Batch Policy Learning under Constraints

Hoang M. Le [1] Cameron Voloshin [1] Yisong Yue [1]

Abstract

When learning policies for real-world domains, two important questions arise: (i) how to efficiently use pre-collected off-policy, non-optimal behavior data; and (ii) how to mediate among different competing objectives and constraints. We thus study the problem of batch policy learning under multiple constraints, and offer a systematic solution. We first propose a flexible meta-algorithm that admits any batch reinforcement learning and online learning procedure as subroutines. We then present a specific algorithmic instantiation and provide performance guarantees for the main objective and all constraints. As part of off-policy learning, we propose a simple method for off-policy policy evaluation (OPE) and derive PAC-style bounds. Our algorithm achieves strong empirical results in different domains, including in a challenging problem of simulated car driving subject to multiple constraints such as lane keeping and smooth driving. We also show experimentally that our OPE method outperforms other popular OPE techniques on a standalone basis, especially in a high-dimensional setting.

1. Introduction

We study the problem of policy learning under multiple constraints. Contemporary approaches to learning sequential decision making policies have largely focused on optimizing some cost objective that is easily reducible to a scalar value function. However, in many real-world domains, choosing the right cost function to optimize is often not a straightforward task. Frequently, the agent designer faces multiple competing objectives. For instance, consider the aspirational task of designing autonomous vehicle controllers: one may care about minimizing the travel time while making sure the driving behavior is safe, consistent, or fuel efficient. Indeed, many such real-world applications require the primary objective function be augmented with an appropriate set of constraints (Altman, 1999).

Contemporary policy learning research has largely focused on either online reinforcement learning (RL) with a focus on exploration, or imitation learning (IL) with a focus on learning from expert demonstrations. However, many real-world settings already contain large amounts of pre-collected data generated by existing policies (e.g., existing driving behavior, power grid control policies, etc.). We thus study the complementary question: *can we leverage this abundant source of (non-optimal) behavior data in order to learn sequential decision making policies with provable guarantees on both primary objective and constraint satisfaction*?

We thus propose and study the problem of batch policy learning under multiple constraints. Historically, batch RL is regarded as a subfield of approximate dynamic programming (ADP) (Lange et al., 2012), where a set of transitions sampled from the existing system is given and fixed. From an interaction perspective, one can view many online RL methods (e.g., DDPG (Lillicrap et al., 2016)) as running a growing batch RL subroutine per round of online RL. In that sense, batch policy learning is complementary to any exploration scheme. To the best of our knowledge, the study of constrained policy learning in the batch setting is novel.

We present an algorithmic framework for learning sequential decision making policies from off-policy data. We employ multiple learning reductions to online and supervised learning, and present an analysis that relates performance in the reduced procedures to the overall performance with respect to both the primary objective and constraint satisfaction.

Constrained optimization is a well studied problem in supervised machine learning and optimization. In fact, our approach is inspired by the work of Agarwal et al. (2018) in the context of fair classification. In contrast to supervised learning for classification, batch policy learning for sequential decision making introduces multiple additional challenges. First, setting aside the constraints, batch policy learning itself presents a layer of difficulty, and the analysis is significantly more complicated. Second, verifying whether the constraints are satisfied is no longer as straightforward as passing the training data through the learned classifier. In sequential decision making, certifying constraint satisfac-

[1] California Institute of Technology, Pasadena, CA. Correspondence to: Hoang M. Le <hmle@caltech.edu>.

Proceedings of the 36^{th} *International Conference on Machine Learning*, Long Beach, California, PMLR 97, 2019. Copyright 2019 by the author(s).

tion amounts to an off-policy policy evaluation problem, which is a challenging problem and the subject of active research. In this paper, we develop a systematic approach to address these challenges, provide a careful error analysis, and experimentally validate our proposed algorithms. In summary, our contributions are:

- We formulate the problem of batch policy learning under multiple constraints, and present the first approach of its kind to solve this problem. The definition of constraints is general and can subsume many objectives. Our approach utilizes multi-level learning reductions, and we show how to instantiate it using various batch RL and online learning subroutines. We show that guarantees from the subroutines provably lift to provide end-to-end guarantees on the original constrained batch policy learning problem.
- While leveraging techniques from batch RL as a subroutine, we provide a refined theoretical analysis for general non-linear function approximation that improves upon the previously known sample complexity result (Munos & Szepesvári, 2008).
- To evaluate off-policy learning performance and constraint satisfaction, we propose a simple new technique for off-policy policy evaluation (OPE), which is used as a subroutine in our main algorithm. We show that it is competitive to other OPE methods.
- We validate our algorithm and analysis with two experimental settings. First, a simple navigation domain where we consider safety constraint. Second, we consider a high-dimensional racing car domain with smooth driving and lane centering constraints.

2. Problem Formulation

We first introduce notation. Let $\mathrm{X} \subset \mathbb{R}^d$ be a bounded and closed d-dimensional state space. Let A be a finite action space. Let $c : \mathrm{X} \times \mathrm{A} \mapsto [0, \overline{C}]$ be the primary objective cost function that is bounded by $\overline{C}$. Let there be m constraint cost functions, $g_i : \mathrm{X} \times \mathrm{A} \mapsto [0, \overline{G}]$, each bounded by $\overline{G}$. To simplify the notation, we view the set of constraints as a vector function $g : \mathrm{X} \times \mathrm{A} \mapsto [0, \overline{G}]^m$ where $g(x, a)$ is the column vector of individual $g_i(x, a)$. Let $p(\cdot|x, a)$ denote the (unknown) transition/dynamics model that maps state/action pairs to a distribution over the next state. Let $\gamma \in (0, 1)$ denote the discount factor. Let χ be the initial states distribution.

We consider the discounted infinite horizon setting. An MDP is defined using the tuple $(\mathrm{X}, \mathrm{A}, c, g, p, \gamma, \chi)$. A policy $\pi \in \Pi$ maps states to actions, i.e., $\pi(x) \in \mathrm{A}$. The value function $C^\pi : \mathrm{X} \mapsto \mathbb{R}$ corresponding to the primary cost function c is defined in the usual way: $C^\pi(x) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t c(x_t, a_t) \mid x_0 = x\right]$, over the randomness of the policy π and transition dynamics p. We similarly define the vector-value function for the constraint costs $G^\pi : \mathrm{X} \mapsto \mathbb{R}^m$ as $G^\pi(x) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t g(x_t, a_t) | x_0 = x\right]$. Define $C(\pi)$ and $G(\pi)$ as the expectation of $C^\pi(x)$ and $G^\pi(x)$, respectively, over the distribution χ of initial states.

2.1. Batch Policy Learning under Constraints

In batch policy learning, we have a pre-collected dataset, $\mathrm{D} = \{(x_i, a_i, x_i', c(x_i, a_i), g_{1:m}(x_i, a_i)\}_{i=1}^n$, generated from (a set of) historical behavioral policies denoted jointly by π_D. The goal of batch policy learning under constraints is to learn a policy $\pi \in \Pi$ from D that minimizes the primary objective cost while satisfying m different constraints:

$$\begin{aligned} \min_{\pi \in \Pi} \quad & C(\pi) \\ \text{s.t.} \quad & G(\pi) \leq \tau \end{aligned} \tag{OPT}$$

where $G(\cdot) = [g_1(\cdot), \ldots, g_m(\cdot)]^\top$ and $\tau \in \mathbb{R}^m$ is a vector of known constants. We assume that (OPT) is feasible. However, the dataset D might be generated from multiple policies that violate the constraints.

2.2. Examples of Policy Learning with Constraints

Counterfactual & Safe Policy Learning. In conventional online RL, the agent needs to "re-learn" from scratch when the cost function is modified. Our framework enables counterfactual policy learning assuming the ability to compute the new cost objective from the same historical data. A simple example is *safe* policy learning (Garcıa & Fernández, 2015). Define safety cost $g(x, a) = \phi(x, a, c)$ as a new function of existing cost c and features associated with current state-action pair. The goal here is to counterfactually avoid undesirable behaviors observed from historical data. We experimentally study this safety problem in Section 5.

Other examples from the literature that belong to this safety perspective include planning under chance constraints (Ono et al., 2015; Blackmore et al., 2011). The constraint here is $G(\pi) = \mathbb{E}[\mathbb{I}(x \in \mathrm{X}_{error})] = \mathrm{P}(x \in \mathrm{X}_{error}) \leq \tau$.

Multi-objective Batch Learning. Traditional policy learning (RL or IL) presupposes that the agent optimizes a single cost function. In reality, we may want to satisfy multiple objectives that are not easily reducible to a scalar objective function. One example is learning fast driving policies under multiple behavioral constraints such as smooth driving and lane keeping consistency (see Section 5).

2.3. Equivalence between Constraint Satisfaction and Regularization

Our constrained policy learning framework accommodates several existing regularized policy learning settings. Regularization typically encodes prior knowledge, and has been used extensively in the RL and IL literature to improve learning performance. Many instances of regularized policy learning can be naturally cast into (OPT):

- *Entropy regularized RL* (Haarnoja et al., 2017; Ziebart, 2010) maps to policy-dependent constraint cost $g(x) = \mathbb{H}\big(\pi(\cdot|x)\big)$, where $\mathbb{H}$ measures conditional entropy.[1]
- *Conservative policy improvement* (Levine & Abbeel, 2014; Schulman et al., 2015; Achiam et al., 2017) is equivalent to $G(\pi) = D_{KL}(\pi, \pi_k)$, where π_k is some “well-behaving” policy.
- *Smooth imitation learning* (Le et al., 2016) is equivalent to $G(\pi) = \min_{h \in \mathrm{H}} \Delta(h, \pi)$, where H is a class of provably smooth policies and Δ is a divergence metric.
- *Regularizing RL with expert demonstration* (Hester et al., 2018) is equivalent to $G(\pi) = \mathbb{E}[\ell(\pi(x), \pi^*(x))]$, where π^* is the expert policy.

We provide further equivalence derivation of the above examples in Appendix A. Of course, some problems are more naturally described using the regularization perspective, while others using constraint satisfaction.

More generally, one can establish the equivalence between regularized and constrained policy learning via a simple appeal to Lagrangian duality as shown in Proposition 2.1 below. This Lagrangian duality also has a game-theoretic interpretation (Section 5.4 of Boyd & Vandenberghe (2004)), which serves as an inspiration for developing our approach.

Proposition 2.1. *Let Π be a convex set of policies. Let $C : \Pi \mapsto \mathbb{R}, C : \Pi \mapsto \mathbb{R}^K$ be value functions. Consider the two policy optimization tasks:*

$$\texttt{Regularization:} \quad \min_{\pi \in \Pi} \quad C(\pi) + \lambda^\top G(\pi)$$

$$\texttt{Constraint:} \quad \min_{\pi \in \Pi} \quad C(\pi) \quad \textit{s.t. } G(\pi) \leq \tau$$

Assume that the Slater’s condition is satisfied in the `Constraint` *problem (i.e., $\exists \pi$ s.t. $G(\pi) < \tau$). Assume also that the constraint cannot be removed without changing the optimal solution, i.e., $\inf_{\pi \in \Pi} C(\pi) < \inf_{\pi \in \Pi : G(\pi) \leq \tau} C(\pi)$. Then $\forall\ \lambda > 0$, $\exists\ \tau$, and vice versa, such that* `Regularization` *and* `Constraint` *share the same optimal solutions. (Proof in Appendix A.)*

3. Proposed Approach

To make use of strong duality, we first *convexify* the policy class Π by allowing stochastic combinations of policies, which effectively expands Π into its convex hull $\texttt{Conv}(\Pi)$. Formally, $\texttt{Conv}(\Pi)$ contains *randomized policies*, which we denote $\pi = \sum_{t=1}^{T} \alpha_t \pi_t$ for $\pi_t \in \Pi$ and $\sum_{t=1}^{T} \alpha_t = 1$. Executing a mixed π consists of first sampling *one* policy π_t from $\pi_{1:T}$ according to distribution $\alpha_{1:T}$, and then executing π_t. Note that we still have $\mathbb{E}[\pi] = \sum_{t=1}^{T} \alpha_t \mathbb{E}[\pi_t]$ for any first-moment statistic of interest (e.g., state distribution, expected cost). It is easy to see that the augmented version of (OPT) over $\texttt{Conv}(\Pi)$ has a solution at least as good as the original (OPT). As such, to lighten the notation, we will equate Π with its convex hull for the rest of the paper.

Algorithm 1 Meta-algo for Batch Constrained Learning

```
for each round $t$ do
    $\pi_t \leftarrow \texttt{Best-response}(\lambda_t)$
    $\widehat{\pi}_t \leftarrow \frac{1}{t}\sum_{t'=1}^{t} \pi_{t'}$, $\widehat{\lambda}_t \leftarrow \frac{1}{t}\sum_{t'=1}^{t} \lambda_{t'}$
    $\mathrm{L}_{\max} = \max_{\lambda} L(\widehat{\pi}_t, \lambda)$
    $\mathrm{L}_{\min} = L(\texttt{Best-response}(\widehat{\lambda}_t), \widehat{\lambda}_t)$
    if $\mathrm{L}_{\max} - \mathrm{L}_{\min} \leq \omega$ then
        Return $\widehat{\pi}_t$
    end if
    $\lambda_{t+1} \leftarrow \texttt{Online-algorithm}(\pi_1, \dots, \pi_{t-1}, \pi_t)$
end for
```

3.1. Meta-Algorithm

The Lagrangian of (OPT) is $L(\pi, \lambda) = C(\pi) + \lambda^\top (G(\pi) - \tau)$ for $\lambda \in \mathbb{R}^m_+$. Clearly (OPT) is equivalent to the min-max problem: $\min_{\pi \in \Pi} \max_{\lambda \in \mathbb{R}^k_+} L(\pi, \lambda)$. We assume (OPT) is feasible and that Slater’s condition holds (otherwise, we can simply increase the constraint τ by a tiny amount). Slater’s condition and policy class convexification ensure that strong duality holds (Boyd & Vandenberghe, 2004), and (OPT) is also equivalent to the max-min problem: $\max_{\lambda \in \mathbb{R}^k_+} \min_{\pi \in \Pi} L(\pi, \lambda)$.

Since $L(\pi, \lambda)$ is linear in both λ and π (due to stochastic mixture[2]) , strong duality is also a consequence of von Neumann’s celebrated convex-concave minimax theorem for zero-sum games (Von Neumann & Morgenstern, 2007). From a game-thoeretic perspective, the problem becomes finding the equilibrium of a two-player game between the $\pi-$player and the $\lambda-$player (Freund & Schapire, 1999). In this repeated game, the $\pi-$player minimizes $L(\pi, \lambda)$ given the current λ, and the $\lambda-$player maximizes it given the current (mixture over) π.

We first present a meta-algorithm (Algorithm 1) that uses any no-regret online learning algorithm (for λ) and batch policy optimization (for π). At each iteration, given λ_t, the π-player runs `Best-response` to get the best response:

$$\begin{aligned} \texttt{Best-response}(\lambda_t) &= \arg\min_{\pi \in \Pi} L(\pi, \lambda_t) \\ &= \arg\min_{\pi \in \Pi} C(\pi) + \lambda_t^\top (G(\pi) - \tau). \end{aligned}$$

This is equivalent to a standard batch reinforcement learning problem where we learn a policy that is optimal with respect to $c + \lambda_t^\top g$. The corresponding mixed strategy is the uniform distribution over all previous π_t. In response to the

[1] Constraint value function $G(\pi)$ can be viewed as the expectation over discounted state visitation distribution. The lack of explicit discount rate does not intefere with our overall approach.

[2] This places no restrictions on the individual policies. Individual policy can be non-linear and cost function can be non-convex.

Algorithm 2 Constrained Batch Policy Learning

```
Input: Dataset $\mathrm{D} = \{x_i, a_i, x'_i, c_i, g_i\}_{i=1}^n \sim \pi_\mathrm{D}$. Online algorithm parameters: $\ell_1$ norm bound $B$, learning rate $\eta$
 1: Initialize $\lambda_1 = (\frac{B}{m+1}, \dots, \frac{B}{m+1}) \in \mathbb{R}^{m+1}$
 2: for each round $t$ do
 3:     Learn $\pi_t \leftarrow$ FQI($c + \lambda_t^\top g$)     // FQI with cost $c + \lambda_t^\top g$
 4:     Evaluate $\widehat{C}(\pi_t) \leftarrow$ FQE($\pi_t, c$)     // Algo 3 with $\pi_t$, cost c
 5:     Evaluate $\widehat{G}(\pi_t) \leftarrow$ FQE($\pi_t, g$)     // Algo 3 with $\pi_t$, cost g
 6:     $\widehat{\pi}_t \leftarrow \frac{1}{t}\sum_{t'=1}^t \pi_{t'}$
 7:     $\widehat{C}(\widehat{\pi}_t) \leftarrow \frac{1}{t}\sum_{t'=1}^t \widehat{C}(\pi_{t'}), \widehat{G}(\widehat{\pi}_t) \leftarrow \frac{1}{t}\sum_{t'=1}^t \widehat{G}(\pi_{t'})$
 8:     $\widehat{\lambda}_t \leftarrow \frac{1}{t}\sum_{t'=1}^t \lambda_{t'}$
 9:     Learn $\widetilde{\pi} \leftarrow$ FQI($c + \widehat{\lambda}_t^\top g$)     // FQI with cost $c + \widehat{\lambda}_t^\top g$
10:     Evaluate $\widehat{C}(\widetilde{\pi}) \leftarrow$ FQE($\widetilde{\pi}, c$), $\widehat{G}(\widetilde{\pi}) \leftarrow$ FQE($\widetilde{\pi}, g$)
11:     $\widehat{\mathrm{L}}_{\max} = \max_{\lambda, \|\lambda\|_1 = B} \left(\widehat{C}(\widehat{\pi}_t) + \lambda^\top \left[(\widehat{G}(\widehat{\pi}_t) - \tau)^\top, 0\right]^\top\right)$
12:     $\widehat{\mathrm{L}}_{\min} = \widehat{C}(\widetilde{\pi}) + \widehat{\lambda}_t^\top \left[(\widehat{G}(\widetilde{\pi}) - \tau)^\top, 0\right]^\top$
13:     if $\widehat{\mathrm{L}}_{\max} - \widehat{\mathrm{L}}_{\min} \leq \omega$ then
14:         Return $\widehat{\pi}_t$
15:     end if
16:     Set $z_t = \left[(\widehat{G}(\pi_t) - \tau)^\top, 0\right]^\top \in \mathbb{R}^{m+1}$
17:     $\lambda_{t+1}[i] = B\frac{\lambda_t[i]e^{-\eta z_t[i]}}{\sum_j \lambda_t[j]e^{-\eta z_t[j]}} \forall i$     // $\lambda[i]$ the $i^{th}$ coordinate
18: end for
```

$\pi-$player, the $\lambda-$player employs `Online-algorithm`, which can be *any* no-regret algorithm that satisfies:

$$\sum_t L(\pi_t, \lambda_t) \geq \max_\lambda \sum_t L(\pi_t, \lambda) - o(T)$$

Finally, the algorithm terminates when the estimated primal-dual gap is below a threshold ω (Lines 7-8).

Leaving aside (for the moment) issues of generalization, Algorithm 1 is guaranteed to converge assuming: (i) `Best-response` gives the best single policy in the class, and (ii) $\mathrm{L}_{\max}$ and $\mathrm{L}_{\min}$ can be evaluated exactly.

Proposition 3.1. *Assuming (i) and (ii) above, Algorithm 1 is guaranteed to stop and the convergence depends on the regret of `Online-algorithm`. (Proof in Appendix B.)*

3.2. Specific Instantiation of Meta-Algorithm

We now focus on a specific instantiation of Algorithm 1. Algorithm 2 is our main algorithm in this paper.

Policy Learning. We instantiate `Best-response` with Fitted Q Iteration (FQI), a model-free off-policy learning approach (Ernst et al., 2005). FQI relies on a series of reductions to supervised learning. The key idea is to approximate the true action-value function Q^* by a sequence $\{Q_k \in \mathrm{F}\}_{k=0}^K$, where F is a chosen function class.

Algorithm 3 Fitted Q Evaluation: FQE(π, c)

```
Input: Dataset $\mathrm{D} = \{x_i, a_i, x'_i, c_i\}_{i=1}^n \sim \pi_\mathrm{D}$. Function class F. Policy $\pi$ to be evaluated
1: Initialize $Q_0 \in \mathrm{F}$ randomly
2: for $k = 1, 2, \dots, K$ do
3:     Compute target $y_i = c_i + \gamma Q_{k-1}(x'_i, \pi(x'_i)) \ \forall i$
4:     Build training set $\widetilde{\mathrm{D}}_k = \{(x_i, a_i), y_i\}_{i=1}^n$
5:     Solve a supervised learning problem:
           $Q_k = \arg\min_{f \in \mathrm{F}} \frac{1}{n}\sum_{i=1}^n (f(x_i, a_i) - y_i)^2$
6: end for
Output: $\widehat{C}^\pi(x) = Q_K(x, \pi(x)) \quad \forall x$
```

In Lines 3 & 9, `FQI`$(c + \lambda^\top g)$ is defined as follows. With Q_0 randomly initialized, for each $k = 1, \dots, K$, we form a new training dataset $\widetilde{\mathrm{D}}_k = \{(x_i, a_i), y_i\}_{i=1}^n$ where:

$$\forall i: \quad y_i = (c_i + \lambda^\top g_i) + \gamma \min_a Q_{k-1}(x'_i, a),$$

and $(x_i, a_i, x'_i, c_i, g_i) \sim \mathrm{D}$ (original dataset). A supervised regression procedure is called to solve for:

$$Q_k = \arg\min_{f \in \mathrm{F}} \frac{1}{n}\sum_{i=1}^n (f(x_i, a_i) - y_i)^2.$$

FQI returns the policy: $\pi_K = \arg\min_a Q_K(\cdot, a)$. FQI has been shown to work well with several empirical domains: spoken dialogue systems (Pietquin et al., 2011), physical robotic soccer (Riedmiller et al., 2009), and cart-pole swing-up (Riedmiller, 2005), and clinical treatment (Prasad et al., 2017). Another possible model-free subroutine is Least-Squares Policy Iteration (LSPI) (Lagoudakis & Parr, 2003). One can also consider model-based alternatives (Ormoneit & Sen, 2002).

Off-policy Policy Evaluation. A crucial difference between constrained policy learning and existing work on constrained supervised learning is the technical challenge of evaluating the objective and constraints. First, estimating $\widehat{L}(\pi, \lambda)$ (Lines 11-12) requires estimating $\widehat{C}(\pi)$ and $\widehat{G}(\pi)$. Second, any gradient-based approach to `Online-learning` requires passing in $\widehat{G}(\pi) - \tau$ as part of gradient estimate (line 15). This problem is known as the off-policy policy evaluation (OPE) problem: we need to evaluate $\widehat{C}(\pi)$ and $\widehat{G}(\pi)$ having only access to data $\mathrm{D} \sim \pi_\mathrm{D}$

There are three main contemporary approaches to OPE: (i) importance weighting (IS) (Precup et al., 2000; 2001), which is unbiased but often has high-variance; (ii) regression-based direct methods (DM), which are typically model-based (Thomas & Brunskill, 2016),and can be biased but have much lower variance than IS; and (iii) doubly-robust techniques (Jiang & Li, 2016; Dudík et al., 2011), which combine IS and DM.

We propose a simple model-free technique using function approximation, called Fitted Q Evaluation (FQE). FQE is based on an iterative reductions scheme similar to FQI, but for the problem of off-policy evaluation. Algorithm 3 lays out the steps. The key difference with FQI is that the *min* operator is replaced by $Q_{k-1}(x'_i, \pi(x'_i))$ (Line 3 of Algorithm 3). Each x'_i comes from the original D. Since we know $\pi(x'_i)$, each $\widetilde{\mathrm{D}}_k$ is well-defined. Note that FQE can be plugged-in as a direct method if one wishes to augment the policy evaluation with a doubly-robust technique.

Online Learning Subroutine. As $L(\pi_t, \lambda)$ is linear in λ, many online convex optimization approaches can be used for `Online-algorithm`. Perhaps the simplest choice is Online Gradient Descent (OGD) (Zinkevich, 2003). We include an instantiation using OGD in Appendix G.

For our main Algorithm 2, similar to (Agarwal et al., 2018), we use Exponentiated Gradient (EG) (Kivinen & Warmuth, 1997), which has a regret bound of $O(\sqrt{\log(m)T})$ instead of $O(\sqrt{mT})$ as in OGD. One can view EG as a variant of Online Mirror Descent (Nemirovsky & Yudin, 1983) with a softmax link function, or of Follow-the-Regularized-Leader with entropy regularization (Shalev-Shwartz et al., 2012). Gradient-based algorithms generally require bounded λ. We thus force $\|\lambda\|_1 \le B$ using hyperparameter B. Solving (OPT) exactly requires $B = \infty$. We will analyze Algorithm 2 with respect to finite B. With some abuse of notation, we augment λ into a $(m+1)-$dimensional vector by appending $B - \|\lambda\|_1$, and augment the constraint cost vector g by appending 0 (Lines 11, 12 & 15 of Algorithm 2).[3]

4. Theoretical Analysis

4.1. Convergence Guarantee

The convergence rate of Algorithm 2 depends on the radius B of the dual variables λ, the maximal constraint value $\overline{G}$, and the number of constraints m. In particular, we can show $O(\frac{B^2}{\omega^2})$ convergence for primal-dual gap ω.

Theorem 4.1 (Convergence of Algorithm 2)**.** *After T iterations, the empirical duality gap is bounded by*

$$\widehat{\mathrm{L}}_{\max} - \widehat{\mathrm{L}}_{\min} \le 2\frac{B\log(m+1)}{\eta T} + 2\eta B\overline{G}^2$$

Consequently, to achieve the primal-dual gap of ω, setting $\eta = \frac{\omega}{4\overline{G}^2 B}$ will ensure that Algorithm 2 converges after $\frac{16B^2\overline{G}^2\log(m+1)}{\omega^2}$ iterations. (Proof in Appendix B.)

Convergence analysis of our main Algorithm 2 is an extension of the proof to Proposition 3.1, leveraging the no-regret property of the EG procedure (Shalev-Shwartz et al., 2012).

4.2. Generalization Guarantee of FQE and FQI

In this section, we provide sample complexity analysis for FQE and FQI as *standalone* procedures for off-policy evaluation and off-policy learning. We use the notion of pseudo-dimension as capacity measure of non-linear function class F (Friedman et al., 2001). Pseudo-dimension $\mathtt{dim}_{\mathrm{F}}$, which naturally extends VC dimension into the regression setting, is defined as the VC dimension of the function class induced by the sub-level set of functions of F: $\mathtt{dim}_{\mathrm{F}} = \mathtt{VC\text{-}dim}(\{(x,y) \mapsto \mathrm{sign}(f(x)-y) : f \in \mathrm{F}\})$. Pseudo-dimension is finite for a large class of function approximators. For example, Bartlett et al. (2017) bounded the pseudo-dimension of piece-wise linear deep neural networks (e.g., with ReLU activations) as $O(WL\log W)$, where W is the number of weights, and L is the number of layers.

Both FQI and FQE rely on reductions to supervised learning to update the value functions. In both cases, the learned policy and evaluation policy induces a different state-action distribution compared to the data generating distribution μ. We use the notion of concentration coefficient for the worst case, proposed by (Munos, 2003), to measure the degree of distribution shift. The following standard assumption from analysis of related ADP algorithms limits the severity of distribution shift over future time steps:

Assumption 1 (Concentrability coefficient of future state-action distribution)**.** *(Munos, 2003; 2007; Munos & Szepesvári, 2008; Antos et al., 2008a;b; Lazaric et al., 2010; 2012; Farahmand et al., 2009; Maillard et al., 2010)* *Let P^{π} be the operator acting on $f : \mathrm{X} \times \mathrm{A} \mapsto \mathbb{R}$ s.t. $(P^{\pi}f)(x,a) = \int_{\mathrm{X}} f(x', \pi(x'))p(dx'|x,a)$. Given data generating distribution μ, initial state distribution χ, for $m \ge 0$ and an arbitrary sequence of stationary policies $\{\pi_m\}_{m\ge 1}$ define the concentration coeffient:*

$$\beta_{\mu}(m) = \sup_{\pi_1,\ldots,\pi_m} \left\| \frac{d(\chi P^{\pi_1} P^{\pi_2} \ldots P^{\pi_m})}{d\mu} \right\|_{\infty}$$

We assume $\beta_{\mu} = (1-\gamma)^2 \sum_{m\ge 1} m\gamma^{m-1}\beta_{\mu}(m) < \infty$.

This assumption is valid for a fairly large class of MDPs (Munos, 2007). For instance β_{μ} is finite for any finite MDP, or any infinite state-space MDP with bounded transition density.[4] Having a finite concentration coefficient is equivalent the top-Lyapunov exponent $\Gamma \le 0$ (Bougerol & Picard, 1992), which means the underlying stochastic system is stable. We show below a simple sufficient condition for Assumption 1 (albeit stronger than necessary).

Example 4.1. Consider an MDP such that for any non-stationary distribution ρ, the marginals over states satisfy $\frac{\rho_x(x)}{\mu_x(x)} \le L$ (i.e., transition dynamics are sufficiently stochastic), and $\exists M : \forall x, a : \mu(a|x) > \frac{1}{M}$ (i.e., the behavior policy is sufficiently exploratory). Then $\beta_{\mu} \le LM$.

Recall that for a given policy π, the Bellman (evaluation) operator is defined as $(\mathbb{T}^{\pi}Q)(x,a) = r(x,a) + \gamma \int_{\mathrm{X}} Q(x', \pi(x'))p(dx'|x,a)$. In general $\mathbb{T}^{\pi}f$ may not belong to F for $f \in \mathrm{F}$. For FQE (and FQI), the main operation in the algorithm is to iteratively project $\mathbb{T}^{\pi}Q_{k-1}$ back to F via $Q_k = \arg\min_{f\in\mathrm{F}} \|f - \mathbb{T}^{\pi}Q_{k-1}\|$. The performance

[3] The $(m+1)^{th}$ coordinate of g is thus always satisfied. This augmentation is only necessary when executing EG.

[4] This assumption ensures sufficient data diversity, even when the executing policy is deterministic. An example of how learning can fail without this assumption is based on the "combination lock" MDP (Koenig & Simmons, 1996). In this deterministic MDP example, β_{μ} can grow arbitrarily large, and we need an exponential number of samples for both FQE and FQI. See Appendix D.

of both FQE and FQI thus depend on how well the function class F approximates the Bellman operator. We measure the ability of function class F to approximate the Bellman evaluation operator via the worst-case Bellman error:

Definition 4.1 (inherent Bellman evaluation error)**.** Given a function class F and policy π, the *inherent Bellman evaluation error* of F is defined as $d_{\mathrm{F}}^{\pi} = \sup_{g\in\mathrm{F}} \inf_{f\in\mathrm{F}} \|f - \mathbb{T}^{\pi} g\|_{\pi}$ where $\|\cdot\|_{\pi}$ is the ℓ_2 norm weighted by the state-action distribution induced by π.

We are now ready to state the generalization bound for FQE:

Theorem 4.2 (Generalization error of FQE)**.** *Under Assumption 1, for $\epsilon > 0$ & $\delta \in (0,1)$, after K iterations of Fitted Q Evaluation (Algorithm 3), for $n = O\big(\frac{\overline{C}^4}{\epsilon^2}(\log\frac{K}{\delta} + \mathtt{dim}_{\mathrm{F}} \log\frac{\overline{C}^2}{\epsilon^2} + \log \mathtt{dim}_{\mathrm{F}})\big)$, we have with probability $1-\delta$:*

$$\big|C(\pi) - \widehat{C}(\pi)\big| \leq \frac{\gamma^{1/2}}{(1-\gamma)^{3/2}}\big(\sqrt{\beta_\mu}\,(2d_{\mathrm{F}}^{\pi} + \epsilon) + \frac{2\gamma^{K/2}\overline{C}}{(1-\gamma)^{1/2}}\big).$$

This result shows a dependency on ϵ of $\widetilde{O}(\frac{1}{\epsilon^2})$, compared to $\widetilde{O}(\frac{1}{\epsilon^4})$ from other related ADP algorithms (Munos & Szepesvári, 2008; Antos et al., 2008b). The price that we pay is a multiplicative constant 2 in front of the inherent error d_{F}^{π}. The error from second term on RHS decays exponentially with iterations K. The proof is in Appendix E.

We can show an analogous generalization bound for FQI. While FQI has been widely used, to the best of our knowledge, a complete analysis of FQI for non-linear function approximation has not been previously reported.[5]

Definition 4.2 (inherent Bellman optimality error)**.** (Munos & Szepesvári, 2008) Recall that the Bellman optimality operator is defined as $(\mathbb{T}Q)(x,a) = r(x,a) + \gamma \int_{\mathrm{X}} \min_{a'\in\mathrm{A}} Q(x',a') p(dx'|x,a)$. Given a function class F, the *inherent Bellman error* is defined as $d_{\mathrm{F}} = \sup_{g\in\mathrm{F}} \inf_{f\in\mathrm{F}} \|f - \mathbb{T}g\|_{\mu}$, where $\|\cdot\|_{\mu}$ is the ℓ_2 norm weighted by μ, the state-action distribution induced by π_{D}.

Theorem 4.3 (Generalization error of FQI)**.** *Under Assumption 1, for $\epsilon > 0$ & $\delta \in (0,1)$, after K iterations of Fitted Q Iteration, for $n = O\big(\frac{\overline{C}^4}{\epsilon^2}(\log\frac{K}{\delta} + \mathtt{dim}_{\mathrm{F}} \log\frac{\overline{C}^2}{\epsilon^2} + \log \mathtt{dim}_{\mathrm{F}})\big)$, we have with probability $1-\delta$:*

$$\big|C^* - C(\pi_K)\big| \leq \frac{2\gamma}{(1-\gamma)^3}\big(\sqrt{\beta_\mu}\,(2d_{\mathrm{F}} + \epsilon) + 2\gamma^{K/2}\overline{C}\big)$$

where π_K is the policy acting greedy with respect to the returned function Q_K. (Proof in Appendix F.)

4.3. End-to-End Generalization Guarantee

We are ultimately interested in the test-time performance and constraint satisfaction of the returned policy from Algorithm 2. We now connect the previous analyses from Theorems 4.1, 4.2 & 4.3 into an end-to-end error analysis.

Since Algorithm 2 uses FQI and FQE as subroutines, the inherent Bellman error terms d_{F} and d_{F}^{π} will enter our overall performance bound. Estimating the inherent Bellman error caused by function approximation is not possible in general (chapter 11 of Sutton & Barto (2018)). We assume existence of a sufficiently expressive F that can generally make d_{F} and d_{F}^{π} arbitrarily small. To simplify our end-to-end analysis, consider $d_{\mathrm{F}} = 0$ and $d_{\mathrm{F}}^{\pi} = 0$, i.e., the function class F is closed under applying the Bellman operator.

Assumption 2 (Bellman operator realizability)**.** *We consider function classes F sufficiently rich so that $\forall f: \mathbb{T}f \in \mathrm{F}$ & $\mathbb{T}^{\pi} f \in \mathrm{F}$ for the policies π returned by Algorithm 2.*

With Assumptions 1 & 2, we have the following error bound:

Theorem 4.4 (Generalization guarantee of Algorithm 2)**.** *Let π^* be the optimal policy to (OPT). Denote $\overline{V} = \overline{C} + B\overline{G}$. Let K be the number of iterations of FQE and FQI. Let $\widehat{\pi}$ be the policy returned by Algorithm 2, with termination threshold ω. For $\epsilon > 0$ & $\delta \in (0,1)$, when $n = O\big(\frac{\overline{V}^4}{\epsilon^2}(\log\frac{K(m+1)}{\delta} + \mathtt{dim}_{\mathrm{F}} \log\frac{\overline{V}^2}{\epsilon^2} + \log \mathtt{dim}_{\mathrm{F}})\big)$, we have with probability at least $1-\delta$:*

$$C(\widehat{\pi}) \leq C(\pi^*) + \omega + \frac{(4+B)\gamma}{(1-\gamma)^3}\big(\sqrt{\beta_\mu}\epsilon + 2\gamma^{K/2}\overline{V}\big),$$

and

$$G(\widehat{\pi}) \leq \tau + 2\frac{\overline{V} + \omega}{B} + \frac{\gamma^{1/2}}{(1-\gamma)^{3/2}}\big(\sqrt{\beta_\mu}\epsilon + \frac{2\gamma^{K/2}\overline{V}}{(1-\gamma)^{1/2}}\big).$$

The proof is in Appendix C. This result guarantees that, upon termination of Algorithm 2, the true performance on the main objective can be arbitrarily close to that of the optimal policy. At the same time, each constraint will be approximately satisfied with high probability, assuming sufficiently large B & K, and sufficiently small ϵ.

5. Empirical Analysis

We perform experiments on two different domains: a grid-world domain (from OpenAI's FrozenLake) under a safety constraint, and a challenging high-dimensional car racing domain (from OpenAI's CarRacing) under multiple behavior constraints. We seek to answer the following questions in our experiments: (i) whether the empirical convergence behavior of Algorithm 2 is consistent with our theory; and (ii) how the returned policy performs with respect to the main objective and constraint satisfaction. Appendix H includes a more detailed discussion of our experiments.

5.1. Frozen Lake.

Environment & Data Collection. The environment is an 8x8 grid. The agent has 4 actions N,S,E,W at each state. The main goal is to navigate from a starting position to

[5] FQI for continuous action space from (Antos et al., 2008a) is a variant of fitted policy iteration and not the version of FQI under consideration. The appendix of (Lazaric & Restelli, 2011) contains a proof of FQI specifically for linear function class.

the goal. Each episode terminates when the agent reaches the goal or falls into a hole. The main cost function is defined as $c = -1$ if goal is reached, otherwise $c = 0$ everywhere. We simulate a non-optimal data gathering policy π_{D} by adding random sub-optimal actions to the shortest path policy from any given state to goal. We run π_D for 5000 trajectories to collect the behavior dataset D (with constraint cost measurement specified below).

Counterfactual Safety Constraint. We augment the main objective c with safety constraint cost defined as $g = 1$ if the agent steps into a hole, and $g = 0$ otherwise. We set the constraint threshold $\tau = 0.1$, roughly 75% of the accumulated constraint cost of behavior policy π_{D}. The threshold can be interpreted as a counterfactually acceptable probability that we allow the learned policy to fail.

Results. The empirical primal dual gap $\widehat{\mathrm{L}}_{\max} - \widehat{\mathrm{L}}_{\min}$ in Figure 1 (left) quickly decreases toward the optimal gap of zero. The convergence is fast and monotonic, supporting the predicted behavior from our theory. The test-time performance in Figure 1 (middle) shows the safety constraint is always satisfied, while the main objective cost also smoothly converges to the optimal value achieved by an online RL baseline (DQN) trained without regard to the constraint. The returned policy significantly outperformed the data gathering policy π_{D} on both main and safety cost.

5.2. Car Racing.

Environment & Data Collection. The car racing environment (OpenAI), is a high-dimensional domain where the state is a $96 \times 96 \times 3$ tensor of raw pixels. The action space $\mathrm{A} = \{\text{steering} \times \text{gas} \times \text{brake}\}$ takes 12 discretized values. The goal in this episodic task is to traverse over 95% of the track, measured by a given number of "tiles" as a proxy for distance coverage. The agent receives a reward (negative cost) for each unique tile crossed and no reward if the agent is off-track. A small positive cost applies at every time step, with maximum horizon of 1000 for each episode. With these costs given by the environment, one can train online RL agent using DDQN (Van Hasselt et al., 2016). We collect ≈ 5000 trajectories from DDQN's randomization, resulting in data set D with ≈ 94000 transition tuples.

Fast Driving under Behavioral Constraints. We study the problem of minimizing environment cost while subject to two behavioral constraints: smooth driving and lane centering. The first constraint G_0 approximates smooth driving by $g_0(x, a) = 1$ if a contains braking action, and 0 otherwise. The second constraint cost g_1 measures the distance between the agent and center of lane at each time step. This is a highly challenging setup since three objectives and constraints are in direct conflict with one another, e.g., fast driving encourages the agent to cut corners and apply frequent brakes to make turns. Outside of this work, we are not aware of previous work in policy learning with 2 or more constraints in high-dimensional settings.

Baseline and Procedure. As a naïve baseline, DDQN achieves low cost, but exhibits "non-smooth" driving behavior (see our supplementary videos). We set the threshold for each constraint to 75% of the DDQN benchmark. We also compare against regularized batch RL algorithms (Farahmand et al., 2009), specifically regularized LSPI. We instantiate our subroutines, FQE and FQI, with multi-layered CNNs. We augment LSPI's linear policy with non-linear features derived from a well-performing FQI model.

Results. The returned mixture policy from our algorithm achieves low main objective cost, comparable with online RL policy trained without regard to constraints. After several initial iterations violating the braking constraint, the returned policy - corresponding to the appropriate λ trade-off - satisties both constraints, while improving the main objective. The improvement over data gathering policy is significant for both constraints and main objective.

Regularized policy learning is an alternative approach to (OPT) (section 2). We provide the regularized LSPI baseline the same set of λ found by our algorithm for one-shot regularized learning (Figures 2 (left & middle)). While regularized LSPI obtains good performance for the main objective, it does not achieve acceptable constraint satisfaction. By default, regularized policy learning requires parameter tuning heuristics. In principle, one can perform a grid-search over a range of parameters to find the right combination - we include such an example for both regularized LSPI and FQI in Appendix H. However, since our objective and constraints are in conflict, main objective and constraint satisfaction of policies returned by one-shot regularized learning are sensitive to step changes in λ. In constrast, our approach is systematic, and is able to avoid the curse-of-dimensionality of brute-force search that comes with multiple constraints.

In practice, one may wish to deterministically extract a single policy from the returned mixture for execution. A de-randomized policy can be obtained naturally from our algorithm by selecting the best policy from the existing FQE's estimates of individual `Best-response` policies.

5.3. Off-Policy Evaluation

The off-policy evaluation by FQE is critical for updating policies in our algorithm, and is ultimately responsible for certifying constraint satisfaction. While other OPE methods can also be used in place of FQE, we find that the estimates from popular methods are not sufficiently accurate in a high-dimensional setting. As a standalone comparison, we select an individual policy and compare FQE against PDIS (Precup et al., 2000), DR (Jiang & Li, 2016) and WDR (Thomas & Brunskill, 2016) with respect to the constraint cost evaluation. To compare both accuracy and

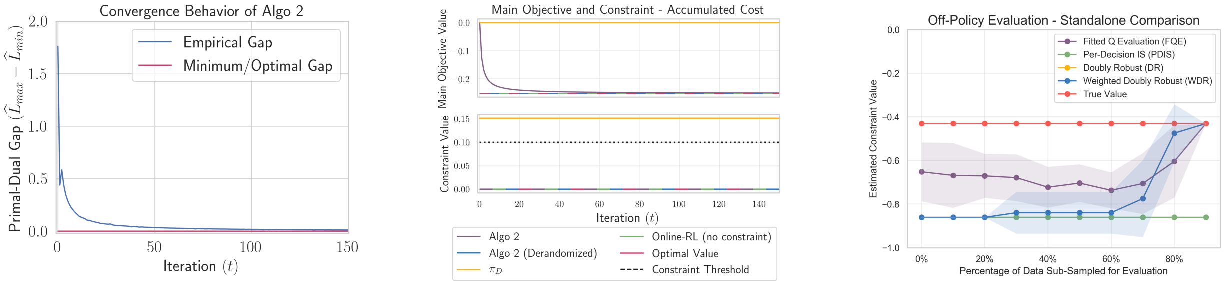

Figure 1. FrozenLake Results. (Left) Empirical duality gap of algorithm 2 vs. optimal gap. *(Middle)* Comparison of returned policy and others w.r.t. (top) main objective value and (bottom) safety constraint value. *(Right)* FQE vs. other OPE methods on a standalone basis.

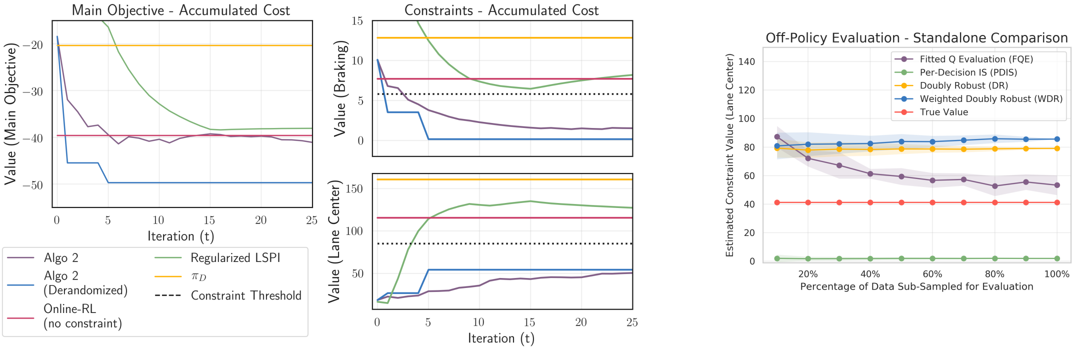

Figure 2. CarRacing Results. (Left) & *(Middle)* (Lower is better) Comparing our algorithm, regularized LSPI, online RL w/o constraints, behavior policy π_D w.r.t. main cost objectives and two constraints. *(Right)* FQE vs. other OPE methods on a standalone basis.

data-efficiency, for each domain we randomly sample different subsets of dataset $\mathbb{D}$ (from 10% to 100% transitions, 30 trials each). Figure 1 (right) and 2 (right) illustrate the difference in quality. In the FrozenLake domain, FQE performs competitively with the top baseline method (DR and WDR), converging to the true value estimate as the data subsample grows close to 100%. In the high-dimensional car domain, FQE signficantly outperforms other methods.

6. Other Related Work

Constrained MDP (CMDP). Among the most important techniques for solving CMDP are the Lagrangian approach and solving the dual LP program via occupation measure(Altman, 1999). However, these approaches require known MDP, and small state dimension so that solving via an LP is tractable. More recently, the constrained policy optimization approach (CPO) by (Achiam et al., 2017) learns a policy when the model is not initially known. The focus of CPO is on online safe exploration, and thus is not directly comparable to our setting. Other approaches (Cheng et al., 2019; Dalal et al., 2018) address safe exploration by building the constraint directly into the policy.

Multi-objective Reinforcement Learning (MORL). (Van Moffaert & Nowé, 2014; Roijers et al., 2013) Approaches to MORL have largely focused on approximating the Pareto frontier that trades-off competing objectives (Van Moffaert & Nowé, 2014; Roijers et al., 2013). The underlying approach to MORL frequently relies on linear or non-linear scalarization of rewards to heuristically turns the problem into a standard RL problem. Our proposed approach represents another systematic paradigm to solve MORL, whether in batch or online settings.

7. Discussion and Conclusion

Our implementation complies with the steps laid out in Algorithm 2. In very large scale or high-dimensional problems, one could consider a noisy update version for both policy learning and evaluation. We leave the theorerical and practical exploration of this extension to future work. In our high-dimensional domain with long horizon, our proposed FQE algorithm for OPE achieves strong results. More extensive comparisons between FQE and other contemporary OPE methods deserve further study.

We have presented a systematic approach for batch policy learning under multiple constraints. Our problem formulation can accommodate general definition of constraints, as partly illustrated by our experiments. We provide guarantees for our algorithm for both the main objective and constraint satisfaction. Our empirical results show a promise of making constrained batch policy learning applicable for real-world domains, where behavior data is abundant.

References

Achiam, J., Held, D., Tamar, A., and Abbeel, P. Constrained policy optimization. In *International Conference on Machine Learning*, pp. 22–31, 2017.

Agarwal, A., Beygelzimer, A., Dudík, M., Langford, J., and Wallach, H. A reductions approach to fair classification. In *International Conference on Machine Learning*, 2018.

Altman, E. *Constrained Markov decision processes*, volume 7. CRC Press, 1999.

Antos, A., Szepesvári, C., and Munos, R. Fitted q-iteration in continuous action-space mdps. In *Advances in neural information processing systems*, pp. 9–16, 2008a.

Antos, A., Szepesvári, C., and Munos, R. Learning near-optimal policies with bellman-residual minimization based fitted policy iteration and a single sample path. *Machine Learning*, 71(1): 89–129, 2008b.

Bartlett, P. L., Harvey, N., Liaw, C., and Mehrabian, A. Nearly-tight vc-dimension bounds for piecewise linear neural networks. In *Proceedings of the 22nd Annual Conference on Learning Theory (COLT 2017)*, 2017.

Blackmore, L., Ono, M., and Williams, B. C. Chance-constrained optimal path planning with obstacles. *IEEE Transactions on Robotics*, 27(6):1080–1094, 2011.

Bougerol, P. and Picard, N. Strict stationarity of generalized autoregressive processes. *The Annals of Probability*, pp. 1714–1730, 1992.

Boyd, S. and Vandenberghe, L. *Convex optimization*. Cambridge university press, 2004.

Cheng, R., Verma, A., Orosz, G., Chaudhuri, S., Yue, Y., and Burdick, J. W. Control regularization for reduced variance reinforcement learning. In *International Conference on Machine Learning*, 2019.

Dalal, G., Dvijotham, K., Vecerik, M., Hester, T., Paduraru, C., and Tassa, Y. Safe exploration in continuous action spaces. *arXiv preprint arXiv:1801.08757*, 2018.

Dudík, M., Langford, J., and Li, L. Doubly robust policy evaluation and learning. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, pp. 1097–1104. Omnipress, 2011.

Ernst, D., Geurts, P., and Wehenkel, L. Tree-based batch mode reinforcement learning. *Journal of Machine Learning Research*, 6(Apr):503–556, 2005.

Farahmand, A. M., Ghavamzadeh, M., Mannor, S., and Szepesvári, C. Regularized policy iteration. In *Advances in Neural Information Processing Systems*, pp. 441–448, 2009.

Farajtabar, M., Chow, Y., and Ghavamzadeh, M. More robust doubly robust off-policy evaluation. *arXiv preprint arXiv:1802.03493*, 2018.

Freund, Y. and Schapire, R. E. Adaptive game playing using multiplicative weights. *Games and Economic Behavior*, 29: 79–103, 1999.

Friedman, J., Hastie, T., and Tibshirani, R. *The elements of statistical learning*. Springer, 2001.

Garcıa, J. and Fernández, F. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, 16(1):1437–1480, 2015.

Guo, Z., Thomas, P. S., and Brunskill, E. Using options and covariance testing for long horizon off-policy policy evaluation. In *Advances in Neural Information Processing Systems*, pp. 2492–2501, 2017.

Györfi, L., Kohler, M., Krzyzak, A., and Walk, H. *A distribution-free theory of nonparametric regression*. Springer Science & Business Media, 2006.

Ha, D. and Schmidhuber, J. World models. *arXiv preprint arXiv:1803.10122*, 2018.

Haarnoja, T., Tang, H., Abbeel, P., and Levine, S. Reinforcement learning with deep energy-based policies. In *International Conference on Machine Learning*, pp. 1352–1361, 2017.

Haussler, D. Sphere packing numbers for subsets of the boolean n-cube with bounded vapnik-chervonenkis dimension. *Journal of Combinatorial Theory, Series A*, 69(2):217–232, 1995.

Henaff, M., Canziani, A., and LeCun, Y. Model-predictive policy learning with uncertainty regularization for driving in dense traffic. *arXiv preprint arXiv:1901.02705*, 2019.

Hester, T., Vecerik, M., Pietquin, O., Lanctot, M., Schaul, T., Piot, B., Horgan, D., Quan, J., Sendonaris, A., Osband, I., et al. Deep q-learning from demonstrations. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Jiang, N. and Li, L. Doubly robust off-policy value evaluation for reinforcement learning. In *International Conference on Machine Learning*, pp. 652–661, 2016.

Kakade, S. and Langford, J. Approximately optimal approximate reinforcement learning. In *ICML*, volume 2, pp. 267–274, 2002.

Kivinen, J. and Warmuth, M. K. Exponentiated gradient versus gradient descent for linear predictors. *Information and Computation*, 132(1):1–63, 1997.

Koenig, S. and Simmons, R. G. The effect of representation and knowledge on goal-directed exploration with reinforcement-learning algorithms. *Machine Learning*, 22(1-3):227–250, 1996.

Lagoudakis, M. G. and Parr, R. Least-squares policy iteration. *Journal of machine learning research*, 4(Dec):1107–1149, 2003.

Lange, S., Gabel, T., and Riedmiller, M. Batch reinforcement learning. In *Reinforcement learning*, pp. 45–73. Springer, 2012.

Lazaric, A. and Restelli, M. Transfer from multiple mdps. In *Advances in Neural Information Processing Systems*, pp. 1746–1754, 2011.

Lazaric, A., Ghavamzadeh, M., and Munos, R. Finite-sample analysis of lstd. In *ICML-27th International Conference on Machine Learning*, pp. 615–622, 2010.

Lazaric, A., Ghavamzadeh, M., and Munos, R. Finite-sample analysis of least-squares policy iteration. *Journal of Machine Learning Research*, 13(Oct):3041–3074, 2012.

Le, H. M., Kang, A., Yue, Y., and Carr, P. Smooth imitation learning for online sequence prediction. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning-Volume 48*, pp. 680–688. JMLR. org, 2016.

Lee, W. S., Bartlett, P. L., and Williamson, R. C. Efficient agnostic learning of neural networks with bounded fan-in. *IEEE Transactions on Information Theory*, 42(6):2118–2132, 1996.

Levine, S. and Abbeel, P. Learning neural network policies with guided policy search under unknown dynamics. In *Advances in Neural Information Processing Systems*, pp. 1071–1079, 2014.

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., and Wierstra, D. Continuous control with deep reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2016.

Liu, Q., Li, L., Tang, Z., and Zhou, D. Breaking the curse of horizon: Infinite-horizon off-policy estimation. In *Advances in Neural Information Processing Systems*, pp. 5361–5371, 2018.

Maillard, O.-A., Munos, R., Lazaric, A., and Ghavamzadeh, M. Finite-sample analysis of bellman residual minimization. In *Proceedings of 2nd Asian Conference on Machine Learning*, pp. 299–314, 2010.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. *Foundations of machine learning*. MIT press, 2012.

Montgomery, W. H. and Levine, S. Guided policy search via approximate mirror descent. In *Advances in Neural Information Processing Systems*, pp. 4008–4016, 2016.

Munos, R. Error bounds for approximate policy iteration. In *ICML*, volume 3, pp. 560–567, 2003.

Munos, R. Performance bounds in l_p-norm for approximate value iteration. *SIAM journal on control and optimization*, 46(2): 541–561, 2007.

Munos, R. and Szepesvári, C. Finite-time bounds for fitted value iteration. *Journal of Machine Learning Research*, 9(May):815–857, 2008.

Nemirovsky, A. S. and Yudin, D. B. *Problem complexity and method efficiency in optimization.* Wiley, 1983.

Oh, J., Guo, Y., Singh, S., and Lee, H. Self-imitation learning. In *International Conference on Machine Learning*, 2018.

Ono, M., Pavone, M., Kuwata, Y., and Balaram, J. Chance-constrained dynamic programming with application to risk-aware robotic space exploration. *Autonomous Robots*, 39(4): 555–571, 2015.

Ormoneit, D. and Sen, Ś. Kernel-based reinforcement learning. *Machine learning*, 49(2-3):161–178, 2002.

Pietquin, O., Geist, M., Chandramohan, S., and Frezza-Buet, H. Sample-efficient batch reinforcement learning for dialogue management optimization. *ACM Transactions on Speech and Language Processing (TSLP)*, 7(3):7, 2011.

Prasad, N., Cheng, L.-F., Chivers, C., Draugelis, M., and Engelhardt, B. E. A reinforcement learning approach to weaning of mechanical ventilation in intensive care units. *arXiv preprint arXiv:1704.06300*, 2017.

Precup, D., Sutton, R. S., and Singh, S. P. Eligibility traces for off-policy policy evaluation. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pp. 759–766. Morgan Kaufmann Publishers Inc., 2000.

Precup, D., Sutton, R. S., and Dasgupta, S. Off-policy temporal difference learning with function approximation. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pp. 417–424. Morgan Kaufmann Publishers Inc., 2001.

Riedmiller, M. Neural fitted q iteration–first experiences with a data efficient neural reinforcement learning method. In *European Conference on Machine Learning*, pp. 317–328. Springer, 2005.

Riedmiller, M., Gabel, T., Hafner, R., and Lange, S. Reinforcement learning for robot soccer. *Autonomous Robots*, 27(1):55–73, 2009.

Roijers, D. M., Vamplew, P., Whiteson, S., and Dazeley, R. A survey of multi-objective sequential decision-making. *Journal of Artificial Intelligence Research*, 48:67–113, 2013.

Ross, S. and Bagnell, J. A. Reinforcement and imitation learning via interactive no-regret learning. *arXiv preprint arXiv:1406.5979*, 2014.

Schulman, J., Levine, S., Abbeel, P., Jordan, M., and Moritz, P. Trust region policy optimization. In *International Conference on Machine Learning*, pp. 1889–1897, 2015.

Shalev-Shwartz, S. et al. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4 (2):107–194, 2012.

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

Swaminathan, A. and Joachims, T. Batch learning from logged bandit feedback through counterfactual risk minimization. *Journal of Machine Learning Research*, 16(1):1731–1755, 2015.

Thomas, P. and Brunskill, E. Data-efficient off-policy policy evaluation for reinforcement learning. In *International Conference on Machine Learning*, pp. 2139–2148, 2016.

Van Hasselt, H., Guez, A., and Silver, D. Deep reinforcement learning with double q-learning. In *AAAI*, volume 2, pp. 5. Phoenix, AZ, 2016.

Van Moffaert, K. and Nowé, A. Multi-objective reinforcement learning using sets of pareto dominating policies. *The Journal of Machine Learning Research*, 15(1):3483–3512, 2014.

Von Neumann, J. and Morgenstern, O. *Theory of games and economic behavior (commemorative edition).* Princeton university press, 2007.

Ziebart, B. D. *Modeling purposeful adaptive behavior with the principle of maximum causal entropy*. PhD thesis, CMU, 2010.

Ziebart, B. D., Maas, A. L., Bagnell, J. A., and Dey, A. K. Maximum entropy inverse reinforcement learning. In *AAAI*, volume 8, pp. 1433–1438. Chicago, IL, USA, 2008.

Zinkevich, M. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pp. 928–936, 2003.