# Explaining Fast Improvement in Online Imitation Learning

**Xinyan Yan**[1] **Byron Boots**[2] **Ching-An Cheng**[3]

[1]Aurora Innovation Inc., Pittsburgh, PA
[2]School of Computer Science and Engineering, University of Washington, Seattle, WA
[3]Microsoft Research, Redmond, WA

## Abstract

Online Imitation Learning (IL) is an algorithmic framework that leverages interactions with expert policies for efficient policy optimization. Here policies are optimized by performing online learning on a sequence of loss functions that encourage the learner to mimic expert actions. If the online learning algorithm has no regret, then the agent can provably learn an expert-like policy. Online IL has demonstrated empirical successes in many applications and interestingly, its policy improvement speed observed in practice is usually much faster than existing theory suggests. In this work, we provide an explanation of this phenomenon. Let $\xi$ denote the policy class bias and assume the online IL loss functions are convex, smooth, and non-negative. We prove that, after $N$ rounds of online IL with stochastic feedback, the policy improves in $\tilde{O}(1/N + \sqrt{\xi/N})$ in both expectation and high probability. In other words, we show that adopting a sufficiently expressive policy class in online IL has two benefits: both the policy improvement speed increases and the performance bias decreases.

## 1 INTRODUCTION

Imitation Learning (IL) is a framework for improving the sample efficiency of policy optimization in sequential decision making. Unlike reinforcement learning (RL) algorithms that optimize policies purely by trial-and-error, IL leverages *expert policies* during training to provide extra feedback to aid policy search (e.g., in the form of supervisory loss functions). These expert policies can represent human demonstrators or resource-intensive engineered solutions which achieve non-trivial performance in the problem domain. By following the guidance of an expert policy, the learner can avoid blindly exploring the problem space and focus on promising directions that lead to expert-like behaviors, so the learning becomes sample efficient.

Online IL, pioneered by Ross et al. [2011], is a well-known algorithm that exploits such expert policies. Given the ability to interact with an expert policy, online IL reduces policy optimization to no-regret online learning [Hazan et al., 2016], for which effective algorithms have been developed. The main idea of online IL is to design an *online learning problem*[1] in which *1)* the decision set is identified with the policy class in the original policy optimization problem; and *2)* the online loss functions are set to encourage the learner to take expert-like actions under its own state distribution, which resemble *a sequence of supervised learning problems*. When these two conditions are met, the reduction follows: the regret rate and the minimum cumulative loss witnessed in the online learning problem determine the *learning speed* and the *performance bias* in the original policy optimization problem, respectively.

Since the work by Ross et al. [2011] was published, significant progress has been made in further developing theory, algorithms, and practical applications of this approach. For certain problems, online IL can learn the optimal policy exponentially faster than any RL algorithm when the expert policy is optimal [Sun et al., 2017]. Several IL algorithms developed recently are based on online IL, such as $f$-divergence minimization [Ke et al., 2019, Ghasemipour et al., 2020] and Integral Probability Metrics minimization [Swamy et al., 2021]. Furthermore, online IL has been validated in a diverse range of real-world applications, including agile off-road driving [Pan et al., 2018], quadrupedal locomotion [Lee et al., 2020], vision-and-language navigation [Anderson et al., 2018], and intelligent edge computing [Ning et al., 2020]. Beyond typical IL scenarios in which the goal is to imitate expert actions, the main prin-

[1]The online decision in the iterative process of online learning should not be confused with the decisions made at each time step in sequential decision making.

*Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence* (UAI 2021), PMLR 161:1874–1884.

ciple of online IL has also been applied to design algorithms for system identification [Venkatraman et al., 2014], model-based RL [Ross and Bagnell, 2012], structured prediction [Ross and Bagnell, 2014, Chang et al., 2015, Sun et al., 2017], and combinatorial search [Song et al., 2018]. Here we collectively call these algorithms *online IL*, since they adopt the same reduction idea and mainly differ in the way the expert policy is defined.

Despite the success of online IL, there is a mismatch between provable theoretical guarantees and the learning phenomenon observed in practice. Because of the design constraint imposed on the online losses mentioned above, the online losses used in the online IL reduction are not fully adversarial, but generated by samples of a sequence of probability distributions that vary slowly as the learner updates its policy [Cheng and Boots., 2018]. This structure makes the performance guarantee given by the classic adversary-style analysis of the regret rate taken by Ross et al. [2011] overly conservative, and motivates a deeper study on theoretical underpinnings of online IL [Cheng and Boots., 2018, Cheng et al., 2019b,a, Lee et al., 2019].

In this work, we are interested in explaining the fast policy improvement of online IL that is observed in practice but not captured by existing theory. When the online loss functions are convex and Lipschitz, typical analyses of regret and martingales [Ross et al., 2011, Cesa-Bianchi et al., 2004] suggest an on-average convergence rate in $O(1/\sqrt{N})$ after $N$ rounds. However, empirically, online IL algorithms learn much faster; e.g., the online IL algorithm DAgger [Ross et al., 2011] learned to mimic a model predictive control policy for autonomous off-road driving in only three rounds in [Pan et al., 2018]. Although the convergence rate improves to $\tilde{O}(1/N)$ when the online losses are strongly convex [Cheng and Boots., 2018], this condition can be difficult to satisfy especially when the policy class is large, such as a linear function class built on high-dimensional features. The empirical effectiveness and sample efficiency of online IL demand alternative explanations.

In this work, we bring a new perspective on the efficacy of online IL: even when learning from convex (but not strongly convex) sampled online losses, the learner in online IL can actually achieve a $\tilde{O}(1/N)$-like rate, because the consistency that the expert to imitate is fixed across different rounds provides a stability effect to learning. Formally, we prove a new bias-dependent convergence rate for online IL that is adaptive to the performance of the best policy in the policy class on the sequence of sampled losses. Interestingly, this new rate shows that an online IL algorithm can learn faster as this performance bias becomes smaller. In other words, adopting a sufficiently expressive policy class in online IL has two benefits: as the policy class becomes *reasonably* but not overly rich, both the *learning speed* increases and the *performance bias* decreases.

Concretely, suppose that the losses in online IL are convex, smooth, and non-negative, which, e.g., includes learning linear policies with quadratic losses as commonly used in continuous control problems. Let $\xi$ denote the policy class bias, which measures the performance of the best policy in the policy class on the sequence of imitation losses. We give a convergence rate in $\tilde{O}(1/N + \sqrt{\xi/N})$ both in expectation and in high probability for online IL algorithms using stochastic feedback. This new result shows a transition from the faster rate of $\tilde{O}(1/N)$ to the usual rate of $\tilde{O}(1/\sqrt{N})$ as the policy class bias $\xi$ increases.

This type of bias-dependent or optimistic convergence rate has been studied in typical machine learning settings, e.g., statistical learning [Srebro et al., 2010, Theorem 1], stochastic convex optimization [Zhang et al., 2017, Liu et al., 2018], and online learning [Srebro et al., 2010, Theorem 2], [Orabona, 2019, Theorem 4.21]. In fact, our new rate in expectation for online IL can be treated, from a technical viewpoint, as a direct consequence of the bias-dependent bound in the online learning literature. However, deriving such a new rate also in *high probability* requires extra technicalities, because the losses in online IL mix non-stationarity and stochasticity together; indeed, previous analyses tackle only one of these two properties and a straightforward combination does not lead to the fast rate desired here (cf. Section 3.3). To prove the desired fast high-probability bound, we propose a new regret decomposition technique for analyzing online IL and leverage a recent martingale concentration result based on path-wise statistics [Rakhlin and Sridharan, 2015, Theorem 3].

We conclude by corroborating the new theoretical findings with experimental results of online IL. The detailed proofs for this paper can be found in the Appendix.

## 2 BACKGROUND: ONLINE IL

### 2.1 POLICY OPTIMIZATION

The objective of policy optimization is to find a high-performance policy in a policy class $\Pi$ for sequential decision making problems. Typically, it models the world as a Markov decision process (MDP), defined by an initial state distribution, transition dynamics, and an instantaneous state-action cost function [Puterman, 2014]. This MDP is often assumed to be unknown to the learning agent; therefore the learning algorithm for policy optimization needs to perform systematic exploration in order to discover good policies in $\Pi$. Concretely, let us consider a policy class $\Pi$ that has a one-to-one mapping to a parameter space $\Theta$, and let $\pi_\theta$ denote the policy associated with the parameter $\theta \in \Theta$. That is, $\Pi = \{\pi_\theta : \theta \in \Theta\}$. The goal of policy optimization is to find a policy $\pi_\theta \in \Pi$ that minimizes the expected cost,

$$J(\pi_\theta) := \mathbb{E}_{s \sim d_{\pi_\theta}} \mathbb{E}_{a \sim \pi_\theta}[c(s, a)], \tag{1}$$

where $s$ and $a$ are the state and the action, respectively, $c$ is the instantaneous cost function and $d_{\pi_\theta}$ denotes the *average state distribution* over the problem horizon induced by executing policy $\pi_\theta$ starting from a state sampled from the initial state distribution. The problem formulation in (1) applies to various settings of problem horizon and discount rate, where the main difference is how the average state distribution is defined; e.g., for a discounted problem, $d_{\pi_\theta}$ is defined by a geometric mean, whereas $d_{\pi_\theta}$ is the stationary state distribution for average infinite-horizon problems.

## 2.2 ONLINE IL ALGORITHMS

Online imitation learning (IL) is a policy optimization technique that leverages interactive experts to efficiently find good policies. It devises a sequence of online loss functions $l_n$ such that *no regret* and *small policy class bias* imply good policy performance in the original sequential decision problem. Concretely, let $\pi_{\mathrm{e}}$ be an interactive expert policy. Instead of minimizing (1) directly, online IL minimizes a surrogate objective that upper bounds the performance difference between the policy $\pi_\theta$ and the expert $\pi_{\mathrm{e}}$:

$$J(\pi_\theta) - J(\pi_{\mathrm{e}}) \le O\Big(\underbrace{\mathbb{E}_{s\sim d_{\pi_\theta}}\mathbb{E}_{a\sim\pi_\theta}[D_{\pi_{\mathrm{e}}}(s,a)]}_{\text{surrogate objective}}\Big), \quad (2)$$

where the function $D_{\pi_{\mathrm{e}}}(s,a)$ represents how similar an action $a$ is to the action taken by expert policy $\pi_{\mathrm{e}}$ at state $s$, measured by statistical distances (e.g., Wasserstein distance and KL divergence) or their upper bounds [Ross et al., 2011, Ross and Bagnell, 2014, Sun et al., 2017].

Although the surrogate objective in (2) resembles (1) (i.e., by replacing $D_{\pi_{\mathrm{e}}}(s,a)$ with $c(s,a)$), the surrogate objective has an additional critical property that its range is normalized [Cheng and Boots., 2018]: regardless of the definition of the cost function $c$ of the original sequential decision problem, if the policy class $\Pi$ has enough capacity to contain the expert policy $\pi_{\mathrm{e}}$, there is a policy $\pi_\theta \in \Pi$ such that, for *all* states,

$$\mathbb{E}_{a\sim\pi_\theta}[D_{\pi_{\mathrm{e}}}(s,a)] = 0. \quad (3)$$

Under the realizability assumption (3), online IL can minimize the surrogate function in (2) by solving an online learning problem: Let parametric space $\Theta$ be the decision set (i.e., the policy class) in online learning; it defines the online loss function in round $n$ as

$$l_n(\theta) = \mathbb{E}_{s\sim d_{\pi_{\theta_n}}}\mathbb{E}_{a\sim\pi_\theta}[D_{\pi_{\mathrm{e}}}(s,a)], \quad (4)$$

where $\theta_n \in \Theta$ is the online decision made by the online algorithm in round $n$.

The main benefit of this indirect iterative approach is that, compared with the surrogate function (2), the average state distribution $d_{\pi_{\theta_n}}$ in the online loss function (4) is not considered as a function of the policy parameter $\theta$, making the online loss function (4) the objective function of a supervised learning problem whose sampled gradient is less noisy than that of the surrogate problem in (2). Because of the realizability assumption (3), the influence of the policy parameter on the change of the average state distribution can be ignored here, and the average regret with respect to the online loss functions in (4) alone [Ross et al., 2011] can upper bound the surrogate function in (2).

**Algorithm 1:** Online Imitation Learning (IL)

**Input:** Initial policy $\pi_{\theta_1}$ and online algorithm $\mathcal{A}$
**Output:** The best policy in the sequence of policies $\{\pi_{\theta_n}\}_{n=1}^N$

```
Initialize 𝒜 with the initial policy π_θ1
for n from 1 to N do
    Design the online loss function l_n based on π_θn
    Execute π_θn in the MDP to gather samples
    Use the samples to build an estimate l̂_n of l_n such
      that for all θ, 𝔼[l̂_n(θ)] = l_n(θ)
    Pass the functional feedback l̂_n to 𝒜 and use the
      return of 𝒜 to update policy to π_θn+1
```

When the expert policy $\pi_{\mathrm{e}}$ is only *nearly realizable* by the policy class $\Pi$ (that is, (3) can only be satisfied up to a certain error), optimizing the policy with this online learning reduction would suffer from an extra performance bias due to using a limited policy class, as we will later discuss in Section 2.3.

**Summary** Online IL can be viewed as a meta algorithm shown in Algorithm 1, where we take into account that in practice the MDP is unknown and therefore the online loss function $l_n$ needs to be further approximated by finite samples as $\hat{l}_n$, such that $\forall\theta \in \Theta$, $\mathbb{E}[\hat{l}_n(\theta)] = l_n(\theta)$. Given an expert policy, it selects a surrogate function to satisfy conditions similar to (2) and (3) (or their approximations). Then a no-regret online learning algorithm $\mathcal{A}$ is used to optimize the policy with respect to the sampled online loss functions $\hat{l}_n$, generating a sequence of policies $\{\pi_{\theta_n}\}_{n=1}^N$. By this reduction, performance guarantees can be obtained for the best policy in this sequence.

**Online IL in General** Before proceeding we note that by following the online IL design protocol above, Algorithm 1 can be instantiated beyond the typical IL setup. By properly choosing the *definition* of expert policies, the online IL reduction can be used to efficiently solve model-based RL and system identification where the samples of the MDP transition dynamics are treated as experts demonstrations [Ross and Bagnell, 2012, Venkatraman et al., 2014], and structured prediction where expert state-action value functions measure how good an action is in the surrogate function in (2) [Ross and Bagnell, 2014, Sun et al., 2017]. Similar reduction ideas are also used in recent RL algorithms [Agarwal et al., 2019, Abbasi-Yadkori et al., 2019].

### 2.3 GUARANTEES OF ONLINE IL

Now that we have reviewed the algorithmic aspects of online IL, we give a brief tutorial of the theoretical foundation of online IL and the known convergence results, which show exactly how regret and policy class bias are related to the performance in the original policy optimization problem.

To this end, let us formally define *1)* the regret and *2)* the policy class bias. For a sequence of online loss functions $\{f_n\}_{n=1}^N$ and decisions $\{\theta_n\}_{n=1}^N$ in an online learning problem, we define the regret as

$$\text{Regret}(f_n) = \sum f_n(\theta_n) - \min_{\theta\in\Theta} \sum f_n(\theta). \qquad (5)$$

Note that, for brevity, the range in $\sum_{n=1}^N$ is omitted in (5) and we will continue to do so below as long as the range is clear from the context. In addition to the regret, we define two problem-dependent biases of the decision set $\Theta$ (the equivalence of the policy class $\Pi$).

**Definition 1** (Problem-dependent biases)**.** For the sampled loss functions $\{\hat{l}_n\}_{n=1}^N$ experienced by running Algorithm 1, we define $\hat{\epsilon} = \frac{1}{N}\min_{\theta\in\Theta}\sum \hat{l}_n(\theta)$ and $\epsilon = \frac{1}{N}\min_{\theta\in\Theta}\sum l_n(\theta)$, where for all $n$ and $\theta$, $l_n(\theta) = \mathbb{E}[\hat{l}_n(\theta)]$.

A typical online IL analysis uses the regret and the policy class biases $\epsilon$ and $\hat{\epsilon}$ to decompose the cumulative loss $\sum l_n(\theta_n)$ to provide policy performance guarantees. Specifically, define $\theta^\star \in \arg\min_{\theta\in\Theta}\sum l_n(\theta)$. By (5) and Definition 1, we can write

$$\sum l_n(\theta_n) = \text{Regret}(\hat{l}_n) + \left(\sum l_n(\theta_n) - \hat{l}_n(\theta_n)\right) + N\hat{\epsilon} \qquad (6)$$

$$\leq \text{Regret}(\hat{l}_n) + \left(\sum l_n(\theta_n) - \hat{l}_n(\theta_n)\right) + \left(\sum \hat{l}_n(\theta^\star) - l_n(\theta^\star)\right) + N\epsilon \qquad (7)$$

where, in both (6) and (7), the first term is the online learning regret, the middle term(s) are the generalization error(s), and the last term is the policy class bias.

Because the surrogate loss $l_n(\theta_n)$ in online IL provides an upper bound on the policy performance in the original sequential decision problem (see (2) and (4)), picking the best policy in a policy sequence $\{\pi_{\theta_n}\}_{n=1}^N$ with a small cumulative loss $\sum l_n(\theta_n)$ guarantees good performance.

In a nutshell, existing convergence results of online IL are applications of (6) and (7) with different upper bounds on the regret and the generalization errors [Ross et al., 2011, Ross and Bagnell, 2012, 2014, Sun et al., 2017]. For example, when the sampled loss functions $\hat{l}_n$ are bounded, the generalization error(s) (i.e., the middle term(s) in (6) and (7)) can be bounded by $\tilde{O}(\sqrt{N})$ with high probability by Azuma's inequality (see [Cesa-Bianchi et al., 2004] or [Hazan et al., 2016, Chapter 9]). Together with an $O(\sqrt{N})$ bound on the regret (which is standard for online convex losses) [Hazan et al., 2016, McMahan, 2017], it implies that the average performance $\frac{1}{N}\sum l_n(\theta_n)$ and the best performance $\min_n l_n(\theta_n)$ converge to $\hat{\epsilon}$ or $\epsilon$ at the speed of $\tilde{O}(1/\sqrt{N})$.

However, the rate above often does not explain the fast improvement of online IL observed in practice [Laskey et al., 2016, Sun et al., 2017, Pan et al., 2018, Cheng et al., 2018], as we will also show experimentally in Section 5. While faster rates in $\tilde{O}(1/N)$ was shown for strongly convex loss functions [Ross et al., 2011, Cheng and Boots., 2018], the strong convexity assumption usually does not hold; for example, the common setting of learning with a policy class $\Pi$ and squared losses can easily break the strong convexity assumption, when the state samples are not diverse enough or when the feature dimension is high. Thus, alternative explanations are needed.

## 3 NEW BIAS-DEPENDENT RATES

In this section, we present new policy convergence rates that are adaptive to the performance biases in Definition 1. The full proofs of these theorems is provided in the Appendix.

### 3.1 SETUP AND ASSUMPTIONS

We suppose the parameter space of the policy class $\Theta$ is a closed convex subset of a Hilbert space $\mathcal{H}$ that is equipped with norm $\|\cdot\|$. Since $\|\cdot\|$ is not necessarily the norm induced by the inner product, we denote its dual norm by $\|\cdot\|_*$, which is defined as $\|x\|_* = \max_{\|y\|=1}\langle x, y\rangle$.

We define admissible algorithms to broaden the scope of online IL algorithms that our analysis covers.

**Definition 2** (Admissible online algorithm)**.** We say an online algorithm $\mathcal{A}$ is *admissible* for a parameter space $\Theta$ if there exists $R_{\mathcal{A}} \in [0,\infty)$ such that given any $\eta > 0$ and any sequence of differentiable convex functions $f_n$, $\mathcal{A}$ can achieve $\text{Regret}(f_n) \leq \text{Regret}(\langle\nabla f_n(\theta_n), \cdot\rangle) \leq \frac{1}{\eta}R_{\mathcal{A}}^2 + \frac{\eta}{2}\sum\|\nabla f_n(\theta_n)\|_*^2$, where $\theta_n$ is the decision made by $\mathcal{A}$ in round $n$.

We assume that Algorithm 1 is realized by an admissible online learning algorithm $\mathcal{A}$. This assumption is satisfied by common online algorithms, such as mirror descent [Nemirovski et al., 2009] and Follow-The-Regularized-Leader [McMahan, 2017], where $\eta$ in Definition 2 corresponds to a constant stepsize that is chosen before seeing the online losses, and $R_{\mathcal{A}}$ measures that size of the decision set $\Theta$.

Finally, we formally define convex, smooth, and non-negative (CSN) functions; we will assume the online loss $l_n$ in online IL and its sampled version $\hat{l}_n$ belong to this class.

**Definition 3** (Convex, smooth, and non-negative (CSN) function)**.** A function $f : \mathcal{H} \to \mathbb{R}$ is *CSN* if $f$ on $\mathcal{H}$ is convex, $\beta$-smooth[2], and non-negative.

Several popular loss functions used in online IL (e.g., squared $\ell_2$-loss and KL-divergence) are indeed CSN (Definition 3) (see Section 4 for examples). If the losses are not smooth, several smoothing techniques in the optimization literature are available to smooth the losses locally, e.g., Nesterov's smoothing [Nesterov, 2005], Moreau-Yosida regularization [Lemaréchal and Sagastizábal, 1997], and randomized smoothing [Duchi et al., 2012].

### 3.2 RATE IN EXPECTATION

Our first contribution is a non-asymptotic bias-dependent convergence rate in expectation by analyzing the online regret and the generalization error in the decomposition in (6) individually. First, under the assumption that sampled losses are CSN (Definition 3) and the online algorithm is admissible (Definition 2), the online regret can be bounded by extending the bias-dependent regret bound stated for mirror descent [Srebro et al., 2010, Theorem 2]. Second, because the generalization error is a martingale difference sequence, it vanishes in expectation.

**Theorem 1.** *In Algorithm 1, suppose $\hat{l}_n$ is CSN and $\mathcal{A}$ is admissible. Let $\hat{\epsilon} = \frac{1}{N}\min_{\theta\in\Theta}\sum \hat{l}_n(\theta)$ be the bias, and let $\hat{E}$ be an upper bound on $\hat{\epsilon}$. Choose the stepsize $\eta$ in $\mathcal{A}$ to be $\frac{1}{2\left(\beta+\sqrt{\beta^2+\frac{1}{2}\beta N\hat{E}R_{\mathcal{A}}^{-2}}\right)}$. Then it holds that*

$$\mathbb{E}\left[\frac{1}{N}\sum l_n(\theta_n) - \hat{\epsilon}\right] \leq \frac{8\beta R_{\mathcal{A}}^2}{N} + \sqrt{\frac{8\beta R_{\mathcal{A}}^2\hat{E}}{N}} \qquad (8)$$

The rate in (8) suggests that an online IL algorithm can learn faster as the policy class bias becomes smaller; this is reflected in the transition from the usual rate $O(1/\sqrt{N})$ to the faster rate $O(1/N)$ when the bias goes to zero. Notably, the rate in (8) does not depend on the dimensionality of $\mathcal{H}$ but only on $R_{\mathcal{A}}$, which one can roughly think of as the largest norm in $\Theta$. Therefore, we can increase the dimension of the policy class to reduce the bias (e.g., by using reproducing kernels [Hofmann et al., 2008]) as long as the diameter of $\Theta$ measured by norm $\|\cdot\|$ (e.g., $\ell_2$-norm) stays controlled.

Although the proof of Theorem 1 is a straightforward extension of the existing bias-dependent regret bounds from online learning literature, Theorem 1 brings a new perspective of online IL, which better explains its fast improvement and suggests directions for designing new algorithms that learn faster. As shown in Theorem 1, the policy learning speed in online IL can be closely connected to the policy class biases in Definition 1 which have been used in the online IL literature as a measure of expressivity.

Importantly, unlike in adversarial online learning, the biases $\hat{\epsilon}$ and $\epsilon$ in online IL is not arbitrarily large, but of constant sizes in most applications. For example, consider a popular application of online IL–learning-to-search by imitating a *deterministic* algorithmic expert $\pi_{\mathrm{e}}$ [Bhardwaj et al., 2017]. Here the goal is to learn a computationally efficient policy in place of the expert policy that relies on intensive computation or information unavailable at test time (e.g., the expert can be a brute-force search algorithm). In each round of learning, a problem instance is drawn from a distribution of problems, and the learner would query for the expert's advice for the state it visits in the sampled problem. If we consider a deterministic learner policy $\pi_{\theta_n}$ parameterized by $\theta_n \in \Theta$, the sampled online loss can be set as $\hat{l}_n(\theta_n) = (\pi_{\theta_n}(s_n) - \pi_{\mathrm{e}}(s_n))^2$, where $s_n$ is the sampled state visited by the learner in round $n$. In these problems, the stochasticity comes from sampling problem instances and the learner's states. But when the expert policy is contained in the class of approximators, there is some $\theta^* \in \Theta$ such that $\hat{l}_n(\theta^*) = 0$ *simultaneously* for all $n$ and all samples, i.e., $\epsilon = \hat{\epsilon} = 0$. Generally, one can show that $\epsilon$ and $\hat{\epsilon}$ are at most the losses incurred by the expert policy plus some distance between the expert and the approximator class. Therefore, our results show that online IL in most useful cases roughly has a $O(1/N)$ rate.

**Online IL with Adaptive Stepsizes** In Theorem 1, the bias-dependent rate (8) holds when the stepsize of the admissible online learning algorithm $\mathcal{A}$ is appropriately tuned. While this seems to be a limitation of Theorem 1, one can show that the rate (8) still holds if the stepsizes are properly adapted online (e.g., using an AdaGrad rule [Duchi et al., 2011] $\eta_n = \frac{R_{\mathcal{A}}}{2\sqrt{\sum_{i=1}^n \|\nabla f_n(\theta_n)\|_*^2}}$) without knowing the constants $\beta, N, \hat{E}$. This is because an online algorithm with adaptive stepsizes can obtain almost the same regret guarantee as an algorithm that would know the optimal constant stepsize in advance. Furthermore, the high-probability bias-dependent rate in Theorem 2 that will be presented in Section 3.3 can also be extended to adaptive stepsizes. Please find details in Appendix D.

### 3.3 RATE IN HIGH PROBABILITY

Next we show that a similar *non-asymptotic*[3] bias-dependent convergence rate to the rate (8) also holds in high probability.

[2]A function $f$ is $\beta$-smooth if its gradient satisfy $\|\nabla f(x) - \nabla f(y)\|_* \leq \beta\|x-y\|$ for $x, y \in \mathcal{H}$.

[3]For compactness, we use the big-O notation to hide the constants in the rate; the exact constants can be found in Appendix C.

**Theorem 2.** *Under the same assumptions and setup of Theorem 1, further assume that there is $G \in [0, \infty)$ such that, for any $\theta \in \Theta$, $\|\nabla \hat{l}_n(\theta)\|_* \leq G$. For any $\delta < 1/e$, with probability at least $1 - \delta$, the following holds*

$$\frac{1}{N} \sum l_n(\theta_n) - \epsilon = O\left(\frac{C\beta R^2}{N} + \sqrt{\frac{C\beta R^2(\hat{E}+\epsilon)}{N}}\right) \quad (9)$$

*where* $R_\Theta = \max_{\theta \in \Theta} \|\theta\|, R = \max(1, R_\Theta, R_\mathcal{A}), C = \log(1/\delta)\log(GRN)$.

We remark that the uniform bound $G$ on the norm of the gradients only appears in logarithmic terms. Therefore, this rate stays reasonable when the loss functions have gradients whose norm grows with the size of $\Theta$, such as the popular squared loss.

To prove Theorem 2, one may attempt to build on top of the proof of Theorem 1 by applying basic martingale concentration properties on the martingale difference sequences (MDSs) in (6), or devise a similar scheme for (7). But taking this direct approach will bring back the usual rate of $O(1/\sqrt{N})$. To the best of our knowledge, sharp concentration inequalities for the counterparts of MDS in other learning settings cannot be adapted here in a straightforward way. Srebro et al. [2010, Theorem 1] prove a fast rate for empirical risk minimizer (ERM) in statistical learning. However, their proof is based on local Rademacher complexities, which do not have obvious extension to non-stationary online losses. Zhang et al. [2017] extend the results of Srebro et al. [2010] to stochastic convex optimization, but the extension relies on an i.i.d. concentration lemma.[4] Kakade and Tewari [2009] show fast converging excess risk of online convex programming algorithms when the loss function is Lipschitz and strongly convex; relaxing the strong convexity assumptions is the goal of this work.

**Convexity Assumption** Both Theorem 1 and Theorem 2 require that the sampled online losses $\hat{l}_n$ are convex. *1)* On one hand, this convexity assumption appears to be restrictive, because our results cannot explain learning with generic neural networks. Nonetheless, expressive linear policy classes that meet the convexity assumption still include many useful cases (such as RKHS [Hofmann et al., 2008] and rich feature sets). Furthermore, convexity has been a central assumption in almost all online learning paradigms; we did not attempt to address this limitation in this work. *2)* On the other hand, the convexity assumption relaxes the strong convexity assumption needed in the online IL literature [Ross et al., 2011, Cheng and Boots., 2018]. This relaxation is important when the number of samples is smaller than the number of policy parameters or when the samples are not diverse enough. In those cases, even if the expected losses $l_n$ are strongly convex, the sampled losses $\hat{l}_n$ may not be strongly convex because the Hessian matrix is singular.

[4] The i.i.d. concentration lemma further depends on a martingale concentration bound [Pinelis, 1994, Theorem 3.4], which relies on an almost-surely upper bound of the second-order statistics. In comparison, the martingale concentration we utilize in this work relies on second-order statistics that are defined on the sample path [Rakhlin and Sridharan, 2015, Theorem 3].

**Related Work in Learning** Similar bias-dependent or optimistic rates have been studied extensively in several more typical learning settings such as contextual bandits [Allen-Zhu et al., 2018], statistical learning [Panchenko et al., 2002, Srebro et al., 2010, Zhang et al., 2017, Liu et al., 2018], online learning with adversarial loss sequences [Srebro et al., 2010, Orabona et al., 2012], and online-to-batch conversion [Littlestone, 1990, Cesa-Bianchi et al., 2004]. Table 1 summarizes these different learning setups. In contrast to bandit settings that focus on discrete actions or simplex geometry, online IL usually leads to online convex losses, a general compact convex decision set, and stochastic functional or gradient feedback. Compared to statistical and online learning, online IL concerns loss functions that are both stochastic and online; we can view statistical and online learning as special cases of online IL. The interactions between noises and non-stationarity make the analysis of online IL especially interesting.

**Specialization to Stochastic Convex Optimization** Because of the generality of the online IL, an online IL algorithm (Algorithm 1) running on a *stationary* loss function can serve as a one-pass learning algorithm for stochastic optimization; that is, we have $l_n = l$ for some $l$ for all $n$; By specializing Theorem 1 and Theorem 2 to the stochastic optimization setting, we can recover the existing bounds in the stochastic optimization literature, i.e., Corollary 3 and Theorem 1 in [Srebro et al., 2010], respectively. These special cases can be derived in a straightforward manner due to the relationship between $\hat{\epsilon}$ and $\epsilon$ when the loss function is fixed (i.e., $l_n = l$): *1)* $\mathbb{E}[\epsilon] = \mathbb{E}[\hat{\epsilon}]$, and *2)* in high probability, $\hat{\epsilon} - \epsilon \leq O(\sqrt{1/N})$. However, we note that for general online IL problems, the sizes of $\hat{\epsilon}$ and $\epsilon$ are not comparable and $\mathbb{E}[\epsilon] \neq \mathbb{E}[\hat{\epsilon}]$.

## 3.4 PROOF SKETCH FOR THEOREM 2

We take a different decomposition of the cumulative loss to avoid the usual $O(1/\sqrt{N})$ rate originating from applying martingale analyses on the MDSs in (6) and (7). Here we construct two *new* MDSs in terms of the gradients: recall $\epsilon = \min_{\theta \in \Theta} \sum l_n(\theta)$ and let $\theta^\star = \arg\min_{\theta \in \Theta} \sum l_n(\theta)$.

Table 1: Comparison of different learning settings. Info.: Information about the loss function available to the learning algorithm in each round. ERM: Empirical risk minimization. Partial FB: Partial feedback.

| Setting | Info. | Stochastic | Non-stationary | Partial FB | Estimator | Excess loss to minimize |
|---|---|---|---|---|---|---|
| Online IL (this work) | $\hat{l}_n$ | Yes | Yes | No | Online | $\sum l_n(\theta_n) - \min \sum l_n(\theta)$ |
| Stochastic bandits | $\hat{l}(\theta_n)$ | Yes | No | Yes | Online | $\sum l(\theta_n) - N \min l(\theta)$ |
| Online learning | $l_n$ | No | Yes | No | Online | $\sum l_n(\theta_n) - \min \sum l_n(\theta)$ |
| Statistical learning | $\hat{l}$ | Yes | No | No | ERM | $l(\theta_{\mathrm{ERM}}) - \min l(\theta)$ |
| Online-to-batch | $\hat{l}$ | Yes | No | No | Online | $\sum l(\theta_n) - N \min l(\theta)$ |

Then by convexity of $l_n$, we can derive

$$\begin{aligned}
&\sum l_n(\theta_n) - N\epsilon \\
&\leq \sum \underbrace{\langle \nabla l_n(\theta_n) - \nabla \hat{l}_n(\theta_n), \theta_n \rangle}_{\text{MDS}} - \qquad (10) \\
&\quad \sum \langle \underbrace{\nabla l_n(\theta_n) - \nabla \hat{l}_n(\theta_n)}_{\text{MDS}}, \theta^\star \rangle + \text{Regret}(\langle \nabla \hat{l}_n(\theta_n), \cdot \rangle)
\end{aligned}$$

Our proof is based on analyzing these three terms. For the MDSs in (10), we notice that, for smooth and non-negative functions, the squared norm of the gradients can be bounded by its function value.

**Lemma 1** (Lemma 3.1 [Srebro et al., 2010])**.** *Suppose a function $f : \mathcal{H} \to \mathbb{R}$ is $\beta$-smooth and non-negative, then for any $x \in \mathcal{H}$, $\|\nabla f(x)\|_*^2 \leq 4\beta f(x)$.*

Lemma 1 enables us to properly control the second-order statistics of the MDSs in (10). By a recent vector-valued martingale concentration inequality that depends only on second-order statistics [Rakhlin and Sridharan, 2015, Theorem 3], we obtain a self-bounding property for (10) to get fast concentration rate.

Besides analyzing the MDSs, we need to bound the regret to the linear functions defined by the gradients (the last term in (10)). Since this last term is linear, not CSN, the bias-dependent online regret in the proof of Theorem 1 does not apply. Nonetheless, because these linear functions are based on the gradients of CSN functions, we discover that their regret rate actually obeys the exact same rate as the regret to the CSN loss functions. This is notable because the regret to these linear functions upper bounds the regret to the CSN loss functions.

Combining the bounds on the MDSs and the regret, we obtain the rate in (9).

# 4 CASE STUDIES

We use two concrete instantiations of the online IL algorithm (Algorithm 1) to show how the new theoretical results in Section 3 improve the existing understanding of the policy improvement speed in these algorithms.

## 4.1 IMITATION LEARNING

The work by Ross et al. [2011] on online IL has demonstrated successes in solving many real-world sequential decision making problems [Laskey et al., 2016, 2017, Pan et al., 2018]. When the action space is discrete, a popular design choice is to set $D_{\pi_e}(s, a)$ in (4) as the hinge loss [Ross et al., 2011]. For continuous domains, $\ell_1$-loss becomes a natural alternative for defining $D_{\pi_e}(s, a)$, which, e.g., is adopted by Pan et al. [2018] for autonomous driving. When the policy is linear in the parameters, one can verify that these loss functions are convex and non-negative, though not strongly convex. Therefore, existing theorems suggest only an $O(1/\sqrt{N})$ rate, which does not reflect the fast experimental rates [Ross et al., 2011, Pan et al., 2018].

Although our new theorems are not directly applicable to these non-smooth loss functions, they can be applied to a smoothed version of these non-negative convex loss functions. For instance, applying the Huber approximation (an instantiation of Nesterov's smoothing) [Nesterov, 2005] to "smooth the tip" of these $\ell_1$-like losses yields a globally smooth function with respect to the $\ell_2$-norm. As the smoothing mainly changes where the loss is close to zero, our new theorems suggest that, when the policy class is expressive enough, learning with these $\ell_1$-like losses would converge in a $\tilde{O}(1/N)$ rate before the policy gets very close to the expert policy during policy optimization.

## 4.2 INTERACTIVE SYSTEM ID FOR MODEL-BASED RL

Interactive system identification (ID) is a technique that interleaves data collection and dynamics model learning for robust model-based RL. Ross and Bagnell [2012] show that interactive system ID can be analyzed under the online IL framework, where the regret guarantee implies learning a dynamics model that mitigates the train-test distribution shift problem [Abbeel and Ng, 2005, Ross and Bagnell, 2012]. Let $T$ and $T_\theta$ denote the true and the learned transition dynamics, respectively. A common online loss for interactive system ID is $l_n(\theta) = \mathbb{E}_{(s,a)\sim \frac{1}{2}d_{T_{\theta_n}} + \frac{1}{2}\nu} [D_{s,a}(T_\theta || T)]$, where $D_{s,a}(T_\theta||T)$ is some distance between $T$ and $T_\theta$ under state $s$ and action $a$, $\nu$ is the state-action distribution of an exploration policy, and $d_{T_{\theta_n}}$ is the state-action distribution induced by running an optimal policy with respect to the model $T_{\theta_n}$. When the model class is expressive enough to contain the $T$, then $l_n(\theta) = 0$ holds for some $\theta \in \Theta$ (cf. (3)).

Suppose that the states and actions are continuous. A common choice for $D_{s,a}(T_\theta||T)$ in learning deterministic dynamics is the squared error $D_{s,a}(T_\theta||T) = \|T_\theta(s,a) - s'\|_2^2$ [Ross and Bagnell, 2012], where the $s'$ is the next state in the true transition of $T$. If $T_\theta$ is linear in $\theta$ or belongs to a reproducing kernel Hilbert space, the sampled loss function $\hat{l}_n$ is CSN. Alternatively, when learning a probabilistic model, $D_{s,a}$ can be selected as the KL-divergence [Ross and Bagnell, 2012]; it is known that if $T_\theta$ belongs to the exponential family of distributions, the KL divergence, and hence $\hat{l}_n$, are smooth and convex [Wainwright and Jordan, 2008]. If the sample size is large enough, $\hat{l}_n$ becomes non-negative in high probability.

As these online losses are CSN, our theoretical results apply and suggest a convergence rate in $\tilde{O}(1/N)$. On the contrary, the finite sample analysis conducted in [Ross and Bagnell, 2012] uses the standard online-to-batch techniques [Cesa-Bianchi et al., 2004] and can only give a rate of $O(1/\sqrt{N})$. Our new results provide a better explanation to justify the fast policy improvement speed observed empirically, e.g., [Ross and Bagnell, 2012, Figure 2].

# 5 EXPERIMENTAL RESULTS

Although the main focus of this paper is the new theoretical insights, we conduct experiments to provide evidence that the fast policy improvement phenomenon indeed exists, as our theory predicts. We verify the change of the policy improvement rate due to policy class capacity by running an imitation learning experiment in a simulated CartPole balancing task. Details can be found in Appendix E.

**MDP Setup** The goal of the CartPole task is to keep the pole upright by controlling the acceleration of the cart. The start state is a configuration with a small uniformly sampled offset from being static and vertical, and the dynamics is deterministic. In each time step, if the pole is maintained within a threshold from being upright, the learner receives an instantaneous reward of one; otherwise, the learner receives zero rewards and the episode terminates. This MDP has a 4-dimensional continuous state space and a 1-dimensional continuous action space.

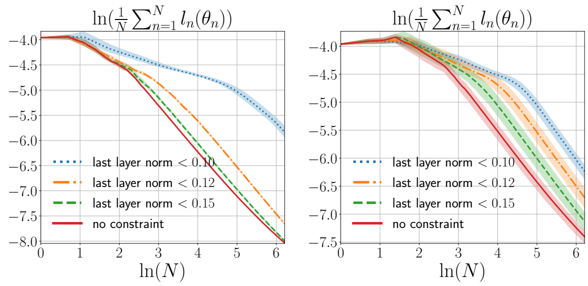
**Expert and Learner Policies** We use a neural network expert policy (with one hidden layer of 64 units and tanh activation) which is trained using policy gradient with GAE [Schulman et al., 2015] and ADAM [Kingma and Ba, 2014]. We let the learner policy be another neural network that shares the same architecture with the expert policy. When learning only the output layer, we copy the weights of the hidden layer from the expert policy and randomly initialize the weights of the output layer; we can view the learner as a *linear* policy using the representation of the expert policy. When learning the full network, we randomly initialize all the weights and biases.

**Online IL Setup** We emulate online IL with unbiased and biased policy classes. To define policy classes with different degrees of bias, we impose $\ell_2$-norm constraints of different sizes on the weights of the learner's output layer. To define the unbiased policy class, we lift this $\ell_2$-norm constraint. We select $l_n(\theta) = \mathbb{E}_{s\sim d_{\pi_{\theta_n}}}[H_\mu(\pi_\theta(s) - \pi_{\mathrm{e}}(s))]$ as the online loss in IL (see Section 2.2), where $H_\mu$ is the Huber function defined as $H_\mu(x) = \frac{1}{2}x^2$ for $|x| \leq \mu$ and $\mu|x| - \frac{1}{2}\mu^2$ for $|x| > \mu$. In the experiments, $\mu$ is set to $0.05$; as a result, $H_\mu$ is linear when its function value is larger than $0.00125$. In the setting of training only the output layer, because the learner's policy is linear, this online loss is CSN (Definition 3) in the unknown weights of the learner. We use AdaGrad [McMahan and Streeter, 2010, Duchi et al., 2011] to optimize the learner policy with constant stepsize $0.01$ and $500$ ($\ln 500 \approx 6.2$) iterations.

**Simulation Results** We compare the results in the unbiased and the biased settings, in terms of how the average loss $\frac{1}{N}\sum_{n=1}^N l_n(\theta_n)$ changes as the number of rounds $N$ in online learning increases. We impose $\ell_2$-norm constraints of sizes $\{0.1, 0.12, 0.15\}$ on the weights of the output layer to simulate biased policy classes. For comparison, when training the output layer, the $\ell_2$-norm of the final policy trained without the constraint is about $0.18$; when training the full network, it is about $0.23$. The experimental results are depicted in Fig. 1. To better visualize the rate of improvement, we plot both the $x$- and $y$-axis in log scale, so that the slope of the curves directly represents the rate: if the slope is $-1$, the rate is $O(1/N)$ and if the slope is $-\frac{1}{2}$ the rate is $O(1/\sqrt{N})$. In Fig. 1a, only the output layer of the learner policy is trained. In this setting, all the assumptions made in our theorems are satisfied. It can be seen that when using a larger norm constraint (i.e., smaller bias), the learner policy improvement becomes faster, moving towards $O(1/N)$. The curve with the constraint of $0.10$ in Fig. 1a gets a rate slightly faster $O(1/\sqrt{N})$, likely because the Huber loss is strongly convex near zero. Interestingly, Fig. 1b shows that this phenomenon happens also in training the full network, which does not meet the assumption required in the theory.

# 6 CONCLUSION

In this paper, we provide an explanation of the fast learning speed of online IL by proving new expected and high-probability convergence rates that depend on the policy class capacity. However, our current results do not explain all the




(a) learning output layer (b) learning full network

Figure 1: The convergence rate of online IL with different policy class biases, where the bias is defined as the $\ell_2$-norm constraint on the weights of the output layer. The curves are plotted using the median over 4 random seeds, and the shaded region represents $10\%$ and $90\%$ percentile.

fast improvements of online IL observed in practice. The analyses here are based on the assumption of using convex and smooth loss functions. This assumption would be violated, for example, with a deep neural network policy based on with ReLU activation; yet Pan et al. [2018] show fast empirical convergence rates of these networks in online IL. Nonetheless, we envision that the insights from this paper can provide a promising direction to better understanding the behaviors of online IL, and to suggest ways for designing new online IL algorithms that proactively leverage these self-bounding regret properties to achieve faster learning. For example, extending the analysis to nonconvex online IL loss functions by studying the convergence to the 2nd-order stationary points is an interesting direction for lifting the convexity assumption.

## References

Yasin Abbasi-Yadkori, Peter Bartlett, Kush Bhatia, Nevena Lazic, Csaba Szepesvari, and Gellért Weisz. Politex: Regret bounds for policy iteration using expert prediction. In *International Conference on Machine Learning*, pages 3692–3702. PMLR, 2019.

Pieter Abbeel and Andrew Y Ng. Exploration and apprenticeship learning in reinforcement learning. In *Proceedings of the 22nd international conference on Machine learning*, pages 1–8, 2005.

Alekh Agarwal, Sham M Kakade, Jason D Lee, and Gaurav Mahajan. On the theory of policy gradient methods: Optimality, approximation, and distribution shift. *arXiv preprint arXiv:1908.00261*, 2019.

Zeyuan Allen-Zhu, Sébastien Bubeck, and Yuanzhi Li. Make the minority great again: First-order regret bound for contextual bandits. In *International Conference on Machine Learning*, pages 186–194. PMLR, 2018.

Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton Van Den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3674–3683, 2018.

Mohak Bhardwaj, Sanjiban Choudhury, and Sebastian Scherer. Learning heuristic search via imitation. In *Conference on Robot Learning*, pages 271–280. PMLR, 2017.

Nicolo Cesa-Bianchi, Alex Conconi, and Claudio Gentile. On the generalization ability of on-line learning algorithms. *IEEE Transactions on Information Theory*, 50(9): 2050–2057, 2004.

Kai-Wei Chang, Akshay Krishnamurthy, Alekh Agarwal, John Langford, and Hal Daumé III. Learning to search better than your teacher. 2015.

Ching-An Cheng and Byron Boots. Convergence of value aggregagtion for imitation learning. In *Proceedings of the 21st International Conference on Artificial Intelligence and Statistics*, 2018.

Ching-An Cheng, Xinyan Yan, Nolan Wagener, and Byron Boots. Fast policy learning through imitation and reinforcement. In *Proceedings of the 34th Conference on Uncertanty in Artificial Intelligence*, pages 845–855, 2018.

Ching-An Cheng, Jonathan Lee, Ken Goldberg, and Byron Boots. Online learning with continuous variations: Dynamic regret and reductions. *arXiv preprint arXiv:1902.07286*, 2019a.

Ching-An Cheng, Xinyan Yan, Evangelos Theodorou, and Byron Boots. Accelerating imitation learning with predictive models. In *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*, 2019b.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

John C Duchi, Peter L Bartlett, and Martin J Wainwright. Randomized smoothing for stochastic optimization. *SIAM Journal on Optimization*, 22(2):674–701, 2012.

Seyed Kamyar Seyed Ghasemipour, Richard Zemel, and Shixiang Gu. A divergence minimization perspective on imitation learning methods. In *Conference on Robot Learning*, pages 1259–1277. PMLR, 2020.

Elad Hazan et al. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016.

Thomas Hofmann, Bernhard Schölkopf, and Alexander J Smola. Kernel methods in machine learning. *The annals of statistics*, pages 1171–1220, 2008.

Sham M Kakade and Ambuj Tewari. On the generalization ability of online strongly convex programming algorithms. In *Advances in Neural Information Processing Systems*, pages 801–808, 2009.

Liyiming Ke, Matt Barnes, Wen Sun, Gilwoo Lee, Sanjiban Choudhury, and Siddhartha Srinivasa. Imitation learning as $f$-divergence minimization. *arXiv preprint arXiv:1905.12888*, 2019.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Michael Laskey, Jonathan Lee, Caleb Chuck, David Gealy, Wesley Hsieh, Florian T Pokorny, Anca D Dragan, and Ken Goldberg. Robot grasping in clutter: Using a hierarchy of supervisors for learning from demonstrations. In *2016 IEEE International Conference on Automation Science and Engineering (CASE)*, pages 827–834. IEEE, 2016.

Michael Laskey, Jonathan Lee, Roy Fox, Anca Dragan, and Ken Goldberg. Dart: Noise injection for robust imitation learning. *arXiv preprint arXiv:1703.09327*, 2017.

Jonathan Lee, Ching-An Cheng, Ken Goldberg, and Byron Boots. Continuous online learning and new insights to online imitation learning. *arXiv preprint arXiv:1912.01261*, 2019.

Joonho Lee, Jemin Hwangbo, Lorenz Wellhausen, Vladlen Koltun, and Marco Hutter. Learning quadrupedal locomotion over challenging terrain. *Science robotics*, 5(47), 2020.

Claude Lemaréchal and Claudia Sagastizábal. Practical aspects of the moreau–yosida regularization: Theoretical preliminaries. *SIAM Journal on Optimization*, 7(2):367–385, 1997.

Nicholas Littlestone. Mistake bounds and logarithmic linear-threshold learning algorithms. 1990.

Mingrui Liu, Xiaoxuan Zhang, Lijun Zhang, Rong Jin, and Tianbao Yang. Fast rates of erm and stochastic approximation: Adaptive to error bound conditions. In *Advances in Neural Information Processing Systems*, pages 4678–4689, 2018.

H Brendan McMahan. A survey of algorithms and analysis for adaptive online learning. *The Journal of Machine Learning Research*, 18(1):3117–3166, 2017.

H Brendan McMahan and Matthew Streeter. Adaptive bound optimization for online convex optimization. *arXiv preprint arXiv:1002.4908*, 2010.

Arkadi Nemirovski, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on optimization*, 19(4):1574–1609, 2009.

Yu Nesterov. Smooth minimization of non-smooth functions. *Mathematical programming*, 103(1):127–152, 2005.

Zhaolong Ning, Kaiyuan Zhang, Xiaojie Wang, Lei Guo, Xiping Hu, Jun Huang, Bin Hu, and Ricky YK Kwok. Intelligent edge computing in internet of vehicles: a joint computation offloading and caching solution. *IEEE Transactions on Intelligent Transportation Systems*, 2020.

Francesco Orabona. A modern introduction to online learning. *arXiv preprint arXiv:1912.13213*, 2019.

Francesco Orabona, Nicolo Cesa-Bianchi, and Claudio Gentile. Beyond logarithmic bounds in online learning. In *Artificial Intelligence and Statistics*, pages 823–831, 2012.

Yunpeng Pan, Ching-An Cheng, Kamil Saigol, Keuntak Lee, Xinyan Yan, Evangelos Theodorou, and Byron Boots. Agile autonomous driving using end-to-end deep imitation learning. In *Robotics: science and systems*, 2018.

Dmitriy Panchenko et al. Some extensions of an inequality of vapnik and chervonenkis. *Electronic Communications in Probability*, 7:55–65, 2002.

Iosif Pinelis. Optimum bounds for the distributions of martingales in banach spaces. *The Annals of Probability*, pages 1679–1706, 1994.

Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

Alexander Rakhlin and Karthik Sridharan. On equivalence of martingale tail bounds and deterministic regret inequalities. *arXiv preprint arXiv:1510.03925*, 2015.

Stephane Ross and J Andrew Bagnell. Agnostic system identification for model-based reinforcement learning. *arXiv preprint arXiv:1203.1007*, 2012.

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

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 627–635, 2011.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015.

Jialin Song, Ravi Lanka, Albert Zhao, Aadyot Bhatnagar, Yisong Yue, and Masahiro Ono. Learning to search via retrospective imitation. *arXiv preprint arXiv:1804.00846*, 2018.

Nathan Srebro, Karthik Sridharan, and Ambuj Tewari. Smoothness, low noise and fast rates. In *Advances in neural information processing systems*, pages 2199–2207, 2010.

Wen Sun, Arun Venkatraman, Geoffrey J Gordon, Byron Boots, and J Andrew Bagnell. Deeply aggrevated: Differentiable imitation learning for sequential prediction. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3309–3318. JMLR. org, 2017.

Gokul Swamy, Sanjiban Choudhury, Zhiwei Steven Wu, and J Andrew Bagnell. Of moments and matching: Tradeoffs and treatments in imitation learning. *arXiv preprint arXiv:2103.03236*, 2021.

Arun Venkatraman, Byron Boots, Martial Hebert, and J Andrew Bagnell. Data as demonstrator with applications to system identification. In *ALR Workshop, NIPS*, 2014.

Martin J Wainwright and Michael I Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1-2):1–305, 2008.

Lijun Zhang, Tianbao Yang, and Rong Jin. Empirical risk minimization for stochastic convex optimization: O (1/n) -and o (1/n$\hat{2}$)-type of risk bounds. *arXiv preprint arXiv:1702.02030*, 2017.