced our problem to quantifying the evolution of $\left\|\hat{d}\right\|_*^2$ in time, which we accomplish with the following theorem.

Theorem 4 (short version) *For sufficiently small $\eta > 0$, the difference $\hat{d}$ of the p.d.f.s for US and BFF is bounded by*

$$\left\|\hat{d}(t)\right\|_* \leq C_1 e^{-C_2 t} + O\left(\epsilon\sqrt{\mathbb{E}[\delta_*^2]}\eta^{C_3}\right)\sqrt{1 - e^{-C_2 t}}, \tag{19}$$

where $\mathbb{E}[\delta_*^2] = \min_\theta \mathbb{E}[\delta^2]$ and C_1, C_2, C_3 are all positive constants.

Remark 5 *In (Zhu et al., 2020), the upper bound for $\left\|\hat{d}(t)\right\|_*$ is $O(\epsilon^2)$ (see Theorem 4 in (Zhu et al., 2020)), which is independent of time. In this paper, we give the tighter and more delicate time-dependent bound of $O(e^{-t} + \epsilon\sqrt{\mathbb{E}[\delta_*^2]})$, where δ_* is the smallest residual unbiased SGD can achieve. This bound tells us that the difference will first decay exponentially and end up at $O(\epsilon\mathbb{E}[\delta_*])$, which gives more information about the evolution of the algorithm compared to (Zhu et al., 2020)'s upper bound. Notice that in (Zhu et al., 2020), the difference between unbiased SGD and BFF in V-value evaluation is $O(\epsilon^2)$, while in Q-value evaluation as in this paper, the difference is $O(\epsilon)$, so the bound becomes $O(\epsilon\mathbb{E}[\delta_*])$ instead of $O(\epsilon^2\mathbb{E}[\delta_*])$.*

The precise version of Theorem 4 is stated in Theorem C.5 of Appendix C, and its proof is also given in Appendix C. This theorem implies that as the algorithm moves on, the difference between BFF and US will decay exponentially. After running the algorithm for sufficiently many steps, the difference will eventually be $O\left(\epsilon\sqrt{\mathbb{E}[\delta_*^2]}\eta^{C_3}\right)$. As long as $\mathbb{E}[\delta_*^2]$ is small, BFF will achieve a minimizer close to US with an error much smaller than $O(\epsilon)$. Note that if $\mathbb{E}[\delta_*^2] = 0$, the difference still does not vanish. Instead, the leading order term of the last term in (19) becomes $O(\epsilon\eta^{C_3+1/2})$, which is shown in Corollary C.2 of Appendix C.

The constant C_1 depends on the initial p.d.f. of the algorithm. The constant C_3 is related to the shape of $\mathbb{E}[\delta_*^2](\theta)$ in the parameter space. The flatter the shape at the minimizer is, the smaller C_3 is. The constant C_2 decreases as η decreases, so the first term increases as η decreases, while the last term $O(\epsilon^2\mathbb{E}[\delta_*^2]\eta^{C_3})$ does the opposite. This suggests that one should set the learning rate η large at first, making the exponential decay faster. As the training progresses, η should be reduced to make the final error smaller.

4. Numerical examples

In each of the settings below, we test the efficiency of learning Q^π via SC and BFF. We use the generalized version of BFF specified by equation (9). The label nBFF in the plots corresponds to using the estimate $\hat{F}^n$ from equation (9); 1BFF corresponds to the standard BFF algorithm (e.g. Algorithms 1 and 3). In each case, we use the uniform weights $\alpha_i = 1/n$. When applicable, we also compare to US and PD. (For the full definition of the PD algorithm, see Appendix E.)[2]

2. Code for reproducing these experiments can be found at https://github.com/zleizzo/bffQ.

4.1. Continuous state space

We consider an MDP with continuous state space equal to the unit circle: $\mathbb{S} = S^1 = \mathbb{R}/2\pi\mathbb{Z}$, and $s \in \mathbb{S}$ represents the angle of a point on the unit circle. The transition dynamics are

$$\Delta s_m = a_m \epsilon + \sigma Z_m \sqrt{\epsilon},$$

where $a_m \in \mathbb{A} = \{\pm 1\}$ is drawn from policy π to be defined later and $Z_m \sim N(0,1)$. We set $\epsilon = \frac{2\pi}{32}$ and $\sigma = 0.2$. The reward function is $r(s_{m+1}, s_m, a_m) = \sin(s_{m+1}) + 1$.

In the first two experiments, we approximate Q^π with a neural network with two hidden layers. Each hidden layer contains 50 neurons and cosine activations. The NN takes a state as input and outputs a vector in $\mathbb{R}^{|\mathbb{A}|}$; the i-th entry of the output vector corresponds to $Q^\pi(s, a_i)$. The CartPole experiment uses a larger network with ReLU activations.

Q-evaluation We first estimate Q^π for the fixed policy $\pi(a|s) = 1/2 + a\sin(s)/5$. We use a neural network with two hidden layers to approximate Q. Each hidden layer has 50 neurons. The activations are $\cos(x)$ for the hidden layers and identity for the output layer.

The training procedure is as follows. We generate a trajectory of length 10^6 and run BFF, SC, and US with batch size $M = 50$ and learning rate $\eta = 0.1$. We also train via PD with $\beta = \eta = 0.1$ and all other hyperparameters identical. We compute the exact Q by running US on a trajectory of length 10^7. The results are plotted in Figure 1. BFF exhibits superior performance compared to SC and PD, with only slightly worse performance than the (impractical) US algorithm.

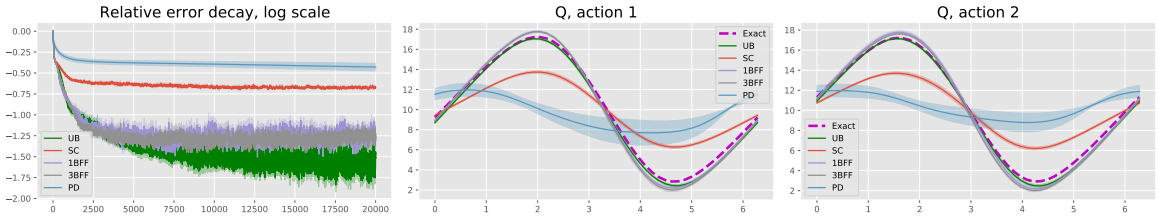

Figure 1: Results of each method for fixed-policy Q-evaluation. The solid lines are the mean of five runs and the shaded region denotes the standard error of the mean. The BFF algorithm performs better than both SC and PD. Changing the number of future steps used to compute the BFF approximation does not have a large impact on its performance in this case.

Q-control In the control case, a fixed behavior policy is used to generate the training trajectory. At each step, the behavior policy samples an action uniformly at random, i.e. $\pi(a|s) = 1/2$ for all $a \in \mathbb{A}$ and $s \in \mathbb{S}$. For the simple case we consider here, we found that this fixed policy worked better for training than an ϵ-greedy policy. Other than this minor difference, the training procedure is identical to the continuous Q-evaluation experiment (with the same hyperparameters, trajectory length, etc.).

The results are shown in Figure 2. Again, BFF has comparable performance to SC and outperforms both SC and PD.

CartPole We tested the BFF algorithm on the CartPole environment from OpenAI gym (Brockman et al. , 2016). It is straightforward to apply BFF to adaptive SGD algorithms such as Adam (Kingma et al. , 2014) and a version of BFF with Adam is used for this experiment.

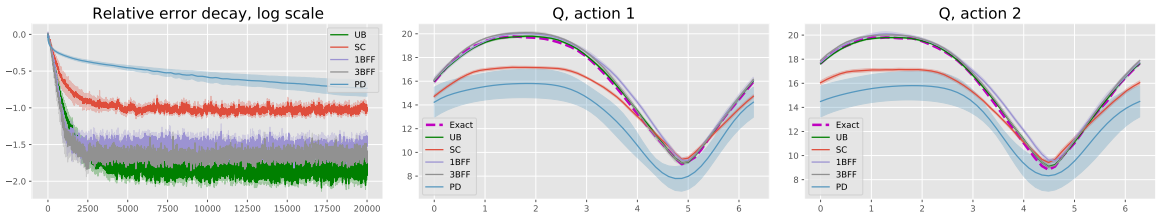

Figure 2: Results of each method for Q-control. The solid lines are the mean of five runs and the shaded region denotes the standard error of the mean. As before, the more accurate gradient estimate from BFF improves our learned approximation for Q. In this case, the variance reduction obtained from 4 future steps improved BFF's performance even more, giving results comparable to US.

We injected a small amount of noise into the environment: the actions correspond to applying a force of $\pm 10 + N(0, 1)$ to the cart. BFF shows improvement over SC even for very small amounts of noise (i.e. variance close to 0). In fact, even in the deterministic base environment, we found that BFF performed better than SC. BFF may promote exploration in some structured way which is helpful for this environment.

We approximate Q with a neural network with a single hidden layer of size 100. The hidden layer has ReLU activations. For both BFF and sample-cloning, we train using Adam with the default settings for β_1 and β_2 ($\beta_1 = 0.9$, $\beta_2 = 0.999$). For all of the methods, we use batch size 50 and experience replay storing the 10,000 most recent experiences in the training trajectory. We train for 200 episodes. We also use an ϵ-greedy approach to generate the trajectory. Initially, we set $\epsilon = 1$ (so the agent acts completely randomly at the beginning of training), and decay ϵ by 0.99 after each parameter update. We stop decaying ϵ when it reaches 0.1, so there is always some randomness in our training actions to prevent getting stuck on an ineffective policy.

We tested learning rates in $\{10^{-k}\}_{k=2}^{4}$ for SC and BFF. The initial learning rates for PD were chosen from $\{10^{-k}\}_{k=1}^{3}$. The batch size, number of training episodes, reward discount factor, and epsilon decay rate were constant across the different methods. There was no additional tuning for BFF. ϵ starts at 1 and is set to $\max(0.1, 0.99\epsilon)$ at the conclusion of each episode.

For the PD algorithm, We tried fixed values for β and η, as well as decaying β and η with different starting values and with the decay recommended in (Wang et al., 2017). The results in Figure 3 have $\beta_k = 0.1 \times k^{-3/4}$ and $\eta_k = 0.1 \times k^{-1/2}$, where β_k and η_k denote the parameters used for the k-th step.

The results are plotted in Figure 3. BFF reaches the max reward (200) faster than SC and achieves it with greater regularity throughout the training process. In contrast to both of these methods, the PD method fails to converge even after the extensive hyperparameter search described above.

4.2. Tabular case

We next consider an MDP with a discrete state space $\mathbb{S} = \{\frac{2\pi k}{n}\}_{k=0}^{n-1}$ and $n = 32$. The transition dynamics are given by

$$\Delta s_m = \frac{2\pi}{n} a_m \epsilon + \sigma Z_m \sqrt{\epsilon}, \tag{20}$$

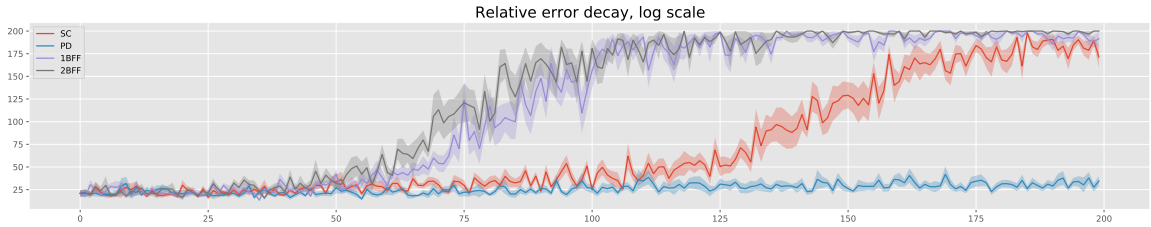

Figure 3: Reward per training episode for the CartPole experiment. The shaded regions show the standard error of the mean over 5 trials. BFF is the first to reach the maximum reward and achieves it more consistently than sample-cloning. It achieves slightly better performance using 2 future steps (2BFF in the plot). Despite an extensive hyperparameter search, PD was not able to learn an effective policy.

where $a_m \in \mathbb{A} = \{\pm 1\}$ is drawn from the policy $\pi(a|s) = 1/2 + a\sin(s)/5$ and $Z_m \sim N(0,1)$, and the addition is performed in $\mathbb{R}/2\pi\mathbb{Z}$. We then set $s_{m+1} = \arg\min_{s\in\mathbb{S}} |s_m + \Delta s_m - s|$. For the experiment below, $\sigma = 1$ and $\epsilon = 1$.

Q-evaluation The training procedure is as follows. We generate a long trajectory of length $T = 10^7$ from the MDP dynamics using a fixed policy $\pi(a|s) = \frac{1}{2} + a\frac{\sin(s)}{5}$. We use a learning rate of $\eta = 0.5$ and a batch size of 50 for each of the methods. We find the exact matrix Q^* by first forming a Monte Carlo estimate of the transition matrix $\mathcal{P}$ based on 50,000 repetitions per entry, then forming the expected reward vector R and solving the Bellman equation based on this estimate for $\mathcal{P}$.

The results are plotted in Figure 4. In this case, BFF is nearly indistinguishable from training via US. The PD method achieves comparable performance in the tabular case.

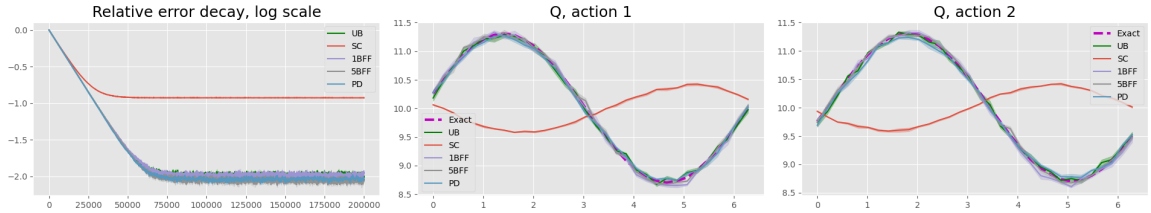

Figure 4: Results of each method for fixed-policy Q-evaluation in the tabular case. The solid lines are the mean of five runs and the shaded region denotes the standard error of the mean. BFF gives a better estimate for the gradient than SC, leading to improved performance. BFF's performance does not change significantly with the number of future steps.

Q-control We find the exact Q^* by first running US on a trajectory of length 10^8 with batch size 1000 and learning rate 0.5 to obtain an approximation Q^1. We then refine Q^1 by training via US on a trajectory of length 10^7 with batch size 10000 and a learning rate of 0.1 to obtain the true Q^*. We confirm the correctness of Q^* via Monte Carlo (not shown).

We test each of the methods (US, SC, and BFF) on a trajectory of length 5×10^7 with a learning rate of 0.5 and a batch size of 100. The results are shown in Figure 5. BFF outperforms SC by a wide margin and has performance comparable to US. Using a greater number of future steps to approximate the BFF gradient improved its performance marginally. The PD method also performs well for the tabular case.

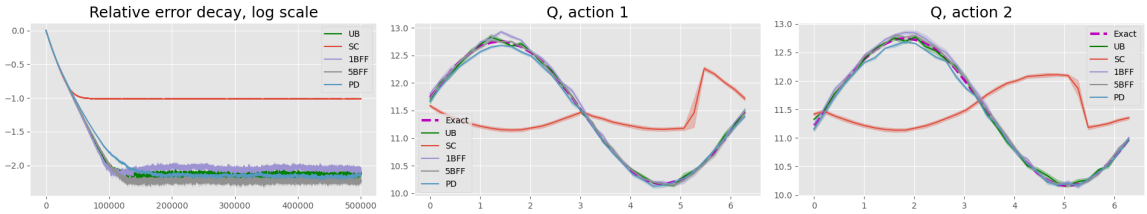

Figure 5: Results of each method for Q-control in the tabular case. The solid lines are the mean of five runs and the shaded region denotes the standard error of the mean. SC is unable to learn an accurate approximation for Q, while BFF's performance is almost indistinguishable from US. Using 5 future steps to compute the BFF approximation helped improve its performance slightly.

5. Conclusion

In this paper, we show that BFF has an advantage over other BRM algorithms for model-free RL problems with continuous state spaces and smooth underlying dynamics. We also prove that the difference between the BFF algorithm and the uncorrelated sampling algorithm first decays exponentially and eventually stabilizes at an error of $O(\epsilon\delta_*)$, where δ_* is the smallest Bellman residual that US can achieve.

We remark that the SDE interpretation of the underlying dynamics (i.e. (2)) is not necessary to apply our algorithms. Similar to the result of Lemma 2, if the underlying transition is smooth in the state space and the variation of the underlying transition w.r.t. the action space is small, this should be sufficient for BFF to perform well. We leave the relaxation of this assumption for future study.

There are several interesting directions for future work. For example, the underlying dynamics of an MDP may exhibit sudden changes at the boundary of the state space. Adapting BFF to work in these settings is important for improving its practical efficacy. Furthermore, in this paper we have restricted ourselves to learning a policy indirectly via the Q function. Applying BFF to direct policy computation methods (e.g. policy gradient) is another promising direction for exploration.

Acknowledgments

The work of L.Y. and Y.Z. is partially supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Scientific Discovery through Advanced Computing (SciDAC) program. The work of L.Y. is also partially supported by the National Science Foundation under award DMS-1818449. Z.I. is supported by a Stanford Interdisciplinary Graduate Fellowship.

References

Leemon Baird. Residual algorithms: Reinforcement learning with function approximation. *Machine Learning Proceedings*, pg. 30–37, 1995.

Shalabh Bhatnagar, Doina Precup, David Silver, Richard S. Sutton, Hamid R. Maei, Csaba Szepesvári. Convergent Temporal-Difference Learning with Arbitrary Smooth Function Approximation. *NIPS*, 2009.

David Bissell, Thomas Birtchnell, Anthony Elliott, and Eric L. Hsu. Autonomous automobilities: The social impacts of driverless vehicles. *Current Sociology*, 2020.

Steven J. Bradtke. Reinforcement learning applied to linear quadratic regulation. *Advances in neural information processing systems*:295–302, 1993.

Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. OpenAI Gym. *arXiv preprint arXiv:1606.01540*, 2016.

Bo Carlsson. Technological systems and economic performance: the case of factory automation. Springer Science & Business Media, 2012.

Bo Dai, Albert Shaw, Lihong Li, Lin Xiao, Niao He, Zhen Liu, Jianshu Chen, Le Song. SBEED: Convergent reinforcement learning with nonlinear function approximation. In International. *ICML*, 2018.

Kenji Doya. Reinforcement learning in continuous time and space. *Neural computation*: (12)1, 2000.

Shixiang Gu, Ethan Holly, Timothy Lillicrap, and Sergey Levine. Deep Reinforcement Learning for Robotic Manipulation with Asynchronous Off-Policy Updates. *ICRA*, 2017.

Hado van Hasselt, Arthur Guez, and David Silver. Deep Reinforcement Learning with Double Q-learning In *AAAI Conference on Artificial Intelligence*, AAAI, 2016.

Wenqing Hu, Chris Junchi Li, Lei Li, and Jian-Guo Liu. On the diffusion approximation of nonconvex stochastic gradient descent. *arXiv preprint arXiv:1705.07562*, 2017.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations*, ICLR, 2015.

Wassily Leontief and Duchin Faye. The Future Impact of Automation on Workers. Oxford University Press, 1986.

Qianxiao Li, Cheng Tai, et al. Stochastic modified equations and adaptive stochastic gradient algorithms. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2101–2110. JMLR. org, 2017.

Bo Liu, Ji Liu, Mohammad Ghavamzadeh, Sridhar Mahadevan and Marek Petrik. Finite-sample analysis of proximal gradient TD algorithms. *UAI*, 2015.

Sridhar Mahadevan, Bo Liu, Philip Thomas, Will Dabney, Steve Giguere, Nicholas Jacek, Ian Gemp, Ji Liu. Proximal reinforcement learning: A new theory of sequential decision making in primal-dual spaces. *arXiv preprint*, arXiv:1405.6757, 2014.

Volodymyr Mnih, Adrià Puigdomènech Badia, Mehdi Mirza, Alex Graves, Timothy P. Lillicrap, Tim Harley, David Silver and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. *ICML*, pp. 1928–1937, 2016.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint*, arXiv:1312.5602, 2013.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518:529–533, 2015.

Grigorios A. Pavliotis. *Stochastic processes and applications: diffusion processes, the Fokker-Planck and Langevin equations*. Springer, 2014.

Mads Lund Pedersen, Michael J Frank and Guido Biele. The drift diffusion model as the choice rule in reinforcement learning. *Psychonomic bulletin & review*, 24(4):1234–1251, 2017.

Gavin A. Rummery and Mahesan Niranjan. *On-line Q-learning using connectionist systems*. University of Cambridge, Department of Engineering Cambridge, UK, 1994

Ahmad El Sallab, Mohammed Abdou, Etienne Perot, and Senthil Yogamani. Deep Reinforcement Learning framework for Autonomous Driving. *Electronic Imaging*, 2017.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587): 484, 2016.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *Nature*, 550(7676):354, 2017.

Richard S Sutton. Learning to predict by the methods of temporal differences. *Machine learning*, 3 (1):9–44, 1988.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

Richard S. Sutton, Hamid Reza Maei, Doina Precup, Shalabh Bhatnagar, David Silver, Csaba Szepesvàri, and Eric Wiewiora. Fast gradient-descent methods for temporal-difference learning with linear function approximation. *ICML*, pp. 993–1000, 2009.

Richard S. Sutton, Csaba Szepesvári and Hamid Reza Maei. A convergent O (n) algorithm for off-policy temporal-difference learning with linear function approximation. *NIPS*, pg. 1609-1616, 2008.

Mengdi Wang. Randomized linear programming solves the Markov decision problem in nearly linear (sometimes sublinear) time. *Mathematics of Operations Research*, 45(2):517–546, 2020.

Mengdi Wang, Ethan X Fang, and Han Liu. Stochastic compositional gradient descent: algorithms for minimizing compositions of expected-value functions. *Mathematical Programming*, 161(1-2):419–449, 2017.

Mengdi Wang, Ji Liu, and Ethan Fang. Accelerating stochastic composition optimization. In *Advances in Neural Information Processing Systems*, pg. 1714–1722, 2016.

Christopher J.C.H. Watkins. Learning from Delayed Rewards. *PhD thesis*, King's College, University of Cambridge, UK, 1989.

Yuhua Zhu and Lexing Ying Borrowing From the Future: An attempt to address double sampling. *MSML*, accepted, 2020.

Appendices

A. Extension and Proof of Lemma 2

Lemma A.1 (Extension of Lemma 2 for Q-evaluation) *Suppoese that $Q^\pi(s,a;\theta)$ is Lipschitz continuous in the $\theta \in \mathbb{R}^{d_\theta}$ and $\partial_s \nabla_\theta Q^\pi(s,a;\theta)$ is continuous in the state space, then the difference between the gradients of the US and BFF algorithms for Q-evaluation is*

$$\mathbb{E}[\hat{F}_m] - \mathbb{E}[F_m] = \mathbb{E}\left[\delta(s_m, a_m; \theta) C(s_m;\theta)\epsilon\right] + o(\epsilon) = O(\mathbb{E}[\delta\epsilon]),$$

where

$$C(s_m;\theta) = \gamma\left(\partial_s \mathbb{E}_{a\sim\pi(a|s_m)}[\nabla_\theta Q^\pi(s_m,a;\theta)]\right) C_2,$$

and C_2 is an upper bound for the variation of the drift term in the action space $|\mu(s_m, a_{m+1}) - \mu(s_m, a_m)| \le C_2$. In addition, if $|\mathbb{E}_a[Q^\pi(s,a;\theta)] - Q(s,a;\theta)|$, $|\mathbb{E}_a[\nabla_\theta Q^\pi(s,a;\theta)] - \nabla_\theta Q(s,a;\theta)|$, $|\mu(s,a) - \mu(s,a')|$, $|r(s,s,a)| \le C$ almost surely for $s \in \mathbb{S}, a \in \mathbb{A}, \theta \in \mathbb{R}^{d_\theta}$, then the difference between the variances can also be bounded by

$$\left|\mathbb{V}[\hat{F}_m] - \mathbb{V}[F_m]\right| = O(\epsilon),$$

where $\mathbb{V}$ stands for the variance and

$$\begin{aligned}
&F_m = j(s_m, a_m, s_{m+1};\theta)\nabla_\theta j(s_m, a_m, s'_{m+1};\theta),\\
&\hat{F}_m = j(s_m, a_m, s_{m+1};\theta)\nabla_\theta j(s_m, a_m, s_m + \Delta s_{m+1};\theta),\\
&\quad j(s_m, a_m, s_{m+1};\theta_m) = r(s_{m+1}, s_m, a_m) + \gamma\int Q^\pi(s_{m+1}, a;\theta)\pi(a|s_{m+1})da \ - Q^\pi(s_m, a_m;\theta),\\
&\quad \delta(s_m, a_m;\theta) = \mathbb{E}\left[j(s_m, a_m, s_{m+1};\theta_m)|\, s_m, a_m\right].
\end{aligned}$$

Note that the above form also works for the discrete action spaces. Specifically, $\pi(a|s)da = \sum_{a_i\in\mathbb{A}} \pi(a_i|s)\delta_{a_i}(a)da$ in the discrete action space, where $\delta_{a_i}(a)$ is the Dirac delta function.

Proof The expectation of the US gradient and the BFF gradient are

$$\mathbb{E}[F_m] = \mathbb{E}[\mathbb{E}[j\nabla_\theta j'|s_m, a_m]], \quad \mathbb{E}[\hat{F}_m] = \mathbb{E}\left[\mathbb{E}\left[j\nabla_\theta \hat{j}|s_m, a_m\right]\right] \tag{A.1}$$

respectively, with $j = j(s_m, a_m, s_{m+1};\theta)$, $j' = j(s_m, a_m, s'_{m+1};\theta)$, $\hat{j} = j(s_m, a_m, s_m + \Delta s_{m+1};\theta)$.

For notational convenience, in what follows we drop the explicit dependence of Q^π on θ. All of the gradients ∇ are taken with respect to θ. For ease of exposition, we consider a one-dimensional state space $\mathbb{S}$. It is straightforward to generalize to the multi-dimensional case. Using a Taylor expansion, we can expand $\nabla Q^\pi(s_{m+1}, a)\pi(a|s_{m+1})$ around $\nabla Q^\pi(s_m, a)\pi(a|s_m)$ by

$$\begin{aligned}
&\nabla Q^\pi(s_{m+1}, a)\pi(a|s_{m+1})\\
=&\nabla Q^\pi(s_m, a)\pi(a|s_m) + \partial_s(\nabla Q^\pi(s_m, a)\pi(a|s_m))\Delta s_m + \frac{1}{2}\partial_s^2(\nabla Q^\pi(s_m, a)\pi(a|s_m))\Delta s_m^2\\
&+ \frac{1}{3}\partial_s^3(\nabla Q^\pi(s^*, a)\pi(a|s^*))\Delta s_m^3,
\end{aligned} \tag{A.2}$$

where $s^* \in (s_m, s_{m+1})$. Since $\Delta s_m = \mu(s_m, a_m)\epsilon + \sigma Z_m \sqrt{\epsilon} = O(\sqrt{\epsilon})$, the last term of the above equation is $o(\epsilon)$. Substituting $\Delta s_m = \mu(s_m, a_m)\epsilon + \sigma Z_m \sqrt{\epsilon}$ yields

$$\begin{aligned}
&\nabla_\theta j' = \gamma \int \nabla Q^\pi(s'_{m+1}, a)\pi(a|s'_{m+1})da - \nabla Q^\pi(s_m, a_m) \\
=&\underbrace{\gamma \int \nabla Q^\pi(s_m, a)\pi(a|s_m)da - \nabla Q^\pi(s_m, a_m)}_{f_0} + \underbrace{\left(\gamma \int \partial_s(\nabla Q^\pi(s_m, a)\pi(a|, s_m))da\right)\mu(s_m, a_m)}_{f_1}\epsilon \\
&+ \underbrace{\left(\gamma \int \partial_s(\nabla Q^\pi(s_m, a)\pi(a|, s_m))da\right)\sigma}_{f_2} Z'_m\sqrt{\epsilon} + \underbrace{\left(\gamma \int \partial_s^2(\nabla Q^\pi(s_m, a)\pi(a|s_m))da\right)\sigma^2}_{f_3}(Z'_m)^2\epsilon + o(\epsilon).
\end{aligned} \tag{A.3}$$

Similarly, we can expand $\nabla Q^\pi(s_m + \Delta s_{m+1}, a)\pi(a|s_m + \Delta s_{m+1})$ around $\nabla Q^\pi(s_m, a)\pi(a|s_m)$. This yields

$$\begin{aligned}
&\nabla Q^\pi(s_{m+1}, a)\pi(a|s_{m+1}) \\
=&\nabla Q^\pi(s_m, a)\pi(a|s_m) + \partial_s(\nabla Q^\pi(s_m, a)\pi(a|s_m))\Delta s_{m+1} + \partial_s^2(Q^\pi(s_m, a)\pi(a|s_m))\Delta s_{m+1}^2 + o(\epsilon).
\end{aligned} \tag{A.4}$$

By Taylor expanding $\mu(s_{m+1}, a_{m+1})$ around $\mu(s_m, a_{m+1})$ and using the fact that $\Delta s_m = O(\sqrt{\epsilon})$, we see that

$$\begin{aligned}
\mu(s_{m+1}, a_{m+1}) &= \mu(s_m, a_{m+1}) + \partial_s\mu(s_m, a_{m+1})\Delta s_m + O(\Delta s_m^2) \\
&= \mu(s_m, a_{m+1}) + o(1).
\end{aligned}$$

Substituting this into the expression for Δs_{m+1} yields

$$\Delta s_{m+1} = \mu(s_{m+1}, a_{m+1})\epsilon + \sigma Z_{m+1}\sqrt{\epsilon} = \mu(s_m, a_{m+1})\epsilon + \sigma Z_{m+1}\sqrt{\epsilon} + o(\epsilon).$$

Combining this expression for Δs_{m+1} with the Taylor expansion of ∇Q^π, we conclude that

$$\begin{aligned}
&\nabla_\theta \hat{j} = \gamma \int \nabla Q^\pi(s_m + \Delta s_{m+1}, a)\pi(a|s_m + \Delta s_{m+1})da - \nabla Q^\pi(s_m, a_m) \\
=&f_0 + \underbrace{\left(\gamma \int \partial_s(\nabla Q^\pi(s_m, a)\pi(a|s_m))da\right)\mu(s_m, a_{m+1})}_{\hat{f}_1}\epsilon + f_2 Z_{m+1}\sqrt{\epsilon} + f_3 Z_{m+1}^2\epsilon + o(\epsilon).
\end{aligned} \tag{A.5}$$

It follows that

$$\nabla\hat{j} - \nabla j' = (\hat{f}_1 - f_1)\epsilon + f_2(Z_{m+1} - Z'_m)\sqrt{\epsilon} + f_3(Z_{m+1}^2 - (Z'_m)^2)\epsilon + o(\epsilon), \tag{A.6}$$

which implies

$$\begin{aligned}
&\mathbb{E}[\hat{F}_m] = \mathbb{E}[\mathbb{E}[j\nabla\hat{j}|s_m, a_m]] \\
=&\mathbb{E}[\mathbb{E}[j(\nabla j' + (\hat{f}_1 - f_1)\epsilon + f_2(Z_{m+1} - Z'_m)\sqrt{\epsilon} + f_3(Z_{m+1}^2 - (Z'_m)^2)\epsilon + o(\epsilon))|s_m, a_m]] \\
=&\mathbb{E}[\mathbb{E}[j\nabla j'|s_m, a_m]] + \mathbb{E}\left[\mathbb{E}\left[j\left((\hat{f}_1 - f_1)\epsilon + f_2(Z_{m+1} - Z'_m)\sqrt{\epsilon} + f_3(Z_{m+1}^2 - (Z'_m)^2)\epsilon + o(\epsilon)\right)|s_m, a_m\right]\right].
\end{aligned} \tag{A.7}$$

Since both $j = j(s_m, a_m, s_m + \mu(s_m, a_m)\epsilon + \sigma\sqrt{\epsilon}Z_m)$ and $f_2 = f_2(s_m)$ are independent of Z_{m+1}, Z'_m, we have

$$\mathbb{E}\left[\mathbb{E}\left[j\left(f_2(Z_{m+1} - Z'_m)\sqrt{\epsilon}\right)|s_m, a_m\right]\right] = \mathbb{E}\left[\mathbb{E}\left[jf_2|s_m, a_m\right]\mathbb{E}\left[(Z_{m+1} - Z'_m)\sqrt{\epsilon}|s_m, a_m\right]\right] = 0.$$

Similarly,

$$\mathbb{E}\left[\mathbb{E}\left[j\left(f_3(Z^2_{m+1} - (Z'_m)^2)\epsilon\right)|s_m, a_m\right]\right] = \mathbb{E}\left[\mathbb{E}\left[jf_3|s_m, a_m\right]\mathbb{E}\left[(Z^2_{m+1} - (Z'_m)^2)\epsilon|s_m, a_m\right]\right] = 0.$$

Hence, (A.7) becomes

$$\mathbb{E}[\hat{F}_m] = \mathbb{E}[F_m] + \mathbb{E}[\mathbb{E}[j((\hat{f}_1 - f_1)\epsilon + o(\epsilon))|s_m, a_m]]$$

Let $C_1(s_m) = \left|\int \partial_s(\nabla Q^\pi(s_m, a)\pi(a|s_m))da\right|$. As $|\mu(s_m, a_{m+1}) - \mu(s_m, a_m)| \leq C_2(s_m)$ by assumption,

$$|\mathbb{E}[\hat{F}_m] - \mathbb{E}[F_m]| \leq \gamma\mathbb{E}[\mathbb{E}[|j||s_m, a_m]C_1(s_m)C_2(s_m)\epsilon] + o(\epsilon) = \mathbb{E}[\delta C_1(s_m)C_2(s_m)\epsilon] + o(\epsilon),$$

which completes the proof for the first part of the lemma.

We now bound the difference of the variance. By the definition of $F_m, \hat{F}_m$ in (6), (8), we have

$$\begin{aligned}
&\left|\mathbb{V}[\hat{F}_m] - \mathbb{V}[F_m]\right| \\
=&\left|\mathbb{E}[j^2((\nabla_\theta\hat{j})^2 - (\nabla_\theta j')^2)] - \left(\mathbb{E}[j\nabla_\theta\hat{j}]^2 - \mathbb{E}[j\nabla_\theta j']^2\right)\right| \\
=&\left|\underbrace{\mathbb{E}[j^2(\nabla_\theta\hat{j} - \nabla_\theta j')(\nabla_\theta\hat{j} + \nabla_\theta j')]}_{I} - \underbrace{\mathbb{E}[j(\nabla_\theta\hat{j} - \nabla_\theta j')]\mathbb{E}[j(\nabla_\theta\hat{j} + \nabla_\theta j')]}_{II}\right|.
\end{aligned} \tag{A.8}$$

Using the same approximations of $\nabla_\theta\hat{j}$, $\nabla_\theta j'$ as in (A.3), (A.5) gives

$$\begin{aligned}
\nabla_\theta\hat{j} - \nabla_\theta j' =&(f_1 - \hat{f}_1)\epsilon + f_2(Z_{m+1} - Z'_m)\sqrt{\epsilon} + f_3(Z^2_{m+1} - Z'^2_m)\epsilon + o(\epsilon) \\
\nabla_\theta\hat{j} + \nabla_\theta j' =&2f_0 + (f_1 + \hat{f}_1)\epsilon + f_2(Z_{m+1} + Z'_m)\sqrt{\epsilon} + f_3(Z^2_{m+1} + Z'^2_m)\epsilon + o(\epsilon).
\end{aligned} \tag{A.9}$$

It follows that

$$\begin{aligned}
&(\nabla_\theta\hat{j})^2 - (\nabla_\theta j')^2 = (\nabla_\theta\hat{j} - \nabla_\theta j')(\nabla_\theta\hat{j} + \nabla_\theta j') \\
=&2f_0(f_1 - \hat{f}_1)\epsilon + 2f_0f_2(Z_{m+1} - Z'_m)\sqrt{\epsilon} + 2f_0f_3(Z^2_{m+1} - Z'^2_m)\epsilon + f_2^2(Z^2_{m+1} - Z'^2_m)\epsilon + o(\epsilon).
\end{aligned}$$

Again, using a Taylor expansion, we can approximate j by

$$\begin{aligned}
j =& \underbrace{r + \gamma\int Q^\pi\pi da - Q^\pi}_{g_0} + \underbrace{\left(\partial_s r + \gamma\int \partial_s(Q^\pi\pi)da\right)\mu\,\epsilon}_{g_1} \\
&+ \underbrace{\left(\partial_s r + \gamma\int \partial_s(Q^\pi\pi)da\right)\sigma}_{g_2} Z_m\sqrt{\epsilon} + \underbrace{\left(\partial_s^2 r + \gamma\int \partial_s^2(Q^\pi\pi)da\right)\sigma^2}_{g_3} Z_m^2\epsilon + o(\epsilon),
\end{aligned} \tag{A.10}$$

where we abbreviate $r(s_m, s_m, a_m), Q^\pi(s_m, a), \pi(a|s_m)$, and $\mu(s_m, a_m)$ by r, Q^π, π, and μ, respectively. It follows that

$$j^2 = g_0^2 + g_2^2 Z_m \epsilon + 2g_0(g_1\epsilon + g_2 Z_m\sqrt{\epsilon} + g_3 Z_m^2 \epsilon) + o(\epsilon).$$

Then,

$$\begin{aligned} I =& \mathbb{E}\left[j^2((\nabla_\theta \hat{j})^2 - (\nabla_\theta j')^2)\right] \\ =& \mathbb{E}[g_0^2(2f_0(f_1 - \hat{f}_1)\epsilon + 2f_0 f_2(Z_{m+1} - Z'_m)\sqrt{\epsilon} + 2f_0 f_3(Z_{m+1}^2 - Z_m'^2)\epsilon + f_2^2(Z_{m+1}^2 - Z_m'^2)\epsilon) \\ &+ 2g_0 g_2 Z_m\sqrt{\epsilon}(2f_0 f_2(Z_{m+1} - Z'_m)\sqrt{\epsilon})] + o(\epsilon). \end{aligned}$$

Similar to (A.7), since Z_{m+1}, Z'_m are independent of g_0, g_2, f_0, f_2, f_3, the above equation becomes,

$$I = 2\mathbb{E}[g_0^2 f_0(f_1 - \hat{f}_1)\epsilon] + o(\epsilon).$$

Futhermore, plugging (A.9) and (A.10) into II gives,

$$II = \mathbb{E}[2f_0 g_0]\mathbb{E}[g_0(\hat{f}_1 - f_1)\epsilon] + o(\epsilon).$$

Combining I and II, we see that

$$\left|\mathbb{V}[\hat{F}_m)] - \mathbb{V}[F_m]\right| = |I - II| \leq 2\text{Cov}(f_0 g_0, g_0(\hat{f}_1 - f_1))\epsilon + o(\epsilon) \leq O(\epsilon)$$

as long as the covariance of $f_0 g_0$ and $g_0(\hat{f}_1 - f_1)$ is bounded. But since $f_0, g_0, \hat{f}_1 - f_1$ are all bounded by the conditions in the second part of the lemma, $\text{Cov}(f_0 g_0, g_0(\hat{f}_1 - f_1))$ must be bounded as well. This concludes the proof. ■

Lemma A.2 (Extension of Lemma 2 for Q-control) *Suppose that $f(s;\theta) = \max_{a\in\mathbb{A}} Q^*(s, a; \theta)$ is Lipschitz continuous in $\theta \in \mathbb{R}^{d_\theta}$ and $\partial_s \nabla_\theta f(s;\theta)$ is continuous, then the difference between the gradients in US and BFF for Q-control is*

$$\mathbb{E}[\hat{F}_m] - \mathbb{E}[F_m] = \mathbb{E}\left[\delta(s_m, a_m)C(s_m;\theta)\epsilon\right] + o(\epsilon) = O(\mathbb{E}[\delta\epsilon]),$$

where

$$C(s_m;\theta) = \gamma|\partial_s \nabla_\theta f(s_m;\theta)|C_2(s_m),$$

where $C_2(s_m)$ is the uppper bound for the variantion of the drift in action space $|\mu(s_m, a_{m+1})] - \mu(s_m, a_m)| \leq C_2(s_m)$. In addition, if $|f(s;\theta) - Q^*(s, a;\theta)|, |\nabla_\theta f(s;\theta) - \nabla_\theta Q^*(s, a;\theta)|$, $\left|\mu(s, a) - \mu(s, a')\right|, |r(s, s, a)| \leq C$ almost surely over $s \in \mathbb{S}, a \in \mathbb{A}, \theta \in \mathbb{R}^{d_\theta}$, then

$$\left|\mathbb{V}[\hat{F}_m] - \mathbb{V}[F_m]\right| \leq O(\epsilon).$$

Here $F_m, \hat{F}_m$ are the same as in Lemma A.1 with j and δ replaced by

$$\begin{aligned} j(s_m, a_m, s_{m+1};\theta) &= r(s_{m+1}, s_m, a_m) + \gamma \max_a Q^*(s_{m+1}, a;\theta) - Q^*(s_m, a_m;\theta); \\ \delta &= \mathbb{E}\left[r(s_{m+1}, s_m, a_m) + \gamma \max_a Q^*(s_{m+1}, a;\theta) - Q^*(s_m, a_m;\theta)|s_m, a_m\right]. \end{aligned} \tag{A.11}$$

Proof

The difference between the two algorithms is

$$\mathbb{E}[\hat{F}_m] - \mathbb{E}[F_m] = \mathbb{E}\left[\mathbb{E}\left[j(\nabla_\theta j' - \nabla_\theta \hat{j})\middle| \, s_m, a_m\right]\right]$$

where

$$j = j(s_m, a_m, s_{m+1}; \theta), \quad \nabla_\theta j' = \nabla_\theta j(s_m, a_m, s'_{m+1}; \theta), \quad \nabla_\theta \hat{j} = \nabla_\theta j(s_m, a_m, s_m + \Delta s_{m+1}; \theta)$$

and

$$\nabla_\theta j(s_m, a_m, s_{m+1}; \theta) = \nabla_\theta \max_{a' \in \mathbb{A}} Q^*(s_{m+1}, a'; \theta) - \nabla_\theta Q^*(s_m, a_m; \theta).$$

Since we have assumed that $f(s;\theta) = \max_{a' \in \mathbb{A}} Q^*(s, a; \theta)$ is continuous in $s \in \mathbb{S}$ and that $\partial_s f(s;\theta), \partial_s^2 f(s;\theta)$ exist almost surely, we can write $\nabla_\theta j$ as

$$\nabla_\theta j(s_m, a_m, s_{m+1}; \theta) = \nabla_\theta f(s_{m+1}; \theta) - \nabla_\theta Q^*(s_m, a_m; \theta).$$

Similarly to the proof of Lemma A.1, we use a Taylor expansion:

$$\begin{aligned}
\nabla_\theta j' =& \underbrace{\gamma \nabla f(s_m) - \nabla Q^\pi(s_m, a_m)}_{f_0} + \underbrace{\gamma \partial_s \nabla f(s_m) \mu(s_m, a_m)}_{f_1} \epsilon \\
&+ \underbrace{\gamma \partial_s \nabla f(s_m) \sigma}_{f_2} Z'_m \sqrt{\epsilon} + \underbrace{\gamma \partial_s^2 \nabla f(s_m) \sigma^2}_{f_3} Z'^2_m \epsilon + o(\epsilon), \\
\nabla_\theta \hat{j} =& f_0 + \underbrace{\gamma \partial_s \nabla f(s_m) \mu(s_m, a_{m+1})}_{\hat{f}_1} \epsilon + f_2 Z_{m+1} \sqrt{\epsilon} + f_3 Z_{m+1}^2 \epsilon + o(\epsilon).
\end{aligned} \tag{A.12}$$

Using the expressions from (A.12) and the independence between Z'_m, Z_{m+1} and j, f_2, f_3, we see that

$$\begin{aligned}
\mathbb{E}[\hat{F}_m] - \mathbb{E}[F_m] =& \mathbb{E}\left[\mathbb{E}[j(f_1 - \hat{f}_1)\epsilon | s_m, a_m]\right] + o(\epsilon) \\
=& \gamma \mathbb{E}\left[\mathbb{E}[j \partial_s \nabla_\theta f(s_m) \left(\mu(s_m, a_{m+1}) - \mu(s_m, a_m)\right) \epsilon | s_m, a_m]\right] + o(\epsilon).
\end{aligned}$$

Let $C_1(s_m;\theta) = |\partial_s \nabla_\theta f(s_m)|$ and by the assumption that $|\mu(s_m, a_{m+1}) - \mu(s_m, a_m)| \leq C_2(s_m)$, one has

$$\mathbb{E}[\hat{F}_m] - \mathbb{E}[F_m] \leq \gamma \mathbb{E}\left[\mathbb{E}[|j| | s_m, a_m] C_1(s_m;\theta) C_2(s_m) \epsilon\right] + o(\epsilon) = \mathbb{E}[\delta C(s_m;\theta)\epsilon] + o(\epsilon),$$

for $C(s_m;\theta)$ defined in the Lemma, which completes the proof for the first part of the lemma.

We next bound the difference of the variance. We have

$$\mathbb{V}[\hat{F}_m] - \mathbb{V}[F_m] = \mathbb{E}[j^2(\nabla_\theta \hat{j} - \nabla_\theta j')(\nabla_\theta \hat{j} + \nabla_\theta j')] - \left(\mathbb{E}[j(\nabla_\theta \hat{j} - \nabla_\theta j')]\mathbb{E}[j(\nabla_\theta \hat{j} + \nabla_\theta j')]\right).$$

Substituting the Taylor expansions of $\nabla_\theta j', \nabla_\theta \hat{j}$ from (A.12), we obtain

$$\begin{aligned}
j =& \underbrace{r(s_m, s_m, a_m) + \gamma f(s_m) - Q^\pi(s_m, a_m)}_{g_0} + \underbrace{\left(\partial_s r(s_m) + \gamma \partial_s f(s_m)\right) \mu}_{g_1} \epsilon \\
&+ \underbrace{\left(\partial_s r(s_m) + \gamma \partial_s f(s_m)\right) \sigma}_{g_2} Z_m \sqrt{\epsilon} + \underbrace{\left(\partial_s^2 r(s_m) + \gamma \partial_s^2 f(s_m)\right) \sigma^2}_{g_3} Z_m^2 \epsilon + o(\epsilon).
\end{aligned} \tag{A.13}$$

Following steps similar to the proof of Lemma A.1, we arrive at

$$\mathbb{V}[\hat{F}_m] - \mathbb{V}[F_m] = O(\epsilon),$$

provided that $g_0, f_0, \hat{f}_1 - f_1$ are all bounded. The boundedness of these quantities is precisely the second set of assumptions in the lemma, so we are done.

■

B. Dynamics with diffusion depending on $\sigma(s_m, a_m)$

In this section, we will show that if the diffusion of the dynamics depends on $\sigma(s_m, a_m)$, i.e.,

$$s_{m+1} = s_m + \mu(s_m, a_m)\epsilon + \sigma(s_m, a_m)\sqrt{\epsilon}Z_m, \tag{B.1}$$

the theoretical results in Lemma A.1 and A.2 still hold as long as the diffusion term $\sigma(s_m, a_m)$ is bounded in the action space.

Lemma B.1 *With the same conditions as in Lemma A.1, the difference between the gradients of the US and BFF algorithms for Q-evaluation is*

$$\mathbb{E}[\hat{F}_m] - \mathbb{E}[F_m] = \mathbb{E}\left[\delta(s_m, a_m; \theta)C(s_m; \theta)\epsilon\right] + o(\epsilon) = O(\mathbb{E}[\delta\epsilon]),$$

where

$$C(s_m; \theta) = \gamma\left(\partial_s \mathbb{E}_{a\sim\pi(a|s_m)}[\nabla_\theta Q^\pi(s_m, a; \theta)]\right) C_2(s_m) + \gamma\left(\partial_s^2 \mathbb{E}_{a\sim\pi(a|s_m)}[\nabla_\theta Q^\pi(s_m, a; \theta)]\right) C_4(s_m),$$

and $C_2(s_m), C_4(s_m)$ are upper bounds for the variation of the drift term and diffusion term in the action space, i.e., $|\mu(s_m, a_{m+1}) - \mu(s_m, a_m)| \le C_2(s_m)$, $|\sigma^2(s_m, a_{m+1}) - \sigma^2(s_m, a_m)| \le C_4(s_m)$.

In addition, if $|\sigma(s, a) - \sigma^2(s, a')| \le C$ almost surely for $s \in \mathbb{S}, a, a' \in \mathbb{A}$, then the difference between the variances can also be bounded by

$$\left|\mathbb{V}[\hat{F}_m] - \mathbb{V}[F_m]\right| = O(\epsilon),$$

Lemma B.2 *With the same conditions as in Lemma A.2, the difference between the gradients in US and BFF for Q-control is*

$$\mathbb{E}[\hat{F}_m] - \mathbb{E}[F_m] = \mathbb{E}\left[\delta(s_m, a_m)C(s_m; \theta)\epsilon\right] + o(\epsilon) = O(\mathbb{E}[\delta\epsilon]),$$

where

$$C(s_m; \theta) = \gamma|\partial_s \nabla_\theta f(s_m; \theta)|C_2(s_m) + \gamma|\partial_s^2 \nabla_\theta f(s_m; \theta)|C_4(s_m),$$

and $C_2(s_m), C_4(s_m)$ are upper bounds for the variation of the drift and the diffusion in action space, i.e., $|\mu(s_m, a_{m+1})] - \mu(s_m, a_m)| \le C_2(s_m)$, $|\sigma(s_m, a_{m+1})] - \sigma(s_m, a_m)| \le C_4(s_m)$.

In addition, if $|\sigma(s, a) - \sigma(s, a')| \le C$ almost surely for $s \in \mathbb{S}, a, a' \in \mathbb{A}$, then

$$\left|\mathbb{V}[\hat{F}_m] - \mathbb{V}[F_m]\right| \le O(\epsilon).$$

We will provide the proof for Lemma B.1. We omit the proof of Lemma B.2 since they are similar.

Proof [Lemma B.1]

The dynamics (2) gives

$$\begin{aligned}\Delta s_m &= \mu(s_m, a_m)\epsilon + \sigma(s_m, a_m) Z_m \sqrt{\epsilon};\\ \Delta s_{m+1} &= \mu(s_m, a_{m+1})\epsilon + \sigma(s_m, a_{m+1}) Z_{m+1}\sqrt{\epsilon} + \partial_s \sigma(s_m, a_{m+1})\sigma(s_m, a_m) Z_m Z_{m+1}\epsilon + o(\epsilon).\end{aligned}$$

Replacing Δs_m in (A.2) and Δs_{m+1} in (A.4), $\nabla_\theta j'$ and $\nabla_\theta \hat{j}$ become

$$\begin{aligned}\nabla_\theta j' =& f_0 + f_1\epsilon + + \underbrace{\left(\gamma \int \partial_s(\nabla Q^\pi(s_m, a)\pi(a|, s_m))da\right)\sigma(s_m, a_m)}_{f_2'} Z_m'\sqrt{\epsilon}\\ &+ \underbrace{\left(\gamma \int \partial_s^2(\nabla Q^\pi(s_m, a)\pi(a|s_m))da\right)\sigma^2(s_m, a_m)}_{f_3'}(Z_m')^2\epsilon + o(\epsilon).\\ \nabla_\theta \hat{j} =& f_0 + \hat{f}_1\epsilon + \underbrace{\left(\gamma \int \partial_s(\nabla Q^\pi(s_m, a)\pi(a|, s_m))da\right)\sigma(s_m, a_{m+1})}_{\hat{f}_2} Z_{m+1}\sqrt{\epsilon}\\ &+ \underbrace{\gamma \int \partial_s^2(\nabla Q^\pi(s_m, a)\pi(a|s_m))da)\sigma^2(s_m, a_{m+1})}_{\hat{f}_3} Z_{m+1}^2\epsilon\\ &+ \underbrace{\left(\gamma \int \partial_s(\nabla Q^\pi(s_m, a)\pi(a|, s_m))da\right)\partial_s\sigma(s_m, a_{m+1})\sigma(s_m, a_m)}_{\hat{f}_4} Z_m' Z_{m+1}\epsilon + o(\epsilon).\end{aligned} \tag{B.2}$$

Similar to (A.6) - (A.7), we have

$$\begin{aligned}&\mathbb{E}[\hat{F}_m] = \mathbb{E}[\mathbb{E}[j(\nabla j' + (\nabla\hat{j} - \nabla j'))|s_m, a_m]]\\ =&\mathbb{E}[\mathbb{E}[j\nabla j'|s_m, a_m]] + \mathbb{E}\left[\mathbb{E}\left[j\left((\hat{f}_1 - f_1)\epsilon + (\hat{f}_2 Z_{m+1} - f_2' Z_m')\sqrt{\epsilon} + (\hat{f}_3 Z_{m+1}^2 - f_3'(Z_m')^2)\epsilon\right.\right.\right.\\ &\left.\left.\left.+\hat{f}_4 Z_m' Z_{m+1}\epsilon + o(\epsilon)\right)|s_m, a_m\right]\right]\\ =&\mathbb{E}[F_m] + \mathbb{E}\left[\mathbb{E}\left[j\left((\hat{f}_1 - f_1)\epsilon + (\hat{f}_3 - f_3')\epsilon + o(\epsilon)\right)|s_m, a_m\right]\right]\end{aligned}$$

where the last equality is because of the independence between $\hat{f}_2$ and Z_{m+1}, f_2' and Z_m', etc. Let $C_1(s_m) = \left|\int \partial_s(\nabla Q^\pi(s_m, a)\pi(a|s_m))da\right|$, $C_3(s_m) = \left|\int \partial_s^2(\nabla Q^\pi(s_m, a)\pi(a|s_m))da\right|$. By the assumption that $|\mu(s_m, a_{m+1}) - \mu(s_m, a_m)| \le C_2(s_m)$, $|\sigma^2(s_m, a_{m+1}) - \sigma^2(s_m, a_m)| \le C_4(s_m)$, one has

$$|\mathbb{E}[\hat{F}_m] - \mathbb{E}[F_m]| \le \gamma\mathbb{E}[\delta\left(C_1(s_m)C_2(s_m) + C_3(s_m)C_4(s_m)\right)\epsilon] + o(\epsilon),$$

which completes the proof for the first part of the lemma.

We now bound the difference of the variance. Plugging the new approximation (B.2) into (A.9) yields

$$\begin{aligned}\nabla_\theta \hat{j} - \nabla_\theta j' =& (\hat{f}_1 - f_1)\epsilon + (\hat{f}_2 Z_{m+1} - f_2' Z_m')\sqrt{\epsilon} + (\hat{f}_3 Z_{m+1}^2 - f_3' Z_m'^2)\epsilon + \hat{f}_4 Z_m' Z_{m+1}\epsilon + o(\epsilon)\\ \nabla_\theta \hat{j} + \nabla_\theta j' =& 2f_0 + (\hat{f}_1 + f_1)\epsilon + (\hat{f}_2 Z_{m+1} + f_2' Z_m')\sqrt{\epsilon} + (\hat{f}_3 Z_{m+1}^2 + f_3' Z_m'^2)\epsilon + \hat{f}_4 Z_m' Z_{m+1}\epsilon + o(\epsilon).\end{aligned}$$

It follows that

$$\begin{aligned}&(\nabla_\theta \hat{j})^2 - (\nabla_\theta j')^2 = (\nabla_\theta \hat{j} - \nabla_\theta j')(\nabla_\theta \hat{j} + \nabla_\theta j')\\ =&2f_0(\hat{f}_1 - f_1)\epsilon + 2f_0(\hat{f}_2 Z_{m+1} - f_2' Z_m')\sqrt{\epsilon} + 2f_0(\hat{f}_3 Z_{m+1}^2 - f_3' Z_m'^2)\epsilon + 2f_0 \hat{f}_4 Z_m' Z_{m+1}\epsilon\\ &+ (\hat{f}_2^2 Z_{m+1}^2 - (f_2')^2 (Z_m')^2)\epsilon + o(\epsilon).\end{aligned}$$

The approximation of j is similar to (A.10):

$$j = g_0 + g_1\epsilon + g_2' Z_m \epsilon + g_3' Z_m^2 \epsilon + o(\epsilon),$$

$$j^2 = g_0^2 + (g_2')^2 Z_m^2 \epsilon + 2g_0(g_1\epsilon + g_2' Z_m \sqrt{\epsilon} + g_3' Z_m^2 \epsilon) + o(\epsilon),$$

where g_2', g_3' are the same as g_2, g_3 except σ depends on s_m, a_m. Plugging these approximations into (A.8) yields

$$\begin{aligned}&I = \mathbb{E}\left[j^2((\nabla_\theta \hat{j})^2 - (\nabla_\theta j')^2)\right]\\ =&\mathbb{E}[g_0^2(2f_0(\hat{f}_1 - f_1)\epsilon + 2f_0(\hat{f}_2 Z_{m+1} - f_2' Z_m')\sqrt{\epsilon} + 2f_0(\hat{f}_3 Z_{m+1}^2 - f_3' Z_m'^2)\epsilon + 2f_0 \hat{f}_4 Z_m' Z_{m+1}\epsilon\\ &+ (\hat{f}_2^2 Z_{m+1}^2 - (f_2')^2 (Z_m')^2)\epsilon) + 2g_0 g_2' Z_m \sqrt{\epsilon}(2f_0(\hat{f}_2 Z_{m+1} - f_2' Z_m')\sqrt{\epsilon})] + o(\epsilon)\\ =&\mathbb{E}[g_0^2(2f_0(\hat{f}_1 - f_1)\epsilon + 2f_0(\hat{f}_3 - f_3')\epsilon + (\hat{f}_2^2 - (f_2')^2)\epsilon)] + o(\epsilon)\end{aligned}$$

and

$$II = \mathbb{E}[g_0((\hat{f}_1 - f_1)\epsilon + (\hat{f}_3 - f_3'))\epsilon]\mathbb{E}[2g_0 f_0] + o(\epsilon)$$

Combining I and II, we see that

$$\begin{aligned}&\left|\mathbb{V}[\hat{F}_m)] - \mathbb{V}[F_m]\right| = |I - II|\\ \leq& 2\text{Cov}(f_0 g_0, g_0(\hat{f}_1 - f_1) + g_0 \hat{f}_3 - f_3'))\epsilon + \mathbb{E}[g_0^2(\hat{f}_2^2 - (f_2')^2)]\epsilon + o(\epsilon) \leq O(\epsilon),\end{aligned}$$

which completes the proof for the second part of the lemma. ■

C. Extension and proof of Theorem 4

For the convenience of writing, we let $\beta = \frac{2}{\eta\xi}$ be the coefficient of the exponential power of p^∞, and we omit the index m of $F_m, \hat{F}_m$.

The proof of Theorem 4 (or equivalently, Theorem C.5) is based on Corollary C.2 and Lemma C.4. Note that although Theorem 4 can only be proved for compact set Ω, Corollary C.2 and Lemma

C.4 holds for both compact set and the whole domain $\theta \in \mathbb{R}^{d_\theta}$. However, for the case of the whole domain, one requires an additional assumption for the loss function $\mathbb{E}[\delta^2]$, that is,

$$\begin{aligned} &1) \quad \lim_{|\theta|\to\infty} \mathbb{E}[\delta^2] \to \infty \quad \text{and} \quad \int e^{-\mathbb{E}[\delta^2]} < \infty, \\ &2) \quad \lim_{|\theta|\to\infty} \left(\frac{|\nabla \mathbb{E}[\delta^2]|}{2} - \Delta \mathbb{E}[\delta^2] \right) = +\infty. \end{aligned} \tag{C.1}$$

The above two assumtions can be removed if we only consider $\theta \in \Omega \subset \mathbb{R}^{d_\theta}$ is in a compact set. These assumptions ensure that the probability measure p^∞ satisfies the Poincare inequality

$$\int f^2 p^\infty d\theta \le \lambda(\beta) \int (\nabla f)^2 p^\infty d\theta, \qquad \forall \int f d\theta = 0, \tag{C.2}$$

where $\lambda(\beta)$ is the Poincare constant depending on β. Typically $\lambda(\beta)$ becomes smaller as β becomes larger.

The following two lemmas hold for any function $\delta(\theta)^2$ on a compact domain $\theta \in \Omega \subset \mathbb{R}^{d_\theta}$, or on an unbounded domain $\theta \in \mathbb{R}^{d_\theta}$ if $\lim_{|\theta|\to\infty} \delta(\theta)^2 \to +\infty$.

Lemma C.1 *Let $f(\theta) = \mathbb{E}[\delta^2]$ and define $f_* = \min f(\theta)$. Suppose that f has only finitely many discrete minimizers, and that all of the minima are strict. Then there exists a constant C (depending on the Hessian $\nabla^2 f$ of f at each of the minimizers) such that for β large enough,*

$$\int f(\theta) e^{-\beta f(\theta)} d\theta \le C \left(f_* \beta^{-\frac{d_\theta}{2}} \right) + C \left(\beta^{-\frac{d_\theta+2}{2}} \right),$$

where d_θ is the dimension of θ.

Proof See Appendix C.1. ■

Corollary C.2 *Suppose that $f(\theta)$ has non-strict minima, i.e. minima at which the Hessian is not strictly positive definite. Define d_{θ_*} as*

$$d_{\theta_*} = \min\{\textit{number of positive eigenvalues of } \nabla^2_\theta f(\theta_*) : \theta_* = \arg\min f(\theta)\}. \tag{C.3}$$

Then there exists a constant C (depending on the Hessian $\nabla^2 f$ of f at each of the minimizers) such that

$$\int f(\theta) e^{-\beta f(\theta)} d\theta \le C \left(f_* \beta^{-\frac{d_{\theta_*}}{2}} \right) + C \left(\beta^{-\frac{d_{\theta_*}+2}{2}} \right). \tag{C.4}$$

Proof The proof of the corollary is similar to the proof of Lemma C.1, so we omit it here. ■

Remark C.3 *Note that the bound (C.4) depends on d_{θ_*},* not *the parameter dimension d_θ. When the dimension of the parameter space is high (i.e. when d_θ is large), it is more likely that there are many minima which are flat in some direction (i.e. $\nabla^2_\theta f(\theta_*)$ is a positive* semi-*definite matrix at these minima). In the above, d_{θ_*} denotes the smallest number of positive eigenvalues of $\nabla^2_\theta f(\theta_*)$ among all minima. As a result, when the dimension d_θ becomes larger, the upper bound $\beta^{-\frac{d_{\theta_*}}{2}}$ does not necessarily become smaller.*

Furthermore, if $f_* = 0$ (i.e. there exists θ_* such that $Q^\pi(s,a;\theta_*)$ exactly satisfies the Bellman equation) then

$$\int f(\theta)e^{-\beta f(\theta)}d\theta \le C\left(\beta^{-\frac{d_{\theta_*}+2}{2}}\right).$$

Lemma C.4 *The solution to (16) (i.e. the approximate p.d.f. of US)*

$$\int \mathbb{E}[\delta^2]\frac{(p(t,\theta)-p^\infty)^2}{p^\infty}d\theta \le C_0 e^{-b(\beta)t},$$

where C_0 is a constant depending on the initial data $(p(0,\theta)-p^\infty)$, $b(\beta) = \frac{2\lambda(\beta)^2}{C+\lambda(\beta)}$ with Poincare constant $\lambda(\beta)$ and $C = \sup_{a,s}|\nabla\delta(a,s)|^2$. In addition,

$$\int \mathbb{E}[\delta^2]\frac{p^2(t,\theta)}{p^\infty}d\theta \le C_0 e^{-b(\beta)t} + O\left(\mathbb{E}[\delta_*^2]\beta^{-\frac{d_{\theta_*}}{2}}\right),$$

where $\lambda(\beta)$ is the Poincare constant defined in (C.2), $\mathbb{E}[\delta_*^2] = \min_\theta \mathbb{E}[\delta^2]$, and d_{θ_*} is defined in (C.3) for $f = \mathbb{E}[\delta^2]$.

Proof See Appendix C.2. ■

Based on Corollary C.2 and Lemma C.4, we are now ready to prove the following theorem.

Theorem C.5 *The difference $\hat{d}$ between the p.d.f.s of the US and BFF algorithms is bounded by*

$$\left\|\hat{d}(t)\right\|_* \le \|p(0)-p^\infty\|_* e^{-\frac{\lambda(\beta)}{4}t} + O(\epsilon)e^{-\frac{b(\beta)}{2}t} + O\left(\epsilon\sqrt{\mathbb{E}[\delta_*^2]}\beta^{-\frac{d_{\theta_*}}{4}}\right)\sqrt{1-e^{-\frac{\lambda(\beta)}{2}t}}, \tag{C.5}$$

where $\lambda(\beta)$ is the Poincare constant defined in (C.2), $b(\beta)$ is the same constant as in Lemma C.4, $\mathbb{E}[\delta_*^2] = \min_\theta \mathbb{E}[\delta^2]$, and d_{θ_*} is defined in (C.3) for $f = \mathbb{E}[\delta^2]$.

Proof By subtracting (17) from (16), the difference of the p.d.f. $\hat{d} = p - \hat{p}$ satisfies

$$\partial_t \hat{d} = \nabla\cdot\left[\mathbb{E}[F]\hat{d} + \frac{\eta}{2}\nabla\cdot\left(\mathbb{V}[F]\hat{d}\right)\right] + \nabla\cdot\left[\left(\mathbb{E}[F]-\mathbb{E}[\hat{F}]\right)\hat{p} + \frac{\eta}{2}\nabla\cdot\left(\left(\mathbb{V}[F]-\mathbb{V}[\hat{F}]\right)\hat{p}\right)\right]. \tag{C.6}$$

Observe that

$$\begin{aligned}\mathbb{E}[F]p - \mathbb{E}[\hat{F}]\hat{p} &= \mathbb{E}[F]\hat{d} + (\mathbb{E}[\hat{F}]-\mathbb{E}[F])\hat{d} + (\mathbb{E}[F]-\mathbb{E}[\hat{F}])p \\ &= \mathbb{E}[F]\hat{d} + E[\delta O(\epsilon)]\hat{d} + E[\delta O(\epsilon)]p.\end{aligned}$$

Similarly, we have

$$\mathbb{V}[F]p - \mathbb{V}[\hat{F}]\hat{p} = \mathbb{V}[F]\hat{d} + O(\epsilon)\hat{d} + O(\epsilon)p,$$

Multiplying (C.6) by $\frac{\hat{d}}{p^\infty}$, then integrating with respect to θ, we have

$$\frac{1}{2}\partial_t \left\|\hat{d}\right\|_*^2 = \underbrace{-\int \left[p^\infty \nabla\left(\frac{\hat{d}}{p^\infty}\right)\right]\cdot\nabla\left(\frac{\hat{d}}{p^\infty}\right)d\theta}_{I} \underbrace{-\int\left(\mathbb{E}[\delta O(\epsilon)]\hat{d} + O(\eta\epsilon)\left|\nabla\hat{d}\right|\right)\cdot\nabla\left(\frac{\hat{d}}{p^\infty}\right)d\theta}_{II}$$
$$\underbrace{-\int\left(\mathbb{E}[\delta O(\epsilon)]p + O(\eta\epsilon)\left|\nabla p\right|\right)\cdot\nabla\left(\frac{\hat{d}}{p^\infty}\right)d\theta}_{III}.$$

We proceed by bounding the terms $I - III$ separately. First, note that

$$I = -\int\left[\nabla\left(\frac{\hat{d}}{p^\infty}\right)\right]^2 p^\infty d\theta \le -\frac{1}{2}\int\left[\nabla\left(\frac{\hat{d}}{p^\infty}\right)\right]^2 p^\infty d\theta - \frac{\lambda}{2}\left\|\hat{d}\right\|_*^2,$$

where we have used the Poincare inequality (C.2). For the second term, we have

$$\begin{aligned}
II \le& O(\epsilon)\int\left|\hat{d}\right|\left|\nabla\left(\frac{\hat{d}}{p^\infty}\right)\right|d\theta + O(\epsilon\eta)\int\left|\nabla\left(\frac{\hat{d}}{p^\infty}p^\infty\right)\right|\left|\nabla\left(\frac{\hat{d}}{p^\infty}\right)\right|d\theta \quad \text{(boundedness of}\mathbb{E}\delta) \\
\le& O(\epsilon)\left\|\hat{d}\right\|_*^2 + \frac{1}{8}\int\left[\nabla\left(\frac{\hat{d}}{p^\infty}\right)\right]^2 p^\infty d\theta + O(\epsilon\eta)\int\left|\nabla\left(\frac{\hat{d}}{p^\infty}\right)\right|^2 p^\infty d\theta \\
&+ O(\epsilon\eta)\int\left|\beta\mathbb{E}[\delta\nabla\delta]\hat{d}\right|\left|\nabla\left(\frac{\hat{d}}{p^\infty}\right)\right|d\theta \qquad \text{(Cauchy-Schwartz Inequality)} \\
\le& O(\epsilon)\left\|\hat{d}\right\|_*^2 + \frac{1}{4}\int\left[\nabla\left(\frac{\hat{d}}{p^\infty}\right)\right]^2 p^\infty d\theta + O(\epsilon\eta)\int\left|\nabla\left(\frac{\hat{d}}{p^\infty}\right)\right|^2 p^\infty d\theta + O(\epsilon^2\eta^2\beta^2)\left\|\hat{d}\right\|_*^2.
\end{aligned}$$

Since $\eta^2\beta^2 = O(1)$, $O(\epsilon^2\eta^2\beta^2) = O(\epsilon^2)$. This yields

$$II \le O(\epsilon)\left\|\hat{d}\right\|_*^2 + \left(\frac{1}{4} + O(\epsilon\eta)\right)\int\left[\nabla\left(\frac{\hat{d}}{p^\infty}\right)\right]^2 p^\infty d\theta.$$

For the last term, by the Cauchy-Schwarz inequality,

$$\begin{aligned}
III \le& O(\epsilon^2)\int\mathbb{E}[\delta^2]\frac{p^2}{p^\infty}d\theta + \frac{1}{16}\int\left|\nabla\left(\frac{\hat{d}}{p^\infty}\right)\right|^2 p^\infty d\theta + O(\epsilon\eta)\int\left|\nabla\left(\frac{p}{p^\infty}p^\infty\right)\right|\left|\nabla\left(\frac{\hat{d}}{p^\infty}\right)\right|d\theta \\
\le& O(\epsilon^2)\int\mathbb{E}[\delta^2]\frac{p^2}{p^\infty}d\theta + O(\epsilon^2\eta^2)\int\left|\nabla\left(\frac{p}{p^\infty}\right)\right|^2 p^\infty d\theta + \frac{2}{16}\int\left[\nabla\left(\frac{\hat{d}}{p^\infty}\right)\right]^2 p^\infty d\theta \\
&+ O(\epsilon^2\eta^2\beta^2)\int\mathbb{E}[\delta^2|\nabla\delta|^2]\frac{p^2}{p^\infty}d\theta + \frac{1}{16}\int\left|\nabla\left(\frac{\hat{d}}{p^\infty}\right)\right|^2 p^\infty d\theta \\
\le& O(\epsilon^2)\int\mathbb{E}[\delta^2]\frac{p^2}{p^\infty}d\theta + O(\epsilon^2\eta^2)\int\left|\nabla\left(\frac{p}{p^\infty}\right)\right|^2 p^\infty d\theta + \frac{3}{16}\int\left[\nabla\left(\frac{\hat{d}}{p^\infty}\right)\right]^2 p^\infty d\theta.
\end{aligned}$$

Combining the above three terms, we have

$$\frac{1}{2}\partial_t \left\|\hat{d}\right\|_*^2 \leq \left(\frac{1}{16} - O(\epsilon\eta)\right) \int \left[\nabla\left(\frac{\hat{d}}{p^\infty}\right)\right]^2 p^\infty d\theta - \left(\frac{\lambda}{2} - O(\epsilon)\right) \left\|\hat{d}\right\|_*^2 + O(\epsilon^2) \int \mathbb{E}[\delta^2] \frac{p^2}{p^\infty} d\theta + O(\epsilon^2\eta^2) \int \left|\nabla\left(\frac{p}{p^\infty}\right)\right|^2 p^\infty d\theta.$$

As long as ϵ, η are small enough, and using the fact that $\nabla\left(\frac{p^\infty}{p^\infty}\right) = 0$, we have

$$\frac{1}{2}\partial_t \left\|\hat{d}\right\|_*^2 \leq -\frac{\lambda}{4}\left\|\hat{d}\right\|_*^2 + O(\epsilon^2) \int \mathbb{E}[\delta^2] \frac{p^2}{p^\infty} d\theta + O(\epsilon^2\eta^2) \int \left|\nabla\left(\frac{p - p^\infty}{p^\infty}\right)\right|^2 p^\infty d\theta. \tag{C.7}$$

Setting $d = p - p^\infty$, it is easy to see that d also satisfies (16). Multiplying (16) by $\frac{p}{p^\infty}$ and integrating with respect to θ, we have

$$\frac{1}{2}\partial_t \|d\|_*^2 = -\int \left|\nabla\left(\frac{d}{p^\infty}\right)\right|^2 p^\infty d\theta \leq -\frac{1}{2}\int \left|\nabla\left(\frac{d}{p^\infty}\right)\right|^2 p^\infty d\theta - \frac{\lambda}{2}\|d\|_*^2. \tag{C.8}$$

Adding equations (C.7) and (C.8) gives

$$\frac{1}{2}\partial_t \left(\left\|\hat{d}\right\|_*^2 + \|d\|_*^2\right) \leq -\frac{\lambda}{4}\left(\left\|\hat{d}\right\|_*^2 + \|d\|_*^2\right) - \frac{\lambda}{4}\|d\|_*^2 + O(\epsilon^2) \int \mathbb{E}[\delta^2] \frac{p^2}{p^\infty} d\theta$$

$$\partial_t \left[e^{\frac{\lambda}{2}t}\left(\left\|\hat{d}\right\|_*^2 + \|d\|_*^2\right)\right] \leq e^{\frac{\lambda}{2}t} O(\epsilon^2) \int \mathbb{E}[\delta^2] \frac{p^2}{p^\infty} d\theta.$$

Applying Lemma C.4, we have

$$\partial_t \left[e^{\frac{\lambda}{2}t}\left(\left\|\hat{d}\right\|_*^2 + \|d\|_*^2\right)\right] \leq O(\epsilon^2) e^{\frac{\lambda}{2}t} e^{-bt} + O\left(\epsilon^2 \mathbb{E}[\delta_*^2] \beta^{-\frac{d_{\theta_*}}{2}}\right) e^{\frac{\lambda}{2}t}.$$

We then integrate the above inequality on both sides to obtain

$$\left(\left\|\hat{d}(t)\right\|_*^2 + \|d(t)\|_*^2\right) \leq e^{-\frac{\lambda}{2}t}\left(\left\|\hat{d}(0)\right\|_*^2 + \|d(0)\|_*^2\right) + O(\epsilon^2)(e^{-bt} + e^{-\frac{\lambda}{2}t}) + O\left(\epsilon^2 \mathbb{E}[\delta_*^2] \beta^{-\frac{d_{\theta_*}}{2}}\right)(1 - e^{-\frac{\lambda}{2}t}).$$

Since $\hat{d}(0) = 0$, the above inequality is equivalent to

$$\left\|\hat{d}(t)\right\|_*^2 \leq e^{-\frac{\lambda}{2}t}\|d(0)\|_*^2 + O(\epsilon^2) e^{-bt} + O\left(\epsilon^2 \mathbb{E}[\delta_*^2] \beta^{-\frac{d_{\theta_*}}{2}}\right)(1 - e^{-\frac{\lambda}{2}t})$$

as desired. ■

C.1. PROOF OF LEMMA C.1

Proof For the unbounded domain, since $\lim_{|\theta|\to\infty} f(\theta) = +\infty$ and $\lim_{f\to+\infty} fe^{-\beta f} = 0$, there always exists a compact domain $\Omega = \{|\theta| \le M\}$ such that

$$\int_{\mathbb{R}^{d_\theta}\setminus\Omega} f(\theta)e^{-\beta f(\theta)}d\theta \le O\left(f(\theta_*)\beta^{-\frac{d_\theta}{2}}\right) + O\left(\beta^{-\frac{d_\theta+2}{2}}\right).$$

We can divide Ω into $\{\Omega_i\}_{i=1}^k$ such there is only one minimizer θ_* in each Ω_i, or else $f(\Omega_i) \equiv 0$. For this latter case, it is trivial to see that $\int_{\Omega_i} f(\theta)e^{-\beta f(\theta)} = 0$.

For the former case, notice that the integral can be separated into two parts,

$$\int_{\Omega_1} f(\theta)e^{-\beta f(\theta)}d\theta = \int_{|\theta-\theta_*|\le\varepsilon} f(\theta)e^{-\beta f(\theta)}d\theta + \int_{\Omega_1\setminus|\theta-\theta_*|\le\varepsilon} f(\theta)e^{-\beta f(\theta)}d\theta.$$

For any $\varepsilon > 0$, we can choose β large enough that the second integral will be smaller than $O(\beta^{-\frac{d_\theta+2}{2}})$. Since θ_* is a minimizer, $\nabla f(\theta_*) = 0$ and we have

$$\begin{aligned}
&\int_{\Omega_1} f(\theta)e^{-\beta f(\theta)}d\theta \\
=&\int_{|\theta-\theta_*|\le\varepsilon} \left(f(\theta_*) + (\theta-\theta_*)^\top\nabla^2 f(\theta_*)(\theta-\theta_*) + O(|\theta-\theta_*|^3)\right) \\
&\exp\left(-\beta f(\theta_*) - \beta(\theta-\theta_*)^\top\nabla^2 f(\theta_*)(\theta-\theta_*) - \beta O(|\theta-\theta_*|^3)\right)d\theta + O(\beta^{-\frac{d_\theta+1}{2}}) \\
=&f(\theta_*)\exp(-\beta f(\theta_*))\int_{|\theta-\theta_*|\le\varepsilon} \exp\left(-\beta(\theta-\theta_*)^\top\nabla^2 f(\theta_*)(\theta-\theta_*)\right)d\theta \\
&+ \exp(-\beta f(\theta_*))\int_{|\theta-\theta_*|\le\varepsilon} (\theta-\theta_*)^\top\nabla^2 f(\theta_*)(\theta-\theta_*)\exp\left(-\beta(\theta-\theta_*)^\top\nabla^2 f(\theta_*)(\theta-\theta_*)\right)d\theta \\
&+ (\text{higher order terms in } \beta).
\end{aligned} \tag{C.9}$$

We will prove later the higher order terms are all smaller than $O(\beta^{-\frac{d_\theta+1}{2}})$. Without loss of generality, we assume $\nabla^2 f(\theta_*)$ is a diagonal matrix. (If it is not, we can simply perform a change of basis.) Since θ_* is a local minimum, $\partial^2_{\theta_i} f(\theta_*) > 0$. Then after making the change of variables $\tilde{\theta} = \theta - \theta_*$, we have

$$\begin{aligned}
&\int_{\Omega_1} f(\theta)e^{-\beta f(\theta)}d\theta \\
=&f(\theta_*)\exp(-\beta f(\theta_*))\int_{\Omega_1}\prod_i \exp\left(-\beta\partial^2_{\theta_i} f(\theta_*)\tilde{\theta}_i^2\right)d\tilde{\theta}_1\cdots d\tilde{\theta}_{d_\theta} \\
&+ \exp(-\beta f(\theta_*))\int_{\Omega_1}\left(\sum_i \partial^2_{\theta_i} f(\theta_*)\tilde{\theta}_i^2\right)\prod_i \exp\left(-\beta\partial^2_{\theta_i} f(\theta_*)\tilde{\theta}_i^2\right)d\tilde{\theta}_1\cdots d\tilde{\theta}_{d_\theta} + O(\beta^{-\frac{d_\theta+1}{2}}).
\end{aligned} \tag{C.10}$$

Since

$$\int_{\mathbb{R}} \exp\left(-\beta\partial^2_{\theta_i} f(\theta_*)\tilde{\theta}_i^2\right)d\theta_i = \sqrt{2\pi}(\beta\partial^2_{\theta_i} f(\theta_*))^{-1/2},$$
$$\int_{\mathbb{R}} \tilde{\theta}_i^2\exp\left(-\beta\partial^2_{\theta_i} f(\theta_*)\tilde{\theta}_i^2\right)d\theta_i = \sqrt{2\pi}(\beta\partial^2_{\theta_i} f(\theta_*))^{-3/2},$$

we have,

$$\begin{aligned}
\int_{\mathbb{R}^{d_\theta}} \prod_i \exp\left(-\beta \partial^2_{\theta_i} f(\theta_*) \tilde{\theta}_i^2\right) d\tilde{\theta}_1 \cdots d\tilde{\theta}_{d_\theta} &= (2\pi)^{d_\theta/2} \beta^{-d_\theta/2} \prod_i \left(\partial^2_{\theta_i} f(\theta_*)\right)^{-1/2} \\
&= O\left(\beta^{-\frac{d_\theta}{2}}\right); \\
\int_{\mathbb{R}^{d_\theta}} \theta_i^2 \prod_i \exp\left(-\beta \partial^2_{\theta_i} f(\theta_*) \tilde{\theta}_i^2\right) d\tilde{\theta}_1 \cdots d\tilde{\theta}_{d_\theta} &= (2\pi)^{d_\theta/2} \beta^{-d_\theta/2-1} (\partial^2_{\theta_i} f(\theta_*))^{-1} \prod_j (\partial^2_{\theta_j} f(\theta_*))^{-1/2} \\
&= O\left(\beta^{-\frac{d_\theta+2}{2}}\right).
\end{aligned}$$

Plugging the above estimate back to (C.10) and recalling that θ_* is the only minimizer in Ω_1, we have

$$\int_{\Omega_1} f(\theta) e^{-\beta f(\theta)} d\theta \leq O\left(f(\theta_*)\beta^{-\frac{d_\theta}{2}}\right) + O\left(\beta^{-\frac{d_\theta+2}{2}}\right). \tag{C.11}$$

Now we will estimate the higher order terms in (C.9),

$$\begin{aligned}
&\text{(higher order terms in } \beta) \\
=& f(\theta_*)\exp(-\beta f(\theta_*)) \int_{|\tilde{\theta}|\leq\varepsilon} \underbrace{\exp\left(-\beta \tilde{\theta}^\top \nabla^2 f(\theta_*) \tilde{\theta}\right)}_{\leq 1} \underbrace{\left(e^{-\beta O(|\tilde{\theta}|^3)} - 1\right)}_{\leq e^{\beta\varepsilon^3}-1} d\theta \\
&+ \exp(-\beta f(\theta_*)) \int_{|\tilde{\theta}|\leq\varepsilon} \underbrace{\tilde{\theta}^\top \nabla^2 f(\theta_*) \tilde{\theta} \exp\left(-\beta \tilde{\theta}^\top \nabla^2 f(\theta_*) \tilde{\theta}\right)}_{\leq \varepsilon^2 \max_i \{\partial^2_{\theta_i} f(\theta_*)\}} \left(e^{-\beta O(|\tilde{\theta}|^3)} - 1\right) d\theta \\
&+ \exp(-\beta f(\theta_*)) \int_{|\tilde{\theta}|\leq\varepsilon} \underbrace{O(|\tilde{\theta}|^3)}_{\leq\varepsilon^3} \underbrace{\exp\left(-\beta \tilde{\theta}^\top \nabla^2 f(\theta_*) \tilde{\theta} - \beta O(|\tilde{\theta}|^3)\right)}_{\leq 1} d\theta \\
\leq& O\left(\mathrm{vol}\left(\{|\theta| \leq \varepsilon\}\right) \left((e^{\beta\varepsilon^3-1}) + \varepsilon^3\right)\right).
\end{aligned}$$

From the above estimates, we see that as long as ε is small enough, the higher order terms are smaller than $O\left(\beta^{-\frac{d_{\theta_*}+2}{2}}\right)$. Since we assumed that the number of discrete minimizers is finite, this completes the proof.

■

C.2. PROOF OF LEMMA C.4

Proof Setting $d = p - p^\infty$, it is easy to see that d also satisfies (16). Multiplying (16) by $\mathbb{E}[\delta^2]\frac{d}{p^\infty}$ and then integrating with respect to θ, we have

$$\begin{aligned}
\frac{1}{2}\partial_t \int \mathbb{E}[\delta^2]\frac{d^2}{p^\infty}d\theta \leq & -\int p^\infty \nabla\left(\frac{d}{p^\infty}\right)\nabla\left(\mathbb{E}[\delta^2]\frac{d}{p^\infty}\right) \\
= & -\int\int p^\infty\left[\nabla\left(\frac{d}{p^\infty}\right)\nabla\left(\delta^2\frac{d}{p^\infty}\right)\right]d\theta d\mu(s,a) \\
= & -\int\int p^\infty\left[\left(\nabla\left(\frac{\delta d}{p^\infty}\right)\right)^2 - \left(\frac{d}{p^\infty}\right)^2(\nabla\delta)^2\right]d\theta d\mu(s,a) \\
\leq & -\int\int p^\infty\left(\nabla\left(\frac{\delta d}{p^\infty}\right)\right)^2 d\theta d\mu(s,a) + C\int\int\left(\frac{d}{p^\infty}\right)^2 p^\infty d\theta d\mu(s,a) \\
\leq & -\lambda\int\int\frac{(\delta d)^2}{p^\infty}d\theta d\mu(s,a) + C\int\frac{d^2}{p^\infty}d\theta d\mu(s,a) \\
\leq & -\lambda\int\mathbb{E}[\delta^2]\frac{d^2}{p^\infty}d\theta + C\left\|d\right\|_*^2.
\end{aligned}$$

Using the fact that $\frac{1}{2}\partial_t\left\|d\right\|_*^2 \leq -\lambda\left\|d\right\|_*^2$, we have

$$\begin{aligned}
\frac{1}{2}\partial_t\left[\int\mathbb{E}[\delta^2]\frac{d^2}{p^\infty}d\theta + \left(\frac{C}{\lambda}+1\right)\left\|d\right\|_*^2\right] \leq & -\lambda\left(\int\mathbb{E}[\delta^2]\frac{d^2}{p^\infty}d\theta + \left\|d\right\|_*^2\right) \\
\leq & -\frac{\lambda^2}{C+\lambda}\left[\int\mathbb{E}[\delta^2]\frac{d^2}{p^\infty}d\theta + \left(\frac{C}{\lambda}+1\right)\left\|d\right\|_*^2\right].
\end{aligned}$$

By Grownwall's inequality,

$$\begin{aligned}
\left[\int\mathbb{E}[\delta^2]\frac{d(t)^2}{p^\infty}d\theta + \left(\frac{C}{\lambda}+1\right)\left\|d(t)\right\|_*^2\right] \leq & e^{-\frac{2\lambda^2}{C+\lambda}t}\left[\int\mathbb{E}[\delta^2]\frac{d(0)^2}{p^\infty}d\theta + \left(\frac{C}{\lambda}+1\right)\left\|d(0)\right\|_*^2\right], \\
\int\mathbb{E}[\delta^2]\frac{d(t)^2}{p^\infty}d\theta \leq & e^{-\frac{2\lambda^2}{C+\lambda}t}\left[\int\mathbb{E}[\delta^2]\frac{d(0)^2}{p^\infty}d\theta + \left(\frac{C}{\lambda}+1\right)\left\|d(0)\right\|_*^2\right].
\end{aligned}$$

which completes the first part of the proof.

While the second part of the Lemma is obtained by inserting $p^2 = (d+p^\infty)^2 \leq 2d^2 + 2(p^\infty)^2$ into the following equation,

$$\int\mathbb{E}[\delta^2]\frac{p(t)^2}{p^\infty}d\theta \leq 2\int\mathbb{E}[\delta^2]\frac{d(t)^2}{p^\infty}d\theta + 2\int\mathbb{E}[\delta^2]p^\infty d\theta = C_0 e^{-\frac{2\lambda^2}{C+\lambda}t} + O(\beta^{-\frac{d_\theta+2}{2}}),$$

where Lemma C.1 is applied to the last equality.

■

D. Difference between SC and US

The SC parameter update is given by

$$\theta_{m+1} = \theta_m - \eta\tilde{F}_m, \quad \tilde{F}_m = j(s_m, a_m, s_{m+1}; \theta_m)\nabla_\theta j(s_m, a_m, s_{m+1}; \theta_m).$$

The definition of j depends on whether we are doing Q-evaluation or Q-control:

$$\begin{aligned} Q\text{-evaluation:} \quad j(s_m, a_m, s_{m+1}; \theta_m) =& r(s_{m+1}, s_m, a_m) + \gamma \int Q^\pi(s_{m+1}, a; \theta)\pi(a|s_{m+1})da \\ & - Q^\pi(s_m, a_m; \theta); \\ Q\text{-control:} \quad j(s_m, a_m, s_{m+1}; \theta_m) =& r(s_{m+1}, s_m, a_m) + \gamma \max_a Q^\pi(s_{m+1}, a; \theta) - Q^\pi(s_m, a_m; \theta). \end{aligned}$$

The expectation of the SC gradient $\tilde{F}_m$ at each step is

$$\mathbb{E}[\tilde{F}_m] = \mathbb{E}\left[\mathbb{E}\left[j\nabla_\theta j | s_m = s, a_m = a\right]\right], \tag{D.1}$$

which is the gradient of the following loss function

$$\tilde{J}(\theta) = \frac{1}{2}\mathbb{E}\left[\mathbb{E}\left[j^2 \middle| s_m = s, a_m = a\right]\right]. \tag{D.2}$$

Note that this is not the same as the desired objective function $J(\theta) = \frac{1}{2}\mathbb{E}[(\mathbb{E}[j|s_m, a_m])^2]$.

D.1. DIFFERENCE AT EACH STEP

In Lemmas D.1 and D.2, we prove that the difference between the gradients used in US and SC is $O(\epsilon)$. The constants hidden by the big-O depend on the square of the diffusion σ^2. In practice, this means that SC will not converge to a good approximation for Q^π.

Lemma D.1 *Suppose that $Q^\pi(s, a; \theta)$ is Lipschitz continuous in the $\theta \in \mathbb{R}^{d_\theta}$ and $\partial_s Q^\pi(s, a; \theta)$, $\partial_s \nabla_\theta Q^\pi(s, a; \theta)$, $\partial_{s'} r(s, s, a)$ are continuous in the state space, then the difference between the gradients of the US and BFF algorithms for Q-evaluation is*

$$\mathbb{E}[\tilde{F}_m] - \mathbb{E}[F_m] = \mathbb{E}\left[C(s_m, a_m)\epsilon + o(\epsilon)\right] = O(\epsilon),$$

where

$$C(s_m, a_m) = \gamma\sigma^2 \partial_s \mathbb{E}_{a\sim\pi(a|s_m)}[\nabla_\theta Q^\pi(s_m, a)]\left(\partial_{s'} r(s_m, s_m, a_m) + \gamma\partial_s \mathbb{E}_{a\sim\pi(a|s_m)}[Q(s_m, a)]\right) = O(\epsilon), \tag{D.3}$$

In addition, if $|\mathbb{E}_a[Q^\pi(s, a; \theta)] - Q(s, a; \theta)|, |\mathbb{E}_a[\nabla_\theta Q^\pi(s, a; \theta)] - \nabla_\theta Q(s, a; \theta)|, |r(s, s, a)| \le C$ almost surely in $\forall s \in \mathbb{S}, a \in \mathbb{A}, \theta \in \mathbb{R}^{d_\theta}$, then the difference between the variances is bounded by

$$\left|\mathbb{V}[\tilde{F}_m] - \mathbb{V}[F_m]\right| \le O(\epsilon),$$

where $\tilde{F}, F, \sigma$ are defined in (D.1) (A.1) and (2) respectively.

Proof The proof is similar to the proof of Lemma A.1. Subtracting (A.1) from (D.1) yields

$$\mathbb{E}\left[\tilde{F}_m - F_m\right] = \mathbb{E}\left[\mathbb{E}[j\left(\nabla j - \mathbb{E}[\nabla j|s_m, a_m]\right)|s_m, a_m]\right]. \tag{D.4}$$

By the approximation of ∇j in (A.3), we have

$$\nabla j - \mathbb{E}[\nabla j|s_m, a_m] = f_2 Z_m\sqrt{\epsilon} + f_3(Z_m^2 - 1)\epsilon + o(\epsilon).$$

Combining this with the approximation of j in (A.10) gives

$$
\begin{aligned}
&\mathbb{E}\left[\tilde{F}_m - F_m\right] = \mathbb{E}[g_2 f_2 \epsilon] + o(\epsilon) \\
=&\gamma \mathbb{E}\left[\partial_s r \int \partial_s(\nabla Q^\pi \pi) da\right] \sigma^2 \epsilon + \gamma^2 \mathbb{E}\left[\int \partial_s(Q^\pi \pi) da \int \partial_s(\nabla Q^\pi \pi) da\right] \sigma^2 \epsilon + o(\epsilon) \\
=&\gamma \sigma^2 \epsilon \mathbb{E}\left[\partial_s \mathbb{E}_{a\sim\pi(a|s_m)}[\nabla_\theta Q^\pi(s_m, a)]\left(\partial_{s'} r(s_m, s_m, a_m) + \gamma \partial_s \mathbb{E}_{a\sim\pi(a|s_m)}[Q(s_m, a)]\right)\right] + o(\epsilon) \\
=&O(\epsilon),
\end{aligned}
$$

which completes the proof for the first part of the lemma. Next, we bound the difference of the variance. We have

$$
\begin{aligned}
&\left|\mathbb{V}[\tilde{F}_m] - \mathbb{V}[F_m]\right| = \mathbb{E}[j^2((\nabla j)^2 - (\nabla j')^2)] - \left(\mathbb{E}[j\nabla j]^2 - \mathbb{E}[j\nabla j']^2\right) \\
=&\underbrace{\mathbb{E}[\mathbb{E}[j^2\left((\nabla j)^2 - \mathbb{E}[(\nabla j)^2|s_m, a_m]\right)|s_m, a_m]}_{I} \\
&- \underbrace{\left(\mathbb{E}[\mathbb{E}[j\nabla j|s_m, a_m]]^2 - \mathbb{E}[\mathbb{E}[j|s_m, a_m]\mathbb{E}[\nabla j|s_m, a_m]]^2\right)}_{II}.
\end{aligned}
\tag{D.5}
$$

Using the approximations for ∇j, j in (A.3), (A.10), we have

$$
\begin{aligned}
&\underbrace{\mathbb{E}[(\nabla j)^2|s_m, a_m] - (\nabla j)^2}_{①} \\
=&\mathbb{E}[f_0^2 + 2f_0 f_2 Z_m \sqrt{\epsilon} + 2f_0 f_1 \epsilon + (2f_0 f_3 + f_2^2) Z_m^2 \epsilon + o(\epsilon)|s_m, a_m] \\
&- \left(f_0^2 + 2f_0 f_2 Z_m \sqrt{\epsilon} + 2f_0 f_1 \epsilon + (2f_0 f_3 + f_2^2) Z_m^2 \epsilon + o(\epsilon)\right) \\
=&-2f_0 f_2 Z_m \sqrt{\epsilon} + (2f_0 f_3 + f_2^2)(1 - Z_m^2)\epsilon + o(\epsilon); \\
&\mathbb{E}[j^2 ①|s_m, a_m] \\
=&\mathbb{E}\left[\left(g_0^2 + 2g_0 g_2 Z_m \sqrt{\epsilon} + 2g_0 g_1 \epsilon + (2g_0 g_3 + g_2^2) Z_m^2 \epsilon + o(\epsilon)\right) ①|s_m, a_m\right] \\
=&-4g_0 g_2 f_0 f_2 \epsilon + o(\epsilon).
\end{aligned}
$$

It follows that

$$
I = -\mathbb{E}[\mathbb{E}[j^2 ①|s_m, a_m]] = 4\epsilon \mathbb{E}[g_0 g_2 f_0 f_2] + o(\epsilon). \tag{D.6}
$$

Furthermore, we have

$$
\begin{aligned}
&\underbrace{\mathbb{E}[j|s_m, a_m]}_{②} = g_0 + (g_1 + g_3)\epsilon + o(\epsilon), \\
&\underbrace{\mathbb{E}[\nabla j|s_m, a_m]}_{③} = f_0 + (f_1 + f_3)\epsilon + o(\epsilon), \\
&\mathbb{E}[②③]^2 = (\mathbb{E}[g_0 f_0] + \mathbb{E}[(g_0(f_1 + f_3) + f_0(g_1 + g_3))\epsilon] + o(\epsilon))^2 \\
=&\mathbb{E}[g_0 f_0]^2 + 2\mathbb{E}[g_0 f_0]\mathbb{E}[(g_0(f_1 + f_3) + f_0(g_1 + g_3))]\epsilon + o(\epsilon),
\end{aligned}
$$

and

$$
\begin{aligned}
&\underbrace{\mathbb{E}[j\nabla j|s_m,a_m]}_{④}\\
=&\mathbb{E}[f_0g_0+f_0g_1\epsilon+f_0g_2Z_m\sqrt{\epsilon}+f_0g_3Z_m^2\epsilon+f_1g_0\epsilon+f_2g_0Z_m\sqrt{\epsilon}+f_2g_2Z_m^2\epsilon\\
&+g_0f_3Z_m^2\epsilon|s_m,a_m]+o(\epsilon)\\
=&f_0g_0+(f_0(g_1+g_3)+g_0(f_1+f_3))\epsilon+f_2g_2\epsilon+o(\epsilon)\\
&\mathbb{E}[④]^2=(\mathbb{E}[f_0g_0]+\mathbb{E}[(f_0(g_1+g_3)+g_0(f_1+f_3))]\epsilon+\mathbb{E}[f_2g_2]\epsilon+o(\epsilon))^2\\
=&\mathbb{E}[f_0g_0]^2+2\mathbb{E}[f_0g_0]\mathbb{E}[(f_0(g_1+g_3)+g_0(f_1+f_3))]\epsilon+2\mathbb{E}[f_0g_0]\mathbb{E}[f_2g_2]\epsilon+o(\epsilon).
\end{aligned}
$$

Combining these shows that

$$
II=\mathbb{E}[④]^2-\mathbb{E}[②③]^2=2\mathbb{E}[f_0g_0]\mathbb{E}[f_2g_2]\epsilon+o(\epsilon). \tag{D.7}
$$

Finally, substituting (D.6) and (D.7) into (D.5) gives,

$$
\left|\mathbb{V}[\tilde{F}_m]-\mathbb{V}[F_m]\right|=2\epsilon\left(2\mathbb{E}[f_0g_0f_2g_2]-\mathbb{E}[f_0g_0]\mathbb{E}[f_2g_2]\right)+o(\epsilon)=O(\epsilon). \tag{D.8}
$$

provided that g_0, f_0, g_2, f_2 are all bounded. The boundedness of these quantities is precisely the second set of assumptions in the lemma, so we are done.

■

Lemma D.2 *Let $f(s;\theta)=\max_{a'\in\mathbb{A}}Q^*(s,a;\theta)$, suppose that $f(s;\theta)$ is Lipschitz continuous in $\theta\in\mathbb{R}^{d_\theta}$ and $\partial_s f(s;\theta),\partial_s\nabla_\theta f(s;\theta)$ is continuous, then the difference between the gradients in US and BFF for Q-control is*

$$
\mathbb{E}[\hat{F}_m]-\mathbb{E}[F_m]=\mathbb{E}\left[C(s_m,a_m)\epsilon+o(\epsilon)\right]=O(\epsilon),
$$

where

$$
C(s_m,a_m)=\gamma\sigma^2\partial_s\nabla_\theta f(s_m)\left(\partial_{s'}r(s_m,s_m,a_m)+\gamma\partial_s f(s_m)\right).
$$

In addition, if $|f(s;\theta)-Q^*(s,a;\theta)|,|\nabla_\theta f(s;\theta)-\nabla_\theta Q^*(s,a;\theta)|,|r(s,s,a)|\le C$ almost surely in $s\in\mathbb{S},a\in\mathbb{A},\theta\in\mathbb{R}^{d_\theta}$, then

$$
\left|\mathbb{V}[\tilde{F}_m]-\mathbb{V}[F_m]\right|\le O(\epsilon).
$$

From the above lemmas, we see that the magnitude of the difference is related to $\partial_s Q^*$, $\partial_s\nabla_\theta Q^*$, and $\partial_{s'}r$. We can control the first two terms through the approximating function space. This implies that if the reward $r(s',s,a)$ changes slowly w.r.t. s', then the sample-cloning algorithm for Q-control performs better.

Proof The proof of this Lemma is almost the same as the one of Lemma D.1, except that f_i, g_i are the ones defined in the proof of Lemma A.2, that is, in (A.12), (A.13). Therefore, we omit the proof here. ■

D.2. DIFFERENCE FOR THE WHOLE PROCESS

The p.d.f. of the parameters during the SC algorithm satisfies the equation

$$\partial_t \tilde{p} = \nabla \cdot \left[\mathbb{E}[\tilde{F}]\tilde{p} + \frac{\eta}{2} \nabla \cdot \left(\mathbb{V}[\tilde{F}]\tilde{p} \right) \right]. \tag{D.9}$$

Therefore, the difference of the p.d.f.s $\tilde{d} = p - \tilde{p}$ satisfies

$$\partial_t \tilde{d} = \nabla \cdot \left[\mathbb{E}[F]\tilde{d} + \frac{\eta}{2} \nabla \cdot \left(\mathbb{V}[F]\tilde{d} \right) \right] + \nabla \cdot \left[\left(\mathbb{E}[F] - \mathbb{E}[\tilde{F}] \right) \tilde{p} + \frac{\eta}{2} \nabla \cdot \left(\left(\mathbb{V}[F] - \mathbb{V}[\tilde{F}] \right) \tilde{p} \right) \right]. \tag{D.10}$$

Using this observation, we can prove the following theorem.

Theorem D.3 *The difference $\tilde{d}$ of the p.d.f. between US and SC satisfies,*

$$\left\| \tilde{d}(t) \right\|_* \leq e^{-\frac{\lambda(\beta)}{4} t} \left\| p(0) - p^\infty \right\|_* + O(\epsilon) \sqrt{1 - e^{-\frac{\lambda(\beta)}{2} t}}.$$

Unlike the evolution of $\hat{d}$ in Theorem C.5, the difference between SC and US will eventually decay to $O(\epsilon)$ instead of $O(\epsilon \sqrt{\mathbb{E}[\delta_*^2]} \eta^{-\frac{d_{\theta_*}}{4}})$. As a result, the error of SC is much larger than that of BFF.

Proof The analysis of $\left\| \tilde{d} \right\|_*^2$ is similar to the analysis of $\left\| \hat{d} \right\|_*^2$ before applying Lemma C.4. Therefore, similar to (C.7), we have

$$\begin{aligned}
\frac{1}{2} \partial_t \left\| \tilde{d} \right\|_*^2 \leq & -\frac{\lambda}{4} \left\| \tilde{d} \right\|_*^2 + O(\epsilon^2) \int \frac{p^2}{p^\infty} d\theta + O(\epsilon^2 \eta^2) \int \left| \nabla \left(\frac{p - p^\infty}{p^\infty} \right) \right|^2 p^\infty d\theta \\
\leq & -\frac{\lambda}{4} \left\| \tilde{d} \right\|_*^2 + O(\epsilon^2) \left\| d \right\|_*^2 + O(\epsilon^2) + O(\epsilon^2 \eta^2) \int \left| \nabla \left(\frac{d}{p^\infty} \right) \right|^2 p^\infty d\theta,
\end{aligned}$$

where $d = p - p^\infty$. Combining the above equation with (C.8) and taking $d = p - p^\infty$, we have

$$\begin{aligned}
\frac{1}{2} \partial_t \left(\left\| \hat{d} \right\|_*^2 + \left\| d \right\|_*^2 \right) \leq & -\frac{\lambda}{4} \left(\left\| \hat{d} \right\|_*^2 + \left\| d \right\|_*^2 \right) + O(\epsilon^2) \\
\partial_t \left[e^{\frac{\lambda}{2} t} \left(\left\| \hat{d} \right\|_*^2 + \left\| d \right\|_*^2 \right) \right] \leq & O(\epsilon^2) e^{\frac{\lambda}{2} t}.
\end{aligned}$$

Integrating the above inequality on both sides leads to

$$\left(\left\| \hat{d}(t) \right\|_*^2 + \left\| d(t) \right\|_*^2 \right) \leq e^{-\frac{\lambda}{2} t} \left(\left\| \hat{d}(0) \right\|_*^2 + \left\| d(0) \right\|_*^2 \right) + O\left(\epsilon^2\right) (1 - e^{-\frac{\lambda}{2} t}).$$

Since $\hat{d}(0) = 0$, the above inequality is equivalent to,

$$\left\| \hat{d}(t) \right\|_*^2 \leq e^{-\frac{\lambda}{2} t} \left\| d(0) \right\|_*^2 + O\left(\epsilon^2\right) (1 - e^{-\frac{\lambda}{2} t}).$$

■

E. The PD algorithm

The primal dual method transfer the minimization problem to a mimimax problem, that is,

$$\min_{\theta} \max_{\omega} \ \mathbb{E}_{(s_m,a_m)}[\delta(s_m,a_m;\theta)y(s_m,a_m;\omega) - \frac{1}{2}y(s_m,a_m;\omega)^2]$$

Therefore SGD applied to the above minimax problem does not have the double sampling problem anymore. The algorithm updates the parameters in the following way,

$$\begin{aligned}
&\omega_{k+1} = \omega_k + \beta(\delta(s_m,a_m;\theta_k))\nabla_\omega y(s_m,a_m;\omega_k) - y(s_m,a_m;\omega_k)\nabla_\omega y(s_m,a_m;\omega_k);\\
&\theta_{k+1} = \theta_k - \eta(\nabla_\theta \delta(s_m,a_m;\theta_k)y(s_m,a_m;\omega_{k+1})).
\end{aligned}$$

We usually set $y(s,a;\omega)$ to be the same model as $Q(s,a;\theta)$.

F. Additional algorithm descriptions

The single step Q-control algorithm for the tabular case is given by Algorithm 5.

Algorithm 5 BFF for Q-control (tabular case)

Require: η: Learning rate
Require: $Q^* \in \mathbb{R}^{|\mathbb{S}|\times|\mathbb{A}|}$: matrix of $Q^*(s,a)$ values
Require: $j(s_m,a_m,s_{m+1}) = r(s_{m+1},s_m,a_m) + \gamma \max_a Q^*(s_{m+1},a) - Q^*(s_m,a_m)$
Require: s_0: Initial state

Sample a_0 with an ϵ-greedy policy from $Q^*(s_0;\theta_0)$
Transition to state s_1 from state s_0 and action a_0
$m \leftarrow 0$
while Q^* not converged **do**
  Sample a_{m+1} with an ϵ-greedy policy from $Q^*(s_{m+1};\theta_m)$
  Transition to state s_{m+2} from state s_{m+1} and action a_{m+1}
  $s'_{m+1} \leftarrow s_m + (s_{m+2} - s_{m+1})$
  $\hat{F}_m \leftarrow 0 \in \mathbb{R}^{|\mathbb{S}|\times|\mathbb{A}|}$
  $\hat{F}_m(s_m,a_m) \leftarrow -j(s_m,a_m,s_{m+1})$
  $a^*_{m+1} \leftarrow \arg\max_a Q^*(s'_{m+1},a)$
  $\hat{F}_m(s'_{m+1},a^*_{m+1}) \leftarrow j(s_m,a_m,s_{m+1})$
  $Q^* \leftarrow Q^* - \eta\hat{F}_m$
  $m \leftarrow m+1$
end while