Regret Guarantees for Online Deep Control

Xinyi Chen
Princeton University, Google AI Princeton

Edgar Minasyan
Princeton University, Google AI Princeton

Jason D. Lee
Princeton University, Google AI Princeton

Elad Hazan
Princeton University, Google AI Princeton

Editors: N. Matni, M. Morari, G. J. Pappas

Abstract

Despite the immense success of deep learning in reinforcement learning and control, few theoretical guarantees for neural networks exist for these problems. Deriving performance guarantees is challenging because control is an online problem with no distributional assumptions and an agnostic learning objective, while the theory of deep learning so far focuses on supervised learning with a fixed known training set.

In this work, we begin to resolve these challenges and derive the first regret guarantees in online control over a neural network-based policy class. In particular, we show sublinear episodic regret guarantees against a policy class parameterized by deep neural networks, a much richer class than previously considered linear policy parameterizations. Our results center on a reduction from online learning of neural networks to online convex optimization (OCO), and can use any OCO algorithm as a blackbox. Since online learning guarantees are inherently agnostic, we need to quantify the performance of the best policy in our policy class. To this end, we introduce the interpolation dimension, an expressivity metric, which we use to accompany our regret bounds. The results and findings in online deep learning are of independent interest and may have applications beyond online control.

1. Introduction

The use of deep neural networks has been highly successful in reinforcement learning (RL) and continuous control problems. However, a theory for deep control and RL remains challenging. The main difficulty in applying the theory developed for supervised learning to the RL domain is the distributional assumptions and realizability goal made in the literature thus far. In control and RL, the environment is inherently online and often nonstochastic, and the goal is usually agnostic learning with respect to a policy class.

In this work, we consider the problem of online episodic control with neural network-based policies. We begin to resolve the aforementioned challenges and derive the first regret bound guarantees in this setting. Provable regret bounds in this domain have thus far been limited to linear
controllers. However, most dynamical systems in the physical world are nonlinear and/or require nonlinear controls. An important tool that allows us to go beyond linear controllers is the emerging paradigm of online nonstochastic control: a methodology for control that is robust to adversarial noise in the dynamics. The important aspect of this paradigm to our study is that it uses policy classes that admit a convex parameterization.

It is natural to consider the online episodic control setting: although it is less challenging from a technical perspective than single-trajectory control, the policy learning procedure in empirical deep control is often done episodically as detailed in the related work section. The main technical challenge to this goal is formalizing online learning over deep neural networks and proving accompanying regret bounds. Given this result, an extension to the single-trajectory setting is possible.

As the major technical component of this work, we propose a black-box reduction from online deep learning to online convex optimization (OCO) that attains provable regret bounds. These bounds apply to the general online learning setting with vector output predictors and arbitrary convex loss functions. Moreover, the regret guarantees are naturally agnostic, i.e. they show performance competitive to the best neural network in our policy class in hindsight without assuming it achieves zero loss. To capture agnostic learning and derive meaningful guarantees for online learning and control, we also introduce a new metric of expressivity, namely the “interpolation dimension”, that accompanies our regret bounds.

An interesting conclusion from this reduction is the unifying view that provable convergence and/or generalization bounds for training deep neural networks can be derived for any OCO method, beyond online and stochastic gradient descent. This includes mirror descent, adaptive gradient methods, follow-the-perturbed leader and other algorithms. Previously, convergence and generalization analyses for neural networks were done in isolation for different optimization algorithms as detailed in the related work section.

Our contributions in this work can be summarized as follows:

• **Online episodic deep control:** We derive the first provable regret guarantees in online episodic control with policies based on deep neural networks. Furthermore, we demonstrate the richness of the considered policy class by showing that it can output the optimal open-loop control sequence of any single episode.

• **Online learning over neural networks:** We give a general reduction from online learning of neural networks to OCO that can use any OCO algorithm as a blackbox.

• **Interpolation dimension:** To state meaningful guarantees in online agnostic learning, we introduce the interpolation dimension as an expressivity metric. It is a fundamental notion and applies to any hypothesis class.

• **Unifying analysis:** Our proposed method applies to any OCO algorithm, including mirror descent and adaptive gradient methods widely used in deep learning. This leads to a unifying framework for optimization in deep learning: the online learning framework implies both convergence and generalization bounds in the supervised learning setting.

1.1. Related work

**Online and nonstochastic control.** Our study focuses on algorithms which enjoy sublinear regret for online control of dynamical systems; that is, whose performance tracks a given benchmark
of policies up to a term which is vanishing relative to the problem horizon. Abbasi-Yadkori and Szepesvári (2011) initiated the study of online control under the regret benchmark for linear time-invariant (LTI) dynamical systems. Bounds for this setting have since been improved and refined in Dean et al. (2018); Mania et al. (2019); Cohen et al. (2019); Simchowitz and Foster (2020). Our work instead adopts the online nonstochastic control setting (Agarwal et al., 2019), that allows for adversarially chosen (e.g. non-Gaussian) noise and general convex costs that may vary with time. This model has been studied for many extended settings, see Hazan and Singh (2021) for a comprehensive survey. Similar to our control framework, online episodic control is also studied in Kakade et al. (2020), but the regret definition differs from ours, the results are only information-theoretic and the system is linear in a kernel space. In terms of nonlinear systems, one common approach in control is iterative linearization which takes the local linear approximation via the gradient of the nonlinear dynamics. One can apply techniques from optimal control to solve the resulting changing linear system. Iterative planning methods such as iLQR (Tassa et al., 2012), iLC (Moore, 2012) and iLQG (Todorov and Li, 2005) fall into this category. Recent works (Roulet et al., 2022; Westenbroek et al., 2021) provide theoretical results and insights to this approach but many theoretical questions about the approach remain open.

The emerging theory of deep learning. For detailed background on the developing theory for deep learning, see the book draft (Arora et al., 2021). Among the various studies on the theory of deep learning, the neural tangent kernel (NTK or linearization) approach has emerged as the most complete and pervasive; it is not currently believed to fully explain the practical success but there is no alternative substantial theory yet. This technique shows that neural networks behave similar to their local linearization and proves that gradient descent converges to a global minimizer of the training loss (Soltanolkotabi et al., 2018; Du et al., 2018a,b; Jacot et al., 2018; Bai and Lee, 2019; Lee et al., 2019). The NTK approach/regime has been expanded to provide various generalization error bounds (Arora et al., 2019; Wei et al., 2019; Cao and Gu, 2019; Ji and Telgarsky, 2020), and adversarial training guarantees (Gao et al., 2019; Zhang et al., 2020). As opposed to our generic approach, a number of different optimization algorithms have been considered in isolation for analyzing deep learning theory in the NTK regime including (Wu et al., 2019; Cai et al., 2019; Wu et al., 2021; Zhang et al., 2019).

The results in this work extend upon the described deep learning theory literature; in particular, we use the same deep learning setup, and follow techniques and results from Gao et al. (2019); Allen-Zhu et al. (2019). Furthermore, several works in the literature (Cao and Gu, 2019; Gao et al., 2019; Zhang et al., 2020) have observed and used online components in their derivations of generalization and adversarial training guarantees. We note that all these works, unlike our contributions, operate in the supervised learning setting.

Online convex optimization and dimensionality notions in learning. The framework of learning in games has been extensively studied as a model for learning in adversarial and nonstochastic environments (Cesa-Bianchi and Lugosi, 2006). Online learning was infused with algorithmic techniques from mathematical optimization into the setting of online convex optimization, see (Hazan, 2019) for a comprehensive introduction. Learnability in the statistical and online learning settings was characterized using various notions of dimensionality, starting from the VC-dimension, fat-shattering dimension, Rademacher complexity, Littlestone dimension and more. For an extensive treatment see (Mohri et al., 2018; Shalev-Shwartz and Ben-David, 2014; Vapnik, 1999). Regarding interpolation, Bubeck and Sellke (2021) establish an inverse relationship between the interpolation
ability and robustness of a function class. The notion of interpolation dimension that we introduce here has found applications in the theory of boosting (Alon et al., 2021).

Deep control. Deep neural networks have advanced the state of the art for continuous control, not only in simulated environments (Tassa et al., 2018; Zhang et al., 2016; Duan et al., 2016), but also in real-world tasks such as robotic manipulation (OpenAI et al., 2018, 2019) and temperature control in office buildings and data centers (Wang et al., 2017; Lazic et al., 2018). In many of these applications, the policy learning procedure is episodic, where the environment resets at the beginning of an episode. For example, (OpenAI et al., 2018, 2019) train an LSTM policy for manipulating a Rubik’s cube with a robotic hand in the following manner: an environment is generated at the beginning of an episode. For example, OpenAI et al. (2018, 2019) train an LSTM policy for manipulating a Rubik’s cube with a robotic hand in the following manner: an environment is generated at the beginning of an episode, which interacts with the current policy for a fixed number of time steps; then, after collecting the episodic trajectory, the policy is updated according to a chosen optimization scheme. This setting is closely related to online episodic control, which we formally describe in Section 2, and motivates our theoretical analysis of neural network-based policies in this framework.

2. Problem Setting and Preliminaries

Notation. Let \(\|\cdot\|\) denote the Euclidean norm and \(\langle\cdot,\cdot\rangle\) the corresponding inner product between two vectors, matrices, or tensors of the same dimension: \(\langle x, y \rangle = \text{vec}(x)^\top \text{vec}(y)\). Let \(S_p = \{x \in \mathbb{R}^p : \|x\| = 1\}\) denote the unit \(p\)-dimensional sphere, and for a convex set \(\mathcal{K}\), let \(\mathbb{P}_K\) denote projection onto \(\mathcal{K}\).

2.1. Deep neural networks and the interpolation dimension

Deep neural networks. Let \(x \in \mathbb{R}^p\) be the \(p\)-dimensional input. We define the depth \(H\) network with ReLU activation and scalar output as follows:

\[
x^0 = Ax, \quad x^h = \sigma_{\text{relu}}(\theta^h x^h-1), \quad h \in [H], \quad f(\theta, x) = a^\top x^H,
\]

where \(\sigma_{\text{relu}}(\cdot)\) is the ReLU function \(\sigma_{\text{relu}}(z) = \max(0, z)\), \(A \in \mathbb{R}^{m \times p}\), \(\theta^h \in \mathbb{R}^{m \times m}\), and \(a \in \mathbb{R}^m\). Let \(\theta = (\theta^1, \ldots, \theta^H)^\top \in \mathbb{R}^{H \times m \times m}\) denote the trainable parameters of the network and the parameters \(A, a\) are fixed after initialization. The initialization scheme is as follows: each entry in \(A\) and \(\theta^h\) is drawn i.i.d. from the Gaussian distribution \(\mathcal{N}(0, \frac{2}{m})\), and each entry in \(a\) is drawn i.i.d. from \(\mathcal{N}(0, 1)\). This setup is common in recent literature and follows that of Gao et al. (2019).

For vector-valued outputs, we consider a scalar output network for each coordinate. Suppose for \(i \in [d]\), \(f_i\) is a deep neural network with a scalar output; with a slight abuse of notation, for input \(x \in \mathbb{R}^p\), denote

\[
f(\theta; x) = (f_1(\theta[1]; x), \ldots, f_d(\theta[d]; x))^\top \in \mathbb{R}^d,
\]

where \(\theta[i] \in \mathbb{R}^{H \times m \times m}\) denotes the trainable parameters for the network \(f_i\) for coordinate \(i\). Let \(\theta = (\theta[1], \theta[2], \ldots, \theta[d]) \in \mathbb{R}^{d \times H \times m \times m}\) denote all the parameters for \(f\).

In the online setting, the neural net receives an input \(x_t \in \mathbb{R}^p\) at each round \(t \in [T]\), and with parameter \(\theta\) suffers loss \(\ell_t(f(\theta; x_t))\). Note that this framework generalizes the supervised learning paradigm. We make the following standard assumptions:

Assumption 1 The input \(x\) has unit norm, i.e. \(x \in S_p, \|x\|_2 = 1\).

Assumption 2 The loss functions \(\ell_t(f(\theta; x))\) are \(L\)-Lipschitz and convex in \(f(\theta; x)\).
Interpolation dimension. Since we aim to prove regret bounds for online learning with families of deep neural networks as the comparator class, we need to ensure these families have non-trivial representation power. To this end, we introduce interpolation dimension, an expressivity metric that can be naturally applied to our setting. In real-valued learning, we say that a hypothesis class has interpolation dimension of at least $k$ if one can assign arbitrary real labels to any $k$ different inputs using a hypothesis from that class.

Definition 1 The interpolation dimension of a hypothesis class $H = \{ h : \mathcal{X} \subseteq \mathbb{R}^p \to \mathbb{R}^d \}$ over input domain $\mathcal{X}$ at non-degeneracy $\gamma > 0$, denoted $\mathcal{I}_{\mathcal{X},\gamma}(H)$, is the largest cardinality $k$ such that for any set of data points $\{(x_j, y_j)\}_{j=1}^k$, satisfying $\min_{j \neq l} \| x_j - x_l \|_2 \geq \gamma$, $y_j \in [-1, 1]^d$, $\forall j \in [k]$, $\inf_{h \in H} \left[ \sum_{j=1}^k \| y_j - h(x_j) \|_2^2 \right] = 0$.

The label bound above is 1 for simplicity, but can be extended to any $B > 0$. Henceforth, we show that over input domain $\mathcal{X} = \mathbb{S}_p$, neural networks that have $\text{poly}(k, \frac{1}{\gamma})$ width have $\mathcal{I}_{\mathcal{X},\gamma}(H) \geq k$. This enables us to derive regret bounds for online agnostic learning over a class of neural networks that has interpolation dimension at least $k$.

In the case of binary classification, interpolation dimension can be seen as the "dual" of the VC dimension. More details on the interpolation dimension in binary classification, connection to VC dimension, and additional examples can be found in Appendix A.1 of the full manuscript: https://xinyi.github.io/submission_1.pdf.

2.2. Online convex optimization

In Online Convex Optimization (OCO), a decision maker sequentially chooses a point in a convex set $\theta_t \in K \subseteq \mathbb{R}^d$, and suffers loss $\ell_t(\theta_t)$ according to a convex loss function $\ell : K \to \mathbb{R}$. The goal of the learner is to minimize her regret, defined as

$$\text{Regret}_T = \sum_{t=1}^T \ell_t(\theta_t) - \min_{\theta^* \in K} \sum_{t=1}^T \ell_t(\theta^*) .$$

A host of techniques from classical optimization are applicable to this setting and give rise to efficient low-regret algorithms. To name a few methods, mirror descent, Newton’s method, Frank-Wolfe and follow-the-perturbed leader all have online analogues, see e.g. Hazan (2019) for a comprehensive treatment.

As an extension to the OCO framework, we show that regret bounds hold analogously for the online optimization of nearly convex functions. As we show in later sections, these regret bounds naturally carry over to the setting of online learning over neural networks.

Definition 2 A function $\ell : \mathbb{R}^n \to \mathbb{R}$ is $\varepsilon$-nearly convex over the convex, compact set $K \subseteq \mathbb{R}^n$ iff $\forall x, y \in K$, $\ell(x) \geq \ell(y) + \nabla \ell(y)^\top (x - y) - \varepsilon$.

The analysis of any algorithm for OCO, including the most fundamental method of online gradient descent (OGD), extends to this case in a straightforward manner. Let $\mathcal{A}$ be any regret minimization algorithm for OCO with a regret bound given by $\text{Regret}_T(\mathcal{A})$. This algorithm $\mathcal{A}$ can be applied on the surrogate loss functions $h_t(\theta) = \ell_t(\theta_t) + \nabla \ell_t(\theta_t)^\top (\theta - \theta_t)$ to obtain regret bounds on the nearly convex losses $\ell_t$ as given below. The described method is presented in Algorithm 3 which along with more details can be found in Appendix A.2 of the aforementioned full manuscript.
Lemma 3  Suppose $\ell_1, \ldots, \ell_T$ are $\varepsilon$-nearly convex, then Algorithm 3 has regret bounded by
\[
\sum_{t=1}^{T} \ell_t(\theta_t) - \min_{\theta^* \in K} \sum_{t=1}^{T} \ell_t(\theta^*) \leq \text{Regret}_T(A) + \varepsilon T.
\]

2.3. Online episodic control

Consider the following online episodic learning problem for nonstochastic control over linear time-varying (LTV) dynamics: there is a sequence of $T$ control problems each with a horizon $K$ and an initial state $x_1 \in \mathbb{R}^{d_x}$. In each episode, the state transition is given by
\[
\forall k \in [1, K], \quad x_{k+1} = A_k x_k + B_k u_k + w_k,
\]
where $x_k \in \mathbb{R}^{d_x}, u_k \in \mathbb{R}^{d_u}$. The system matrices $A_k \in \mathbb{R}^{d_x \times d_x}, B_k \in \mathbb{R}^{d_x \times d_u}$ along with the next state $x_{k+1}$ are revealed to the learner after taking the action $u_k$. The disturbances $w_k \in \mathbb{R}^{d_d}$ are unknown and adversarial but can be a posteriori computed by the learner $w_k = x_{k+1} - A_k x_k - B_k u_k$.

An episode loss is defined cumulatively over the rounds $k \in [1, K]$ according to the convex cost functions $c_k : \mathbb{R}^{d_x} \times \mathbb{R}^{d_d} \to \mathbb{R}$ of state and action: for a policy $\pi$, the loss is $J(\pi; x_1, c_1:K) = \sum_{k=1}^{K} c_k(x_k^\pi, w_k^\pi)$. Like the system matrices, the cost function $c_k$ is also revealed after taking action $u_k$. The transition matrices $(A_k, B_k)_{1:K}$, initial state $x_1$, disturbances $w_{1:K}$ and costs $c_{1:K}$ can change arbitrarily over different episodes. The goal of the learner is to minimize episodic regret by adapting its output policies $\pi_t$ for $t \in [1, T]$,
\[
\text{Regret}_T(\Pi) = \sum_{t=1}^{T} J_t(\pi_t; x_1^t, c_1^t:K) - \min_{\pi \in \Pi} \sum_{t=1}^{T} J_t(\pi; x_1^t, c_1^t:K),
\]
where $\Pi$ denotes the class of policies the learner competes against.

The model above is presented in its utmost generality: the system in an episode is LTV and these LTVs are allowed to change arbitrarily throughout episodes. Results for this model can be applied to derive guarantees for: (1) a simpler setting, learning to control a single LTV episodically; (2) a more complex setting, first-order guarantees in control or planning over nonlinear dynamics by taking the Jacobian linearization of the dynamics (Ahn et al., 2007; Westenbroek et al., 2021; Roulet et al., 2022). We make the following basic assumptions about the dynamical system in each episode that are common in the nonstochastic control literature (Agarwal et al., 2019).

Assumption 3  The disturbances satisfy $\forall k \in [K], \|w_k\|_2 \leq W$.

Assumption 4 (Sequential stability) 1  There exist $C_1, C_2 \geq 1$, $0 < \rho_1 < 1$ such that the system matrices satisfy:
\[
\forall k \in [K], \forall n \in [1, k], \quad \left\| \prod_{i=k}^{k-n+1} A_i \right\|_{op} \leq C_1 \cdot \rho_1^n, \quad \|B_k\|_{op} \leq C_2.
\]

Assumption 5  Each cost function $c_k : \mathbb{R}^{d_x} \times \mathbb{R}^{d_d} \to \mathbb{R}$ is jointly convex and satisfies a generalized Lipschitz condition $\|\nabla c_k(x, u)\| \leq L_c \max\{1, \|x\| + \|u\|\}$ for some $L_c > 0$.

1. This condition is relaxed to sequential stabilizability in Appendix E.
The performance of the learner given by (2.3) directly depends on the policy class II. In this work, we focus on disturbance based policies, i.e. policies that take past perturbations as input $u_k = f(w_{k-1})$, which are parameterized w.r.t. policy-independent inputs. This is in contrast to the commonly used state feedback policy $u_k = f(x_k)$. For example, the Disturbance Action Control (DAC) policy class, shown to be more general than linear state feedback policies (Agarwal et al., 2019), outputs controls linear in past finite disturbances, resulting in a convex parameterization of the state/control and enabling the design of efficient provable online methods. Our work expands the comparator class by considering policies that are nonlinear in the past disturbances, represented by neural networks.

**Definition 4 (Disturbance Neural Feedback Control)** Let $\pi^\theta_{\text{dnn}}$ denote the policy with control outputs $u_k$ given by

$$\forall k \in [K], \quad u_k = f_\theta(w_{k-1}, w_{k-2}, \ldots, w_1) \in \mathbb{R}^{du},$$

where $f_\theta(\cdot) = f(\theta; \cdot)$ is a neural network defined in (2.1). The policy class is defined as $\Pi_{\text{dnn}}(f; \Theta) = \{\pi^\theta_{\text{dnn}} : \theta \in \Theta\}$ with $\Theta$ being the set of permissible parameters.

### 3. Online learning of deep neural networks

We present our technical results in the following two sections; due to space constraints, all proofs are included in the full manuscript https://xinyi.github.io/submission_1.pdf. We first present the general framework of online learning with deep neural networks and state the accompanying regret guarantees. Our framework can use any OCO algorithm as a black-box as in Algorithm 1, but for our main result, we use projected Online Gradient Descent (OGD). Projected OGD has explicit regret bounds and variants of GD are widely used in practice.

**Algorithm 1** Online Learning over Neural Networks

**Input:** OCO algorithm $A$, neural network $f(\cdot; \cdot)$, initial $\theta_1$, parameter set $\Theta = B(R; \theta_1)$.

**for** $t = 1 \ldots T$ **do**

- Play $\theta_t$, receive loss $\ell_t(\theta) = \ell_t(f(\theta; x_t))$.
- Construct $h_t(\theta) = \ell_t(\theta_t) + \nabla \ell_t(\theta_t) \top (\theta - \theta_t)$.
- Update $\theta_{t+1} = A(h_1, \ldots, h_t) \in \Theta$.

**end**

The main technical result, provided in Theorem 5, gives a regret bound on the online agnostic learning of deep neural networks. The benchmark hypothesis class is a class of deep neural networks with interpolation dimension of at least $k$ where $k$ is decided a priori and used in the construction of the network.

**Theorem 5** Suppose Assumptions 1 and 2 hold, and let $\mathcal{H}_{\text{NN}}(R; \theta_1) = \{f(\theta; \cdot) : \theta \in \Theta\}$ denote the class of neural networks $f(\theta; \cdot)$ as in (2.1) with parameter set $\Theta = B(R; \theta_1) = \{\theta : ||\theta[i] - \theta_1[i]||_F \leq R, \forall i \in [d]\}$ and $\mathcal{X} = \mathbb{S}_p$. Suppose $\gamma \in (0, O\left(\frac{1}{\sqrt{p}}\right))$, take $R = O\left(\frac{k^3 \log m}{\gamma \sqrt{m}}\right)$, then for $m \geq O\left(\frac{k^3 (k^2 H^{12} \log^8 m + d)^{3/2}}{\gamma \sqrt{m}}\right)$, with probability $1 - O(H + d)e^{-\Omega(\log^2 m)}$ over the random initialization,

- The function class $\mathcal{H}_{\text{NN}}(R; \theta_1)$ has interpolation dimension $\mathcal{I}_{\mathcal{X}, \gamma}(\mathcal{H}_{\text{NN}}(R; \theta_1)) \geq k$. 


• Algorithm 1 using OGD with $\eta_t = \frac{2R\sqrt{d}}{LH\sqrt{m}} \cdot t^{-1/2}$ for $A$ attains regret bound
\[
\sum_{t=1}^{T} \ell_t(f_t; x_t) \leq \min_{g \in \mathcal{H}_{NN}(R; \theta_1)} \sum_{t=1}^{T} \ell_t(g(x_t)) + \tilde{O}\left(\frac{k^3 L^2 H \sqrt{dT}}{\gamma} + \frac{k^4 L^5/2 \sqrt{dT}}{\gamma^{4/3} m^{1/6}}\right),
\]
where $\tilde{O}(\cdot)$ hides terms polylogarithmic in $m$.

The above theorem indicates that the average regret can be minimized up to arbitrary precision: for any $\varepsilon > 0$, if one chooses sufficiently large network width $m = \Omega(\varepsilon^{-6})$ and sufficiently large number of iterations $T = \Omega(\varepsilon^{-2})$, the average regret is bounded by $\varepsilon$. The interpolation dimension bound is established due to the seminal work Allen-Zhu et al. (2019), spelled out in the following lemma and proven in Appendix A.1.

**Lemma 6** Let $\mathcal{H}_{NN}(R; \theta_1) = \{f(\theta; \cdot) : \theta \in \Theta\}$ denote the class of neural networks as in (2.1) where $\Theta = B(R; \theta_1)$ and $\mathcal{X} = \mathbb{S}_p$. Suppose $\gamma \in (0, O(\frac{1}{T})], m \geq \Omega\left(\frac{k^2 H^2 \log^2 m}{\gamma^8}\right)$ and $R = O\left(\frac{k^2 \log m}{\gamma m}\right)$, then with probability $1 - d \cdot e^{-\Omega(\log^2 m)}$ over random initialization of $\theta_1$,

\[
I_{X, \gamma}(\mathcal{H}_{NN}(R; \theta_1)) \geq k.
\]

**3.1. Proof Sketch**

Due to space constraints, we give a proof sketch here; for a more detailed analysis outline, see Appendix C, and for the full proof see Appendix D. There are 3 steps to the proof of Theorem 5. First, we show that the considered loss functions $\ell_t : \Theta \to \mathbb{R}, \ell_t(\theta) = \ell_t(f(\theta; x_t))$ are nearly convex with respect to the parameter $\theta$. This is due to the observation that in the overparameterized regime, neural networks behave similarly to their local linearization.

Second, we can use the near convexity of the loss functions $\ell_t(\theta)$ for all $\theta \in B(R; \theta_1)$, and Lemma 3 to show a regret bound over the parameter set $\Theta = B(R; \theta_1)$. The bound is comprised of the sublinear regret of the OCO algorithm used for parameter update, and the worst-case linear penalty of near convexity $\varepsilon_{nc} \cdot T$, where $\varepsilon_{nc}$ is in terms of $R$ and $m$.

Finally, we use Lemma 6 to ensure that our choice of $R$ and $m$ give the desired interpolation dimension, and derive the final regret guarantee in terms of $k, \gamma$ and $m$.

**4. Online episodic control with neural network controllers**

The online episodic control problem described in Section 2.3 with the policy class $\Pi = \Pi_{dnn}(f; \Theta)$ can be reduced to online learning for neural networks. This reduction is done by following the current policy each episode, constructing the episode loss, and updating the policy via an OCO algorithm. Algorithm 2 below uses projected OGD but as in the previous section, any OCO algorithm can be used instead.

**Theorem 7** Suppose Assumptions 3, 4, 5 hold and let $\Pi_{dnn}(f; \Theta)$ denote the policy class given by Definition 4 with $\Theta = B(R; \theta_1)$. Take $R = O\left(\frac{k^3 (2KW + H) \log m}{\sqrt{m}}\right)$, then for $m \geq \Omega(K^{16} H^{20} W^{8} (d_{x}d_{u})^{3/2} \log^{12} m)$
Define the optimal open-loop control sequence over a single episode in the following way. For simplicity, drop the episode index. To demonstrate the capacity of the benchmark policy class \( \Pi = \Omega(T \log \epsilon) \) episodic regret is attained with a large network width \( m \). This theorem statement, analogous to Theorem 5, implies that arbitrarily small \( \epsilon > 0 \) average episodic regret is attained with a large network width \( m = \Omega(\epsilon^{-6}) \) and large number of iterations \( T = \Omega(\epsilon^{-2}) \). The regret bound is against the benchmark policy class \( \Pi_{\text{dnn}}(f; \Theta) \) which is chosen such that the neural network class has interpolation dimension \( k = K \). This implies open-loop control optimality over a single episode in the following way. For simplicity, drop the episode index \( t \in [T] \) and define the optimal open-loop control sequence of any episode as detailed below.

**Algorithm 2** Deep Neural Network Episodic Control with OGD

**Input:** stepsize \( \eta_t > 0 \), initial parameter \( \theta_1 \), parameter set \( \Theta = B(R; \theta_1) \).

**for** \( t = 1 \ldots T \) **do**

**for** \( k = 1 \ldots K \) **do**

- Observe \( z_k^t = \text{vec}([w_{k-1}^t, \ldots, w_1^t, 0, \ldots, 0, k]) \in \mathbb{R}^{Kd_x+1} \).
- Normalize \( z_k^t = \frac{z_k^t}{\|z_k^t\|} \), and play \( u_k^t = f(\theta_t, z_k^t) \).

**end**

- Construct loss function \( L_t(\theta) = \sum_{k=1}^K c_k(x_k^t, f(\theta_t, z_k^t)) \).
- Perform gradient update \( \theta_{t+1} = \prod_{\Theta} [\theta_t - \eta_t \nabla_\theta L_t(\theta_t)] \).

**end**

with probability at least \( 1 - O(H + d_u)e^{-\Omega(\log^2 m)} \) over the randomness of initialization \( \theta_1 \), Algorithm 2 with \( \eta_t = \Theta(\sqrt{\tau_{\text{dnn}}/}\sqrt{m}t^{-1/2}) \) satisfies

\[
\text{Regret}_T(\Pi_{\text{dnn}}(f; \Theta)) \leq \tilde{O} \left( K^{10} L_c H^4 W^2 d_u q_x^{1/2} \cdot \sqrt{T} + \frac{K^{12} L_c H^6 W^3 d_u q_x^{1/2}}{m^{1/6}} \cdot T \right),
\]

where \( \Pi_{\text{dnn}}(f; \Theta) \) can output the optimal open-loop control sequence \( u_{1:K}^* \in [-1, 1]^{K \times d_u} \) of any episode and \( \tilde{O}(\cdot) \) hides terms polylogarithmic in \( m \).

This theorem statement, analogous to Theorem 5, implies that arbitrarily small \( \epsilon > 0 \) average episodic regret is attained with a large network width \( m = \Omega(\epsilon^{-6}) \) and large number of iterations \( T = \Omega(\epsilon^{-2}) \). The regret bound is against the benchmark policy class \( \Pi_{\text{dnn}}(f; \Theta) \) which is chosen such that the neural network class has interpolation dimension \( k = K \). This implies open-loop control optimality over a single episode in the following way. For simplicity, drop the episode index \( t \in [T] \) and define the optimal open-loop control sequence of any episode.

**Definition 8** Define the optimal open-loop control sequence \( u_{1:K}^* \in [-1, 1]^{K \times d_u} \) to be

\[
u_{1:K}^* = \arg \min_{\forall k, u_k \in [-1, 1]^{d_u}} \left\{ J(u_{1:K}; x_1, c_{1:K}) = \sum_{k=1}^K c_k(x_k, u_k) \right\}.
\]

To demonstrate the capacity of the benchmark policy class \( \Pi_{\text{dnn}}(f; \Theta) \) with \( \Theta = B(R; \theta_1) \) we show that it can output the optimal open-loop control sequence of any single episode as detailed below.

**Lemma 9** Take \( R = O \left( \frac{K^3 \log m (2KW + H)}{\sqrt{m}} \right) \), suppose \( m \geq \Omega \left( K^{24} H^{12} \log^5 m (2KW + H)^8 \right) \), then with probability \( 1 - d_u e^{-\Omega(\log^2 m)} \) over the random initialization of \( \theta_1 \), \( \Pi_{\text{dnn}}(f; \Theta) \) can output any open-loop control sequence \( u_{1:K}^* \in [-1, 1]^{K \times d_u} \):

\[
\inf_{\pi_{\text{dnn}}^* \in \Pi_{\text{dnn}}(f; \Theta)} \left[ \sum_{k=1}^K \| u_k^\theta - u_k^* \|^2 \right] = 0.
\]
4.1. Proof Sketch

To extend the online learning results of Theorem 5 to the online episodic control setting, we ensure the control setting satisfies the corresponding assumptions. For each \( k \in [K] \), denote the padded input \( z_k = \text{vec}(w_{k-1}, \ldots, w_1, 0, \ldots, 0, k) \in \mathbb{R}^{K \cdot d_u + 1} \) where the index is padded to ensure inputs are separable (Definition 1). To satisfy Assumption 1, normalize the network inputs \( \bar{z}_k = \frac{z_k}{\|z_k\|_2} \in \mathbb{S}_{K \cdot d_u + 1} \).

For a policy \( \pi_{\text{dnn}}^\theta \) the episode loss \( \mathcal{L}(\theta) = J(\pi_{\text{dnn}}^\theta; x_1, c_{1:K}) \) depends on the parameter \( \theta \) through all the \( K \) controls \( u_k^\theta = f(\theta; \bar{z}_k) \). Denote \( \bar{f}(\theta) = [u_1^\theta, \ldots, u_K^\theta]^\top \in \mathbb{R}^{K \times d_u} \) and let \( \mathcal{L}(\theta) = \mathcal{L}(\bar{f}(\theta)) \) by abuse of notation. We demonstrate that the reduction to the online learning setting is achieved by showing that \( \mathcal{L}(\bar{f}(\theta)) \) satisfies the convexity (Lemma 23) and Lipschitz (Lemma 26) conditions.

Hence, for each episode \( t \in [T] \), the episode loss \( \mathcal{L}_t(\theta) = J_t(\pi_{\text{dnn}}^\theta; x_t, c_{t:K}) \) satisfies Assumption 2 and the rest of the derivation is analogous to that of Theorem 5. Finally, Lemma 9 uses the interpolation dimension property of the neural network class to conclude the open-loop optimality stated in the theorem. See Appendix E for full details.

5. Conclusions and Future Work

In this work, we derive the first regret guarantees for neural network based controllers in online control. Our results are in the online episodic control setting, which is motivated by empirical research in control and deep reinforcement learning. We propose algorithms that obtain sublinear episodic regret against the optimal open-loop control sequence of any episode, which relies on a general reduction from online deep learning to regret minimization.

We also introduce a new metric for the expressive power of a hypothesis class and use it for characterizing the expressivity of the benchmark neural network class. The definition of interpolation dimension enables this characterization to be isolated to neural networks but is in no way specific to them. Many intriguing questions about this expressivity notion remain, such as its broader connection to statistical learning theory given its close relationship to the VC dimension.

We use the NTK paradigm to derive the control and online learning results in this work. However, there still are open questions to understand the empirical success of neural networks. As deep learning theory advances in this direction, the question of extending these results to reinforcement learning and control problems remains open too.
References


Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization, 2019.


Eran Malach, Gilad Yehudai, Shai Shalev-Schwart, and Ohad Shamir. The connection between approximation, depth separation and learnability in neural networks. In Mikhail Belkin and Samory


