Large-time asymptotics in deep learning

Carlos Esteve, Borjan Geshkovski, Dario Pighin, Enrique Zuazua

LARGE-TIME ASYMPTOTICS IN DEEP LEARNING

CARLOS ESTEVE, BORJAN GESHKOVSKI, DARIO PIGHIN, AND ENRIQUE ZUAZUA

Abstract. It is by now well-known that practical deep supervised learning may roughly be cast as an optimal control problem for a specific discrete-time, nonlinear dynamical system called an artificial neural network. In this work, we consider the continuous-time formulation of the deep supervised learning problem, and study the latter's behavior when the final time horizon increases, a fact that can be interpreted as increasing the number of layers in the neural network setting.

When considering the classical regularized empirical risk minimization problem, we show that, in long time, the optimal states converge to zero training error, namely approach the zero training error regime, whilst the optimal control parameters approach, on an appropriate scale, minimal norm parameters with corresponding states precisely in the zero training error regime. This result provides an alternative theoretical underpinning to the notion that neural networks learn best in the overparametrized regime, when seen from the large layer perspective.

We also propose a learning problem consisting of minimizing a cost with a state tracking term, and establish the well-known *turnpike property*, which indicates that the solutions of the learning problem in long time intervals consist of three pieces, the first and the last of which being transient short-time arcs, and the middle piece being a long-time arc staying exponentially close to the optimal solution of an associated static learning problem. This property in fact stipulates a quantitative estimate for the number of layers required to reach the zero training error regime.

Both of the aforementioned asymptotic regimes are addressed in the context of continuous-time and continuous space-time neural networks, the latter taking the form of nonlinear, integro-differential equations, hence covering residual neural networks with both fixed and possibly variable depths.

Contents

Date: August 6, 2020.

Keywords. Deep Learning, Supervised Learning, Optimal Control, Residual Neural Networks, Neural ODEs, Turnpike property.

Funding: B.G. and E.Z. have received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No.765579-ConFlex. D.P., C.E. and E.Z. have received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement NO. 694126-DyCon). The work of E. Z. has been supported by the Alexander von Humboldt-Professorship program, the Transregio 154 Project "Mathematical Modelling, Simulation and Optimization Using the Example of Gas Networks" of the German DFG, grant MTM2017-92996-C2-1-R COSNET of MINECO (Spain) and by the Air Force Office of Scientific Research (AFOSR) under Award NO. FA9550-18-1-0242.

1. Introduction

Modern machine learning, and more specifically *supervised learning*, addresses the problem of predicting from data, which roughly consists in approximating an unknown function $f : \mathbb{R}^d \to \mathbb{R}^m$ from N known samples $\{\vec{x}_i, \vec{y}_i = f(\vec{x}_i)\}_{i=1}^N$. Depending on the nature of the *labels* $\vec{y}_i$, we distinguish two types of supervised learning tasks, namely that of *classification* (labels take values in a finite set of $\mathbb{R}^m$) and *regression* (the labels take continuous values). As per this nomenclature, f is referred to as a *classifier* or *regressor* for the respective task. In most practical applications, the dimension d of each sample $\vec{x}_i$ is very big – commonly in the order of millions for images or audio and text signals.

A plethora of methods for finding $f(\cdot)$ efficiently with theoretical and empirical guarantees have been developed and investigated in the machine learning literature in recent decades. Prominent examples, to name a few, include linear classification methods (e.g. linear or logistic regression), kernel-based methods (e.g. support vector machines), tree-based methods (e.g. decision trees) and so on. We refer to the book [Goodfellow et al., 2016] for a comprehensive presentation and references on these topics.

Deep neural networks are parametrized computational architectures which propagate each individual sample of the input data $\{\vec{x}_i\}_{i=1}^N \in \mathbb{R}^{d\times N}$ across a sequence of linear parametric operators and simple nonlinearities. The so-called *residual* architectures may – in the simplest scenarios – be cast as schemes of the mould

$$\begin{cases} \mathbf{x}_i^{k+1} = \mathbf{x}_i^k + \sigma(w^k \mathbf{x}_i^k + b^k) & \text{for } k \in \{0, \dots, N_{\text{layers}} - 1\} \\ \mathbf{x}_i^0 = \vec{x}_i \in \mathbb{R}^d \end{cases} \tag{1.1}$$

for all $i \in \{1, \dots, N\}$. The unknowns are the *states* $\mathbf{x}_i^k \in \mathbb{R}^d$ for any $i \in \{1, \dots, N\}$, while σ is an explicit, globally Lipschitz continuous nonlinear function (see Fig. 2.1), $\left\{w^k, b^k\right\}_{k=0}^{N_{\text{layers}}-1}$ are optimizable control parameters (*weights* and *biases*) with $w^k \in \mathbb{R}^{d\times d}$ and $b^k \in \mathbb{R}^d$, and $N_{\text{layers}} \geq 1$ designates the number of layers, commonly called the *depth*. This formulation leads to viewing neural networks as dynamical systems, and the procedure of *training* consists in finding optimal control parameters $\left\{w^k, b^k\right\}_{k=0}^{N_{\text{layers}}-1}$ steering all of the states $\mathbf{x}_i^{N_{\text{layers}}}$ as close as possible to the corresponding labels $\vec{y}_i \in \mathbb{R}^m$ for all i, namely solving

$$\min_{\left\{w^k, b^k\right\}_{k=0}^{N_{\text{layers}}-1}} \frac{1}{N} \sum_{i=1}^{N} \text{loss}\left(\varphi\left(\mathbf{x}_i^{N_{\text{layers}}}\right), \vec{y}_i\right),$$

generally done numerically via stochastic gradient descent and backpropagation, whilst guaranteeing reliable performance on unseen data (ensuring *generalization*).

Here

$$\mathrm{loss}(\cdot,\cdot) : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}_+$$

is a given continuous function – for instance $\mathrm{loss}(x,y) := \|x-y\|_{\ell^p}^p$ for $p = 1, 2$, or $\mathrm{loss}(x,y) := \log\left(1 + e^{-\langle x,y\rangle}\right)$ (logistic loss), while $\varphi : \mathbb{R}^d \to \mathbb{R}^m$ is a parametrized map – which is a linear projection if the labels $\vec{y}_i$ take continuous values (regression tasks), or is a softmax normalization nonlinearity applied to this projection if $\vec{y}_i$ take discrete values (classification tasks) see (2.8), – serves to flatten the d–dimensional states $\mathbf{x}_i^{N_{\mathrm{layers}}}$ onto $\mathbb{R}^m$.

Supervised learning may thus be recast as an *optimal control problem*, the sequence of parameters $\{w^k, b^k\}_{k=0}^{N_{\mathrm{layers}}-1}$ playing the role of *controls*, as seen and discussed in more detail in Section 2.

Deep neural networks have been shown to achieve impressive experimental results for both classification and regression tasks where data is structured and available in large amounts (see [LeCun et al., 2015] for a survey). In particular, convolutional neural networks (CNNs), introduced in [LeCun et al., 1990], implemented with linear convolutions followed by nonlinearities over several layers, have shown to give state of the art performances for image classification with several thousands of classes [Krizhevsky et al., 2012], speech recognition [Hinton et al., 2012], bio-medical applications [Leung et al., 2014], natural language processing [Sutskever et al., 2014], and in many other domains. Even though deep neural networks often have far more trainable parameters than the number of samples they are trained on (an notion called *overparametrization*), they empirically exhibit remarkable generalization properties.

However, many aspects of the working mechanisms leading to these experimental results need to be better understood. Indeed, the lack of generic accuracy guarantees, the lack of understanding on how regularization techniques precisely affect generalization, as well as the ad hoc nature of architecture design and choice of hyper-parameters (e.g. number of layers, number of neurons per layer) result in deep neural networks sometimes acting as black-box algorithms. A better understanding of these aspects, even in simple scenarios, would render neural networks more transparent and thus interpretable, and lead to more principled and reliable architecture design.

Due to the inherent dynamical systems nature of residual neural networks, several recent works have aimed at studying the continuous-time formulation in some detail in order to obtain a better understanding of the choice of the aforementioned hyper-parameters and generate better performing models, a trend started with the works [E, 2017, Haber and Ruthotto, 2017]. This perspective is motivated by the simple observation that for any $i \in \{1, \dots, N\}$, (1.1) is roughly the forward Euler scheme for the ordinary differential equation (ODE)

$$\begin{cases} \dot{\mathbf{x}}_i(t) = \sigma(w(t)\mathbf{x}_i(t) + b(t)) & \text{for } \ t \in (0,T) \\ \mathbf{x}_i(0) = \vec{x}_i \in \mathbb{R}^d, \end{cases} \tag{1.2}$$

where $T > 0$ is given. The continuous-time formulation has also been used to great effect in experimental contexts, in particular as more general adaptive ODE solvers

can be used for improved training performance, as per the works [Chen et al., 2018, Benning et al., 2019].

The role of the final time horizon $T > 0$ however, which may play a key role in the control of dynamical systems, is, up to the best of our knowledge, not discussed in the machine learning context. As each time-step of a discretization to (1.2) represents a different layer of the derived neural network (e.g. (1.1)), and the time horizon $T > 0$ in (1.2) thus serves as an indicator of the number of layers N_{layers} in the discrete-time context (1.1), a good a priori knowledge of the dynamics of the learning problem over longer time horizons is needed. Such an understanding would lead to potential rules for choosing the number of layers, as well as enlighten the generalization properties when the number of layers is large.

Through this work, we aim to bridge this gap by leveraging the added degree of freedom represented by the time horizon T, and propose several novel insights on the relevance of the latter from a more analytical point of view, in view of further hybridizing the topics of deep supervised learning, optimal control, and numerical analysis.

1.1. **Our contributions.** Let us assume that we are given a training dataset $\{\vec{x}_i, \vec{y}_i\}_{i=1}^N$, with $\vec{x}_i \in \mathbb{R}^d$ and $\vec{y}_i \in \mathbb{R}^m$ for any i. We will consider the continuous-time supervised learning problem, which is roughly an optimal control problem for continuous-time neural networks including (but not restricted to) ones of the form (1.2). We bring forth the following results and insights.

1. In Section 3, we consider the classical supervised learning problem, namely that of regularized empirical risk minimization:

$$\inf_{\substack{[w,b]^\top \in H^k(0,T;\mathbb{R}^{d_u}) \\ \text{subject to (1.2)}}} \underbrace{\frac{1}{N}\sum_{i=1}^{N} \text{loss}\big(\varphi(\mathbf{x}_i(T)),\, \vec{y}_i\big)}_{\text{training error}} + \underbrace{\frac{\alpha}{2}\, \big\|[w,b]^\top\big\|^2_{H^k(0,T;\mathbb{R}^{d_u})}}_{\text{regularization}} \tag{1.3}$$

where $H^k(0,T;\mathbb{R}^{d_u})$ is the Sobolev[1] space of square integrable functions from $(0,T)$ to $\mathbb{R}^{d_u}$ with k square integrable derivatives (henceforth, we only consider $k = 0, 1$), whereas for any $i \in \{1, \dots, N\}$, $\mathbf{x}_i \in C^0([0,T];\mathbb{R}^d)$ is the unique solution to (1.2) corresponding to the datum $\vec{x}_i \in \mathbb{R}^d$. The *weight decay* $\alpha > 0$ is fixed, and serves as an overfitting impediment by regulating the oscillations of the control parameters $w(t)$ and $b(t)$.

In Theorem 3.1, we show that solutions $\left[w^T, b^T\right]^\top$ to the minimization problem (1.3), converge, on a suitable scale, to a solution $[w^1, b^1]^\top$ of the minimization problem

$$\inf_{\substack{[w,b]^\top \in H^k(0,1;\mathbb{R}^{d_u}) \\ \text{subject to (1.2) with } T=1 \\ \text{and} \\ \mathbf{x}(1) \in \arg\min_{\mathbb{R}^{d\times N}} \sum_{i=1}^{N} \text{loss}\left(\varphi(\cdot), \vec{y}_i\right)}} \frac{\alpha}{2}\, \big\|[w,b]^\top\big\|^2_{H^k(0,1;\mathbb{R}^{d_u})}$$

[1]We make precise the necessity of considering Sobolev regularization, namely $k = 1$, in the context of (1.2) in Remark 2.

when $T \to +\infty$. Here $\mathbf{x}(1) := [\mathbf{x}_1(1), \dots, \mathbf{x}_N(1)]^\top$, with each $\mathbf{x}_i \in C^0([0,1]; \mathbb{R}^d)$ being the unique solution to (1.2) with $T = 1$, corresponding to the datum $\vec{x}_i$. Furthermore, the stacked vector of states $\mathbf{x}^T(T) = \left[\mathbf{x}_1^T(T), \dots, \mathbf{x}_N^T(T)\right]^\top$ corresponding to the optimal parameters $\left[w^T, b^T\right]^\top$, itself converges to some[2] minimizer of the training error as $T \to +\infty$.

Theorem 3.1 thus stipulates that optimizing with $T \gg 1$, which in the discrete-time residual network case corresponds to a large number of layers, has the practically desirable effect of making the training error close to zero, but in the optimal way, namely, with parameters having the least oscillations possible. Heuristically, this somewhat provides an alternative theoretical basis to the notion that neural networks learn best in the overparametrized regime – we refer to the discussion succeeding Theorem 3.1 for more detail.

2. Parallel to (1.3), in Section 4 we minimize a slightly different cost wherein we add a tracking term accounting for the error between the vector of state trajectories $\mathbf{x}(t) = [\mathbf{x}_1(t), \dots, \mathbf{x}_N(t)]^\top \in \mathbb{R}^{d \times N}$ and a prescribed time-independent target $\mathbf{x}_d \in \mathbb{R}^{d \times N}$:

$$\inf_{\substack{[w,b]^\top \in H^k(0,T;\mathbb{R}^{d_u}) \\ \text{subject to (1.2)}}} \frac{1}{N} \sum_{i=1}^N \text{loss}\left(\varphi(\mathbf{x}_i(T)), \vec{y}_i\right) + \frac{\alpha}{2} \left\|[w,b]^\top\right\|^2_{H^k(0,T;\mathbb{R}^{d_u})} + \underbrace{\frac{\beta}{2} \int_0^T \|\mathbf{x}(t) - \mathbf{x}_d\|^2 \, \mathrm{d}t}_{\text{tracking term}}, \tag{1.4}$$

where for every $i \in \{1, \dots, N\}$, $\mathbf{x}_i \in C^0([0,T]; \mathbb{R}^d)$ is the unique solution to (1.2) corresponding to the datum $\vec{x}_i \in \mathbb{R}^d$, while $\alpha, \beta > 0$ are both fixed.

The presence of the tracking term plays a key role in this context. In Theorem 4.1, we show *the turnpike property*, which indicates that in long time intervals, outside of two short intervals near $t = 0$ and $t = T$, the optimal parameters $[w^T, b^T]$ and corresponding vector of state trajectories $\mathbf{x}^T = [\mathbf{x}_1^T, \dots, \mathbf{x}_N^T]^\top$, stay close to the steady-state optimal solution $[w^s, b^s]^\top$ and $\mathbf{x}^s = [\mathbf{x}_1^s, \dots, \mathbf{x}_N^s]^\top$ (called *the turnpike*) of the associated static minimization problem:

$$\inf_{\substack{[w^s,b^s]^\top \in \mathbb{R}^{d_u} \\ \text{subject to} \\ \sigma(w^s \mathbf{x}_i^s + b^s) = 0}} \frac{\alpha}{2} \left\|[w^s, b^s]^\top\right\|^2 + \frac{\beta}{2} \|\mathbf{x}^s - \mathbf{x}_d\|^2. \tag{1.5}$$

When $\sigma(0) = 0$, any $\overline{\mathbf{x}} \in \mathbb{R}^d$ with $[\overline{w}, \overline{b}]^\top \equiv [0,0]^\top$ is such that $\sigma(\overline{w}\,\overline{\mathbf{x}} + \overline{b}) = 0$. Whence, $[w^s, b^s]^\top \equiv [0,0]^\top$ and $\mathbf{x}^s = \mathbf{x}_d$ designate the solution to (1.5). In particular, our theorem will indicate that the vector of trajectories $\mathbf{x}^T(t)$ will be $\mathcal{O}\left(e^{-\mu t} + e^{-\mu(T-t)}\right)$–close to the turnpike $\mathbf{x}_d$, for some $\mu > 0$ independent of T, and for any $t \in [0,T]$.

Consequently we see in Corollary 4.1 that when $d = m$, and we rather consider the training problem (1.4) with loss $\equiv 0$ and $\mathbf{x}_d = [\vec{y}_1, \dots, \vec{y}_N]^\top$, then $\mathbf{x}^T(T)$ is $\mathcal{O}(e^{-\mu T})$–close to the zero training error regime. This result is in line with our first contribution Theorem 3.1, but with a quantitative

[2]Deep learning is a highly non-convex optimization problem, hence minimizers of the training error are not unique.

rate of convergence, and thus an estimate of the number of layers needed to be ε–close to the zero training error regime for a given $\varepsilon > 0$.

In Section 4.3, we also consider (under more restrictions on the underlying dynamics) the training problem with L^1–parameter regularization. We prove in Theorem 4.2 that the optimal controls are of bang-bang type, and in addition, if the time horizon is sufficiently large, $\mathbf{x}^T(T)$ reaches the zero training error regime in finite time. Hence, considering a larger time-horizon does not have any effect in the optimal parameters and the corresponding trajectories. This represents a different situation compared to the behavior described in Theorem 3.1, in which the zero training error is theoretically achieved only after letting the time-horizon go to infinity.

3. In Section 5, we propose a couple of results illuminating the properties of the control parameters needed to reach the zero training error regime. Namely, in Theorem 5.1 we give a lower bound for the weights steering the optimal trajectories to zero training error in terms of the distribution of the input data, while in Theorem 5.2, we show that there indeed exist such control parameters under smallness conditions on the data (a local controllability result).

4. In most of the literature on supervised learning via residual neural networks such as (1.1) and continuous-time analogs, *fixed width* cases are generally considered, namely, $\mathbf{x}_i^k \in \mathbb{R}^d$ at every layer k. The *width* $d \geq 1$ indicates the number of *neurons* within each layer. To address more general scenarios motivated by multi-layer perceptrons and convolutional neural networks, in Section 6 we propose a continuous space-time neural network taking the form of a scalar non-local partial differential equation (PDE):

$$\begin{cases} \partial_t \mathbf{z}_i(t,x) = \sigma\left(\displaystyle\int_\Omega w(t,x,\xi)\mathbf{z}_i(t,\xi)\,\mathrm{d}\xi + b(t,x)\right) & \text{for } (t,x) \in (0,T)\times\Omega \\ \mathbf{z}_i(0,x) = \mathbf{z}_i^{\mathrm{in}}(x) & \text{for } x \in \Omega \end{cases} \tag{1.6}$$

for any $i \in \{1,\dots,N\}$. Here $\Omega \subset \mathbb{R}^{d_\Omega}$ is a bounded domain, $d_\Omega \geq 1$ is chosen based on the nature[3] of the inputs $\{\vec{x}_i\}_{i=1}^N$, whereas $\mathbf{z}_i^{\mathrm{in}} \in C^0(\overline{\Omega})$ interpolates $\vec{x}_i$ for any i. By means of some simple numerical analysis arguments, in Section 6 we show that (1.6) is generic in the sense that by taking initial data as a linear combination of Dirac masses, one recovers continuous-time neural networks such as (1.2), while by imposing a specific structure on the weight $w(t,x,\xi)$, it allows for deducing various forms of convolutional neural networks as well.

In Theorem 6.1 (resp. Theorem 6.2), we moreover show that our finite-dimensional conclusions from Theorem 3.1 (resp. Theorem 4.1) transfer to the infinite-dimensional analogs of (1.3) (resp. (1.4)) for (1.6).

1.2. **Introductory bibliographical overview.** The study of supervised machine learning as function approximation via the flow of a dynamical system, and its formulation as an open loop optimal control problem, has been presented in [E, 2017].

[3]For instance, $d_\Omega = 3$ if $\vec{x}_i \in \mathbb{R}^{d_1\times d_2\times d_{\mathrm{ch}}}$ in the context of image data, and $d_\Omega = 1$ for vectorized data.

These observations were motivated by the introduction of residual neural networks in [He et al., 2016]. Since then, there has been a flurry of works that share our continuous-time perspective of deep supervised learning.

We do note however that some variants of continuous-time deep learning have been investigated much earlier, going at least as back as the 1980s; the neural network model proposed in [Hopfield, 1982] is a differential equation, and the works [Pineda, 1987, LeCun et al., 1988], in which the idea of back-propagation is connected to the adjoint method arising in optimal control. These techniques have been used to study several problems such as identifying the weights from data [Albertini and Sontag, 1993b, Albertini and Sontag, 1993a, Albertini et al., 1993], the controllability of continuous-time recurrent networks [Sontag and Sussmann, 1997, Sontag and Qiao, 1999], and stability issues [Michel et al., 1989, Hirsch, 1989].

Several recent works consider the aforementioned continuous-time viewpoint of deep learning with different directions, including an infinite-data-like interpretation via mean-field arguments, e.g. [E et al., 2019, Hu et al., 2019, Jabir et al., 2019, Ma et al., 2019, Lu et al., 2020, Conforti et al., 2020], indirect training algorithms based on the Pontryagin Maximum Principle [Li et al., 2017, Benning et al., 2019], while the rigorous limit from discrete to continuous learning is addressed and studied in [Thorpe and van Gennip, 2018, Avelin and Nyström, 2020].

In [Haber and Ruthotto, 2017, Ruthotto and Haber, 2019], the authors combine monotone operator theory and Courant-Friedrich-Levy-like conditions to address stability issues for the continuous-time forward dynamics, in the sense that a small perturbation of the neural network input yields a small perturbation of the output. Of course, in the continuous-time ODE setting, such kinds of estimates are very closely linked to those provided by the continuous dependence of the ODE solutions with respect to the initial data. See also [Zhang and Schaeffer, 2019] for a recent contribution. The works [Haber and Ruthotto, 2017, Ruthotto and Haber, 2019] in particular stipulate that these stability estimates can be interpreted as an input-output stability property under possible adversarial perturbations for neural networks, a topic of central interest in modern deep learning [Goodfellow et al., 2014].

In the machine learning community, continuous-time neural networks such as (1.2) are called *neural ordinary differential equations*. The numerical advantages and efficiency of these continuous models with respect to the discrete-time residual neural networks are demonstrated in several works, including [Lu et al., 2018, Chen et al., 2018, Dupont et al., 2019, Benning et al., 2019], where time-step adaptive ODE solvers are used to solve the underlying neural ODE, with evident numerical advantages which are somewhat reflected by our theoretical results. Indeed, given a fixed network depth, the memory consumed by neural ODEs is significantly smaller than a standard ResNet during training. However only constant or appended-time weights are considered in the first three works, whereas our experiments cover full generality, as done in [Benning et al., 2019], where, given a fixed number of layers, include the time step as an additional parameter to be optimized.

The Neural ODE perspective has been used to great effect in practical applications. Examples include irregular time series modeling [Rubanova et al., 2019], mean field games [Ruthotto et al., 2020], and – due to time reversibility – generative modeling through normalizing flows [Grathwohl et al., 2018, Chen et al., 2019]. They have also been adapted to the stochastic setting [Tzen and Raginsky, 2019].

1.3. **Outline.** This paper is organized in five parts. In Section 2, we give a brief but comprehensive presentation on the topic of deep supervised learning from the perspective of continuous-time optimal control. In Section 3, we present and prove our first main result, Theorem 3.1, along with its interpretation and a greedy pre-training algorithm. In Section 4, we present our main turnpike results, in both the L^2 (Tikhonov) in Theorem 4.1 – Corollary 4.1, and L^1 (Lasso) parameter regularization in Theorem 4.2. In Section 5, we present a couple of results illustrating a lower bound on the size of the controls needed to reach the zero training error regime (Theorem 5.1), as well as a local exact controllability result (Theorem 5.2). Finally in Section 6, we present the continuous analog of residual neural networks with variable widths, illustrate some possible approaches for passing from the continuous to the discrete case, and present possible extensions of Theorem 3.1 and Theorem 4.1 in this context.

Notation. Given any $a \in \mathbb{R}^n$, we denote by $a^\top$ its transpose. We insist on this notation for the transpose, as we use the notation $\mathbf{x}^T$ and u^T to make specific the dependence of the state and control on the time horizon T. We denote by $\|\cdot\|$ the standard euclidean norm: $\|a\| = \left(\sum_{j=1}^n a_j^2\right)^{\frac{1}{2}}$ for $a \in \mathbb{R}^n$. We denote by $\mathrm{Lip}(\mathbb{R})$ the set of functions $f : \mathbb{R} \to \mathbb{R}$ which are globally Lipschitz continuous, and by $L^2(0,T;\mathbb{R}^n)$ (resp. $H^1(0,T;\mathbb{R}^n)$) the Lebesgue (resp. Sobolev) space consisting of all functions $f : (0,T) \to \mathbb{R}^n$ which are square integrable (resp. square integrable and with a square integrable derivative). When $f : (0,T) \to X$ with X an infinite-dimensional Banach space, we define the integral of f via the Bochner integral. Given two Banach spaces X and Y, we denote by $\mathcal{L}(X,Y)$ the space of linear and bounded operators from X to Y. We also use $\dot{\mathbf{x}}(t) := \frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t}(t)$, as well as, whenever the dependence on parameters of a constant is not specified, denote $f \lesssim_S g$ whenever a constant $C \geq 1$, depending only on the set of parameters S, exists such that $f \leq Cg$.

2. A roadmap to continuous-time supervised learning

The unknown classifier/regressor $f : \mathbb{R}^d \to \mathbb{R}^m$ which supervised learning aims to approximate, maps inputs in $\mathbb{R}^d$ (e.g. images, time-series, points) to labels in $\mathbb{R}^m$ (categories, numerical predictions). Given a collection of N sample input-label pairs $\{\vec{x}_i, \vec{y}_i = f(\vec{x}_i)\}_{i=1}^N$, one aims to approximate f using these data points, in such a way that the obtained approximation provides reliable results on points which were not part of the sample dataset.

In this section, we give a brief overview on the continuous-time optimal control viewpoint of deep supervised learning.

2.1. **Artificial neural networks.** Artificial neural networks are dynamical systems whose flow map provides a candidate approximation for the unknown f.

The canonical example of a neural network architecture is the so-called *multi-layer perceptron (MLP)*, which generally takes the form

$$\begin{cases} \mathbf{x}_i^{k+1} = \sigma(w^k \mathbf{x}^k + b^k) & \text{for } \ k \in \{0, \dots, N_{\text{layers}} - 1\} \\ \mathbf{x}_i^0 = \vec{x}_i \in \mathbb{R}^d \end{cases} \tag{2.1}$$

for $i \in \{1, \dots, N\}$. The number of step-sizes $N_{\text{layers}} \geq 1$ is the *depth* of the neural network (2.1), and each time-step k is called a *layer*. For any i, the vector $\mathbf{x}_i^k \in \mathbb{R}^{d_k}$ designates the state at the layer k, while each d_k is referred to as the *width* of the layer k. The optimizable parameters $w^k \in \mathbb{R}^{d_{k+1} \times d_k}$ and $b^k \in \mathbb{R}^{d_k}$ are respectively called the *weights* and *biases* of the network (2.1). Finally, $\sigma \in \text{Lip}(\mathbb{R})$ is a fixed nonlinear *the activation function*. By abuse of notation, we define the vector-valued analog of σ component-wise, namely, $\sigma : \mathbb{R}^d \to \mathbb{R}^d$ is defined by

$$\sigma(\mathbf{x})_j := \sigma(\mathbf{x}_j) \qquad \text{for } j \in \{1, \dots, d\}.$$

Common choices include *sigmoids* such as $\sigma(x) = \tanh(x)$ or $\sigma(x) = \frac{1}{1+e^{-x}}$, and *rectifiers*: $\sigma(x) = \max\{x, ax\}$ for a fixed $0 \leq a < 1$. Whereas rectifiers have several computational benefits (e.g. non-vanishing gradients), in practice, the activation σ is generally selected using cross-validation.

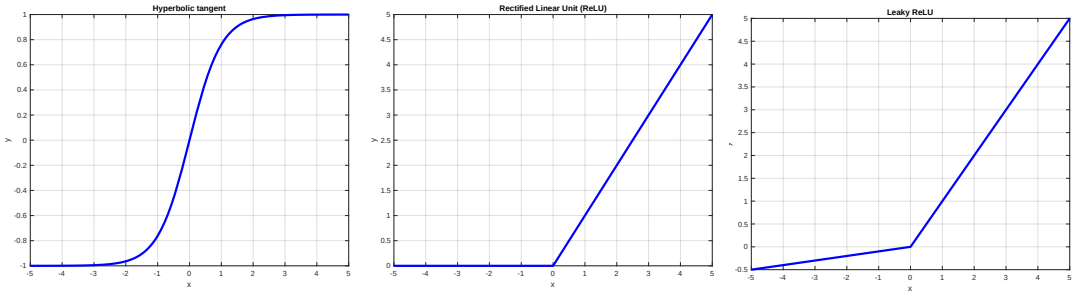

FIGURE 1. Commonly used activation functions include *sigmoids* such as $\sigma(x) = \tanh(x)$ (left), and *rectifiers* such as ReLU: $\sigma(x) = \max\{x, 0\}$ (middle) and Leaky ReLU: $\sigma(x) = \max\{x, 0.1x\}$ (right). All three examples share the property $\sigma(0) = 0$, and a key property which we exhibit in our results is the fact that the rectifiers are positively homogeneous of degree 1: $\max\{\lambda x, \lambda a x\} = \lambda \max\{x, ax\}$ for $\lambda > 0$.

It can readily be seen that the formulation (2.1) coincides with the more conventional formulation of neural networks as compositional structures of parametric linear operators and nonlinearities, as namely $\mathbf{x}_i^{N_{\text{layers}}} = (\sigma \circ \Lambda^k \circ \dots \circ \sigma \circ \Lambda^0)(\vec{x}_i)$, with $\Lambda^k \vec{x} := w^k \vec{x} + b^k$ for $k \in \{0, \dots, N_{\text{layers}}\}$.

Note that the iterative nature of the MLP (2.1) stimulates permuting the order of the parametric linear maps and the nonlinearity σ, to the effect of considering the equivalent, but somewhat simpler system

$$\begin{cases} \mathbf{x}_i^{k+1} = w^k \sigma(\mathbf{x}^k) + b^k & \text{for } \; k \in \{0, \dots, N_{\text{layers}} - 1\} \\ \mathbf{x}_i^0 = \vec{x}_i \in \mathbb{R}^d. \end{cases} \tag{2.2}$$

We will henceforth concentrate on a specific, but rather general class of neural networks called *residual neural networks (ResNets)*. Contrary to the multi-layer perceptrons (2.1) – (2.2), one typically needs to assume that the width d_k is fixed over every layer k, namely $d_k = d$ for every k. We refer to Section 6 for variable width ResNets.

In the fixed width context, a residual neural network generally takes the form

$$\begin{cases} \mathbf{x}_i^{k+1} = \mathbf{x}_i^k + g(u^k, \mathbf{x}_i^k) & \text{for } k \in \{0, \dots, N_{\text{layers}} - 1\} \\ \mathbf{x}_i^0 = \vec{x}_i \in \mathbb{R}^d \end{cases} \tag{2.3}$$

for $i \in \{1, \dots, N\}$, where $\mathbf{x}_i^k \in \mathbb{R}^d$ for any i, k, $u^k := [w^k, b^k]^\top \in \mathbb{R}^{d \times d+d}$ and g is as in (2.1) or (2.2). In this paper we focus on residual networks. This being said, as explained in [Lu et al., 2018], other classes of networks (including specific subclasses of CNNs) can be fit into the residual network framework.

One may readily see that (2.3) corresponds, modulo a scaling factor $\Delta t = \frac{T}{N_{\text{layers}}}$, to the forward Euler discretization of the ordinary differential equation

$$\begin{cases} \dot{\mathbf{x}}_i(t) = g(u(t), \mathbf{x}_i(t)) & \text{in } (0, T) \\ \mathbf{x}_i(0) = \vec{x}_i \in \mathbb{R}^d, \end{cases} \tag{2.4}$$

for $i \in \{1, \dots, N\}$. Here $T > 0$ is a given time horizon. In other words, (2.4) represents the infinite depth/layer analog of (2.3). The parameters $u(t) := [w(t), b(t)]^\top \in \mathbb{R}^{d \times d+d}$ appearing in (2.4) play the role of *controls*. As per what precedes, the nonlinearity g in (2.4) generally takes the form

$$g(u(t), \mathbf{x}_i(t)) := \sigma(w(t)\mathbf{x}_i(t) + b(t)) \tag{2.5}$$

or

$$g(u(t), \mathbf{x}_i(t)) = w(t)\sigma(\mathbf{x}_i(t)) + b(t). \tag{2.6}$$

for $i \in \{1, \dots N\}$. We will address both cases in our analytical study, and emphasize the stark differences between the two. In some literature – e.g. [Chen et al., 2018] – (2.4) is referred to as a *neural ordinary differential equation*.

The above parametrizations are not the lone considered in practice. In fact, one may consider, for instance, combinations of (2.5) and (2.6) which allow intermediate exploration (bottlenecks) in higher dimensions:

$$g(u(t), \mathbf{x}_i(t)) := w_2(t)\sigma(w_1(t)\mathbf{x}_i(t) + b_1(t)) + b_2(t) \tag{2.7}$$

where now, the control is of the form $u(t) := [w_1(t), w_2(t), b_1(t), b_2(t)]^\top$ with $w_1(t) \in \mathbb{R}^{d_{\text{hid}} \times d}$, $w_2(t) \in \mathbb{R}^{d \times d_{\text{hid}}}$, $b_1(t) \in \mathbb{R}^{d_{\text{hid}}}$ and $b_2(t) \in \mathbb{R}^d$. In fact, after a forward Euler discretization, (2.3) with g as in (2.7) roughly coincides with the original ResNet neural network first presented in [He et al., 2016].

Remark 1. Depending on the topological properties of the dataset $\{\vec{x}_i\}_{i=1}^N \subset \mathbb{R}^d$, it may be desirable to consider the dynamical system (2.4) in a bigger dimension than d. Indeed, if we consider time-independent control parameters $u(t) \equiv \overline{u}$, one may only solve a binary classification task via the flow map of (2.4) only if the dataset is a priori linearly separable (i.e. separable by a hyperplane) in the input space $\mathbb{R}^d$. This is due to the fact that two trajectories of an autonomous ODE may not intercept, as observed in [Dupont et al., 2019]. In such a case, for each i one may embed the initial data $\vec{x}_i$ in $\mathbb{R}^{d_{\text{aug}}}$ with $d_{\text{aug}} \geq d$, for instance, by simply concatenating $d_{\text{aug}} - d$ zeroes. This leads to considering the dynamical system (2.4) in $\mathbb{R}^{d_{\text{aug}}}$ instead of $\mathbb{R}^d$. This does not affect the statements of our results, as we may simply relabel the initial data.

2.2. **Learning.** For an input sample $\vec{x}_i$, the *prediction* of the neural network (2.4) is a flattening of the states at time T, of the form $\varphi(\mathbf{x}_i(T))$ for a parametrized smooth map $\varphi : \mathbb{R}^d \to \mathbb{R}^m$. In general,

$$\begin{aligned} \varphi(x) &:= \operatorname{softmax}(\theta_1 x + \theta_2) \qquad && \text{(classification)}, \\ \varphi(x) &:= \theta_1 x + \theta_2 && \text{(regression)}, \end{aligned} \tag{2.8}$$

where $\theta_1 \in \mathbb{R}^{m\times d}$ and $\theta_2 \in \mathbb{R}^m$ are optimizable parameters, and $\operatorname{softmax}(z)_j = \frac{e^{z_j}}{\sum_{\ell=1}^m e^{z_\ell}}$ for $z \in \mathbb{R}^m$ and $j \in \{1, \dots, m\}$. In the context of binary classification, namely $m = 1$ with $\vec{y}_i = \pm 1$, one may also use $\varphi(x) := \tanh(\theta_1 x + \theta_2)$.

The aim of modern deep supervised learning consists in choosing the control parameters w, b so that $\varphi(\mathbf{x}_i(T))$ most closely resembles $\vec{y}_i$ for $i \in \{1, \dots, N\}$. To this end, an open-loop (i.e. offline) optimal control approach is usually considered. The supervised learning problem in the continuous-time dynamical systems framework may then be stated as[4]

$$\inf_{\substack{[w,b]^\top \\ \text{subject to (2.4)}}} \frac{1}{N} \sum_{i=1}^N \operatorname{loss}\big(\varphi(\mathbf{x}_i(T)), \vec{y}_i\big). \tag{2.9}$$

Here $\operatorname{loss}(\cdot,\cdot) : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}_+$ is a given continuous function – examples include $\operatorname{loss}(x, y) := \|x - y\|_{\ell^p}^p$ for $p = 2$ (mean squared error) and $p = 1$ (sparsity), as well as $\operatorname{loss}(x, y) := \log\left(1 + e^{-\langle x,y\rangle}\right)$ (logistic loss). Problem (2.9) is called *empirical risk minimization*.

Problem (2.9) is a special case of a class of general deterministic optimal control problems for nonlinear ODEs, see [Bertsekas, 1995, Trélat, 2005]. However, the ultimate goal of supervised learning is to construct a function which will not only fit well the dataset $\{\vec{x}_i.\vec{y}_i\}$ but also perform (i.e. *generalize*) reliably on other points $\vec{x}$ outside of the training dataset. This is reflected in the fact that (2.9) represents an approximation for the original stochastic, *expected risk minimization* problem

$$\inf_{\substack{[w,b]^\top \\ \text{subject to (2.4)}}} \int_{\mathbb{R}^d\times\mathbb{R}^m} \operatorname{loss}\left(\varphi(\mathbf{x}_{\vec{x}}(T)), \vec{y}\right) \, \mathrm{d}\rho(\vec{x}, \vec{y}) = \inf_{\substack{[w,b]^\top \\ \text{subject to (2.4)}}} \mathbb{E}_\rho\Big[\operatorname{loss}\left(\varphi(\mathbf{x}_\cdot(T)), \cdot\right)\Big],$$

with $\mathbf{x}_{\vec{x}}$ denoting the solution to (2.4) with initial datum $\vec{x}$. Here $\rho : \mathbb{R}^d \times \mathbb{R}^m \to [0, 1]$ is an unknown probability distribution, from which one samples the training dataset $\{\vec{x}_i, \vec{y}_i\}_{i=1}^N$.

To have a clearer understanding of overfitting phenomena (see the subsection just below as well as Theorem 5.1 for a justification of explicit regularization), it is desirable to consider the *regularized* empirical risk minimization problem

$$\inf_{\substack{[w,b]^\top \\ \text{subject to (2.4)}}} \frac{1}{N} \sum_{i=1}^N \operatorname{loss}\big(\varphi(\mathbf{x}_i(T)), \vec{y}_i\big) + \int_0^T \ell\left(\mathbf{x}(t), [w(t), b(t)]^\top\right) \, \mathrm{d}t.$$

[4]As θ_1, θ_2 appearing in φ are constant, we omit them from the statement of the optimization problem for the sake of notation simplicity, as in this work we are principally interested in the dynamics of the time-dependent parameters. This is done without loss of generality, as we may always append them to the time-dependent control parameters and relabel a posteriori.

Here $\ell \in C^\infty(\mathbb{R}^{d\times N} \times \mathbb{R}^{d_u}; \mathbb{R}_+)$ is a regularizer, such as those appearing in (1.3) or (1.4). Whereas in most of the machine learning literature a regularization of the stacked state $\mathbf{x}(t) = [\mathbf{x}_1(t), \dots, \mathbf{x}_N(t)]^\top$ over the entire time horizon $[0, T]$ is generally not considered, we will address both cases in what follows.

The need for regularization: existence of minimizers. As indicated just above, the training of a neural network may include explicit regularization (penalization) of the parameters, and several different regularizers have been proposed in the literature (see [Goodfellow et al., 2016, Section 7]). In order to shed some light on the necessity of explicit regularization, we henceforth follow [Thorpe and van Gennip, 2018, Celledoni et al., 2020], where the following elementary but illustrative example is presented.

We use the ResNet (2.3) with $N = d = m = N_{\text{layers}} = 1$, $\text{loss}(z) := (z-1)^2$, $x_1 = 0$ and $\sigma \equiv \tanh$. The training problem (2.9) simplifies to

$$\min_{[w,b]^\top \in \mathbb{R}\times\mathbb{R}} (\tanh(b) - 1)^2. \tag{2.10}$$

Since $\tanh(\mathbb{R}) = (-1, 1)$ we see that fundamental problems appear: (2.10) does not admit a solution, as in particular, minimizing sequences are not bounded. Indeed, let $w^n := 0, b^n := n$ and $\mathcal{J}(w, b) := (\tanh(b) - 1)^2$, then $\lim_{n\to+\infty} \mathcal{J}(w^n, b^n) = 0$ but $\{w^n, b^n\}_{n=1}^{+\infty}$ is unbounded and does not even contain a convergent subsequence. Thus one cannot a priori expect the training algorithm to converge.

To overcome the aforementioned problem, it is sufficient to add a regularization of the controls which is *coercive*. This would imply that exhibited minimizing sequences are bounded, which in turn, in reflexive Banach spaces, is sufficient to guarantee at least a convergent subsequence by the Banach-Alaoglu theorem.

The regularization term may impose additional properties on the optimized control parameters: common examples in the discrete-time neural network context such as (2.1) – (2.2) – (2.3) include the squared ℓ^2–norm $\|u\|_2^2 = \sum_i |u_i|^2$ (referred to as *Tikhonov* regularization or *weight-decay*), where controls manifest small coefficients, and the ℓ^1–norm $\|u\|_1 = \sum_i |u_i|$ (called *Lasso* in statistical contexts), which induces sparse coefficients, [Ng, 2004, Ranzato et al., 2007]. The continuous-time dynamical system interpretation of residual networks also motivates other norms such as the squared H^1-norm and its discrete counterpart (see e.g. [Haber and Ruthotto, 2017, Thorpe and van Gennip, 2018], and also what follows), which enhance the regularity of parameters across layers.

As discussed in [Zhang et al., 2016], explicit control parameter regularization does neither completely or adequately explain the generalization capabilities of neural networks (as stipulated in subsequent works on the implicit bias of gradient descent, as discussed in Section 3), although it has been empirically shown to improve generalization performance and, as reported by [Krizhevsky et al., 2012], ℓ^2-regularization may aid the optimization process, showing the lack of fundamental understanding regarding its role.

Returning to the continuous-time setting, we note that by using the classical direct method, we may readily prove the existence of minimizer for a class of the learning problems we considered in this work. We nonetheless sketch the proof as to indicate the intrinsic difference between the neural networks (3.3) and (3.2), in

particular, to indicate why even only L^2–parameter regularization may a priori not be sufficient. For the sake of cohesion, we will concentrate our efforts on a specific form of regularization, although the arguments are significantly more general.

Proposition 2.1. *Let $T > 0$, $\alpha > 0$ and $\beta \geq 0$. Let $\mathbf{x}^0 \in \mathbb{R}^{d_x}$ and $\mathbf{x}_d \in \mathbb{R}^{d_x}$ be fixed, and let $\phi \in C^0(\mathbb{R}^{d_x}; \mathbb{R}_+)$ be given. Consider the functional $\mathcal{J}_T : H^k(0,T;\mathbb{R}^{d_u}) \to \mathbb{R}_+$ defined by*

$$\mathcal{J}_T(w,b) := \frac{1}{N}\sum_{i=1}^{N} \mathrm{loss}\big(\varphi(\mathbf{x}_i(T)), \vec{y}_i\big) + \frac{\alpha}{2}\left\|[w,b]^\top\right\|^2_{H^k(0,T;\mathbb{R}^{d_u})} + \frac{\beta}{2}\int_0^T \|\mathbf{x}(t) - \mathbf{x}_d\|^2 \,\mathrm{d}t,$$

where $\mathbf{x} \in C^0([0,T];\mathbb{R}^{d_x})$ is the associated solution to (3.3) with $k = 0$, or (3.2) with $k = 1$. The functional $\mathcal{J}_T$ admits a global minimizer $\left[w^\dagger, b^\dagger\right]^\top \in H^k(0,T;\mathbb{R}^{d_u})$.

For the sake of completeness, we sketch the proof, postponed to Appendix A.1.

Remark 2 (Sobolev regularization)**.** As per Proposition 2.1, the optimization problems for continuous-time neural networks of the form (3.2) – (3.3) can be shown to admit a solution. We stress however the need for considering a H^1–regularization in the case of (3.2), as otherwise, we may not a priori guarantee the existence of a global minimizer. Indeed, an issue arises due to the specific nonlinear form of the neural network (3.2), which may be an impediment for passing to the limit in the equation using only weak convergences.

To be more precise, when considering only an L^2–regularization of the controls for (3.2), we see that the nonlinearity σ may be an impediment in applying the weak convergences of the minimizing sequences $\{w_n\}_{n=1}^{+\infty}$ and $\{b_n\}_{n=1}^{+\infty}$. We can illustrate this with a simple example as the one just above: if $\sigma(x) = \max\{x, 0\}$, we recall that $\sin(nx) \rightharpoonup 0$ weakly in $L^2(0, 2\pi)$, but $|\sin(nx)| = 2\sigma(\sin(nx)) - \sin(nx)$ and $\int_0^{2\pi} |\sin(nx)| = 4$, whence $\sigma(\sin(nx)) \rightharpoonup 0$ cannot hold. In view of this example, we see that to conclude on the existence of a minimizer in the case where the dynamics are given by (3.2), further a priori analysis on the oscillations of the minimizers is required. This issue is specific to the continuous-time setting, as in the discrete-time thus finite dimensional optimization setting, weak and strong convergences coincide.

3. Asymptotics without tracking

Throughout the paper, we will focus on continuous-time neural networks (neural ODEs) given by (2.4) with g as in (2.5) or (2.6). The results can thence be extrapolated to the case when g is parametrized by (2.7) whenever w_1 and b_1 (resp. w_2, b_2) are time-independent. As it will be rather convenient to work with the full stacked state trajectory $\mathbf{x}(t) = [\mathbf{x}_1(t), \dots, \mathbf{x}_N(t)]^\top$, we introduce some further notation. We shall henceforth denote

$$d_u := d \times d + d, \qquad d_x := d \times N.$$

Moreover, given $w \in \mathbb{R}^{d\times d}$ and $b \in \mathbb{R}^d$, we shall write

$$\mathbf{w} := \begin{bmatrix} w & & \\ & \ddots & \\ & & w \end{bmatrix} \in \mathbb{R}^{d_x \times d_x}, \qquad \mathbf{b} := \begin{bmatrix} b \\ \vdots \\ b \end{bmatrix} \in \mathbb{R}^{d_x}. \tag{3.1}$$

In view of the above discussion and noting (3.1), we will consider stacked neural ODEs in $\mathbb{R}^{d_x}$ such as

$$\begin{cases} \dot{\mathbf{x}}(t) = \sigma(\mathbf{w}(t)\mathbf{x}(t) + \mathbf{b}(t)) & \text{for } t \in (0,T) \\ \mathbf{x}(0) = \mathbf{x}^0 \in \mathbb{R}^{d_x}, \end{cases} \tag{3.2}$$

and

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{w}(t)\sigma(\mathbf{x}(t)) + \mathbf{b}(t) & \text{for } t \in (0,T) \\ \mathbf{x}(0) = \mathbf{x}^0 \in \mathbb{R}^{d_x}. \end{cases} \tag{3.3}$$

Throughout the remainder of this work, we will work under the following couple of assumptions.

Assumption 1. *We henceforth assume that we are given a training dataset*

$$\{\vec{x}_i, \vec{y}_i\}_{i=1}^N \subset \mathbb{R}^{d\times N} \times \mathbb{R}^{m\times N},$$

with $\vec{x}_i \neq \vec{x}_j$ for $i \neq j$, as well as a time horizon $T > 0$. Any initial data $\mathbf{x}^0 \in \mathbb{R}^{d\times N}$ for the systems under consideration will thence take the form $\mathbf{x}^0 = [\vec{x}_1, \dots, \vec{x}_N]^\top$.

Assumption 2. *Unless stated otherwise, we fix an activation function σ satisfying*

$$\sigma \in \mathrm{Lip}(\mathbb{R}) \quad \textit{and} \quad \sigma(0) = 0.$$

In this section, we consider the common setting of modern supervised learning, namely the problem of regularized empirical risk minimization. For simplicity of notation, we henceforth denote the training error by

$$\phi(\mathbf{x}(T)) := \frac{1}{N}\sum_{i=1}^N \mathrm{loss}\big(\varphi(\mathbf{x}_i(T)), \vec{y}_i\big), \tag{3.4}$$

where $\varphi \in C^\infty(\mathbb{R}^d;\mathbb{R}^m)$ is as in (2.8). For fixed $\alpha > 0$ we will study the long-time behavior of global minimizers to the functional

$$\mathcal{J}_T(w,b) = \phi(\mathbf{x}(T)) + \frac{\alpha}{2}\left\|[w,b]^\top\right\|^2_{H^k(0,T;\mathbb{R}^{d_u})} \tag{3.5}$$

where $\mathbf{x} \in C^0([0,T];\mathbb{R}^{d_x})$ is the unique solution to either (3.3) or (3.2) corresponding to the control parameters $[w,b]^\top \in H^k(0,T;\mathbb{R}^{d_u})$, noting (3.1). Our results will only require $\mathrm{loss}(\cdot,\cdot) \in C^0(\mathbb{R}^m \times \mathbb{R}^m;\mathbb{R}_+)$.

As noted in Remark 2, we stress the need for considering a H^1–regularization in the case of (3.2), as otherwise, due to the specific nonlinear form of the neural ODE in (3.2), we may not a priori guarantee the existence of a global minimizer for $\mathcal{J}_T$. This issue is specific to the continuous-time setting, as in the discrete-time thus finite dimensional optimization setting, weak and strong convergences coincide.

We begin by laying out a couple of relevant definitions.

Definition 3.1 (Reachable set)**.** For any $\mathbf{x}^0 \in \mathbb{R}^{d_x}$ and any $T > 0$, we define the *reachable set* from $\mathbf{x}^0$ in time T by

$$\mathcal{R}_T(\mathbf{x}^0) := \left\{\mathbf{x}^1 \in \mathbb{R}^{d_x} \ : \ \exists u := [w,b]^\top \in H^k(0,T;\mathbb{R}^{d_u}) \text{ such that } \ \mathbf{x}(T) = \mathbf{x}^1\right\},$$

where $\mathbf{x} \in C^0([0,T];\mathbb{R}^{d_x})$ is the solution to (3.3) (resp. (3.2)), with $[\mathbf{w},\mathbf{b}]$ as in (3.1), and $k = 0$ in the case of (3.3) (resp. $k = 1$ in the case of (3.2)).

As per its name, the reachable set $\mathcal{R}_T(\mathbf{x}^0)$ is the set of points in $\mathbb{R}^{d_x}$ which can be reached by some trajectory of (3.3) or (3.2). We note that there are cases where the reachable set is a strict subset of $\mathbb{R}^{d_x}$. For instance, consider (3.2) with ReLU activation: for any $t \in [0,T]$,

$$\mathbf{x}(t) = \mathbf{x}^0 + \int_0^t \sigma(\mathbf{w}(\tau)\mathbf{x}(\tau) + \mathbf{b}(\tau))\,\mathrm{d}\tau \geq \mathbf{x}^0,$$

whence the reachable subset is a subset of a cone of $\mathbb{R}^{d_x}$. This is in fact a general artefact of (3.2) with an activation function $\sigma(\mathbb{R}) \subsetneq \mathbb{R}$.

Definition 3.2 (Minimal cost)**.** For any $\mathbf{x}^0 \in \mathbb{R}^{d_x}$, $\mathbf{x}^1 \in \mathbb{R}^{d_x}$ and $T > 0$ satisfying $\mathbf{x}^1 \in \mathcal{R}_T(\mathbf{x}^0)$, we define the *minimal cost* of steering a trajectory from $\mathbf{x}^0$ to $\mathbf{x}^1$ in time T by

$$\kappa_T(\mathbf{x}^0,\mathbf{x}^1) := \inf_{\substack{[w,b]^\top \in H^k(0,T;\mathbb{R}^{d_u}) \\ \text{subject to (3.2) (resp. (3.3))} \\ \text{and} \\ \mathbf{x}(0)=\mathbf{x}^0,\ \mathbf{x}(T)=\mathbf{x}^1}} \left\|[w,b]^\top\right\|^2_{H^k(0,T;\mathbb{R}^{d_u})},$$

where $k = 0$ in the case of (3.3) and $k = 1$ in the case of (3.2).

We note that κ_T is not necessarily symmetric, so a priori it does not define a distance.

We may state the main result of this section.

Theorem 3.1. *Let $\mathbf{x}^0 = [\vec{x}_1,\dots,\vec{x}_N]^\top \in \mathbb{R}^{d_x}$, $\alpha > 0$, and assume that $\phi \in C^0(\mathbb{R}^{d_x};\mathbb{R}_+)$ satisfies*

$$\mathcal{R}_{T_0}(\mathbf{x}^0) \cap \operatorname{argmin}(\phi) \neq \varnothing$$

for some some time $T_0 > 0$. For any $T > 0$, let $\mathbf{x}^T \in C^0([0,T];\mathbb{R}^{d_x})$ be the unique solution to (3.3) (resp. (3.2) with σ positively homogeneous[5] of degree 1) associated to control parameters $\left[w^T,b^T\right]^\top \in H^k(0,T;\mathbb{R}^{d_u})$ minimizing (3.5), where $k = 0$ in the case of (3.3), and $k = 1$ in the case of (3.2).

Then, there exists a sequence $\{T_n\}_{n=1}^{+\infty}$, with $T_n > 0$ and $T_n \to +\infty$ as $n \to +\infty$, and $\mathbf{x}^\dagger \in \operatorname{argmin}(\phi)$ such that

$$\left\|\mathbf{x}^{T_n}(T_n) - \mathbf{x}^\dagger\right\| \longrightarrow 0 \qquad \text{as } n \to +\infty. \tag{3.6}$$

For any $n \geq 1$, set

$$w_n(t) := \frac{T_n}{T_0} w^{T_n}\left(t\frac{T_n}{T_0}\right) \qquad \text{for } t \in [0,T_0],$$
$$b_n(t) := \frac{T_n}{T_0} b^{T_n}\left(t\frac{T_n}{T_0}\right) \qquad \text{for } t \in [0,T_0].$$

Then

$$\left\|[w_n,b_n]^\top - \left[w^\dagger,b^\dagger\right]^\top\right\|_{H^k(0,T_0;\mathbb{R}^{d_u})} \longrightarrow 0 \qquad \text{as } n \to +\infty,$$

[5]Meaning $\sigma(\lambda\cdot) = \lambda\sigma(\cdot)$ for all $\lambda > 0$.

where $[w^\dagger, b^\dagger]^\top \in H^k(0,T_0;\mathbb{R}^{d_u})$ *is a solution to the minimization problem*

$$\inf_{\substack{[w,b]^\top \in H^k(0,T_0;\mathbb{R}^{d_u}) \\ \textit{subject to (3.2) (resp. (3.3))} \\ \textit{and} \\ \mathbf{x}(T_0) \in \operatorname{argmin}(\phi)}} \frac{\alpha}{2} \left\| [w,b]^\top \right\|^2_{H^k(0,T_0;\mathbb{R}^{d_u})}. \tag{3.7}$$

Discussion. A common writing in the machine learning literature that neural networks operating in the *overparametrization* regime – namely, neural networks with significantly more trainable parameters than the number N of training data – perform well experimentally precisely because they fit the entire training dataset, namely the training error ϕ is minimal ($\sim$ zero).

When the underlying neural ODE is discretized using a sufficiently small time-step (e.g. ResNet, namely an explicit Euler scheme with $\Delta t = \frac{T}{N_{\text{layers}}}$ with, say, $N_{\text{layers}} \sim T^q$ with $q \geq 1$), Theorem 3.1 stipulates that when $T \gg 1$, which designates an overparametrized regime, residual neural networks fit almost all the training data as they indeed approach a minimizer of the training error ϕ, but do so in the *optimal* way[6], namely, by using control parameters having the smallest possible L^2–norm. Heuristically, this stipulates that the trained predictor $\vec{x}_i \mapsto \varphi(\mathbf{x}_i(T))$ is the one which has the least variations possible among the ones trained by minimizing the L^2–regularized cost, thus indicating a possible *generalization*–like property for deep residual networks in this context. Our techniques are however purely analytical and do not provide any estimates of statistical nature on the generalization error via commonly used metrics such as the VC dimension [Vapnik, 2013], Rademacher complexity [Bartlett and Mendelson, 2002], and uniform stability [Mukherjee et al., 2006, Bousquet and Elisseeff, 2002, Poggio et al., 2004].

The insight from Theorem 3.1 could perhaps be compared to other convergence results of generalization nature such as the *implicit bias property of gradient descent* [Zhang et al., 2016, Soudry et al., 2018, Gunasekar et al., 2018, Chizat and Bach, 2018, Chizat and Bach, 2020]. This property indicates that in the overparametrized regime, after training a neural network with gradient-based methods until zero training error, without requiring any explicit parameter regularization, among the many classifiers which overfit on the training dataset, the algorithm selects the one which performs best on the test dataset. Unlike our a priori qualitative and quantitative study of global minimizers of the cost functional, these kinds of results rely specifically on the descent scheme for finding the optimal parameters. Examples of "best solutions" include the minimal squared ℓ^2–norm solution for linear regression solved via gradient descent or, in the context of linear logistic regression trained on linearly separable data, the max margin support vector machine solution [Soudry et al., 2018]. Such minimum norm or maximum margin solutions are very special among all solutions that fit the training data, and in particular can ensure generalization [Bartlett and Mendelson, 2002, Kakade et al., 2009]. In [Chizat and Bach, 2018, Chizat and Bach, 2020] the overparametrization notion is approached from the point of view of the width of the neural network, unlike our depth-inspired perspective. The authors consider a 2-layer MLP with ReLU activation, and exhibit the Wasserstein gradient flow formulation of the descent scheme

[6]In fact, at least in the case of (3.3), the quantity appearing in (3.7) can be interpreted as the *geodesic distance* from $\mathbf{x}^0$ to $\arg\min(\phi)$ in the reachable set $\mathcal{R}_{T_0}(\mathbf{x}^0)$.

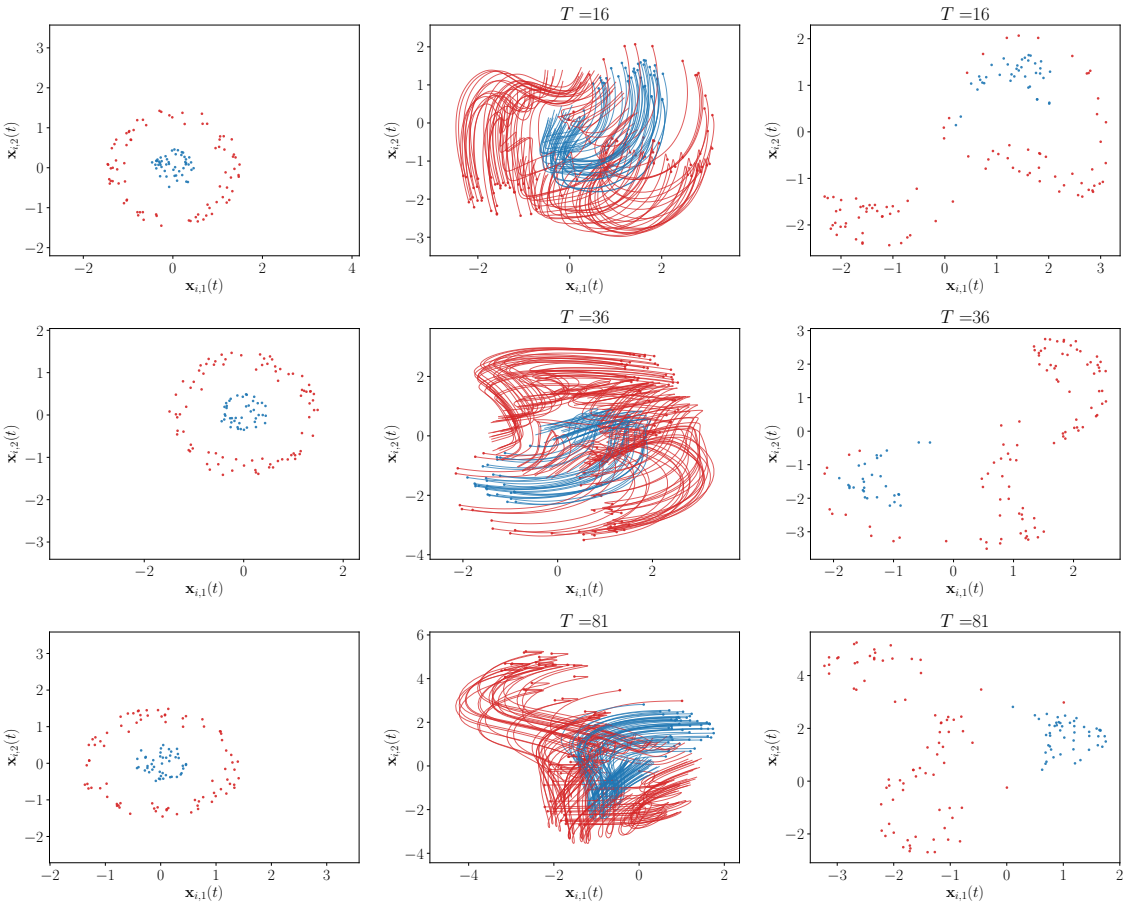

FIGURE 2. We see a manifestation of Theorem 3.1 on a simple binary classification task. Namely, we observe that when the time horizon increases, the outputs $\mathbf{x}_i^T(T)$ (right) of learned trajectories $\mathbf{x}_i^T$ (middle) separate increasingly more of the input data $\vec{x}_i$ (left) for $1 \leq i \leq 128$, and hence are nearer to the zero training error regime, as desired. Here we took $N_{\text{layers}} = \left\lfloor T^{\frac{3}{2}} \right\rfloor$ and thus $\Delta t = \frac{1}{\sqrt{T}}$, and we consider $\alpha = 1$. Noticeably, T is inversely proportionate to α.

yielding controls, and they consequently prove that these controls approach global minimizers of the cost functional when the width increases, with the global minimizer being characterized as a max-margin classifier in a certain non-Hilbertian space of functions, leading to generalization bounds.

The nature of our proof is however different from the existing results such as those in the works cited above, whereas the nature of the result is as well, since we clearly exhibit the explicit L^2–regularization of the control parameters. Moreover, Theorem 3.1 is an *a priori* result, as it is independent of the optimization method chosen for finding a minimizer.

Let us also briefly comment on works addressing the related issue of *deep limits*, e.g. [Thorpe and van Gennip, 2018] (see also [Avelin and Nyström, 2020]). Notably in [Thorpe and van Gennip, 2018], the authors show, via Γ-convergence arguments,

that the optimal control parameters in the discrete-time context converge to those of the continuous-time context when the time-step converges to 0. The latter is interpreted as an infinite layer limit when the final time horizon T in the continuous-time context is fixed (equal to 1). Our result is of different nature. Rather than aim to prove that the discrete-time controls converge to the continuous-time ones, we exhibit the continuous-time neural ODE representation, for which the final time horizon clearly commands the number of layers for the associated time-discretization, and aim to characterize the possible phenomena which arise whenever this time horizon increases.

Idea of proof. The proof of Theorem 3.1 may be found in Section 3.2.2. Let us motivate the main underlying idea. For simplicity, let us assume that $T_0 = 1$.

The key point is the fact that, under the assumptions made in the statement, both of the underlying dynamics (3.2) and (3.3) will be positively homogeneous with respect to the control parameters $w(t)$ and $b(t)$. Namely, both (3.2) and (3.3) (noting (3.1)) can be written as

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), w(t), b(t)) & \text{in } (0,T) \\ \mathbf{x}(0) = \mathbf{x}^0, \end{cases} \tag{3.8}$$

where $\mathbf{f}(\mathbf{x}, \lambda w, \lambda b) = \lambda \mathbf{f}(\mathbf{x}, w, b)$ for $\lambda > 0$. Whilst in the case of (3.3) this homogeneity property holds for any activation function σ, we require σ to be positively homogeneous of degree 1 for neural networks such as (3.2). This includes rectifiers, but excludes sigmoids.

Now a simple computation (see Lemma 3.1) leads to noting that, given some control parameters $u^1 := [w^1, b^1]^\top$ and the solution $\mathbf{x}^1$ to

$$\begin{cases} \dot{\mathbf{x}}^1(t) = \mathbf{f}(\mathbf{x}^1(t), w^1(t), b^1(t)) & \text{in } (0,1) \\ \mathbf{x}^1(0) = \mathbf{x}^0, \end{cases} \tag{3.9}$$

the control $u^T(t) := \frac{1}{T} u^1(\frac{t}{T})$ for $t \in [0,T]$ is such that $\mathbf{x}^T(t) := \mathbf{x}^1(\frac{t}{T})$ solves (3.8). Whence, considering the case of (3.3) and thus $k = 0$ for simplicity, we see that

$$\begin{aligned} &\inf_{\substack{u^T = [w^T, b^T]^\top \in L^2(0,T;\mathbb{R}^{d_u}) \\ \text{subject to (3.8)}}} \phi(\mathbf{x}^T(T)) + \frac{\alpha}{2} \int_0^T \left\| u^T(t) \right\|^2 \, \mathrm{d}t \\ &= \frac{1}{T} \inf_{\substack{u^T = [w^T, b^T]^\top \in L^2(0,T;\mathbb{R}^{d_u}) \\ \text{subject to (3.8)}}} T\phi(\mathbf{x}^T(T)) + \frac{\alpha}{2} \int_0^1 \left\| T u^T(sT) \right\|^2 \, \mathrm{d}s \qquad (3.10) \\ &= \frac{1}{T} \inf_{\substack{u^1 = [w^1, b^1]^\top \in L^2(0,1;\mathbb{R}^{d_u}) \\ \text{subject to (3.9)}}} T\phi(\mathbf{x}^1(1)) + \frac{\alpha}{2} \int_0^1 \left\| u^1(s) \right\|^2 \, \mathrm{d}s \qquad (3.11) \end{aligned}$$

Neglecting the factor $\frac{1}{T}$ for the time being, we see from (3.10) that when $T \gg 1$, the states $\mathbf{x}^T(T)$ ought to approach a minimizer of ϕ, whereas from (3.11), that the rescaled control parameters in the right integrand of (3.10) ought to approach the solution to (3.7). Our proof follows these lines, whilst using common compactness arguments to justify the stated convergences.

3.1. **An algorithm inspired by Theorem 3.1.** In view of Theorem 3.1, the following learning strategy, which applies both to the continuous-time as well as to the discrete-time setting, can be naturally deduced. The idea is to start the training with a shallow neural network and increase the depth progressively until the training error is close to zero.

In the discretized setting, we fix the weight decay parameter $\alpha > 0$, the time-step $\Delta t > 0$ and a starting number of layers $N_0 \geq 1$. The algorithm would consist of the following steps:

Step 1. Train the (probably) underparametrized neural network with N_0 layers by solving the optimal control problem

$$\inf_{\substack{[w_0,b_0]^\top \in \mathbb{R}^{d_u \times N_0} \\ \text{subject to (2.3)}}} \frac{1}{N} \sum_{i=1}^{N} \text{loss}\left(\varphi\left(\mathbf{x}_i^{N_0}\right), \vec{y}_i\right) + \frac{\alpha}{2} \left\|[w_0, b_0]^\top\right\|^2. \tag{3.12}$$

We recall that in the discrete-time setting, the control parameters take the form

$$[w_0, b_0]^\top = \left[\left[w_0^0, b_0^0\right]^\top, \ldots, \left[w_0^{N_0-1}, b_0^{N_0-1}\right]^\top\right]^\top \in \mathbb{R}^{d_u \times N_0},$$

where w_0^k and b_0^k are the weights and the biases in each layer $k \in \{0, \ldots, N_0\}$ of the neural network.

Since the number of layers – and thus the number of parameters – is not too large, we would likely obtain a big training error due to the presence of the parameter regularization term. However, the small number of parameters would enhance the training speed.

Step 2. We use the parameters obtained in the previous step to initialize the learning problem of a deeper neural network with $N_1 > N_0$ layers, in order to reduce the training error.

Because of the number of layers N_0 and the fixed time-step Δt, (3.12) can be considered as an approximation of the corresponding continuous-time version with time-horizon $T_0 := \Delta t\, N_0$. By using, for instance, an affine interpolation, we can obtain a control pair $\left[\widehat{w}_0(\cdot), \widehat{b}_0(\cdot)\right]^\top \in C^0([0, T_0]; \mathbb{R}^{d_u})$ such that

$$\left[\widehat{w}_0(k\,\Delta t), \widehat{b}_0(k\,\Delta t)\right]^\top = [w_0^k, b_0^k]^\top, \qquad \text{for all } k \in \{0, \ldots, N_0 - 1\}.$$

Using the scaling arguments of Lemma 3.1, we can then obtain a control defined in the new time-interval $[0, T_1]$, with $T_1 := \Delta t\, N_1$ by setting

$$[\widehat{w}_1(t), \widehat{b}_1(t)]^\top := \frac{T_0}{T_1} \left[\widehat{w}_0\left(t\frac{T_0}{T_1}\right), \widehat{b}_0\left(t\frac{T_0}{T_1}\right)\right]^\top.$$

Then, we train the new neural network which has N_1 layers, using as initial parameters

$$\left[w_1^k,\, b_1^k\right]^\top := \left[\widehat{w}_1(k\,\Delta t), \widehat{b}_1(k\,\Delta t)\right]^\top, \qquad \text{for all } k \in \{0, \ldots, N_1 - 1\}.$$

Step 2 is then iteratively applied by increasing the number of layers until the training error is sufficiently small (say, up to a user-specified tolerance).

We stress that following this procedure, Theorem 3.1 ensures that as we increase T, the algorithm approaches the overparametrized regime as well as the zero training error regime, and it ensures that the obtained control parameters in this overparametrized regime are of minimal L^2–norm.

This strategy has two main advantages:

- On one hand, the training procedure is started with few parameters, since we are considering a shallow neural network, whose depth we then increase progressively.
- On the other hand, the number of layers is increased only until the training error is near zero, hence we avoid the implementation of unnecessary supplementary layers, which would increase the number of parameters beyond what is strictly necessary.

The algorithm presented just above is in fact a *greedy* algorithm, more precisely greedy in terms of the number of layers.

Greedy, layer-adaptive algorithms have already been investigated in the machine learning literature, sometimes under the common umbrella of *pre-training* algorithms [Goodfellow et al., 2016, Chapter 15]. Indeed, directly training a neural network to solve a complex task can be computationally unfeasible, and it is thus more effective and faster to train a simpler (shallower) network. They come with a strong theoretical backbone particularly in the parameter-identification framework, with convergence rates characterized by the Kolmogorov width (see e.g. [Barron et al., 2008, Binev et al., 2011, Cohen and DeVore, 2015]).

Greedy algorithms, whilst not guaranteeing a globally optimal solution, are computationally much cheaper – because of the fact that they break the full problem into many subcomponents, with each one being solved for the optimal solution in isolation – and still provide acceptable results.

Greedy (supervised) pre-training algorithms are ubiquitous in deep learning, dating back to the original idea proposed in [Bengio et al., 2007], where each subproblem consists of a supervised learning training task involving only a subset of the layers used in the final neural network. Another purpose of greedy algorithms may be to provide an initialization for the complete training algorithm, in order to greatly speed up this training and improve the quality of the solution. Such ideas have been exploited to great effect in [Simonyan and Zisserman, 2014]. Another application of greedy algorithms is proposed in [Yu et al., 2010], where the authors use the outputs of the previously trained MLPs with few layers, combined with the input samples of the training dataset, as inputs for each added step. We refer to [Goodfellow et al., 2016, Chapter 15] for a more detailed presentation and references.

3.2. **Proof of Theorem 3.1.** We note that both (3.3) and (3.2) can be written in the compact form

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}(w(t), b(t), \mathbf{x}(t)) & \text{in } (0,T) \\ \mathbf{x}(0) = \mathbf{x}^0 \in \mathbb{R}^{d_x}, \end{cases} \tag{3.13}$$

with

$$\mathbf{f}(0,0,\mathbf{x}) = 0, \qquad \mathbf{f}(\lambda w, \lambda b, \mathbf{x}) = \lambda \mathbf{f}(w,b,\mathbf{x}) \quad \text{for } \lambda > 0. \tag{3.14}$$

For the sake of simplicity, we will sometimes refer to $u := [w, b]^\top$ as *the control* of the dynamical system, in accordance with control theory vocabulary.

3.2.1. *Preliminaries.* We begin by setting forth the following short but key lemma.

Lemma 3.1. *Let $T_0 > 0$ and $\left[w^{T_0}, b^{T_0}\right]^\top \in L^2(0, T_0; \mathbb{R}^{d_u})$ be given, and let $\mathbf{x}^{T_0} \in C^0([0, T_0]; \mathbb{R}^{d_x})$ be the unique solution to*

$$\begin{cases} \dot{\mathbf{x}}^{T_0}(t) = \mathbf{f}(w^{T_0}(t), b^{T_0}(t), \mathbf{x}^{T_0}(t)) & \text{in } (0, T_0) \\ \mathbf{x}^{T_0}(0) = \mathbf{x}^0 \in \mathbb{R}^{d_x}, \end{cases} \tag{3.15}$$

(i.e. (3.13) on $(0, T_0)$) with $\mathbf{f}$ as in either (3.3) or (3.2), thus satisfying (3.14). Let $T > 0$, and define

$$w^T(t) := \frac{T_0}{T} w^{T_0}\left(t\frac{T_0}{T}\right), \qquad b^T(t) := \frac{T_0}{T} b^{T_0}\left(t\frac{T_0}{T}\right) \qquad \text{for } t \in [0, T], \tag{3.16}$$

and

$$\mathbf{x}^T(t) := \mathbf{x}^{T_0}\left(t\frac{T_0}{T}\right) \qquad \text{for } t \in [0, T]. \tag{3.17}$$

Then $\mathbf{x}^T \in C^0([0, T]; \mathbb{R}^{d_x})$ is the unique solution to (3.13) (with the same $\mathbf{f}$ as in (3.15)) associated to $\left[w^T, b^T\right]^\top$.

This sort of time-scaling in the context of *driftless control affine* systems is commonly used in control theoretical contexts – a canonical example is the proof of the Chow-Rashevskii controllability theorem, see [Coron, 2007, Chapter 3, Section 3.3]. We sketch the short proof for completeness.

Proof. Using the fact that $\mathbf{x}^{T_0}$ is the solution to (3.15), the change of variable $\tau = s\frac{T}{T_0}$ as well as (3.14), we have

$$\begin{aligned} \mathbf{x}^T(t) := \mathbf{x}^{T_0}\left(t\frac{T_0}{T}\right) &= \mathbf{x}^0 + \int_0^{t\frac{T_0}{T}} \mathbf{f}(w^{T_0}(s), b^{T_0}(s), \mathbf{x}^{T_0}(s))\,\mathrm{d}s \\ &= \mathbf{x}^0 + \int_0^t \frac{T_0}{T}\mathbf{f}\left(w^{T_0}\left(\tau\frac{T_0}{T}\right), b^{T_0}\left(\tau\frac{T_0}{T}\right), \mathbf{x}^{T_0}\left(\tau\frac{T_0}{T}\right)\right)\mathrm{d}\tau \\ &= \mathbf{x}^0 + \int_0^t \mathbf{f}\left(w^T(\tau), b^T(\tau), \mathbf{x}^T(\tau)\right)\mathrm{d}\tau. \end{aligned}$$

It follows that $\mathbf{x}^T$ solves (3.13), and we conclude by uniqueness. □

The following corollary is an immediate consequence.

Corollary 3.1. *Let $\mathbf{x}^0 \in \mathbb{R}^{d_x}$. If $\mathbf{x}^1 \in \mathbb{R}^{d_x}$ is reachable for (3.3) (resp. (3.2) with σ positively homogeneous of degree 1) in some time $T_0 > 0$ (in the sense of Definition 3.1), then $\mathbf{x}^1$ is reachable for (3.3) (resp. (3.2)) in any time $T > 0$.*

Proof. Let $\mathbf{x}^0 \in \mathbb{R}^{d_x}$ be any initial datum and let $\mathbf{x}^1 \in \mathcal{R}_{T_0}(\mathbf{x}^0)$. Then there exists a control $u^{T_0} := \left[w^{T_0}, b^{T_0}\right]^\top \in H^k(0, T_0; \mathbb{R}^{d_u})$, with $k = 0$ for (3.3) and $k = 1$ for (3.2), such that the corresponding solution $\mathbf{x}^{T_0} \in C^0([0, T_0]; \mathbb{R}^{d_x})$ to (3.13) satisfies $\mathbf{x}^{T_0}(T_0) = \mathbf{x}^1$. Now, let $T > 0$ and consider $u^T := \left[w^T, b^T\right]^\top$ defined in (3.16). The corresponding solution $\mathbf{x}^T$ to (3.13) is thus given by (3.17), and we may now observe that $\mathbf{x}^T(T) = \mathbf{x}^{T_0}\left(T\frac{T_0}{T}\right) = \mathbf{x}^{T_0}(T_0) = \mathbf{x}^1$. This concludes the proof. □

We make the following observation regarding the scaling of the optimal parameters in the reachability regime. Let $\mathbf{x}^0 \in \mathbb{R}^{d_x}$ and $\mathcal{M}$ be a closed subset of $\mathbb{R}^{d_x}$ such that $\mathcal{R}_{T_0}(\mathbf{x}^0) \cap \mathcal{M} \neq \varnothing$ for some $T_0 > 0$. Then, the set

$$\mathcal{U}_{T,\mathcal{M}} := \{u = [w, b]^\top \in L^2(0, T; \mathbb{R}^{d_u}) \colon \mathbf{x}(T) \in \mathcal{M}\} \tag{3.18}$$

where $\mathbf{x}$ is the solution to (3.3) (resp. (3.2) with σ positively homogeneous of degree 1) associated to u, is non-empty whenever $T > 0$. Furthermore,

$$\inf_{u \in \mathcal{U}_{T,\mathcal{M}}} \|u\|^2_{L^2(0,T;\mathbb{R}^{d_u})} = \frac{T_0}{T} \inf_{u \in \mathcal{U}_{T_0,\mathcal{M}}} \|u\|^2_{L^2(0,T_0;\mathbb{R}^{d_u})}$$

and whenever $\inf_{u \in \mathcal{U}_{T_0,\mathcal{M}}} \|u\|^2_{L^2(0,T_0;\mathbb{R}^{d_u})}$ is achieved at u^{T_0}, then $\inf_{u \in \mathcal{U}_{T,\mathcal{M}}} \|u\|^2_{L^2(0,T;\mathbb{R}^{d_u})}$ is achieved at

$$u^T(t) := \frac{T_0}{T} u^{T_0}\left(t \frac{T_0}{T}\right) \qquad \text{for } \ t \in [0, T].$$

3.2.2. *Proof of Theorem 3.1.* We are now in a position to prove the main result of this section.

Proof of Theorem 3.1. We will henceforth, for notational convenience, extensively make use of the notation $u := [w, b]^\top$. We will focus on the neural ODE (3.3) and hence $k = 0$. The case (3.2) and $k = 1$ follows exactly the same arguments, and we will comment on the key differences at the end of the proof.

Part 1. We begin by proving (3.6). To this end, we will first show that $\{\mathbf{x}^T(T)\}_{T>0}$ is a bounded subset of $\mathbb{R}^{d_x}$. This will allow us to extract a converging sequence, whose limit will be shown to lie in $\arg\min(\phi)$.

Consider any $u^0 = [w^0, b^0]^\top \in L^2(0, T_0; \mathbb{R}^{d_u})$ such that the corresponding solution $\mathbf{x}^{T_0} \in C^0([0, T_0]; \mathbb{R}^{d_x})$ to (3.3), with $T = T_0$, satisfies $\mathbf{x}^{T_0}(T_0) \in \arg\min(\phi)$ and

$$\frac{1}{2} \int_0^{T_0} \left\|u^0\right\|^2 \, \mathrm{d}s \leq \kappa_{T_0}(\mathbf{x}^0, \mathbf{x}^1) + 1,$$

with $\kappa_{T_0}(\mathbf{x}^0, \mathbf{x}^1)$ as in Definition 3.2. Such a u^0 can always be found by the reachability assumption $\mathcal{R}_{T_0}(\mathbf{x}^0) \cap \arg\min(\phi) \neq \varnothing$. For $T > 0$, set

$$u^0_T(t) := \frac{T_0}{T} u^0\left(t \frac{T_0}{T}\right) \qquad \text{for } \ t \in [0, T].$$

Making use of Lemma 3.1, and since $\mathbf{x}^{T_0}(T_0) \in \arg\min(\phi)$, we see that

$$\mathcal{J}_T\left(u^0_T\right) = \phi\left(\mathbf{x}^{T_0}(T_0)\right) + \frac{1}{2}\frac{T_0}{T}\left\|u^0\right\|^2_{L^2(0,T_0;\mathbb{R}^{d_u})} = \min_{\mathbb{R}^{d_x}} \phi + \frac{1}{2}\frac{T_0}{T}\left\|u^0\right\|^2_{L^2(0,T_0;\mathbb{R}^{d_u})}. \tag{3.19}$$

Now using the fact that u^T minimizes $\mathcal{J}_T$, we obtain

$$\mathcal{J}_T\left(u^0_T\right) - \min_{\mathbb{R}^{d_x}} \phi \geq \mathcal{J}_T\left(u^T\right) - \min_{\mathbb{R}^{d_x}} \phi \geq \frac{1}{2}\left\|u^T\right\|^2_{L^2(0,T;\mathbb{R}^{d_u})}. \tag{3.20}$$

Combining (3.20) and (3.19), along with the properties of u^0, we deduce

$$\frac{1}{2}\left\|u^T\right\|^2_{L^2(0,T;\mathbb{R}^{d_u})} \leq \frac{1}{2}\frac{T_0}{T}\left\|u^0\right\|^2_{L^2(0,T_0;\mathbb{R}^{d_u})} \leq \frac{T_0}{T}\Big(\kappa_{T_0}(\mathbf{x}^0, \mathbf{x}^1) + 1\Big). \tag{3.21}$$

We again make use of the formulation (3.13): for any $t \in [0,T]$,

$$\mathbf{x}^T(t) = \mathbf{x}^0 + \int_0^t \mathbf{f}\left(w^T(s), b^T(s), \mathbf{x}^T(s)\right)\, \mathrm{d}s,$$

and so, using the specific form of $\mathbf{f}$ given in (3.3), the fact that σ is globally Lipschitz continuous with constant $L_\sigma > 0$ and satisfies $\sigma(0) = 0$, we find that

$$\begin{aligned}\left\|\mathbf{x}^T(t) - \mathbf{x}^0\right\| &\leq N \int_0^t \left(L_\sigma \left\|w^T(s)\right\| \left\|\mathbf{x}^T(s)\right\| + \left\|b^T(s)\right\|\right) \mathrm{d}s \\ &\leq N L_\sigma \int_0^t \left\|w^T(s)\right\| \left\|\mathbf{x}^T(s)\right\| \,\mathrm{d}s + N \left\|b^T\right\|_{L^1(0,T;\mathbb{R}^d)}.\end{aligned}$$

Hence, by using the Grönwall inequality, we obtain

$$\left\|\mathbf{x}^T(T) - \mathbf{x}^0\right\| \leq N \left\|b^T\right\|_{L^1(0,T;\mathbb{R}^d)} \exp\left(N L_\sigma \int_0^T \left\|w^T(s)\right\| \,\mathrm{d}s\right),$$

while by Cauchy-Schwarz it follows that

$$\left\|\mathbf{x}^T(T) - \mathbf{x}^0\right\| \leq \sqrt{T} N \left\|b^T\right\|_{L^2(0,T;\mathbb{R}^d)} \exp\left(\sqrt{T} N L_\sigma \left\|w^T\right\|_{L^2(0,T;\mathbb{R}^{d\times d})}\right).$$

At this point, employing (3.21), we deduce

$$\left\|\mathbf{x}^T(T) - \mathbf{x}^0\right\| \leq \sqrt{T_0} N \sqrt{\kappa_{T_0}(\mathbf{x}^0, \mathbf{x}^1) + 1} \exp\left(\sqrt{T_0} N L_\sigma \sqrt{\kappa_{T_0}(\mathbf{x}^0, \mathbf{x}^1) + 1}\right).$$

Hence, the set $\{\mathbf{x}^T(T)\}_{T>0}$ is bounded.

Let us now, as a second step, prove that

$$\phi\left(\mathbf{x}^T(T)\right) \longrightarrow \min_{\mathbb{R}^{d_x}} \phi \qquad \text{as} \quad T \to +\infty. \tag{3.22}$$

Let u^0 and $\mathbf{x}^{T_0}$ be the same as before. Since u^T is a minimizer of $\mathcal{J}_T$ and using again the scaling relations from Lemma 3.1, for all $T > 0$, we have

$$\mathcal{J}_T\left(u^T\right) = \phi\left(\mathbf{x}^T(T)\right) + \frac{1}{2}\left\|u^T\right\|^2_{L^2(0,T;\mathbb{R}^{d_u})} \leq \phi(\mathbf{x}^{T_0}(T_0)) + \frac{1}{2}\frac{T_0}{T}\left\|u^0\right\|^2_{L^2(0,T_0;\mathbb{R}^{d_u})},$$

which in turn, since $\mathbf{x}^{T_0}(T_0) \in \arg\min(\phi)$, implies

$$\min_{\mathbb{R}^{d_x}} \phi \leq \phi\left(\mathbf{x}^T(T)\right) \leq \min_{\mathbb{R}^{d_x}} \phi + \frac{1}{2}\frac{T_0}{T}\left\|u^0\right\|^2_{L^2(0,T_0;\mathbb{R}^{d_u})}, \qquad \text{for all } T > 0. \tag{3.23}$$

Estimate (3.23) clearly implies (3.22).

Now, since the set $\{\mathbf{x}^T(T)\}_{T>0}$ is bounded, there exists a sequence $\{T_n\}_{n=1}^{+\infty}$ with $T_n > 0$ and $T_n \to +\infty$ as $n \to +\infty$ and some $\mathbf{x}^\dagger \in \mathbb{R}^{d_x}$, such that

$$\mathbf{x}^{T_n}(T_n) \longrightarrow \mathbf{x}^\dagger \qquad \text{as} \quad n \to +\infty. \tag{3.24}$$

Since $\phi\left(\mathbf{x}^{T_n}(T_n)\right) \to \min_{\mathbb{R}^{d_x}} \phi$ as $n \to +\infty$ by (3.22), by the lower semicontinuity of ϕ, we have

$$\phi\left(\mathbf{x}^\dagger\right) \leq \min_{\mathbb{R}^{d_x}} \phi,$$

whence $\mathbf{x}^\dagger \in \operatorname{argmin}(\phi)$. This concludes the proof of (3.6).

Part 2. We now address the second statement of the theorem. To this end, we will first show that the sequence $\{u_n\}_{n=1}^{+\infty}$ defined in the statement is bounded in $L^2(0,T_0;\mathbb{R}^{d_u})$.

Let $\mathbf{x}^\dagger$ be the same as above, and now let $u^0 := [w^0, b^0]^\top \in L^2(0,T_0;\mathbb{R}^{d_u})$ be any solution to

$$\inf_{\substack{u=[w,b]^\top \in L^2(0,T_0;\mathbb{R}^{d_u}) \\ \text{subject to (3.3)} \\ \text{and} \\ \mathbf{x}(T_0)\in \arg\min(\phi)}} \frac{1}{2}\int_0^{T_0} \|u(t)\|^2 \,\mathrm{d}t. \tag{3.25}$$

Denote by $\mathbf{x}^{T_0}$ the corresponding state, namely the solution to (3.3) with $T = T_0$. We claim that

$$\|u_n\|_{L^2(0,T_0;\mathbb{R}^{d_u})} \leq \left\|u^0\right\|_{L^2(0,T_0;\mathbb{R}^{d_u})}, \qquad \text{for all } n \geq 1. \tag{3.26}$$

Indeed, assume that we had $\|u^0\|_{L^2(0,T_0;\mathbb{R}^{d_u})} < \|u_n\|_{L^2(0,T_0;\mathbb{R}^{d_u})}$ for some $n \geq 1$. We consider

$$u^0_{T_n}(t) := \frac{T_0}{T_n} u^0\left(t\frac{T_0}{T_n}\right) \qquad \text{for } \ t \in [0,T_n],$$

whose corresponding state trajectory $\mathbf{x}^0_{T_n}$, solution to (3.3) with $T = T_n$, satisfies $\mathbf{x}^0_{T_n}(T_n) = \mathbf{x}^{T_0}(T_0) \in \arg\min(\phi)$ by Lemma 3.1. It follows that

$$\begin{aligned}\mathcal{J}_{T_n}\left(u^0_{T_n}\right) &= \min_{\mathbb{R}^{d_x}} \phi + \frac{1}{2}\frac{T_0}{T_n}\|u^0\|^2_{L^2(0,T_n;\mathbb{R}^{d_u})} \\ &< \phi\left(\mathbf{x}^{T_n}(T_n)\right) + \frac{1}{2}\frac{T_0}{T_n}\|u_n\|^2_{L^2(0,T_0;\mathbb{R}^{d_u})} = \mathcal{J}_{T_n}\left(u^{T_n}\right),\end{aligned}$$

which contradicts the fact that $u^{T_n} := \left[w^{T_n}, b^{T_n}\right]^\top$ minimizes $\mathcal{J}_{T_n}$. Hence, (3.26) holds, and $\{u_n\}_{n=1}^{+\infty}$ is bounded in $L^2(0,T_0;\mathbb{R}^{d_u})$. Consequently, by the Banach-Alaoglu theorem, there exists $u^\dagger = \left[w^\dagger, b^\dagger\right]^\top \in L^2(0,T_0;\mathbb{R}^{d_u})$ such that

$$u_n \rightharpoonup u^\dagger \qquad \text{weakly in } L^2(0,T_0;\mathbb{R}^{d_u}),$$

along some subsequence as $n \to +\infty$. Moreover, using the properties of equation (3.3) (see the arguments in the proof of Proposition 2.1), we deduce that the trajectory $\mathbf{x}_n$ associated to u_n satisfies

$$\mathbf{x}_n \longrightarrow \mathbf{x}^{T_0,\dagger} \qquad \text{strongly in } \ C^0([0,T_0];\mathbb{R}^{d_x}) \tag{3.27}$$

as $n \to +\infty$, where $\mathbf{x}^{T_0,\dagger}$ is the solution to (3.3) with $T = T_0$, associated to $u^\dagger$. On another hand, note that by Lemma 3.1, $\mathbf{x}^{T_n}(t) = \mathbf{x}_n(\frac{t}{T_n})$ for $t \in [0,T_n]$, whence $\mathbf{x}^{T_n}(T_n) = \mathbf{x}_n(1)$ and thus, combining (3.27) and (3.24), we see that $\mathbf{x}^{T_0,\dagger}(T_0) = \mathbf{x}^\dagger$. Consequently, $u^\dagger$ is a control such that $\mathbf{x}^{T_0,\dagger}(T_0) = \mathbf{x}^\dagger \in \arg\min(\phi)$, thus satisfying the constraint in (3.25). In view of this, we may also use (3.26) and the weak lower semicontinuity of the L^2–norm to write

$$\begin{aligned}\left\|u^0\right\|_{L^2(0,T_0;\mathbb{R}^{d_u})} \leq \left\|u^\dagger\right\|_{L^2(0,T_0;\mathbb{R}^{d_u})} &\leq \liminf_{n\to+\infty} \|u_n\|_{L^2(0,T_0;\mathbb{R}^{d_u})} \\ &\leq \lim_{n\to+\infty} \|u_n\|_{L^2(0,T_0;\mathbb{R}^{d_u})} \\ &\leq \limsup_{n\to+\infty} \|u_n\|_{L^2(0,T_0;\mathbb{R}^{d_u})} \\ &\leq \|u^0\|_{L^2(0,T_0;\mathbb{R}^{d_u})},\end{aligned} \tag{3.28}$$

clearly implying that

$$\lim_{n\to+\infty} \|u_n\|_{L^2(0,T_0;\mathbb{R}^{d_u})} = \left\|u^\dagger\right\|_{L^2(0,T_0;\mathbb{R}^{d_u})}.$$

Hence, as weak convergence and convergence of the norms in L^2 implies strong convergence in L^2, we deduce that

$$u_n \longrightarrow u^\dagger \qquad \text{strongly in} \quad L^2(0,T_0;\mathbb{R}^{d_u})$$

as $n \to +\infty$. Moreover, from (3.28) we deduce that, since u^0 is a solution to (3.25) and since $u^\dagger$ satisfies the constraints therein, $u^\dagger$ is a solution to (3.25) as well, which concludes the proof for (3.3) and $k = 0$..

In the case (3.2) and $k = 1$, one may clearly repeat the above reasoning, replacing $L^2(0,T;\mathbb{R}^{d_u})$ by $H^1(0,T;\mathbb{R}^{d_u})$ throughout, with some key additions.

In Part 1, we first note that instead of (3.19), one has

$$\begin{aligned}\mathcal{J}_T(u_T^0) &= \phi(\mathbf{x}^{T_0}(T_0)) + \frac{1}{2}\frac{T_0}{T}\left\|u^0\right\|^2_{L^2(0,T_0;\mathbb{R}^{d_u})} + \frac{1}{2}\frac{T_0^3}{T^3}\left\|\dot{u}^0\right\|^2_{L^2(0,T_0;\mathbb{R}^{d_u})}\\ &= \min_{\mathbb{R}^{d_x}}\phi + \frac{1}{2}\frac{T_0}{T}\left\|u^0\right\|^2_{L^2(0,T_0;\mathbb{R}^{d_u})} + \frac{1}{2}\frac{T_0^3}{T^3}\left\|\dot{u}^0\right\|^2_{L^2(0,T_0;\mathbb{R}^{d_u})}.\end{aligned}$$

This is not an impediment to (3.20), which remains true, and one can clearly deduce that $\{\mathbf{x}^T(T)\}_{T>0}$ is bounded as well. Similarly, (3.23) holds with a bound of the form

$$\min_{\mathbb{R}^{d_x}}\phi \leq \phi(\mathbf{x}^T(T)) \leq \frac{1}{2}\frac{T_0}{T}\left\|u^0\right\|^2_{L^2(0,T;\mathbb{R}^{d_u})} + \frac{1}{2}\frac{T_0^3}{T^3}\left\|\dot{u}^0\right\|^2_{L^2(0,T;\mathbb{R}^{d_u})}.$$

Whence the remainder of Part 1 holds in this context as well.

In Part 2, we emphasise the sole key difference between (3.3) and (3.2) – the weak L^2–convergence of $\{u_n\}_{n=1}^{+\infty}$ is a priori not sufficient to entail the strong convergence in (3.27) in the case of (3.2). However, by the Rellich-Kondrachov compactness theorem, the weak H^1–convergence of $\{u_n\}_{n=1}^{+\infty}$ implies a strong L^2–convergence along a subsequence, which would yield (3.27) by arguing just as in the proof of Proposition 2.1.

This concludes the proof. □

4. Asymptotics with tracking

Before proceeding, we simply recall once again that we are addressing dynamics given by stacked neural ODEs of the form

$$\begin{cases}\dot{\mathbf{x}}(t) = \mathbf{w}(t)\sigma(\mathbf{x}(t)) + \mathbf{b}(t) & \text{in } (0,T)\\ \mathbf{x}(0) = \mathbf{x}^0 \in \mathbb{R}^{d_x},\end{cases} \tag{4.1}$$

or

$$\begin{cases}\dot{\mathbf{x}}(t) = \sigma(\mathbf{w}(t)\mathbf{x}(t) + \mathbf{b}(t)) & \text{in } (0,T)\\ \mathbf{x}(0) = \mathbf{x}^0 \in \mathbb{R}^{d_x},\end{cases} \tag{4.2}$$

with $\sigma \in \mathrm{Lip}(\mathbb{R})$ with $\sigma(0) = 0$ as well as $\sigma(\lambda\cdot) = \lambda\,\sigma(\cdot)$ for $\lambda > 0$ (positive homogeneity of degree 1) in the latter system, is defined componentwise for multi-dimensional entries. Most importantly, we recall that the optimizable parameters/controls $[w,b]^\top$ enter the system in the following way:

$$\mathbf{w} := \begin{bmatrix} w & & \\ & \ddots & \\ & & w\end{bmatrix} \in \mathbb{R}^{d_x\times d_x}, \qquad \mathbf{b} := \begin{bmatrix} b\\ \vdots \\ b\end{bmatrix} \in \mathbb{R}^{d_x}.$$

We will henceforth distinguish two cases depending on the regularization of the control parameters, namely that of Tikhonov–Sobolev regularization (as done in Section 3), but also L^1–regularization.

4.1. **Tikhonov–Sobolev regularization.** Given the projector $\varphi \in C^\infty(\mathbb{R}^d;\mathbb{R}^m)$ as in (2.8), with the training error ϕ as in (3.4), and having fixed a couple of regularization hyper-parameters $\alpha, \beta > 0$, in this section consider the non-negative functional

$$\mathcal{J}_T(w,b) := \phi(\mathbf{x}(T)) + \frac{\alpha}{2}\left\|[w,b]^\top\right\|^2_{H^k(0,T;\mathbb{R}^{d_u})} + \frac{\beta}{2}\int_0^T \|\mathbf{x}(t)-\mathbf{x}_d\|^2\,\mathrm{d}t. \tag{4.3}$$

Here ϕ denotes the training error defined in (3.4), $k = 0$ for (4.1) and $k = 1$ for (4.2), $\mathbf{x}_d \in \mathbb{R}^{d_x}$ is a fixed running target, and $\mathbf{x} \in C^0([0,T];\mathbb{R}^{d_x})$ is the unique solution to either (3.3) or (3.2) corresponding to the control parameters $[w,b]^\top \in H^k(0,T;\mathbb{R}^{d_u})$, noting (3.1). We emphasize that, in modern machine learning, one generally minimizes (4.3) with $\beta = 0$, but, as discussed in what precedes, the case $\beta > 0$ carries several interesting features and consequences.

An important observation regarding both (3.3), and (3.2) when σ is positively homogeneous, is that any constant vector $\mathbf{x} \in \mathbb{R}^{d_x}$ is a steady state with control parameters $[w,b]^\top \equiv 0_{\mathbb{R}^{d_u}}$ in (3.1).

We may now state the first main result of this section, namely the *turnpike property* for the learning problem with a tracking term.

Theorem 4.1. *Let $\mathbf{x}^0 = [\vec{x}_1,\dots,\vec{x}_N]^\top \in \mathbb{R}^{d_x}$, and let $\mathbf{x}_d \in \mathcal{R}_{T_0}(\mathbf{x}^0)$ for some $T_0 > 0$ be given. Assume that there exist $L_N > 0$ and $r > 0$ such that*

$$\kappa_{T_0}(\mathbf{x},\mathbf{x}_d) \le L_N^2\,\|\mathbf{x}-\mathbf{x}_d\|^2 \quad \text{and} \quad \kappa_{T_0}(\mathbf{x}_d,\mathbf{x}) \le L_N^2\,\|\mathbf{x}-\mathbf{x}_d\|^2$$

for all $\mathbf{x} \in \{\mathbf{x} \in \mathbb{R}^{d_x} : \|\mathbf{x}-\mathbf{x}_d\| \le r\}$. Let $T \ge 2T_0$ be fixed and let $\mathbf{x}^T \in C^0([0,T];\mathbb{R}^{d_x})$ be the unique solution to (3.2) (resp. (3.3)) with parameters $\left[w^T,b^T\right]^\top \in H^k(0,T;\mathbb{R}^{d_u})$ minimizing (4.3), where $k = 0$ in the case of (3.3), and $k = 1$ in the case of (3.2).

Then there exist $C = C(\alpha,\beta,\mathbf{x}_d,\mathbf{x}^0,N) > 0$, $\gamma = \gamma(\alpha,\beta,\mathbf{x}_d,\mathbf{x}^0,N) > 0$ and $\mu = \mu(\alpha,\beta,N) > 0$ such that

$$\left\|\left[w^T,b^T\right]^\top\right\|_{H^k(0,T;\mathbb{R}^{d_u})} \le C\left(\|\mathbf{x}_d-\mathbf{x}^0\| + \sqrt{\phi(\mathbf{x}_d)}\right) \tag{4.4}$$

and

$$\left\|\mathbf{x}^T(t)-\mathbf{x}_d\right\| \le \gamma\left(e^{-\mu t} + e^{-\mu(T-t)}\right) \tag{4.5}$$

hold for all $t \in [0,T]$.

The proof is done in Section 4.4.1, and relies on constructing auxiliary controls by which one aims to repeatedly estimate $\mathcal{J}_T(u^T)$, combined with a new iterative strategy for obtaining the exponential estimate (see Proposition 4.1).

We note that Theorem 4.1 is a *global result*, namely, we make no restrictive smallness assumptions on the vector of training data $\mathbf{x}^0$ – which serves as an initial datum for (3.2) or (3.3) – or on the running target $\mathbf{x}_d$, as our proof does not rely on linearization arguments. In [Trélat and Zuazua, 2015] the authors present a similar, but local result, which would impose a severe restriction on the training dataset.

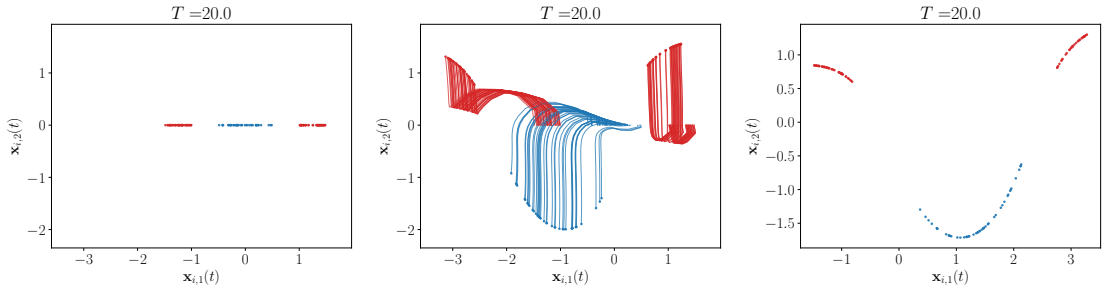

FIGURE 3. We visualize the optimal trajectories $\mathbf{x}_i^T$ of solutions to the learning problem (4.3); the algorithm learns a simple flow, separates the points, and ensures the turnpike property (4.5) – (4.8), as seen in Figure 4. Here $T = 20$ and $N_{\text{layers}} = 50$, the running target is $\mathbf{x}_{d,i} = [2,2]^\top \mathbf{1}_{\vec{y}_i=1} + [-2,-2]^\top \mathbf{1}_{\vec{y}_i=-1}$ for $1 \leq i \leq N$, with $\alpha = 2$ and $\beta = 100$.

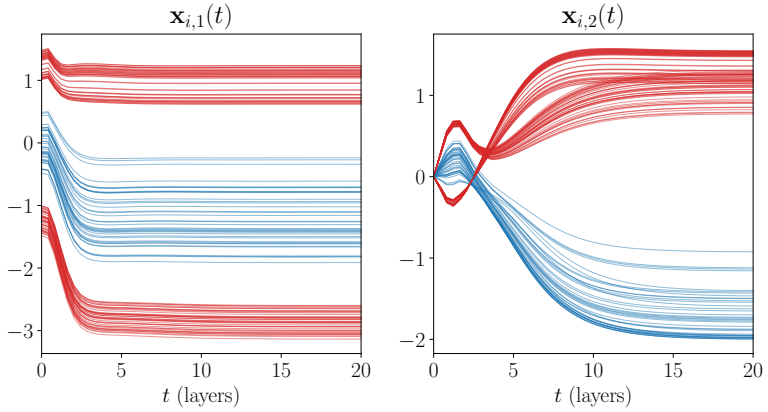

FIGURE 4. We observe the appearance of the turnpike property (4.10) for the learning problem (4.9) at the level of both components $\left[\mathbf{x}_{i,1}^T, \mathbf{x}_{i,2}^T\right]^\top$ of every copy $\mathbf{x}_i^T$ for $1 \leq i \leq 128$.

Let us note that Theorem 4.1 is different in nature to Theorem 3.1. This is because the integral tracking term introduces a newer and stronger time-scale in the behavior of the optimization problem as $T \to +\infty$. To see this, consider the neural ODE (3.3) (whence $k = 0$) for simplicity, and as in (3.8),

$$\begin{aligned}
&\inf_{\substack{u^T \in L^2(0,T;\mathbb{R}^{d_u}) \\ \text{subject to (3.8)}}} \phi\left(\mathbf{x}^T(T)\right) + \frac{1}{2}\int_0^T \left\|u^T(t)\right\|^2 \, \mathrm{d}t + \frac{1}{2} \int_0^T \|\mathbf{x}^T(t) - \mathbf{x}_d\|^2 \, \mathrm{d}t \\
&= \inf_{\substack{u^T \in L^2(0,T;\mathbb{R}^{d_u}) \\ \text{subject to (3.8)}}} \phi\left(\mathbf{x}^T(T)\right) + \frac{1}{2T}\int_0^1 \left\|Tu^T(sT)\right\|^2 \, \mathrm{d}s + \frac{T}{2} \int_0^1 \left\|\mathbf{x}^T\left(\frac{s}{T}\right) - \mathbf{x}_d\right\|^2 \mathrm{d}s \\
&= \inf_{\substack{u^1 \in L^2(0,1;\mathbb{R}^{d_u}) \\ \text{subject to (3.9)}}} \phi\left(\mathbf{x}^1(1)\right) + \frac{1}{2T} \int_0^1 \left\|u^1(s)\right\|^2 \mathrm{d}s + \frac{T}{2} \int_0^1 \|\mathbf{x}^1(s) - \mathbf{x}_d\|^2 \, \mathrm{d}s. \quad (4.6)
\end{aligned}$$

We see that, unlike Theorem 3.1, the rightmost term in (4.6) carries the most significance when $T \gg 1$, somewhat motivating the appearance of the turnpike property.

If we consider the learning problem without the training error (final cost) at time T in the functional (4.3), namely

$$\mathcal{J}_T(w,b) := \frac{\alpha}{2}\left\|[w,b]^\top\right\|^2_{H^k(0,T;\mathbb{R}^{d_u})} + \frac{\beta}{2}\int_0^T \|\mathbf{x}(t)-\mathbf{x}_d\|^2\,\mathrm{d}t, \tag{4.7}$$

then it is possible to improve the estimate (4.5). In this case in fact, it can actually be shown that the optimal trajectory at time $t = T$ is *exponentially* close to the running target $\mathbf{x}_d$.

Corollary 4.1. *Under the assumptions of Theorem 4.1 and considering the functional (4.7) instead of (4.3), there exist constants $\gamma = \gamma(\alpha,\beta,\mathbf{x}_d,\mathbf{x}^0,N) > 0$ and $\mu = \mu(\alpha,\beta,N) > 0$ such that for any control $\left[w^T,b^T\right]^\top \in H^k(0,T;\mathbb{R}^{d_u})$ minimizing (4.7), the corresponding state $\mathbf{x}^T \in C^0([0,T];\mathbb{R}^{d_x})$ satisfies*

$$\left\|\mathbf{x}^T(T)-\mathbf{x}_d\right\| \leq \gamma\, e^{-\mu T}. \tag{4.8}$$

Discussion. Our main reason for considering the tracking term $\int_0^T \|\mathbf{x}(t)-\mathbf{x}_d\|^2\,\mathrm{d}t$, Theorem 4.1 and consequently Corollary 4.1, is motivated by the sensible choice of a cost of the form

$$\frac{\alpha}{2}\left\|[w,b]^\top\right\|^2_{H^k(0,T;\mathbb{R}^{d_u})} + \frac{\beta}{2}\int_0^T \phi(\mathbf{x}(t))\,\mathrm{d}t, \tag{4.9}$$

where, rather than (4.1), we would expect an estimate of the form

$$\operatorname{dist}\left(\mathbf{x}^T(T), \arg\min(\phi)\right) \leq Ce^{-\mu T} \tag{4.10}$$

for all $t \in [0,T]$. Note that (4.10) stipulates that in the neural network setting, the optimal learned trajectories approach the zero training error regime exponentially in terms of the number of layers. In fact, Corollary 4.1 is a manifestation of (4.10) in the case where the input dimension d matches the output dimension m, and $\mathbf{x}_d = [\vec{y}_1,\dots,\vec{y}_N]^\top$, and hence, the same conclusion holds in this scenario. Whilst precisely this turnpike result is left without proof, we do observe this turnpike phenomenon in our numerical experiments and Theorem 4.1 – Corollary 4.1 serves as a strong indicator that such a property should hold.

Another, more heuristic interpretation of Theorem 4.1 in particular is that it provides a robust indication that, rather than considering standard neural network architectures such as ResNets (1.1), one ought to directly use an adaptive ODE solver (e.g. Dormand-Prince, or other adaptive Runge-Kutta scheme) for the underlying continuous neural network (e.g. (1.2)), as done in [Chen et al., 2018, Benning et al., 2019], and in particular, consider large time horizons (number of layers) in view of *stretching* the time grid and capturing the relevant time scales. Heuristically, the turnpike property indicates an intrinsic notion of distance between the different layers of a neural network, namely, it indicates a way to choose where to localize the different time-steps (i.e. layers), whence, the layers near $t = 0$ and $t = T$ carry, in some sense, more relevance than those in the middle. This is a

priori not clear if one trains a discrete neural network such as (1.1) via empirical risk minimization, as turnpike is a staple of the regularized cost functional.

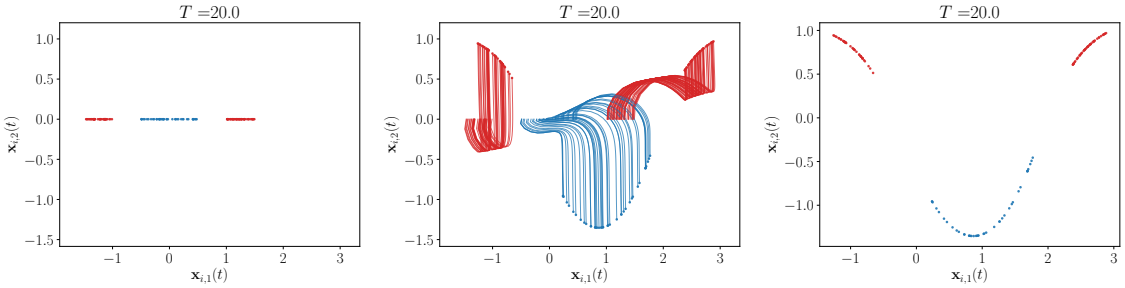

FIGURE 5. Similar to the result we prove in this work, we also visualize the optimal trajectories of solutions to the learning problem (4.9) to similar outcome; the algorithm learns a simple flow, separates the points, and ensures the turnpike property (4.10), as seen in Figure 6. Here $T = 20$, $N_{\text{layers}} = 50$, $\alpha = 2$ and $\beta = 1$.

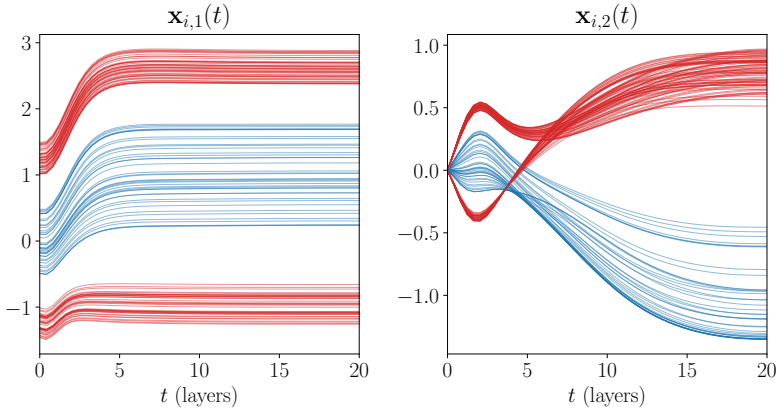

FIGURE 6. We observe the appearance of the turnpike property (4.10) for the learning problem (4.9) at the level of both components $\left[\mathbf{x}_{i,1}^T, \mathbf{x}_{i,2}^T\right]^\top$ of every copy $\mathbf{x}_i^T$ for $1 \leq i \leq 128$.

Biblographical overview. The idea that optimal strategies, when considered over long time periods, are constant for most of the time, traces back to work of John von Neumann in 1930s and 40s [von Neumann, 1945]. The terminology *turnpike* was introduced in the context of economics by Nobel Prize winner Paul Samuelson and collaborators in [Dorfman et al., 1958] to interpret the full evolutionary phenomenon. Several turnpike theorems have subsequently been derived in the 1960s for discrete-time optimal control problems arising in econometrics (see, e.g., [McKenzie, 1963]). Preliminary continuous versions have been proved in [Haurie, 1976] motivated by economic growth models. These works generally stipulate that the solution of an optimal control problem in large time should spend most of its time near a steady-state, while in infinite horizon, the solution should converge to that steady-state. Such steady states are referred to as *von Neumann points*. We refer the reader to [Carlson et al., 2012] for an comprehensive overview of these continuous turnpike

results (see also [Zaslavski, 2006]). Historically, as per [McKenzie, 1976] (see also Chapter 1 therein for a seminal explanation), it appears that the first turnpike result was discovered in [Dorfman et al., 1958, Chapter 12].

As indicated in [Rockafellar, 1973, Samuelson, 1972], turnpike properties can be interpreted as consequences of the Hamiltonian nature of the backward-forward equations derived from the Pontryagin Maximum Principle. These results relate the turnpike property with the asymptotic stability of the solutions of the Hamiltonian system provided suitable convexity properties of the Hamiltonian. More recently, via the dynamic programming principle, the turnpike property has been linked to the long time asymptotic behavior of the Hamilton-Jacobi-Bellman equation satisfied by the value function [Esteve et al., 2020].

In [Wilde and Kokotovic, 1972], turnpike is shown for linear quadratic problems under the Kalman rank condition, subsequently extended to nonlinear control-affine systems in [Anderson and Kokotovic, 1987] for globally Lipschitz vector fields. In both cases, the initial and final conditions for the trajectory are prescribed. Recent turnpike works include [Rapaport and Cartigny, 2004, Rapaport and Cartigny, 2005, Grüne et al., 2019, Grüne and Müller, 2016, Grüne and Guglielmi, 2018], to name a few. A rather complete turnpike theory, combining Pontryagin Maximum Principle, linearization arguments and precise estimates on Riccati equations, and covering a wide variety of nonlinear optimal control problems is developed and presented in [Trélat and Zuazua, 2015]. The study in [Trélat and Zuazua, 2015] is somewhat closely motivated to the interpretation that turnpike is due to a general hyperbolicity phenomenon, any trajectory of a given hyperbolic dynamical system in a neighborhood of a saddle point, which is constrained to remain in this neighborhood in large time will spend most of the time near the saddle point.

In recent years the turnpike property has been extensively studied in the infinite-dimensional context, namely, where the underlying dynamics is governed by a PDE. Motivated by the works [Cardaliaguet et al., 2012, Cardaliaguet et al., 2013] on the long time behavior of mean field games, in [Porretta and Zuazua, 2013] the authors address linear quadratic control problems for linear PDEs, clearly distinguishing assumptions based on the dissipativity (or lack thereof) of the underlying dynamics, and they prove an exponential turnpike property. These results have subsequently been extended in a series of works in both linear and nonlinear contexts, see e.g. [Porretta and Zuazua, 2016, Trélat et al., 2018, Cardaliaguet and Porretta, 2019], and numerical algorithms are discussed in [Grüne et al., 2020]. In the nonlinear case, typically some smallness conditions are imposed, the proof strategies being based on linearization and fixed point. Recently, in [Pighin, 2020, Pighin and Sakamoto, 2020] several global results without such smallness conditions are provided.

4.2. **An algorithm inspired by Theorem 4.1.** Much like we did following Theorem 3.1 in Section 3.1, we also propose an alternative greedy learning strategy, inspired by Theorem 4.1, which applies to the continuous-time as well as the discrete-time setting. We henceforth adopt the compact formulation (3.13) for the stacked neural ODEs (4.1) – (4.2).

Let us henceforth fix an auxiliary horizon $T_1 > 0$ (or number of layers N_1, which may be small), a counter $i = 1$ and a tolerance $\varepsilon > 0$.

Step 1. First minimize (4.9) with $T = T_1$ and initial datum $\mathbf{x}^0$, namely

- Minimize

$$\mathcal{J}_{T_1}(w,b) := \frac{\alpha}{2}\left\|[w,b]^\top\right\|^2_{H^k(0,T_1;\mathbb{R}^{d_u})} + \frac{\beta}{2}\int_0^{T_1}\phi(\mathbf{x}(t))\,\mathrm{d}t$$

subject to the neural ODE dynamics

$$\begin{cases}\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{w}(t),\mathbf{b}(t),\mathbf{x}(t)) & \text{for } t\in(0,T_1)\\ \mathbf{x}(0) = \mathbf{x}^0.\end{cases}$$

This gives an optimal control $u^1 := [w^1,b^1]^\top$, and corresponding optimal state $\mathbf{x}^1$.

- If

$$\left|\phi\left(\mathbf{x}^1(T_1)\right) - \min_{\mathbb{R}^{d_x}}\phi\right| < \varepsilon \tag{4.11}$$

holds, we have obtained the desired neural network ((4.11) is a tolerance threshold).

- Else, we set $i \coloneqq 2$ and proceed with Step i below.

Step i. Minimize (4.9) with initial datum $\mathbf{x}^{i-1}((i-1)T_1)$. Namely,

- Set $i_{\text{old}} \coloneqq i$.
- Minimize

$$\mathcal{J}^i_{T_1}(w,b) := \frac{\alpha}{2}\left\|[w,b]^\top\right\|^2_{H^k((i-1)T_1,iT_1;\mathbb{R}^{d_u})} + \frac{\beta}{2}\int_{(i-1)T_1}^{iT_1}\phi(\mathbf{x}(t))\,\mathrm{d}t,$$

subject to the neural ODE dynamics

$$\begin{cases}\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{w}(t),\mathbf{b}(t),\mathbf{x}(t)) & \text{for } t\in((i-1)T_1,iT_1)\\ \mathbf{x}\left((i-1)T_1\right) = \mathbf{x}^{i-1}\left((i-1)T_1\right).\end{cases}$$

This gives an optimal control $u^i := [w^i,b^i]^\top$, and corresponding optimal state $\mathbf{x}^i$.

- If

$$\left|\phi\left(\mathbf{x}^i(T_1)\right) - \min_{\mathbb{R}^{d_x}}\phi\right| < \varepsilon$$

holds, we have then obtained the desired neural network with the control

$$\widehat{u}(t) \coloneqq u^j(t), \qquad \text{for } t\in((j-1)T_1,jT_1)$$

and for $j\in\{1,\dots,i\}$.

- Else, we set $i \coloneqq i_{\text{old}}+1$ and proceed with Step i.

We note that as a consequence of Theorem 4.1 and Corollary 4.1, one of the most distinguished characteristics of (4.3) is that the time-horizon T needed to get ε–close to any given target is in fact implicitly defined in the cost functional. At the level of classical neural networks, this means that the required number of layers needed to fit the data up to ε-error is given by the cost itself. The goal of the above algorithm is to take advantage of this artefact, and represents a greedy algorithm which uses only the number of layers strictly needed, thus avoiding unnecessary ones.

Remark 3 (Shooting method)**.** As remarked in [Trélat and Zuazua, 2015], Theorem 4.1 also indicates the correct initialization of the shooting method when using indirect methods (i.e. first optimize then discretize) for solving the optimization problem.

4.3. L^1**–regularization.** Let us now consider the learning problem with a state tracking term and L^1–parameter regularization (commonly referred to as *Lasso* in statistical contexts), namely the problem consisting of minimizing the nonnegative functional

$$\mathcal{J}_T(w,b) = \frac{\alpha}{2}\left\|[w,b]^\top\right\|_{L^1(0,T;\mathbb{R}^{d_u})} + \int_0^T \phi(\mathbf{x}(t))\,\mathrm{d}t. \tag{4.12}$$

In this case, we need to impose the following inequality constraint on the control parameters: for some $M>0$,

$$\left\|[w(t),b(t)]^\top\right\| \leq M, \qquad \text{for a.e. } \ t\in(0,T),$$

in order to be able to guarantee the existence of a minimizer. In this case, we can prove the optimal control is of bang-bang form, and in addition, if T is sufficiently large, the optimal parameters can actually overfit the training dataset. This is the goal of the second main result of this section.

Theorem 4.2. *Let* $\mathbf{x}^0 = [\vec{x}_1,\dots,\vec{x}_N]^\top \in \mathbb{R}^{d_x}$, $\alpha>0$, *and assume that* $\phi \in C^0(\mathbb{R}^{d_x};\mathbb{R}_+)$ *is locally Lipschitz continuous and satisfies*

$$\mathcal{R}_{T_0}(\mathbf{x}^0)\cap\{\mathbf{x}\in\mathbb{R}^{d_x} \,:\, \phi(\mathbf{x})=0\}\neq\varnothing \tag{4.13}$$

for some some $T_0>0$. *For a fixed* $M>0$, *consider the optimization problem*

$$\inf_{\substack{u:=[w,b]^\top\in L^1(0,T;\mathbb{R}^{d_u}),\\ \operatorname{ess\,sup}\|u\|\leq M\\ \textit{subject to } (3.3)}} \mathcal{J}_T(w,b) \tag{4.14}$$

with $\mathcal{J}_T$ *defined in* (4.12).

Then, there exists a time $T_M>0$ *such that whenever* $T>T_M$, *any optimal control parameters* $\left[w^T,b^T\right]^\top \in L^1(0,T;\mathbb{R}^{d_u})$ *and corresponding state* $\mathbf{x}^T\in C^0([0,T];\mathbb{R}^{d_x})$, *unique solution to* (3.3), *satisfy*

$$\phi(\mathbf{x}^T(t))=0, \qquad \textit{for all } \ t\in[T',T]$$

and

$$\begin{aligned}\left\|\left[w^T(t),b^T(t)\right]^\top\right\| &= M, \qquad \textit{for a.e. } t\in(0,T')\\ \left\|\left[w^T(t),b^T(t)\right]^\top\right\| &= 0, \qquad \textit{for a.e. } t\in(T',T).\end{aligned}$$

for some $0<T'\leq T_M$.

Theorem 4.2 shows, in particular, that if one considers the learning problem with L^1 parameter regularization and parameters of norm $\leq M$ at every time (layer), then there exists a horizon $T_M>0$ after which any larger time horizon does not have any effect on the optimal parameters. We can then say that there is a finite optimal depth of the corresponding neural network, which depends on the choice of M, and is bounded from above by T_M. This is in fact a different situation compared to the behavior described in Theorem 3.1, where in general, the zero training error regime is in theory exactly reached only by considering the limit as T goes to infinity.

4.4. **Proofs.** We conclude this section by providing the proofs to the results stated above.

4.4.1. *Proof of Theorem 4.1.* The proof of Theorem 4.1 requires several preliminary results. Firstly, we will study the turnpike property for an optimization problem in which, rather than consider the final time cost ϕ as part of the cost functional, we minimize a running cost over a set with a terminal time condition for the state trajectory.

Under the reachability assumption for $\mathbf{x}^1$ in time $T_0 > 0$ by the dynamics (4.1) (resp. (4.2)), the set

$$\mathcal{U}_{T,\mathbf{x}^1} := \left\{u = [w,b]^\top \in L^2(0,T;\mathbb{R}^{d_u})\colon \mathbf{x}(T) = \mathbf{x}^1\right\} \tag{4.15}$$

where $\mathbf{x}$ is the solution to (4.1) (resp. (4.2)) associated to u, is non-empty whenever $T > 0$ by Corollary 3.1.

Let us consider the functional $\mathcal{J}_{T,\mathrm{ex}} : \mathcal{U}_{T,\mathbf{x}^1} \to \mathbb{R}_+$ defined by

$$\mathcal{J}_{T,\mathrm{ex}}(u) := \frac{\alpha}{2}\int_0^T \|u(t)\|^2\,\mathrm{d}t + \frac{\beta}{2}\int_0^T \|\mathbf{x}(t) - \mathbf{x}_d\|^2\,\mathrm{d}t,$$

where $\mathbf{x}$ is the solution to (4.1) (resp. (4.2)) associated to $u = [w,b]^\top \in \mathcal{U}_{T,\mathbf{x}^1}$. By a straightforward adaptation of the techniques used in the proof of Proposition 2.1, one can prove the existence of a minimizer of $\mathcal{J}_{T,\mathrm{ex}}$ in $\mathcal{U}_{T,\mathbf{x}^1}$, namely a solution to the minimization problem

$$\inf_{\substack{u\in\mathcal{U}_{T,\mathbf{x}^1}\\ \text{subject to (4.1)(resp.(4.2))}}} \mathcal{J}_{T,\mathrm{ex}}(u). \tag{4.16}$$

We begin by establishing estimates – uniform with respect to $T > 0$ – for the solutions to (4.16).

Lemma 4.1. *Let $\mathbf{x}_d \in \mathcal{R}_{T_0}(\mathbf{x}^0)$ and $\mathbf{x}^1 \in \mathcal{R}_{T_0}(\mathbf{x}_d)$ for some $T_0 > 0$ be given. Let $T \geq 2T_0$ be fixed, and let $u^T = [w^T, b^T]^\top \in \mathcal{U}_{T,\mathbf{x}^1}$ be a solution to (4.16), with $\mathbf{x}^T \in C^0([0,T];\mathbb{R}^{d_x})$ denoting the associated solution to (4.1) (resp. (4.2)). Assume there exist constants $L_N > 0$ and $r > 0$ such that*

$$\kappa_{T_0}(\mathbf{x},\mathbf{x}_d) \leq L_N^2\|\mathbf{x}-\mathbf{x}_d\|^2 \quad \text{and} \quad \kappa_{T_0}(\mathbf{x}_d,\mathbf{x}) \leq L_N^2\|\mathbf{x}-\mathbf{x}_d\|^2$$

for any $\mathbf{x} \in \mathbb{R}^{d_x}$ satisfying $\|\mathbf{x}-\mathbf{x}_d\| \leq r$. Assume that $\|\mathbf{x}^i - \mathbf{x}_d\| \leq r$ for $i = 0, 1$. Then, there exists a constant $C = C(\alpha,\beta,T_0,\mathbf{x}_d,\sigma,N) > 0$ independent of $T > 0$ such that

$$\left\|u^T\right\|^2_{H^k(0,T;\mathbb{R}^{d_u})} + \left\|\mathbf{x}^T - \mathbf{x}_d\right\|^2_{L^2(0,T;\mathbb{R}^{d_x})} + \left\|\mathbf{x}^T(t) - \mathbf{x}_d\right\|^2 \leq C\left(\left\|\mathbf{x}^0 - \mathbf{x}_d\right\|^2 + \left\|\mathbf{x}^1 - \mathbf{x}_d\right\|^2\right)$$

holds for all $t \in [0,T]$, where $k = 0$ in the case of (4.1), and $k = 1$ in the case of (4.2).

Remark 4. Before proceeding with the proof, let us simply note that Lemma 4.1 also stipulates that there exists some $\delta \in (0,r)$ such that whenever $\|\mathbf{x}^i - \mathbf{x}_d\| \leq \delta$ for $i = 0, 1$, then

$$\left\|\mathbf{x}^T(t) - \mathbf{x}_d\right\| \leq r$$

for all $t \in [0,T]$. This property will be of use in what follows.

Proof of Lemma 4.1. For notational convenience, we henceforth make use of the formulation (3.13), where for $u = [w, b]^\top$ we set $\mathbf{f}(u, \mathbf{x}) := \mathbf{f}(w, b, \mathbf{x})$.

The key point of the proof lies in the construction of an auxiliary control (steering $\mathbf{x}^0$ to $\mathbf{x}^1$ in time T whilst remaining at $\mathbf{x}_d$ over an interval of length $T - 2T_0$) in view of estimating each individual addend of $\mathcal{J}_{T,\mathrm{ex}}(u^T)$, which is the minimal value of the functional $\mathcal{J}_{T,\mathrm{ex}}$. This construction will yield the desired result.

Using the reachability and smallness assumptions, we know the following.

(i) There exist control parameters $u^\dagger = [w^\dagger, b^\dagger]^\top \in H^k(0, T_0; \mathbb{R}^{d_u})$ satisfying

$$\left\|u^\dagger\right\|^2_{H^k(0,T_0;\mathbb{R}^{d_u})} \leq L_N^2 \left\|\mathbf{x}^0 - \mathbf{x}_d\right\|^2, \tag{4.17}$$

and which are such that the corresponding solution $\mathbf{x}^\dagger$ to

$$\begin{cases} \dot{\mathbf{x}}^\dagger(t) = \mathbf{f}\left(u^\dagger(t), \mathbf{x}^\dagger(t)\right) & \text{in } (0, T_0) \\ \mathbf{x}^\dagger(0) = \mathbf{x}^0 \end{cases} \tag{4.18}$$

satisfies $\mathbf{x}^\dagger(T_0) = \mathbf{x}_d$. By using Grönwall's inequality, we see that

$$\begin{aligned} \left\|\mathbf{x}^\dagger(t)\right\| &\lesssim_{\sigma,N} \left(\left\|\mathbf{x}^0\right\| + \left\|b^\dagger\right\|_{L^2(0,T_0;\mathbb{R}^d)}\right) \exp\left(N \left\|w^\dagger\right\|_{L^2(0,T_0;\mathbb{R}^{d\times d})}\right) \\ &\lesssim_{\sigma,N} \left(\left\|\mathbf{x}^0\right\| + L\left\|\mathbf{x}^0 - \mathbf{x}_d\right\|\right) \exp\left(LN\left\|\mathbf{x}^0 - \mathbf{x}_d\right\|\right) \\ &\lesssim_{\sigma,N} \left(\left\|\mathbf{x}^0\right\| + Lr\right) \exp\left(LNr\right) \\ &\lesssim_{\sigma,N} \left(\left\|\mathbf{x}_d\right\| + (L+1)\, r\right) \exp\left(LNr\right) \end{aligned} \tag{4.19}$$

for every $t \in (0, T_0)$. Then, by definition of solution to (4.18), the Lipschitz property of σ as well as (4.19), we have

$$\left\|\mathbf{x}^\dagger(t) - \mathbf{x}_d\right\| \lesssim_{\sigma,N,\mathbf{x}_d,L,r} \left\|\mathbf{x}^0 - \mathbf{x}_d\right\| + \int_0^t \left(\left\|w^\dagger(s)\right\| + \left\|b^\dagger(s)\right\|\right) \mathrm{d}s,$$

whence, by Cauchy-Schwarz applied in $[0, T_0]$ and (4.17), we have

$$\begin{aligned} \left\|\mathbf{x}^\dagger(t) - \mathbf{x}_d\right\| &\lesssim_{\sigma,N,\mathbf{x}_d,L,r,T_0} \left\|\mathbf{x}^0 - \mathbf{x}_d\right\| + \left\|u^\dagger\right\|_{L^2(0,T_0;\mathbb{R}^{d_u})} \\ &\lesssim_{\sigma,N,\mathbf{x}_d,L,r,T_0} \left\|\mathbf{x}^0 - \mathbf{x}_d\right\|. \end{aligned} \tag{4.20}$$

(ii) There exist control parameters $u^\ddagger = [w^\ddagger, b^\ddagger]^\top \in H^k(0, T_0; \mathbb{R}^{d_u})$ satisfying

$$\left\|u^\ddagger\right\|^2_{H^k(0,T_0;\mathbb{R}^{d_u})} \leq L_N^2 \left\|\mathbf{x}_d - \mathbf{x}^1\right\|^2, \tag{4.21}$$

and which are such that the corresponding solution $\mathbf{x}^\ddagger$ to

$$\begin{cases} \dot{\mathbf{x}}^\ddagger(t) = \mathbf{f}\left(u^\ddagger(t), \mathbf{x}^\ddagger(t)\right) & \text{in } (0, T_0) \\ \mathbf{x}^\ddagger(0) = \mathbf{x}_d \end{cases}$$

satisfies $\mathbf{x}^\ddagger(T_0) = \mathbf{x}^1$. By Grönwall's inequality, we see that

$$\begin{aligned}\left\|\mathbf{x}^\ddagger(t)\right\| &\lesssim_{\sigma,N} \left(\|\mathbf{x}_d\| + \left\|b^\ddagger\right\|_{L^2(0,T_0;\mathbb{R}^d)}\right) \exp\left(N\left\|w^\ddagger\right\|_{L^2(0,T_0;\mathbb{R}^{d\times d})}\right)\\ &\lesssim_{\sigma,N} \left(\|\mathbf{x}_d\| + L\left\|\mathbf{x}_d - \mathbf{x}^1\right\|\right) \exp\left(LN\left\|\mathbf{x}_d - \mathbf{x}^1\right\|\right)\\ &\lesssim_{\sigma,N} \left(\|\mathbf{x}_d\| + Lr\right) \exp\left(LNr\right) \end{aligned} \tag{4.22}$$

for every $t \in (0, T_0)$. Then by definition of solution to (4.18), the Lipschitz property of σ as well as (4.22), we have

$$\left\|\mathbf{x}^\ddagger(t) - \mathbf{x}_d\right\| \lesssim_{\sigma,N,\mathbf{x}_d,L,r} \int_0^t \left(\left\|w^\ddagger(s)\right\| + \left\|b^\ddagger(s)\right\|\right) \mathrm{d}s,$$

whence, by Cauchy-Schwarz applied in $[0, T_0]$ and (4.17), we have

$$\begin{aligned}\left\|\mathbf{x}^\ddagger(t) - \mathbf{x}_d\right\| &\lesssim_{\sigma,N,\mathbf{x}_d,L,r} \left\|u^\ddagger\right\|_{L^2(0,T_0;\mathbb{R}^{d_u})}\\ &\lesssim_{\sigma,N,\mathbf{x}_d,L,r} \left\|\mathbf{x}_d - \mathbf{x}^1\right\|. \end{aligned} \tag{4.23}$$

Now set

$$u^{\mathrm{aux}}(t) := \begin{cases} u^\dagger(t) & \text{in } (0, T_0)\\ 0 & \text{in } (T_0, T - T_0)\\ u^\ddagger\left(t - (T - T_0)\right) & \text{in } (T - T_0, T), \end{cases}$$

and let $\mathbf{x}^{\mathrm{aux}}$ be the corresponding solution to (3.13) on $(0, T)$. By construction, we have $\mathbf{x}^{\mathrm{aux}}(t) = \mathbf{x}^\dagger(t)$ on $[0, T_0]$ and thus

$$\mathbf{x}^{\mathrm{aux}}(t) = \mathbf{x}_d \qquad \text{for all } t \in [T_0, T - T_0], \tag{4.24}$$

whereas we also have $\mathbf{x}^{\mathrm{aux}}(T) = \mathbf{x}^1$, whence $u^{\mathrm{aux}} \in \mathcal{U}_{T,\mathbf{x}^1}$, with $\mathcal{U}_{T,\mathbf{x}^1}$ defined in (4.15).

We now evaluate $\mathcal{J}_{T,\mathrm{ex}}$ at u^{aux}, which by virtue of a simple change of variable as well as (4.24), (4.17), (4.20), (4.21) and (4.23), leads us to

$$\begin{aligned}\mathcal{J}_{T,\mathrm{ex}}(u^{\mathrm{aux}}) =& \frac{\alpha}{2}\left\|u^\dagger\right\|_{H^k(0,T_0;\mathbb{R}^{d_u})} + \frac{\alpha}{2}\left\|u^\ddagger\right\|_{H^k(0,T_0;\mathbb{R}^{d_u})}\\ &+ \frac{\beta}{2}\int_0^{T_0}\left\|\mathbf{x}^\dagger(t) - \mathbf{x}_d\right\|^2 \mathrm{d}t + \frac{\beta}{2}\int_0^{T_0}\left\|\mathbf{x}^\ddagger(t) - \mathbf{x}_d\right\|^2 \mathrm{d}t\\ &\lesssim_{\alpha,\beta,\sigma,N,\mathbf{x}_d,L,r} \left(\left\|\mathbf{x}_d - \mathbf{x}^0\right\|^2 + \left\|\mathbf{x}_d - \mathbf{x}^1\right\|^2\right). \end{aligned} \tag{4.25}$$

Hence, the given solution $u^T \in \mathcal{U}_{\mathrm{ad,ex}}$ to the minimization problem (4.16) is uniformly bounded with respect to $T > 0$, namely we have

$$\begin{aligned}\left\|u^T\right\|^2_{H^k(0,T;\mathbb{R}^{d_u})} &\le \frac{2}{\alpha}\mathcal{J}_{T,\mathrm{ex}}\left(u^T\right) \le \frac{2}{\alpha}\mathcal{J}_{T,\mathrm{ex}}\left(u^{\mathrm{aux}}\right)\\ &\lesssim_{\alpha,\beta,\sigma,N,\mathbf{x}_d,L,r} \left(\left\|\mathbf{x}_d - \mathbf{x}^0\right\|^2 + \left\|\mathbf{x}_d - \mathbf{x}^1\right\|^2\right). \end{aligned}$$

On another hand, the form of $\mathcal{J}_{T,\mathrm{ex}}$ and (4.25) give

$$\begin{aligned}\left\|\mathbf{x}^T - \mathbf{x}_d\right\|^2_{L^2(0,T;\mathbb{R}^{d_x})} &\le \frac{2}{\beta}\mathcal{J}_{T,\mathrm{ex}}\left(u^T\right) \le \frac{2}{\beta}\mathcal{J}_{T,\mathrm{ex}}\left(u^{\mathrm{aux}}\right)\\ &\lesssim_{\alpha,\beta,\sigma,N,\mathbf{x}_d,L,r} \left(\left\|\mathbf{x}_d - \mathbf{x}^0\right\|^2 + \left\|\mathbf{x}_d - \mathbf{x}^1\right\|^2\right). \end{aligned}$$

An application of Lemma A.1 combined with the uniform boundedness of $\|u^T\|_{L^2(0,T;\mathbb{R}^{d_u})}$ with respect to $T > 0$ is sufficient to conclude. □

We will need the following useful lemma.

Lemma 4.2. *Let X be a real Banach space, $T > 0$ and $f \in L^2(0,T;X)$. For any $\tau \leq \frac{T}{2}$, there exist $t_1 \in [0,\tau)$ and $t_2 \in (T-\tau, T]$ such that*

$$\|f(t_i)\|_X \leq \varepsilon(\tau) \qquad \text{for } i = 1,2$$

where $\varepsilon(\tau) := \dfrac{\|f\|_{L^2(0,T;X)}}{\sqrt{\tau}}$.

Proof. We argue by contradiction. Assume that either

$$\|f(t)\|_X > \varepsilon(\tau) \qquad \text{for all } t \in [0,\tau)$$

or

$$\|f(t)\|_X > \varepsilon(\tau) \qquad \text{for all } t \in (T-\tau, T].$$

hold. Then we have

$$\int_0^T \|f(t)\|_X^2 \,\mathrm{d}t \geq \int_0^\tau \|f(t)\|_X^2 \,\mathrm{d}t + \int_{T-\tau}^T \|f(t)\|_X^2 \,\mathrm{d}t > \tau\varepsilon(\tau)^2.$$

Hence

$$\varepsilon(\tau)^2 < \frac{1}{\tau}\int_0^T \|f(t)\|_X^2 \,\mathrm{d}t = \varepsilon(\tau)^2,$$

which yields a contradiction. This concludes the proof. □

We now prove the following key local turnpike result.

Proposition 4.1. *Let $\mathbf{x}_d \in \mathcal{R}_{T_0}(\mathbf{x}^0)$ and $\mathbf{x}^1 \in \mathcal{R}_{T_0}(\mathbf{x}_d)$ for some $T_0 > 0$ be given. Assume there exist constants $L_N > 0$ and $r > 0$ such that*

$$\kappa_{T_0}(\mathbf{x}, \mathbf{x}_d) \leq L_N^2 \|\mathbf{x} - \mathbf{x}_d\|^2 \quad \text{and} \quad \kappa_{T_0}(\mathbf{x}_d, \mathbf{x}) \leq L_N^2 \|\mathbf{x} - \mathbf{x}_d\|^2$$

for all $\mathbf{x} \in \mathbb{R}^{d_x}$ such that $\|\mathbf{x} - \mathbf{x}_d\| \leq r$. Assume that $\|\mathbf{x}^i - \mathbf{x}_d\| \leq \delta$ where $\delta \in (0,r)$ is as in Remark 4. Then, for any

$$T^\dagger > 2T_0 + 32C^4,$$

where $C = C(\alpha, \beta, T_0, \mathbf{x}_d, N, \sigma) > 0$ appears in Lemma 4.1, there exist constants $\Lambda = \Lambda(C, L, T_0, T^\dagger, N) > 0$ and $\mu = \mu(C, L, T_0, T^\dagger, N) > 0$ such that for all $T \geq T^\dagger$, whenever $u^T := \left[w^T, b^T\right]^\top \in \mathcal{U}_{T,\mathbf{x}^1}$ is a solution to (4.16), the corresponding state trajectory $\mathbf{x}^T \in C^0([0,T];\mathbb{R}^{d_x})$, solution to (4.1) (resp. (4.2)), satisfies

$$\left\|\mathbf{x}^T(t) - \mathbf{x}_d\right\| \leq \Lambda\left(e^{-\mu t} + e^{-\mu(T-t)}\right)\left(\left\|\mathbf{x}_d - \mathbf{x}^0\right\| + \left\|\mathbf{x}_d - \mathbf{x}^1\right\|\right)$$

for any $t \in [0,T]$.

Proof. We set

$$\tau := \frac{T^\dagger}{2} - T_0.$$

We start by proving that, for all $n \in \mathbb{N}^*$ satisfying

$$n \leq \frac{1}{\tau}\left(\frac{T}{2} - T_0\right),$$

one has

$$\left\|\mathbf{x}^T(t)-\mathbf{x}_d\right\| \leq \frac{1}{2}\left(\frac{4C^2}{\sqrt{\tau}}\right)^n \Big(\left\|\mathbf{x}_d-\mathbf{x}^0\right\| + \left\|\mathbf{x}_d-\mathbf{x}^1\right\| \Big), \quad \text{for } \ t\in[n\tau, T-n\tau]. \tag{4.26}$$

We proceed in proving (4.26) by induction, beginning with the case $n=1$. Applying Lemma 4.2 to the function $f(\cdot) := \mathbf{x}^T(\cdot)-\mathbf{x}_d \in C^0([0,T];X)$, with $X=\mathbb{R}^{d_x}$, and using the estimate from Lemma 4.1, we see that there exist $t_1\in[0,\tau)$ and $t_2\in(T-\tau,T]$ such that

$$\begin{aligned}\left\|\mathbf{x}^T(t_i)-\mathbf{x}_d\right\| &\leq \frac{\|\mathbf{x}-\mathbf{x}_d\|_{L^2(0,T;\mathbb{R}^{d_x})}}{\sqrt{\tau}}\\ &\leq \frac{C}{\sqrt{\tau}}\Big(\|\mathbf{x}_d-\mathbf{x}^0\|+\|\mathbf{x}_d-\mathbf{x}^1\|\Big)\end{aligned}$$

for $i=1,2$. Now observe that, if $u^T := \left[w^T, b^T\right]^\top$ is a solution to (4.16) with $\mathbf{x}^T$ its associated state, then the control $u^T|_{[t_1,t_2]}$ is a solution to the problem

$$\inf_{\substack{u:=[w,b]^\top\in\mathcal{U}_{t_1,t_2}\\ \text{subject to (4.28)}}} \frac{\alpha}{2}\int_{t_1}^{t_2}\|u(t)\|^2\,\mathrm{d}t + \frac{\beta}{2}\int_{t_1}^{t_2}\|\mathbf{z}(t)-\mathbf{x}_d\|^2\,\mathrm{d}t \tag{4.27}$$

where

$$\mathcal{U}_{t_1,t_2} := \{u=[w,b]^\top\in L^2(t_1,t_2;\mathbb{R}^{d_u})\colon \mathbf{z}(t_2)=\mathbf{x}^T(t_2)\}$$

and

$$\begin{cases}\dot{\mathbf{z}}(t)=\mathbf{f}(u(t),\mathbf{z}(t)) & \text{in } (t_1,t_2)\\ \mathbf{z}(t_1)=\mathbf{x}^T(t_1).\end{cases} \tag{4.28}$$

Of course, for the control $u^T|_{[t_1,t_2]}$, the corresponding state is $\mathbf{z}=\mathbf{x}^T|_{[t_1,t_2]}$. Hence, for t_1 and t_2 as above, we apply the estimate from Lemma 4.1 to (4.27) – (4.28) to obtain

$$\begin{aligned}\left\|\mathbf{x}^T(t)-\mathbf{x}_d\right\| &\leq C\Big(\left\|\mathbf{x}^T(t_1)-\mathbf{x}_d\right\|+\left\|\mathbf{x}^T(t_2)-\mathbf{x}_d\right\|\Big)\\ &\leq \frac{2C^2}{\sqrt{\tau}}\Big(\|\mathbf{x}_d-\mathbf{x}^0\|+\|\mathbf{x}_d-\mathbf{x}^1\|\Big),\end{aligned}$$

for all $t\in[\tau,\,T-\tau]$. Note that the use of Lemma 4.1 is justified since

$$t_2-t_1 \geq T-2\tau = T-T^\dagger+2T_0 \geq 2T_0.$$

Thus, (4.26) holds for $n=1$.

Now, let us suppose that (4.26) holds for some $n\in\mathbb{N}^*$, and suppose that

$$n+1 \leq \frac{1}{\tau}\left(\frac{T}{2}-T_0\right). \tag{4.29}$$

As before, we look to apply Lemma 4.2 to the function $f(\cdot)=\mathbf{x}^T(\cdot)-\mathbf{x}_d \in C^0([0,T];\mathbb{R}^{d_x})$, but this time in the interval $[n\tau, T-n\tau]$. Observe that inequality (4.29) clearly implies

$$n+1 \leq \frac{T}{2\tau},$$

which is itself equivalent to

$$\tau \leq \frac{T-2n\tau}{2}.$$

In view of Lemma 4.2, there exist $t_1' \in [n\tau,\, (n+1)\tau)$ and $t_2' \in (T-(n+1)\tau,\, T-n\tau]$ such that

$$\begin{aligned}\left\|\mathbf{x}^T(t_i') - \mathbf{x}_d\right\| &\leq \frac{\left\|\mathbf{x}^T - \mathbf{x}_d\right\|_{L^2(n\tau, T-n\tau;\mathbb{R}^{d_x})}}{\sqrt{\tau}} \\ &\leq \frac{C}{\sqrt{\tau}}\Big(\left\|\mathbf{x}_d - \mathbf{x}^T(n\tau)\right\| + \left\|\mathbf{x}_d - \mathbf{x}^T(T-n\tau)\right\| \Big)\end{aligned}$$

for $i = 1, 2$. Here, as before, we used Lemma 4.1 for the problem (4.27) – (4.28), with $t_1 = n\tau$ and $t_2 = T - n\tau$. Observe that (4.29) implies

$$t_2 - t_1 = T - 2n\tau \geq T - 2(n+1)\tau \geq 2T_0.$$

Using the fact that (4.26) holds at stage n, we obtain

$$\left\|\mathbf{x}^T(t_i') - \mathbf{x}_d\right\| \leq \left(\frac{4C}{\sqrt{\tau}}\right)^n \frac{C}{\sqrt{\tau}}\Big(\|\mathbf{x}_d - \mathbf{x}^0\| + \|\mathbf{x}_d - \mathbf{x}^1\|\Big), \qquad \text{for } i = 1, 2. \quad (4.30)$$

Finally, using (4.30) and by applying Lemma 4.1 to problem (4.27) – (4.28) once again, this time with $t_1 = t_1'$ and $t_2 = t_2'$, we obtain

$$\begin{aligned}\left\|\mathbf{x}^T(t) - \mathbf{x}_d\right\| &\leq C\Big(\left\|\mathbf{x}^T(t_1') - \mathbf{x}_d\right\| + \left\|\mathbf{x}^T(t_2') - \mathbf{x}_d\right\| \Big) \\ &\leq \frac{1}{2}\left(\frac{4C^2}{\sqrt{\tau}}\right)^n \frac{4C^2}{\sqrt{\tau}}\Big(\|\mathbf{x}_d - \mathbf{x}^0\| + \|\mathbf{x}_d - \mathbf{x}^1\|\Big),\end{aligned}$$

for all $t \in [(n+1)\tau,\, T-(n+1)\tau] \subset [t_1', t_2']$. The application of Lemma 4.1 is again justified by (4.29) since

$$t_2' - t_1' \geq T - 2(n+1)\tau \geq 2T_0.$$

Statement (4.26) is hence proven.

Now for any $t \in [0, T]$, we set

$$n(t) := \min\left\{ \left\lfloor \frac{t}{\tau M} \right\rfloor, \left\lfloor \frac{T-t}{\tau M} \right\rfloor \right\},$$

where

$$M := \frac{T^\dagger}{T^\dagger - 2T_0} = \frac{T^\dagger}{2\tau}$$

We will address the cases $n(t) \geq 1$ and $n(t) = 0$ separately.

Case 1: Assume $n(t) \geq 1$, or equivalently, $t \in [\tau M,\, T - \tau M]$. It is not difficult to see that, for any such t we have

$$n(t) \leq \frac{T}{2\tau M} = \frac{T}{2\tau}\frac{T^\dagger - 2T_0}{T^\dagger} \leq \frac{T}{2\tau}\frac{T-2T_0}{T} = \frac{1}{\tau}\left(\frac{T}{2} - T_0\right),$$

hence we can use (4.26). Moreover, since $M > 1$, any $t \in [0, T]$ satisfies

$$n(t)\tau \leq t \leq T - n(t)\tau.$$

Whence we can apply (4.26) to obtain

$$\left\|\mathbf{x}^T(t)-\mathbf{x}_d\right\| \leq \frac{1}{2}\left(\frac{4C^2}{\sqrt{\tau}}\right)^{n(t)}\Big(\|\mathbf{x}_d-\mathbf{x}^0\|+\|\mathbf{x}_d-\mathbf{x}^1\|\Big) \tag{4.31}$$

for all $t\in[\tau M,\, T-\tau M]$. Now, since $T^*>2T_0+32C^4$, we deduce

$$\frac{4C^2}{\sqrt{\tau}}=\frac{4C^2}{\sqrt{\frac{T^\dagger}{2}-T_0}}<1.$$

Hence, using the fact that either $n(t)\geq \dfrac{t}{\tau M}-1$ or $n(t)\geq \dfrac{T-t}{\tau M}-1$, from (4.31), it follows that

$$\begin{aligned}\left\|\mathbf{x}^T(t)-\mathbf{x}_d\right\| &\leq \frac{1}{2}\Bigg[\exp\left\{-\left(\frac{t}{\tau M}-1\right)\log\left(\frac{\sqrt{\tau}}{4C^2}\right)\right\}\\ &\qquad+\exp\left\{-\left(\frac{T-t}{\tau M}-1\right)\log\left(\frac{\sqrt{\tau}}{4C^2}\right)\right\}\Bigg]\Big(\|\mathbf{x}_d-\mathbf{x}^0\|+\|\mathbf{x}_d-\mathbf{x}^1\|\Big)\\ &=\frac{\sqrt{\tau}}{8C^2}\left(e^{-\mu t}+e^{-\mu(T-t)}\right)\Big(\|\mathbf{x}_d-\mathbf{x}^0\|+\|\mathbf{x}_d-\mathbf{x}^1\|\Big)\end{aligned}$$

for all $t\in[\tau M,\, T-\tau M]$, where

$$\mu:=\frac{1}{\tau M}\log\left(\frac{\sqrt{\tau}}{4C^2}\right)=\frac{2}{T^\dagger}\log\left(\frac{\sqrt{\tau}}{4C^2}\right).$$

Case 2: If $t\in[0,\,\tau M)\cup(T-\tau M,\,T]$, then by Lemma 4.1 we have

$$\begin{aligned}\left\|\mathbf{x}^T(t)-\mathbf{x}_d\right\| &\leq C\Big(\|\mathbf{x}_d-\mathbf{x}^0\|+\|\mathbf{x}_d-\mathbf{x}^1\|\Big)\\ &\leq Ce^{\mu\tau M}\left(e^{-\mu t}+e^{-\mu(T-t)}\right)\Big(\|\mathbf{x}_d-\mathbf{x}^0\|+\|\mathbf{x}_d-\mathbf{x}^1\|\Big).\end{aligned}$$

Observe that

$$C\exp(\mu\tau M)=C\exp\left(\mu\frac{T^\dagger}{2}\right)=\frac{\sqrt{\tau}}{4C}.$$

The desired conclusion thence holds with $\Lambda:=\frac{\sqrt{\tau}}{4C}\max\left\{\frac{1}{2C},\,1\right\}$. □

We now come back to the proof of Theorem 4.1, which consists in showing a turnpike phenomenon for the minimizers of $\mathcal{J}_T$ defined in (4.3). The proof of Theorem 4.1 will follow by virtue of combining Proposition 4.1 with the following lemma.

Lemma 4.3. *Let $\mathbf{x}_d\in\mathcal{R}_{T_0}(\mathbf{x}^0)$ for some $T_0>0$ be given. Let $T>0$ and let $u^T=[w^T,b^T]^\top\in H^k(0,T;\mathbb{R}^{d_u})$ be a global minimizer of $\mathcal{J}_T$ defined in (4.3), and denote by $\mathbf{x}^T\in C^0([0,T];\mathbb{R}^{d_x})$ the associated state, solution to (4.1) where $k=0$ (resp. (4.2) where $k=1$). Then, there exists a constant $C=C(\alpha,\beta,T_0,\mathbf{x}^0,\mathbf{x}_d,\sigma,N)>0$ independent of $T>0$ such that*

$$\left\|u^T\right\|_{H^k(0,T;\mathbb{R}^{d_u})}+\left\|\mathbf{x}^T-\mathbf{x}_d\right\|_{L^2(0,T;\mathbb{R}^{d_x})}+\left\|\mathbf{x}^T(t)-\mathbf{x}_d\right\|\leq C$$

holds for all $t\in[0,T]$.

Proof. We begin by considering the case $T \leq T_0$. The inequalities $\mathcal{J}_T(u^T) \leq \mathcal{J}_T(0)$ and $T \leq T_0$ give

$$\frac{\alpha}{2}\left\|u^T\right\|^2_{H^k(0,T;\mathbb{R}^{d_u})} + \frac{\beta}{2}\left\|\mathbf{x}^T - \mathbf{x}_d\right\|^2_{L^2(0,T;\mathbb{R}^{d_x})} \leq \phi(\mathbf{x}^0) + T_0\|\mathbf{x}^0 - \mathbf{x}_d\|^2. \tag{4.32}$$

Combining (4.32) with Lemma A.1, we see that

$$\left\|u^T\right\|_{H^k(0,T;\mathbb{R}^{d_u})} + \left\|\mathbf{x}^T - \mathbf{x}_d\right\|_{L^2(0,T;\mathbb{R}^{d_x})} + \left\|\mathbf{x}^T(t) - \mathbf{x}_d\right\| \leq C_1$$

holds for some $C_1 = C_1\left(\alpha, \beta, \sigma, N, \mathbf{x}^0, \mathbf{x}_d, T_0\right) > 0$ and all $t \in [0,T]$.

We now consider the case $T \geq T_0$. We will begin by showing that there exists a constant $C_2 = C_2\left(\alpha, \beta, \mathbf{x}^0, \mathbf{x}_d, T_0, \sigma, N\right) > 0$ independent of $T > 0$ such that

$$\left\|u^T\right\|_{H^k(0,T;\mathbb{R}^{d_u})} \leq C_2.$$

By virtue of the reachability assumption, there exists control parameters $u^{T_0} = [w^{T_0}, b^{T_0}]^\top \in H^k(0,T_0;\mathbb{R}^{d_u})$ satisfying

$$\left\|u^{T_0}\right\|_{H^k(0,T_0;\mathbb{R}^{d_u})} \leq \kappa_{T_0}(\mathbf{x}^0, \mathbf{x}_d), \tag{4.33}$$

and which are such that the corresponding solution $\mathbf{x}^{T_0} \in C^0([0,T_0];\mathbb{R}^{d_x})$ to (4.1) (resp. (4.2)) set on $(0,T_0)$ satisfies $\mathbf{x}^{T_0}(T_0) = \mathbf{x}_d$. Set

$$u^{\text{aux}}(t) := \begin{cases} u^{T_0}(t) & \text{in } (0,T_0) \\ 0 & \text{in } (T_0,T), \end{cases}$$

and denote by $\mathbf{x}^{\text{aux}} \in C^0([0,T];\mathbb{R}^{d_x})$ the associated solution to (4.1) (resp. (4.2)) set on $(0,T)$. By construction, we have $\mathbf{x}^{\text{aux}}(t) = \mathbf{x}_d$ for $t \in [T_0,T]$, and also $\mathbf{x}^{\text{aux}}(t) = \mathbf{x}^{T_0}(t)$ for $t \in [0,T_0]$. Hence, using (4.33), we obtain

$$\mathcal{J}_T\left(u^{\text{aux}}\right) = \frac{\alpha}{2}\left\|u^{T_0}\right\|^2_{H^k(0,T_0;\mathbb{R}^{d_u})} + \frac{\beta}{2}\int_0^{T_0}\left\|\mathbf{x}^{T_0}(t) - \mathbf{x}_d\right\|^2 \mathrm{d}t + \phi(\mathbf{x}_d) \leq C_3 \tag{4.34}$$

for some $C_3 = C_3\left(\alpha, \beta, \mathbf{x}^0, \mathbf{x}_d, T_0\right) > 0$ independent of $T > 0$. Consequently,

$$\left\|u^T\right\|^2_{H^k(0,T;\mathbb{R}^{d_u})} \leq \frac{2}{\alpha}\,\mathcal{J}_T\left(u^T\right) \leq \frac{2}{\alpha}\mathcal{J}_T\left(u^{\text{aux}}\right) \lesssim_\alpha C_3.$$

Hence $\|u^T\|_{L^2(0,T;\mathbb{R}^{d_u})}$ is bounded uniformly in $T > 0$. On the other hand, by (4.34), we also have

$$\left\|\mathbf{x}^T - \mathbf{x}_d\right\|^2_{L^2(0,T;\mathbb{R}^{d_x})} \leq \frac{2}{\beta}\mathcal{J}_T\left(u^T\right) \leq \frac{2}{\beta}\mathcal{J}_T\left(u^{\text{aux}}\right) \lesssim_\beta C_3.$$

These last two estimates combined with Lemma A.1 yield the stated estimate, as desired. □

We are now in a position to conclude the proof of Theorem 4.1.

Proof of Theorem 4.1. First of all, by Lemma 4.3, we have

$$\left\|u^T\right\|^2_{L^2(0,T;\mathbb{R}^{d_u})} + \left\|\mathbf{x}^T - \mathbf{x}_d\right\|^2_{L^2(0,T;\mathbb{R}^{d_x})} + \left\|\mathbf{x}^T(t) - \mathbf{x}_d\right\| \leq C, \tag{4.35}$$

for some $C = C(\alpha, \beta, T_0, \mathbf{x}^0, \mathbf{x}_d, \sigma, N) > 0$ and for any $t \in [0,T]$, whence (4.4) follows.

Let us now prove (4.5). Let $\delta \in \left(0, \min\left\{\frac{1}{2}, r\right\}\right)$ be as in Remark 4. We apply Lemma 4.2, with $\tau := \frac{2C}{\delta^2}$, to the effect of obtaining $t_1 \in [0, \tau)$ and $t_2 \in (T-\tau, T]$ such that for $i = 1, 2$

$$\left\|\mathbf{x}^T(t_i) - \mathbf{x}_d\right\| \leq \frac{\left\|\mathbf{x}^T - \mathbf{x}_d\right\|_{L^2(0,T;\mathbb{R}^{d_x})}}{\sqrt{\tau}} < \delta. \tag{4.36}$$

Now note that $u^T|_{[t_1,t_2]}$ is a solution to the problem

$$\inf_{\substack{u:=[w,b]^\top \in \mathcal{U}_{t_1,t_2} \\ \text{subject to (4.28)}}} \frac{\alpha}{2}\int_{t_1}^{t_2} \|u(t)\|^2\,\mathrm{d}t + \frac{\beta}{2}\int_{t_1}^{t_2} \|\mathbf{z}(t) - \mathbf{x}_d\|^2\,\mathrm{d}t$$

where

$$\mathcal{U}_{t_1,t_2} := \left\{u = [w,b]^\top \in L^2(t_1,t_2;\mathbb{R}^{d_u})\colon \mathbf{z}(t_2) = \mathbf{x}^T(t_2)\right\}$$

and, employing the notation (3.13), the underlying dynamics are

$$\begin{cases} \dot{\mathbf{z}}(t) = \mathbf{f}(u(t), \mathbf{z}(t)) & \text{in } (t_1, t_2) \\ \mathbf{z}(t_1) = \mathbf{x}^T(t_1). \end{cases}$$

Of course, for the control $u^T|_{[t_1,t_2]}$, the corresponding state is $\mathbf{z} = \mathbf{x}^T|_{[t_1,t_2]}$. Hence, by (4.36), we are in position to apply Proposition 4.1 in $[t_1, t_2]$, thus obtaining for any

$$T^\dagger > 2T_0 + 32C^4,$$

the existence of constants $\Lambda = \Lambda(C, L, T_0, T^\dagger) > 0$ and $\mu = \mu(C, L, T_0, T^\dagger) > 0$ such that, for all $T \geq T^\dagger + 2\tau$, whenever $u^T := \left[w^T, b^T\right]^\top \in \mathcal{U}_{T,\mathbf{x}^1}$ is a solution to (4.16), the corresponding state trajectory $\mathbf{x}^T \in C^0([0,T];\mathbb{R}^{d_x})$, solution to (4.1) (resp. (4.2)), satisfies

$$\begin{aligned} \left\|\mathbf{x}^T(t) - \mathbf{x}_d\right\| &\leq \Lambda\Big(\left\|\mathbf{x}_d - \mathbf{x}^T(t_1)\right\| + \left\|\mathbf{x}_d - \mathbf{x}^T(t_2)\right\|\Big)\left(e^{-\mu(t-t_1)} + e^{-\mu(t_2-t+t_1)}\right) \\ &\leq 2\Lambda\delta\left(e^{-\mu(t-t_1)} + e^{-\mu(t_2-t+t_1)}\right) \\ &\leq \Lambda e^{\frac{2C\mu}{\delta^2}}\left(e^{-\mu t} + e^{-\mu(T-t)}\right), \end{aligned} \tag{4.37}$$

for any $t \in [t_1, T-t_2]$, where we have used $\delta < \frac{1}{2}$, $t_1 \in [0,\tau)$ and $t_2 \in (T-\tau, T]$, with $\tau := \frac{2C}{\delta^2}$. Now set

$$\gamma := \max\{C, \Lambda\} \exp\left(\mu\left(\frac{2C}{\delta^2} + T^\dagger\right)\right), \tag{4.38}$$

where C is given by (4.35). Then, on one hand, by (4.37), for any $t \in [t_1, T-t_2]$, we have

$$\left\|\mathbf{x}^T(t) - \mathbf{x}_d\right\| \leq e^{\frac{2C\mu}{\delta^2}}\Lambda\left(e^{-\mu t} + e^{-\mu(T-t)}\right) \leq \gamma\left(e^{-\mu t} + e^{-\mu(T-t)}\right). \tag{4.39}$$

On the other hand, by (4.35), for any $t \in [0, t_1]$,

$$\left\|\mathbf{x}^T(t) - \mathbf{x}_d\right\| \leq C \leq C\exp\left(\mu\left(\frac{2C}{\delta^2} - t\right)\right) \leq \gamma\exp\left(-\mu t\right) \tag{4.40}$$

and for any $t \in [T - t_2, T]$

$$\left\|\mathbf{x}^T(t) - \mathbf{x}_d\right\| \leq C \leq C \exp\left(\mu\left(\frac{2C}{\delta^2} - T + t\right)\right) \leq \gamma \exp\left(-\mu\left(T - t\right)\right). \tag{4.41}$$

By (4.39), (4.40) and (4.41), statement (4.5) holds for $T \geq T^\dagger + 2\tau$. For $T < T^\dagger + 2\tau$, by (4.35), the definitions of $\tau = \frac{2C}{\delta^2}$ and γ (4.38)

$$\left\|\mathbf{x}^T(t) - \mathbf{x}_d\right\| \leq C \leq C \exp\left(\mu\left(\frac{2C}{\delta^2} + T^\dagger - t\right)\right) \leq \gamma \exp\left(-\mu t\right),$$

whence (4.5) follows. This concludes the proof. □

Let us now provide a proof to Corollary 4.1.

Proof of Corollary 4.1. Firstly note that the functional (4.7) is a particular case of (4.3) with $\phi \equiv 0$ (due to the specific form of ϕ (3.4), this amounts to loss $\equiv 0$). Hence the conclusions of Theorem 4.1 hold in this context as well.

We claim that

$$\left\|\mathbf{x}^T(T) - \mathbf{x}_d\right\| \leq \left\|\mathbf{x}^T(t) - \mathbf{x}_d\right\| \qquad \text{for all } t \in [0, T]. \tag{4.42}$$

To this end, first notice that since $t \mapsto \mathbf{x}^T(t)$ is a continuous function, $t \mapsto \left\|\mathbf{x}^T(t) - \mathbf{x}_d\right\|$ attains its minimum in $[0, T]$. Let us thus define[7]

$$t' := \max\left\{ t \in [0, T] \ : \ t \in \arg\min_{[0,T]} \left\|\mathbf{x}^T(\cdot) - \mathbf{x}_d\right\| \right\}.$$

Claim (4.42) thus holds if and only if $t' = T$. Suppose by contradiction that $t' < T$. Then (recall that $u^T := \left[w^T, b^T\right]^\top$ is a global minimizer) consider $u^{\text{aux}} = \left[w^{\text{aux}}, b^{\text{aux}}\right]^\top$ defined by

$$u^{\text{aux}}(t) := \begin{cases} u^T(t) & \text{for } \ t \in [0, t'] \\ 0 & \text{for } t \in [t', T]. \end{cases}$$

The state trajectory $\mathbf{x}^{\text{aux}}$, solution to (4.1) (resp. (4.2)) associated to u^{aux} is precisely

$$\mathbf{x}^{\text{aux}}(t) = \begin{cases} \mathbf{x}^T(t) & \text{for } \ t \in [0, t'] \\ \mathbf{x}^T(t') & \text{for } \ t \in [t', T]. \end{cases}$$

By definition of t', whenever $t \in (t', T]$, there exists some $s \in [0, T]$ such that

$$\left\|\mathbf{x}^T(t) - \mathbf{x}_d\right\| > \left\|\mathbf{x}^T(s) - \mathbf{x}_d\right\| \geq \left\|\mathbf{x}^T(t') - \mathbf{x}_d\right\|$$

and hence $\mathcal{J}_T\left(u^{\text{aux}}\right) < \mathcal{J}_T\left(u^T\right)$, which contradicts the optimality of u^T. Claim (4.42) thus follows.

To conclude, it suffices to use (4.42) with $t = \frac{T}{2}$ and apply Theorem 4.1 to obtain

$$\left\|\mathbf{x}^T(T) - \mathbf{x}_d\right\| \leq \left\|\mathbf{x}^T\left(\frac{T}{2}\right) - \mathbf{x}_d\right\| \leq 2Ce^{-\mu\frac{T}{2}}.$$ □

[7]The max is clearly well defined, as the set in question is bounded, and also closed as the preimage of the singleton $\left\{ \min_{s\in[0,T]} \left\|\mathbf{x}^T(s) - \mathbf{x}_d\right\| \right\}$ under the continuous map $t \mapsto \left\|\mathbf{x}^T(t) - \mathbf{x}_d\right\|$.

4.4.2. *Proof of Theorem 4.2.* We finish this section by providing the proof to Theorem 4.2. We first note that the existence of a solution to (4.14) can be obtained by adapting the techniques of Theorem 2.1 and observing that the bilateral constraints give weak pre-compacteness of any minimizing sequence in L^2.

Proof of Theorem 4.2. We split the proof in a succession of three chained parts.

Part 1: We shall first show that there exists $T' \in [0,T]$ such that

$$\begin{aligned} \phi\left(\mathbf{x}^T(t)\right) &> \phi\left(\mathbf{x}^T(T')\right) \quad \text{for all } t \in [0,T'), \\ \text{and } \ \phi\left(\mathbf{x}^T(t)\right) &= \phi\left(\mathbf{x}^T(T')\right) \quad \text{for all } t \in [T',T]. \end{aligned} \tag{4.43}$$

We proceed by proving this claim. As $\mathbf{x}^T \in C^0([0,T];\mathbb{R}^{d_x})$ and $\phi \in C^0(\mathbb{R}^{d_x})$, the map $t \mapsto \phi(\mathbf{x}^T(t))$ attains its minimum in the interval $[0,T]$. Let us define[8]

$$T' := \min\left\{ t \in [0,T] : \ \phi(\mathbf{x}^T(t)) = \min_{s\in[0,T]} \phi\left(\mathbf{x}^T(s)\right) \right\}.$$

By definition of T', the first part in (4.43) immediately follows. For the second part, consider $u^\dagger = \left[w^\dagger, b^\dagger\right]^\top$ defined by

$$u^\dagger(t) := \begin{cases} u^T(t) & \text{for } t \in (0,T') \\ 0 & \text{for } t \in (T',T). \end{cases}$$

In view of (3.3), we can easily see that the state $\mathbf{x}^\dagger$ associated to the above-defined control $u^\dagger$ is precisely

$$\mathbf{x}^\dagger(t) = \begin{cases} \mathbf{x}^T(t) & \text{for } t \in [0,T'] \\ \mathbf{x}^T(T') & \text{for } t \in [T',T]. \end{cases}$$

Now, since by definition of T', one has $\phi\left(\mathbf{x}^T(T')\right) \le \phi\left(\mathbf{x}^T(t)\right)$ for all $t \in [0,T]$, we deduce that $\mathcal{J}_T\left(u^\dagger\right) \le \mathcal{J}_T\left(u^T\right)$, which in particular, by the optimality of u^T, implies that $\left\|u^T(t)\right\| = 0$ for a.e. $t \in (T',T)$, and consequently $\mathbf{x}^T(t) = \mathbf{x}^T(T')$ for all $t \in [T',T]$. This completes the proof of Part 1.

Part 2: We shall now show that the optimal control parameters $u^T = \left[w^T, b^T\right]^\top$ satisfy

$$\begin{aligned} \left\|u^T(t)\right\| &= M \quad \text{for a.e. } t \in (0,T'), \\ \text{and } \ \left\|u^T(t)\right\| &= 0 \quad \text{for a.e. } t \in (T',T). \end{aligned} \tag{4.44}$$

The fact that $\left\|u^T(t)\right\| = 0$ for a.e. $t \in (T',T)$ follows from Part 1. In order to prove the first part in (4.44), we will argue by contradiction, employing Lemma 4.4. Let us thus suppose that there exists $0 < \omega < 1$ such that, the set

$$E_\omega := \left\{ t \in (0,T') : \ \left\|u^T(t)\right\| \le (1-\omega)M \right\}$$

has positive Lebesgue measure, i.e. $|E_\omega| > 0$. Then by the continuity of the Lebesgue measure, there exists a sufficiently small $\delta > 0$ such that the set

$$E'_\omega := \left\{ t \in (0,T') : \ \left\|u^T(t)\right\| \le (1-\omega)M \right\} \cap (0, T' - \delta)$$

[8]We see that T' is well defined by arguing as in the proof of Corollary 4.1 just above.

also has positive Lebesgue measure, i.e. $|E'_\omega| := \lambda > 0$. By definition of T', there exists $\gamma > 0$ such that

$$\phi(\mathbf{x}^T(t)) - \gamma \geq \phi_{\min} := \min_{s\in[0,T]} \phi(\mathbf{x}^T(s)), \qquad \text{for all } t \in (0, T' - \delta].$$

By Lebesgue measure theory, for all $\varepsilon > 0$, there exists a countable collection of disjoint nonempty intervals $\{(t_i, t'_i)\}_{i=1}^{+\infty} \subset (0, T' - \delta)$ such that

$$\left|\bigcup_{i=1}^{+\infty}(t_i, t'_i) \setminus E'_\omega\right| < \varepsilon \quad \text{and} \quad \left|E'_\omega \setminus \bigcup_{i=1}^{+\infty}(t_i, t'_i)\right| < \varepsilon. \tag{4.45}$$

This implies in particular that for all $\varepsilon > 0$,

$$\left|\bigcup_{i=1}^{+\infty}(t_i, t'_i)\right| > \lambda - \varepsilon. \tag{4.46}$$

Let $\varepsilon > 0$ be fixed and to be chosen later, and for the corresponding collection of intervals $\{(t_i, t'_i)\}_{i=1}^{+\infty}$ satisfying (4.45), and for $n \geq 1$ to be chosen later, set

$$u^n_\varepsilon(t) := \begin{cases} u^T(t) & \text{for } t \in (0,T) \setminus \left(\displaystyle\bigcup_{i=1}^{n}(t_i, t'_i) \setminus E_\omega\right) \\ \\ 0 & \text{for } t \in \displaystyle\bigcup_{i=1}^{n}(t_i, t'_i) \setminus E_\omega. \end{cases}$$

The above control is admissible, namely

$$\|u^n_\varepsilon(t)\| \leq M, \quad \text{for a.e. } t \in (0,T).$$

Let $\mathbf{x}_\varepsilon$ be the solution to (3.3) associated to u^n_ε, with initial datum $\mathbf{x}_0$. By using the boundedness of the admissible controls in L^∞ and the Grönwall inequality, we have, for any $t \in [0, T']$,

$$\left\|\mathbf{x}^n_\varepsilon(t) - \mathbf{x}^T(t)\right\| \leq C(T') \int_0^{T'} \left\|u^n_\varepsilon(s) - u^T(s)\right\| \, \mathrm{d}s. \tag{4.47}$$

On the other hand, by (4.45), we have

$$\int_0^T \left\|u^n_\varepsilon(s) - u^T(s)\right\| \, \mathrm{d}s < M\varepsilon_n, \tag{4.48}$$

where $\{\varepsilon_n\}_{n=1}^{+\infty}$ is a sequence satisfying $\varepsilon_n \to \varepsilon$ as $n \to \infty$. Hence, using (4.47) and (4.48), together with the fact that $\mathbf{x}^n_\varepsilon$ and $\mathbf{x}^T$ are constant in the interval $[T', T]$, we have, for any $t \in [0, T]$

$$\left\|\mathbf{x}^T(t) - \mathbf{x}^n_\varepsilon(t)\right\| < C(T')\,\varepsilon_n,$$

whence, by the Lipschitz continuity of ϕ, we have

$$\left\|\phi(\mathbf{x}^T(t)) - \phi(\mathbf{x}^n_\varepsilon(t))\right\| \leq L_\phi C(T')\varepsilon_n, \qquad \text{for all } t \in [0, T], \tag{4.49}$$

which in particular implies

$$\phi(\mathbf{x}^n_\varepsilon(t)) - \phi_{\min} \geq \gamma - L_\phi C(T')\varepsilon_n, \qquad \text{for all } t \in [0, T' - \delta].$$

By taking ε small enough and n sufficiently big, we can ensure that $\gamma - L_\phi C(T')\varepsilon > 0$. Then, setting

$$\omega^\dagger := \min\{\omega, \gamma - L_\phi C(T')\varepsilon_n\},$$

we observe that that the control u_ε^n satisfies

$$\|u_\varepsilon^n(t)\| \leq (1-\omega^\dagger)M, \quad \text{for a.e. } t \in \bigcup_{i=1}^n (t_i, t_i'),$$

and

$$\phi(\mathbf{x}_\varepsilon^n(t)) - \phi_{\min} \geq \omega^\dagger, \quad \text{for all } t \in \bigcup_{i=1}^n (t_i, t_i').$$

We can now apply Lemma 4.4, which ensures the existence of an admissible control $\overline{u}_\varepsilon^n$ such that

$$\mathcal{J}_T(\overline{u}_\varepsilon^n) \leq \mathcal{J}_T(u_\varepsilon^n) - (\omega^\dagger)^2 \left| \bigcup_{i=1}^n (t_i,\, t_i') \right|. \tag{4.50}$$

On the other hand, as a consequence of (4.48) and (4.49) we have

$$\mathcal{J}_T(u_\varepsilon^n) \leq \mathcal{J}_T\left(u^T\right) + \left(\frac{\omega M}{2} + L_\phi C(T')T\right)\varepsilon_n,$$

which, together with (4.50) and (4.46), gives

$$\mathcal{J}_T(\overline{u}_\varepsilon^n) \leq \mathcal{J}_T(u^T) + C\,\varepsilon_n - (\omega^\dagger)^2(\lambda - \varepsilon_n).$$

By choosing $\varepsilon > 0$ sufficiently small and $n \geq 1$ sufficiently big, we obtain a contradiction with the optimality of u^T. Part 2 thus holds.

Part 3: We shall finally show that there exists $T_M > 0$ such that if $T > T_M$, then

$$T' \leq T_M \quad \text{and} \quad \phi(\mathbf{x}^T(T')) = 0.$$

To proceed, first note that by assumption (4.13) in the statement of the theorem, there exist parameters $u^\dagger = [w^\dagger, b^\dagger]^\top \in L^1(0, T_0; \mathbb{R}^{d_u})$ such that the associated trajectory solution to (3.3) satisfies $\phi(\mathbf{x}^*(T_0)) = 0$. For any $T \geq T_0$, let us denote by $u_T^\dagger$ the same control $u^\dagger$, extended by 0 in the interval (T_0, T). Observe that

$$\mathcal{J}_T(u_T^\dagger) = \mathcal{J}_{T_0}(u^\dagger) =: K_0, \quad \text{for all } T > T_0.$$

For any $T > T_0$, let u^T be a minimizer of $\mathcal{J}_T$. Using Parts 1 and 2, we have

$$\mathcal{J}_T(u^T) = M\,T' + \int_0^{T'} \phi(\mathbf{x}(t))\,\mathrm{d}t \leq \mathcal{J}_T(u_T^\dagger) = K_0. \tag{4.51}$$

This implies that $T' \leq \frac{K_0}{M}$, where K_0 and M are of course independent of T. It follows that, if $T > T_M := \max\left\{T_0, \frac{K_0}{M}\right\}$, then the optimal control vanishes for all $t \geq \frac{K_0}{M}$. Hence, we deduce that for all $T_1, T_2 > T_M$,

$$u^T \text{ minimizes } \mathcal{J}_{T_1} \quad \text{if and only if} \quad u^T \text{ minimizes } \mathcal{J}_{T_2}.$$

Now, let $\tilde{u}$ be a minimizer of $\mathcal{J}_{\tilde{T}}$ for some $\tilde{T} > T_M$. Then, it is also a minimizer of $\mathcal{J}_T$ for all $T > T_M$. Using (4.51), we obtain

$$K_0 \geq \mathcal{J}_T(\tilde{u}) \geq M\,T' + (T - T')\phi(\tilde{\mathbf{x}}(T')),$$

which in particular implies

$$K_0 \geq (T - T')\phi(\tilde{\mathbf{x}}(T')) \geq \left(T - \frac{K_0}{M}\right)\phi(\tilde{\mathbf{x}}(T')), \quad \text{for all } T > \tilde{T}.$$

Finally, letting $T \to +\infty$, we deduce that $\phi(\tilde{\mathbf{x}}(T')) = 0$. This concludes the proof. □

To conclude this section, we provide the statement and proof of Lemma 4.4 used in the proof above.

Lemma 4.4. *Under the hypotheses of Theorem 4.2, for any $\omega \in (0,1)$, let u_ω be a given control and $\mathbf{x}_\omega$ its corresponding state, such that there exists finite collection of disjoint intervals $\bigcup_{i=1}^{I}(t_i, t_i') \subset (0,T)$, wherein it holds*

$$\|u_\omega(t)\| \leq (1-\omega)M \quad \text{for a.e.} \quad t \in \bigcup_{i=1}^{I}(t_i, t_i'), \tag{4.52}$$

and

$$\phi(\mathbf{x}_\omega(t)) - \phi_{\min} \geq \omega, \quad \text{for all } t \in \bigcup_{i=1}^{I}(t_i, t_i'), \tag{4.53}$$

where $\phi_{\min} := \min_{t\in[0,T]} \phi(\mathbf{x}(t))$. Then, there exists a control $\overline{u}_\omega \in L^1(0,T;\mathbb{R}^{d_u})$ such that $\|\overline{u}_\omega(t)\| \leq M$ for a.e. $t \in (0,T)$ and

$$\overline{u}_\omega(t) = u_\omega(t), \quad \text{for all } (0,T) \setminus \left(\bigcup_{i=1}^{I}(t_i, t_i')\right)$$

and

$$\mathcal{J}_T(\overline{u}_\omega) \leq \mathcal{J}_T(u_\omega) - \omega^2 \sum_{i=1}^{I}(t_i' - t_i).$$

Proof. Consider the control

$$\overline{u}_\omega(t) := \begin{cases} u_\omega(t) & \text{for } \ t \in (0,T) \setminus \left(\bigcup_{i=1}^{I}(t_i, t_i')\right) \\ \dfrac{t_i' - t_i}{t_i'' - t_i} u_\omega\left((t-t_i)\dfrac{t_i' - t_i}{t_i'' - t_i} + t_i\right) & \text{for } \ t \in [t_i,\, t_i'') \\ 0 & \text{for } t \in [t_i'',\, t_i'), \end{cases}$$

where $t_i'' \in (t_i, t_i')$ is chosen so that $\dfrac{t_i' - t_i}{t_i'' - t_i}(1-\omega) = 1$. Observe that, as a consequence of (4.52), the control $\overline{u}_\omega$ still satisfies the constraint $\|\overline{u}_\omega(t)\| \leq M$, for a.e. $t \in (0,T)$.

Using the scaling result in Lemma 3.1, one can check that the trajectory associated to the control $\overline{u}_\omega$ is given by

$$\overline{\mathbf{x}}_\omega(t) := \begin{cases} \mathbf{x}_\omega(t) & \text{for } \ t \in (0,T) \setminus \left(\bigcup_{i=1}^{I}(t_i, t_i')\right) \\ \mathbf{x}_\omega\left((t-t_i)\dfrac{t_i' - t_i}{t_i'' - t_i} + t_i\right) & \text{for } \ t \in [t_i,\, t_i'') \\ \mathbf{x}_\omega(t_i') & \text{for } \ t \in [t_i'',\, t_i'). \end{cases}$$

Let us now evaluate the functional $\mathcal{J}_T$ at the control $\overline{u}_\omega$. We start by computing the L^1-norm of $\overline{u}$:

$$\begin{aligned}\|\overline{u}\|_{L^1(0,T;\mathbb{R}^{d_u})} &= \int_{(0,T)\setminus\left(\bigcup_{i=1}^I (t_i,t_i')\right)} \|u_\omega(t)\|\,\mathrm{d}t \\ &\quad + \sum_{i=1}^I \frac{t_i'-t_i}{t_i''-t_i}\int_{t_i}^{t_i''}\left\|u_\alpha\left((t-t_i)\frac{t_i'-t_i}{t_i''-t_i}+t_i\right)\right\|\,\mathrm{d}t \\ &= \|u_\omega\|_{L^1(0,T;\mathbb{R}^{d_u})}, \end{aligned} \tag{4.54}$$

where we used the following chain of change of variables

$$s \mapsto (s-t_i)\frac{t_i'-t_i}{t_i''-t_i}+t_i \qquad \text{for } i \in \{1,2,\dots,I\}.$$

In view of the assumption (4.53), the same chain of change of variables can be used to estimate the tracking term:

$$\begin{aligned}\int_0^T (\phi(\overline{\mathbf{x}}(t)-\phi_{\min})\,\mathrm{d}t &= \int_{(0,T)\setminus\left(\bigcup_{i=1}^I (t_i,t_i')\right)} \left(\phi(\mathbf{x}^T(t))-\phi_{\min}\right)\,\mathrm{d}t \\ &\quad + \sum_{i=1}^I \underbrace{\frac{t_i''-t_i}{t_i'-t_i}}_{1-\omega}\int_{t_i}^{t_i'} (\phi(\mathbf{x}_\omega(s))-\phi_{\min})\,\mathrm{d}s \\ &\le \int_0^T \left(\phi(\mathbf{x}^T(t)-\phi_{\min}\right)\,\mathrm{d}t - \omega^2\sum_{i=1}^I (t_i'-t_i).\end{aligned}$$

By combining this inequality with (4.54), the conclusion follows. □

5. The zero training error regime

The majority our results stated in the preceding sections stipulate whether and how the neural network prediction approaches the zero training error regime ($\phi = 0$ with ϕ given in (3.4)) when the number of layers increases. It is thus of interest to also illuminate the properties of the control parameters which allow the neural network prediction to reach precisely a minimizer of the training error ϕ.

We retain our continuous-time, neural ODE perspective to the supervised learning problem, and, by means of a simple continuous-dependence argument, we first show the following illustrative result, which stipulates a lower bound for the cost of the weights w – key in the supervised learning problem – in terms of the way the dataset is "spread out".

Theorem 5.1. *Let $\varphi \in C^\infty(\mathbb{R}^d;\mathbb{R}^m)$ be as in (2.8), and let $T>0$. Assume that for some control parameters $u := [w,b]^\top$, the solution $\mathbf{x} = [\mathbf{x}_1,\dots,\mathbf{x}_N]^\top$ to either (3.3) or (3.2) satisfies*

$$\varphi(\mathbf{x}_i(T)) = \vec{y}_i \qquad \text{for all } i \in \{1,\dots,N\}.$$

Then we have

$$\|w\|_{L^1(0,T;\mathbb{R}^{d_u})} \geq L_\sigma \max_{\substack{(i,j)\in\{1,\dots,N\}^2 \\ i\neq j}} \inf_{\substack{\mathbf{x}_i^1\in\varphi^{-1}(\{\vec{y}_i\}) \\ \mathbf{x}_j^1\in\varphi^{-1}(\{\vec{y}_j\})}} \log\left(\frac{\left\|\mathbf{x}_i^1-\mathbf{x}_j^1\right\|}{\left\|\mathbf{x}_i^0-\mathbf{x}_j^0\right\|}\right) \tag{5.1}$$

and

$$\|w\|_{L^2(0,T;\mathbb{R}^{d_u})} \geq \frac{L_\sigma}{\sqrt{T}} \max_{\substack{(i,j)\in\{1,\dots,N\}^2 \\ i\neq j}} \inf_{\substack{\mathbf{x}_i^1\in\varphi^{-1}(\{\vec{y}_i\}) \\ \mathbf{x}_j^1\in\varphi^{-1}(\{\vec{y}_j\})}} \log\left(\frac{\left\|\mathbf{x}_i^1-\mathbf{x}_j^1\right\|}{\left\|\mathbf{x}_i^0-\mathbf{x}_j^0\right\|}\right), \tag{5.2}$$

where $L_\sigma > 0$ is the Lipschitz constant of σ.

Note that for most of the common activation functions, namely sigmoids and rectifiers, one has $L_\sigma = 1$.

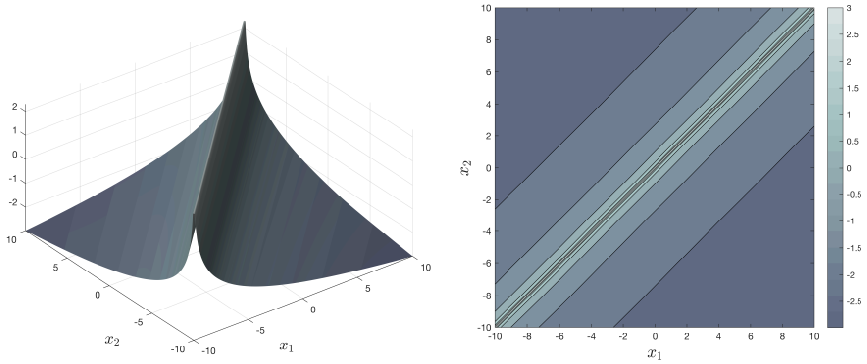

FIGURE 7. We display the lower bound of the cost of classifying a pair of 1D points $\vec{x}_1, \vec{x}_2$, having fixed the respective different labels. Observe that if $\vec{x}_1 = \vec{x}_2$, the classification is impossible. Whereas this picture only represents the 1D case, Theorem 5.1 stipulates the same effect in arbitrary dimensions.

Our arguments are somewhat similar in nature to the stability under adversarial perturbations estimates provided in [Haber and Ruthotto, 2017], but herein we insist on the interpretation in terms of the size of the parameters.

The above theorem assumes the existence of control parameters steering the endpoint $\mathbf{x}(T)$ of the neural ODE exactly to the minimizer of the training error ϕ, namely the endpoint $\mathbf{x}_i(T)$ of the trajectory associated to every datum to the preimage of every label $\vec{y}_i$ under the (possibly nonlinear) projector φ. To complete this section, we state the following local simultaneous controllability result, which namely contains an estimate on the control with respect to the distance of the target and the initial datum, which somewhat enhances the validity of the mild reachability assumptions we made in Theorem 3.1 and Theorem 4.1. While such an estimate is standard in the linear control setting (in both finite and infinite dimensions), it is not provided by sufficient controllability conditions for nonlinear systems such as the Chow-Rashevski theorem [Coron, 2007, Chapter 3, Section 3.3]. Our technique is constructive and differs from those in the works discussed just below.

Theorem 5.2. *Let $T > 0$ and assume that $N \leq d$. Let $\mathbf{x}^1 \in \mathbb{R}^{d_x}$ be given, and assume that the activation function $\sigma \in C^1(\mathbb{R})$ is such that*

$$\left\{\sigma\left(\mathbf{x}_1^1\right), \ldots, \sigma\left(\mathbf{x}_i^1\right), \ldots, \sigma\left(\mathbf{x}_N^1\right)\right\} \tag{5.3}$$

is a system of linearly independent vectors in $\mathbb{R}^d$. Then, there exist $r > 0$ and $C > 0$ such that for any initial datum $\mathbf{x}^0 \in \mathbb{R}^{d_x}$ satisfying $\|\mathbf{x}^0 - \mathbf{x}^1\| \leq r$, there exists a $u = [w, 0]^\top \in L^\infty(0, T; \mathbb{R}^{d_u})$ whose associated state $\mathbf{x}$, unique solution to

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{w}(t)\sigma(\mathbf{x}(t)) & \text{in } (0, T) \\ \mathbf{x}(0) = \mathbf{x}^0, \end{cases}$$

satisfies

$$\mathbf{x}(T) = \mathbf{x}^1,$$

and the estimate

$$\|u\|_{L^\infty(0,T;\mathbb{R}^{d_u})} \leq \frac{C}{T}\left\|\mathbf{x}^0 - \mathbf{x}^1\right\|,$$

holds for some $C > 0$ independent of T.

We postpone the proof until the end of the section.

Remark 5. The following observations are in order.

- For simplicity of presentation, we have not exhibited the bias parameter, namely the additive time-dependent control b. One can readlily check that, in the presence of this additional control, the assumption $N \leq d$ can be relaxed to $N \leq d + 1$.
- One could adapt the argument in the proof of Theorem 5.2 (given just below) to obtain a global result, assuming the existence of a continuous arc γ linking $\mathbf{x}^0$ and $\mathbf{x}^1$, such that

  $$\left\{\sigma\left(\gamma_1(s)\right), \ldots, \sigma\left(\gamma_i(s)\right), \ldots, \sigma\left(\gamma_N(s)\right)\right\}$$

  is a system of linearly independent vectors in $\mathbb{R}^d$ for any $s \in [0, 1]$. Problems arise however whenever this condition is not satisfied. In any case, in view of the uniqueness results for ODEs and Theorem 5.1, we have to assume that $\mathbf{x}_i^0 \neq \mathbf{x}_j^0$ and $\mathbf{x}_i^1 \neq \mathbf{x}_j^1$, for $i \neq j$.
- The case $N > d + 1$ may be treated by linearizing around a non-steady trajectory. Note that in [Coron, 2007, Section 3.1, Theorem 3.6], the controllability of the linearized problem around a general trajectory suffices.
- Given $\mathbf{x}^1 \in \mathbb{R}^{d_x}$, the proof of the above result also gives a rule for determining the activation σ, namely by checking that (5.3) is a system of linearly independent vectors in $\mathbb{R}^d$.

Discussion. In the discrete-time context of neural networks such as (2.1) or (2.3), the analog property has been addressed and well explored in the literature, and is commonly called *finite sample expressivity* [Zhang et al., 2016], with an additional interest in estimating the number of parameters – referred to as *the memorization capacity* – needed to manifest this property. For instance, in [Zhang et al., 2016], the authors use a ReLU network with two layers and $2N + d$ parameters to interpolate any labeling of size N in d dimensions. Their network inevitably has large width, but a network of depth $N_{\text{layers}} \geq 2$ can be conceived, in which each individual layer

has only $\mathcal{O}\left(\frac{N}{N_{\text{layers}}}\right)$ parameters In [Livni et al., 2014], a similar result is obtained with $\mathcal{O}(dN)$ parameters. For more recent and improved results, we refer the reader to [Yun et al., 2019, Kidger and Lyons, 2020, Montanari and Zhong, 2020].

The property of finite sample expressivity is closely related to the *universal approximation theory* (a common notion of neural network expressivity), which in general show which kinds of functions f – those from which the training dataset is sampled – can be approximated by means of neural network flow maps when either the depth or width grows. There is a substantial literature on this topic, see e.g. [Cybenko, 1989, Hornik et al., 1989, Pinkus, 1999, Burger and Neubauer, 2001, Daubechies et al., 2019, Bölcskei et al., 2019], to only name a few. Universal approximation theorems are density results, and in the simplest cases can be interpreted in terms of the elementary building blocks of measure theory such as the density of simple functions. It is possible to relate such results with the finite sample expressivity results using uniform convergence theorems. However, such uniform convergence bounds would require the dataset sample size to be polynomially large in the dimension of the input and exponential in the depth of the network, an unrealistic requirement in practice.

In the continous-time dynamical system context, the property of finite sample expressivity finds its analog in the complete or robust or simultaneous controllability, wherein one requires only 1 pair of controls to steer N trajectories of the same system to N prescribed targets – this is the property we show in Theorem 5.2. This definition is somewhat reminiscent to the concept of *simultaneous controllability*, which was perhaps mathematically instigated by Lions [Lions, 1988], and has been studied in a plethora of contexts for linear systems such as networks of strings [Dáger and Zuazua, 2006], see also [Lohéac and Zuazua, 2016] and the references therein. Motivated by the machine learning applications, there have been some works on such controllability results of neural ODEs, mostly relying on geometrical techniques such as Lie brackets and an application of the Chow-Rashevski theorem (see [Coron, 2007, Chapter 3, Section 3.3]), providing specific constraints on the activations function (see e.g. [Cuchiero et al., 2019, Tabuada and Gharesifard, 2020]). This is due to the specific driftless control affine nature of neural ODEs such as (3.3), and (3.2) with a positively homogeneous activation function.

5.1. **Proofs.** We finish this section with the proofs of Theorem 5.1 and Theorem 5.2.

Proof of Theorem 5.1. For simplicity of presentation but without any loss of generality, we will henceforth concentrate on system (3.3).

Let $u := [w, b]^\top \in \mathcal{U}_{T,\{\mathbf{x}^1\}}$ where $\mathcal{U}_{T,\{\mathbf{x}^1\}}$ is defined in (3.18). Then for any $t \in [0, T]$, $i \in \{1, \dots, N\}$ and $j \in \{1, \dots, N\}$, we have

$$\mathbf{x}_i(t) - \mathbf{x}_j(t) = \mathbf{x}_i^0 - \mathbf{x}_j^0 + \int_0^t w(\tau)\Big(\sigma(\mathbf{x}_i(\tau)) - \sigma(\mathbf{x}_j(\tau))\Big)\,\mathrm{d}\tau.$$

Using the Lipschitz character of σ, we get

$$\begin{aligned}\|\mathbf{x}_i(t)-\mathbf{x}_j(t)\| &\leq \left\|\mathbf{x}_i^0-\mathbf{x}_j^0\right\| + \int_0^t \|w(\tau)\| \left\|\sigma\left(\mathbf{x}_i(\tau)\right)-\sigma\left(\mathbf{x}_j(\tau)\right)\right\| \,\mathrm{d}\tau \\ &\leq \left\|\mathbf{x}_i^0-\mathbf{x}_j^0\right\| + L_\sigma \int_0^t \|w(\tau)\| \left\|\mathbf{x}_i(\tau)-\mathbf{x}_j(\tau)\right\| \,\mathrm{d}\tau.\end{aligned}$$

At this point, we apply the Grönwall inequality to the effect of

$$\|\mathbf{x}_i(t)-\mathbf{x}_j(t)\| \leq \exp\left(L_\sigma \int_0^t \|w(\tau)\| \,\mathrm{d}\tau\right) \left\|\mathbf{x}_i^0-\mathbf{x}_j^0\right\|.$$

We evaluate the above expression at final time $t=T$ and obtain

$$\left\|\mathbf{x}_i^1-\mathbf{x}_j^1\right\| \leq \exp\left(L_\sigma \int_0^T \|w(\tau)\| \,\mathrm{d}\tau\right) \left\|\mathbf{x}_i^0-\mathbf{x}_j^0\right\|,$$

for some $\mathbf{x}_i^1 \in \varphi^{-1}\left(\{\vec{y}_i\}\right)$ and $\mathbf{x}_j^1 \in \varphi^{-1}\left(\{\vec{y}_j\}\right)$, whence

$$\exp\left(L_\sigma \int_0^T \|w(\tau)\| \,\mathrm{d}\tau\right) \geq \frac{\left\|\mathbf{x}_i^1-\mathbf{x}_j^1\right\|}{\left\|\mathbf{x}_i^0-\mathbf{x}_j^0\right\|}.$$

Taking the log on both sides we obtain (5.1), whereas (5.2) follows by Cauchy–Schwarz. □

The following short functional analysis lemma will be of use in the Proof of Theorem 5.2.

Lemma 5.1. *Let $\mathcal{H}_1$ and $\mathcal{H}_2$ be two real Hilbert spaces. Let*

$$\mathcal{L} : \mathcal{H}_1 \longrightarrow \mathcal{H}_2$$

be a bounded and surjective linear operator. Then

$$\begin{aligned}\Gamma : \mathcal{H}_2 &\longrightarrow \mathcal{H}_1 \\ y &\longmapsto \underset{x\in\mathcal{L}^{-1}(\{y\})}{\arg\min} \|x\|_{\mathcal{H}_1}^2\end{aligned}$$

is linear and bounded.

Lemma 5.1 can be proved by using the open mapping theorem (see e.g. [Brezis, 2010, Theorem 2.6, pp. 35]).

Proof of Theorem 5.2. Inspired by the techniques in [Coron and Trélat, 2004] and the so-called "staircase" argument introduced in [Pighin and Zuazua, 2018] (see also [Ruiz-Balet and Zuazua, 2019]), we define the continuous arc

$$\begin{aligned}\gamma : [0,1] &\longrightarrow \mathbb{R}^{d_x} \\ s &\longmapsto (1-s)\mathbf{x}^0 + s\mathbf{x}^1.\end{aligned}$$

By assumption,

$$\left\{\sigma\left(\mathbf{x}_1^1\right), \ldots, \sigma\left(\mathbf{x}_i^1\right), \ldots, \sigma\left(\mathbf{x}_N^1\right)\right\}$$

is a linearly independent system of vectors in $\mathbb{R}^d$ for any $s \in [0,1]$. Thus, by using the continuity of γ, we see that there exists an $\eta > 0$, such that whenever $\|\mathbf{x}^1-\mathbf{x}^0\| \leq \eta$,

$$\left\{\sigma\left(\gamma_1(s)\right), \ldots, \sigma\left(\gamma_i(s)\right), \ldots, \sigma\left(\gamma_N(s)\right)\right\} \tag{5.4}$$

is also a system of linearly independent vectors in $\mathbb{R}^d$ for any $s \in [0,1]$. Following the framework of Lemma 5.1, for any $s \in [0,1]$, set

$$\begin{aligned} \mathcal{L}_s : \mathbb{R}^{d\times d} &\longrightarrow \mathbb{R}^{d_x} \\ w &\longmapsto w\sigma\left(\gamma\left(s\right)\right). \end{aligned}$$

By the linear independence of the system of vectors (5.4), $\mathcal{L}_s$ is surjective for any $s \in [0,1]$. Hence, using Lemma 5.1, we see that

$$\begin{aligned} \Gamma_s : \mathbb{R}^{d_x} &\longrightarrow \mathbb{R}^{d\times d} \\ y &\longmapsto \underset{w\in\mathcal{L}_s^{-1}(\{y\})}{\arg\min} \|w\|, \end{aligned}$$

is a linear and bounded operator for any $s \in [0,1]$, and, since (5.4) is independent and the arc γ is continuous, $\{\Gamma_s\}_{s\in[0,1]}$ is uniformly bounded in operator norm:

$$\|\Gamma_s\|_{\mathcal{L}\left(\mathbb{R}^{d_x};\mathbb{R}^{d\times d}\right)} \leq C \tag{5.5}$$

for some $C > 0$ independent of $T > 0$. Now, for $t \in [0,T]$, set

$$w(t) \coloneqq \Gamma_{s_t}\left(\frac{\mathbf{x}^1 - \mathbf{x}^0}{T}\right), \tag{5.6}$$

with $s_t \coloneqq \dfrac{t}{T}$. Note that for any $t \in [0,T]$, the vector $w(t) \in \mathbb{R}^{d\times d}$ solves the linear system of equations

$$w(t)\sigma\left(\mathbf{x}_i(t)\right) = \dot{\mathbf{x}}_i(t) \quad \text{ for } i \in \{1,\dots,N\},$$

where

$$\mathbf{x}(t) \coloneqq \gamma\left(\frac{t}{T}\right) = \left(1 - \frac{t}{T}\right)\mathbf{x}^0 + \frac{t}{T}\mathbf{x}^1.$$

Hence, $\mathbf{x}(t)$ solves

$$\begin{cases} \dot{\mathbf{x}}_i(t) = \mathbf{w}(t)\sigma(\mathbf{x}_i(t)) & \text{for } t \in (0,T) \\ \mathbf{x}_i(0) = \gamma(0) = \mathbf{x}_i^0 \\ \mathbf{x}_i\left(T\right) = \gamma(1) = \mathbf{x}_i^1, \end{cases}$$

for any $i \in \{1,\dots,N\}$. This thus demonstrates the existence of a control w steering the stacked dynamics from $\mathbf{x}^0$ to $\mathbf{x}^1$ in time T.

Let us now show that w satisfies the stated estimate. By the definition of w in (5.6) as well as (5.5), for any $t \in [0,T]$ we have

$$\|w(t)\| = \left\|\Gamma_t\left(\frac{\mathbf{x}^1 - \mathbf{x}^0}{T}\right)\right\| \leq \frac{C}{T}\left\|\mathbf{x}^1 - \mathbf{x}^0\right\|,$$

as desired. □

6. Continuous space-time neural networks

We now come back to the scheme (2.3) defining a residual neural network with $N_{\text{layers}} \geq 2$ layers (i.e. of depth N_{layers}). For simplicity, let us assume that g is parametrized as in (2.5) (what follows is analogous for other parametrizations), whence (2.3) writes as

$$\begin{cases} \mathbf{x}_i^{k+1} = \mathbf{x}_i^k + \sigma(w^k \mathbf{x}_i^k + b^k) & \text{for } k \in \{0, \dots, N_{\text{layers}} - 1\} \\ \mathbf{x}_i^0 = \vec{x}_i \end{cases} \tag{6.1}$$

for any $i \in \{1, \dots, N\}$. Note that the dimension (i.e. the *width*) of the weights $w^k \in \mathbb{R}^{d \times d}$, biases $b^k \in \mathbb{R}^d$ and states (features) $\mathbf{x}_i^k \in \mathbb{R}^d$, remains the same at each layer k.

Whilst such residual neural networks are widely used in practice and provide reliable results, in the discrete-time context, they do not take into account variations of the dimensions of the weights and states over layers. Such variations may arise when considering *convolutional* and/or *pooling* layers, which are ubiquitous in tasks in computer vision. In such tasks, it is moreover of interest to view the data itself as being continuum objects.

To be more specific, we note that in the simplest nonlinear context, a residual network with variable dimensions analog to (2.3) takes the form (see [He et al., 2016])

$$\begin{cases} \mathbf{z}_i^{k+1} = P^k \mathbf{z}_i^k + \sigma(w^k \mathbf{z}_i^k + b^k) & \text{for } k \in \{0, \dots, N_{\text{layers}} - 1\} \\ \mathbf{z}_i^0 = \vec{x}_i. \end{cases} \tag{6.2}$$

Here, contrary to (2.3), we have $w^k \in \mathbb{R}^{d_{k+1} \times d_k}$ and $b^k \in \mathbb{R}^{d_{k+1}}$, and thus $\mathbf{z}^k \in \mathbb{R}^{d_k}$ for $k \in \{0, \dots, N_{\text{layers}}\}$, where $\{d_k\}_{k=0}^{N_{\text{layers}}}$ are given positive integers, called widths of the layers k. One imposes $d_0 = d$, and $P^k \in \mathbb{R}^{d_{k+1} \times d_k}$ is a projection/embedding operator which serves to match dimensions. Much like in the fixed width case, we may also write the residual network when g is parametrized as in (2.6), which reads

$$\begin{cases} \mathbf{z}_i^{k+1} = P^k \mathbf{z}_i^k + w^k \sigma(\mathbf{z}_i^k) + b^k & \text{for } k \in \{0, \dots, N_{\text{layers}} - 1\} \\ \mathbf{z}_i^0 = \vec{x}_i. \end{cases} \tag{6.3}$$

6.1. The continuous space-time network.

It is not immediately obvious how one can see (6.2) or (6.3) as a numerical scheme for some continuous-time dynamical system in the flavor of (2.4). Nevertheless, this can be achieved by viewing the changing dimension over time-steps as an additional (spatial) variable, thus yielding an integro-differential equation in the continuum.

To be more precise, for any $i \in \{1, \dots, N\}$ we consider the non-local equation

$$\begin{cases} \partial_t \mathbf{z}_i(t,x) = \sigma\left(\displaystyle\int_\Omega w(t,x,\xi) \mathbf{z}_i(t,\xi) \, \mathrm{d}\xi + b(t,x) \right) & \text{for } (t,x) \in (0,T) \times \Omega \\ \mathbf{z}_i(0,x) = \mathbf{z}_i^{\text{in}}(x) & \text{for } x \in \Omega. \end{cases} \tag{6.4}$$

Here $\Omega \subset \mathbb{R}^{d_\Omega}$ is a bounded domain, where $d_\Omega \geq 1$. We emphasize that $\mathbf{z}_i(t,x) \in \mathbb{R}$ for $(t,x) \in (0,T) \times \Omega$, and similarly, $w(t,x,\xi) \in \mathbb{R}$ and $b(t,x) \in \mathbb{R}$ for $(x,\xi) \in \Omega \times \Omega$. The initial datum $\mathbf{z}_i^{\text{in}} \in C^0(\overline{\Omega})$ is such that there exist $\{x_j\}_{j=1}^d \subset \Omega$ such that

$\mathbf{z}_i^{\text{in}}(x_j) = (\vec{x}_i)_j$. Such a datum can always be found (e.g. by interpolation). A variant of the continuum model (6.4) is suggested in [E, 2017] albeit in a slightly different context, and several theoretical results are given in [Liu and Markowich, 2019], including the general well-posedness result stated in Lemma 6.1 below.

We distinguish two typical cases for choosing the shape of Ω as well as d_Ω.

1. **Variable-width ResNets.** If in the discretized level, we seek to simply obtain a variable-width residual network such as (6.2) (or even the standard ResNet analog (6.1)), it suffices to consider $\Omega = (0,1)$, thus $d_\Omega = 1$. We give more detail on possible possible discretizations in Section 6.3 and Remark 6.

2. **Convolutional Neural Networks.** The situation is slightly more delicate in the case of CNNs, which are typically (but not exclusively) used in computer vision. We provide a proposal covering the continuous-time analog of CNNs with partial generality.

   For simplicity, we will assume that the dataset $\{\vec{x}_i\}_{i=1}^N$ consists of N images: $\vec{x}_i \in \mathbb{R}^{d_1 \times d_2 \times d_{\text{ch}}}$ for any i; here d_1 (resp. d_2) denote the number of horizontal (resp. vertical) pixels in the image $\vec{x}_i$, whereas d_{ch} denotes the number of channels, i.e. the color format (e.g. $d_{\text{ch}} = 1$ for grayscale, $d_{\text{ch}} = 3$ for RGB).

   In this case, we consider $\Omega := \Omega_{\text{img}} \times (0,1)$, where $\Omega_{\text{img}} \subset \mathbb{R}^2$ is a rectangle. Thus $d_\Omega = 3$. Moreover, we assume that the weights w in (6.4) are compactly supported and of a specific "convolutional" form (as indicated in most works, this is more so a cross-correlation form), namely, for any i, the equation takes the form

$$\partial_t \mathbf{z}_i(t,x,\zeta) = \sigma\left(\int_0^1 \int_{\Omega_{\text{img}}} w(t, x+\xi, \omega)\mathbf{z}_i(t,\xi,\omega)\,\mathrm{d}\xi\,\mathrm{d}\omega + b(t,x,\zeta)\right)$$

   for $(t,x,\zeta) \in (0,T) \times \Omega_{\text{img}} \times (0,1)$. We note that the variable $x \in \Omega_{\text{img}}$ denotes a pixel, whereas $\zeta \in (0,1)$ is a continuous variable indicating, when discretized, the number of extracted features (namely the number of filters). The bias parameter b can be omitted in this case, if needed.

   One possible way to discretize the above continuous-time model and obtain a CNN as in [Fan et al., 2019] is to follow the arguments in Section 6.3, where one would use a time-dependent grid for discretizing with respect to the variable $\zeta \in (0,1)$ as well, as the number of filters commonly varies over layers in CNNs. By discretizing Ω_{img} with a "shrinking" or "expanding" time-dependent rectangular grid, some effects of padding or pooling (but not max-pooling a priori) may also be considered. However, a full CNN–applicable theory is out of the scope of this work.

   The mathematical theory explaining the structural properties of CNNs is well-established. In particular, [Mallat, 2012, Bruna and Mallat, 2013, Mallat, 2016] provide, via a concept of Lipschitz stability to the action of diffeomorphisms, a characterization of of invariance and stabilitiy properties of input images, shown by using the so-called scattering transform, based on Fourier and microlocal analysis techniques. They in particular define explicitly the weight kernels w by means of specific wavelets motivated by

the fact that CNNs are specifically designed to exploit the prior properties of image data, and thus no optimization is involved. This differs significantly from the commonly used CNNs however, which adapt filters to training data.

Remark 6. Observe that the continuous space-time model (6.4) (resp. (6.5)) is more general and englobes (2.4) – (2.5) (resp. (2.4) – (2.6)), where only the time variable is considered to be continuous. Indeed, fix d different points $\{x_1, \dots, x_d\} \in \Omega$, and let δ_{x_j} denote the Dirac mass centered at x_j. For any $i \in \{1, \dots, N\}$, we consider the initial datum

$$\mathbf{z}_i^{\text{in}}(x) := \sum_{j=1}^{d} (\vec{x}_i)_j \, \delta_{x_j}(x) \qquad \text{for} \quad x \in \Omega.$$

We write the weight w as

$$w(t, x, \zeta) := \sum_{j=1}^{d} \sum_{\ell=1}^{d} w_{j,\ell}(t) \delta_{x_j}(x) \delta_{x_\ell}(\zeta) \qquad \text{for} \quad (t, x, \zeta) \in (0, T) \times \Omega \times \Omega,$$

yielding the matrix $[w_{j,\ell}(t)]_{1 \le j,\ell \le d}$ of weights at time t, whereas the bias $b(t, x)$ is written as

$$b(t, x) := \sum_{j=1}^{d} b_j(t) \delta_{x_j}(x) \qquad \text{for} \quad (t, x) \in (0, T) \times \Omega,$$

yielding the vector $[b_j(t)]_{1 \le j \le d}$ of biases at time t. As $\mathbf{z}_i^{\text{in}}$, w and b are all linear combinations of Dirac masses, by plugging them in (6.4), we rewrite the integrals as sums, and setting, for any $i \in \{1, \dots, N\}$,

$$(\mathbf{z}_i)_j(t) := \int_\Omega \mathbf{z}_i(t, x) \; \mathrm{d}\delta_{x_j}(x)$$

for $j \in \{1, \dots, d\}$, we see that $(\mathbf{z}_i)_j$ solves

$$\begin{cases} (\dot{\mathbf{z}}_i)_j(t) = \sigma \left(\displaystyle\sum_{\ell=1}^{d} w_{j,\ell}(t) \, (\mathbf{z}_i)_\ell(t) + b_j(t) \right) & \text{for } t \in (0, T) \\ (\mathbf{z}_i)_j(0) = (\vec{x}_i)_j. \end{cases}$$

This is just the j–th equation of the (2.4) – (2.5) for $i \in \{1, \dots, N\}$.

We state the following existence and uniqueness of solutions results, which can be found in [Liu and Markowich, 2019]. The proof is based on an elementary Picard iteration, much like for the finite dimensional case.

Lemma 6.1. *Let $T > 0$ and $\mathbf{z}^{\text{in}} \in L^2(\Omega)$ be given. Let $w \in L^1(0, T; L^2(\Omega \times \Omega))$ and $b \in L^1(0, T; L^2(\Omega))$ be given. Assume that $\sigma \in \mathrm{Lip}(\mathbb{R})$. Then there exists a unique solution $\mathbf{z} \in C^0([0, T]; L^2(\Omega))$ to (6.4).*

Correspondingly for (6.3), for $i \in \{1, \dots, N\}$ we may consider

$$\begin{cases} \partial_t \mathbf{z}_i(t, x) = \displaystyle\int_0^1 w(t, x, \xi) \sigma(\mathbf{z}_i(t, \xi)) \, \mathrm{d}\xi + b(t, x) & \text{in } (0, T) \times \Omega \\ \mathbf{z}_i(0, x) = \mathbf{z}_i^{\text{in}}(x) & \text{in } \Omega. \end{cases} \tag{6.5}$$

All of the above discussions as well as Lemma 6.1 also apply for this system.

6.2. **The supervised learning problem.** Given a training dataset $\{\vec{x}_i, \vec{y}_i\}_{i=1}^N$ with $\vec{x}_i \in \mathbb{R}^d$ and $\vec{y}_i \in \mathbb{R}^m$ for any i, and a time horizon $T > 0$, just as in the finite dimensional context, we begin by writing the equation satisfied by the stacked vector of states $\mathbf{z} := [\mathbf{z}_1, \dots, \mathbf{z}_N]^\top$ corresponding to the stacked vector of data $\mathbf{z}^{\text{in}} := [\mathbf{z}_1^{\text{in}}, \dots, \mathbf{z}_N^{\text{in}}]^\top$, where each $\mathbf{z}_i$ is the solution to either (6.4) or (6.5) corresponding to the datum $\mathbf{z}_i^{\text{in}}$, and control parameters $[w, b]^\top$ which are the same for all i.

The stacked continuous space-time neural networks we consider are thus either

$$\begin{cases} \partial_t \mathbf{z}(t,x) = \sigma\left(\displaystyle\int_\Omega \mathbf{w}(t,x,\xi)\mathbf{z}(t,\xi)\,\mathrm{d}\xi + \mathbf{b}(t,x)\right) & \text{in } (0,T)\times\Omega \\ \mathbf{z}(0,x) = \mathbf{z}^{\text{in}}(x) & \text{in } \Omega \end{cases} \tag{6.6}$$

or

$$\begin{cases} \partial_t \mathbf{z}(t,x) = \displaystyle\int_\Omega \mathbf{w}(t,x,\xi)\sigma(\mathbf{z}(t,\xi))\,\mathrm{d}\xi + \mathbf{b}(t,x) & \text{in } (0,T)\times\Omega \\ \mathbf{z}(0,x) = \mathbf{z}^{\text{in}}(x) & \text{in } \Omega. \end{cases} \tag{6.7}$$

Just as in the finite-dimensional case, the key point is to note how the controls $[w(t,x,\xi), b(t,x)]^\top$ for $(t,x,\xi) \in (0,T)\times\Omega\times\Omega$ enter the systems:

$$\mathbf{w}(t,x,\xi) := \begin{bmatrix} w(t,x,\xi) & & \\ & \ddots & \\ & & w(t,x,\xi) \end{bmatrix} \in \mathbb{R}^{N\times N}, \quad \mathbf{b}(t,x) := \begin{bmatrix} b(t,x) \\ \vdots \\ b(t,x) \end{bmatrix} \in \mathbb{R}^N. \tag{6.8}$$

As before, we first consider the consider the regularized empirical risk minimization problem

$$\inf_{\substack{[w,b]^\top \in H^k(0,T;\mathcal{U}) \\ \text{subject to (6.6) (resp. (6.7))}}} \phi(\mathbf{z}(T,\cdot)) + \frac{\alpha}{2}\left\|[w,b]^\top\right\|^2_{H^k(0,T;\mathcal{U})}, \tag{6.9}$$

where $\alpha > 0$ is fixed, $k = 0$ for (6.7) and $k = 1$ for (6.6),

$$\mathcal{U} := L^2(\Omega\times\Omega) \times L^2(\Omega),$$

and we again define the training error as

$$\phi(\mathbf{z}(T,\cdot)) := \frac{1}{N}\sum_{i=1}^N \text{loss}\left(\varphi(\mathbf{z}_i(T,\cdot), \vec{y}_i\right),$$

with $\text{loss}(\cdot,\cdot) \in \mathcal{L}(L^2(\Omega)\times L^2(\Omega); \mathbb{R}_+)$, and we recall the definition of φ in (3.4). We note that the optimization problem (6.9) admits a solution – the argument follows the same lines as the proof of Proposition 2.1.

Before proceeding, we recall Definition 3.1, this time in the infinite dimensional setting.

Definition 6.1 (Reachable set)**.** For any $\mathbf{x}^0 \in L^2(\Omega)^N$ and any $T_0 > 0$, we define the *reachable set* from the initial datum $\mathbf{x}^0$ in time T_0 as

$$\mathcal{R}_{T_0}(\mathbf{x}^0) := \left\{\mathbf{x}^1 \in L^2(\Omega)^N \ : \ \exists u := [w,b]^\top \in H^k(0,T_0;\mathcal{U}) \text{ such that } \mathbf{x}(T_0,\cdot) = \mathbf{x}^1(\cdot) \text{ in } \Omega\right\},$$

where $\mathbf{x} \in C^0([0,T_0]; L^2(\Omega)^N)$ is the solution to (6.7) (resp. (6.6)), with $[\mathbf{w}, \mathbf{b}]$ as in (6.8), and $k = 0$ in the case of (3.3) (resp. $k = 1$ in the case of (3.2)).

In view of the rather generic nature of the proof to Theorem 3.1 in the finite-dimensional case, one may in fact roughly repeat the exact same proofs at most points, replacing throughout the finite dimensional euclidean spaces $\mathbb{R}^{d_x}$ and $\mathbb{R}^{d_u}$, by $L^2(\Omega)^N$ and $\mathcal{U}$ respectively. Whence, we state the infinite-dimensional partial analog to Theorem 3.1.

Theorem 6.1. *Let $\mathbf{z}^{\text{in}} \in \left(C^0(\overline{\Omega})\right)^N$ be such that $\mathbf{z}_i^{\text{in}}(x_j) = (\vec{x}_i)_j$, and assume that*

$$\mathcal{R}_{T_0}(\mathbf{z}^{\text{in}}) \cap \underset{L^2(\Omega)^N}{\arg\min}(\phi) \neq \varnothing$$

for some time $T_0 > 0$. Assume moreover that $\phi \in \mathcal{L}(L^2(\Omega)^N; \mathbb{R}_+)$ is convex. For any $T > 0$, let $\mathbf{z}^T \in C^0([0,T]; L^2(\Omega)^N)$ be the unique solution to (6.6) (resp. (6.7) with σ positively homogeneous of degree 1), associated to control parameters $u^T := [w^T, b^T]^\top \in H^k(0,T;\mathcal{U})$ solving the minimization problem (6.9), where $k = 0$ in the case of (6.7) and $k = 1$ in the case of (6.6).

Then, there exists a sequence $\{T_n\}_{n=1}^{+\infty}$, with $T_n > 0$ and $T_n \to +\infty$ as $n \to +\infty$, and $\mathbf{z}^\dagger \in \underset{L^2(\Omega)^N}{\arg\min}(\phi)$ such that

$$\phi\left(\mathbf{z}^{T_n}(T_n)\right) \longrightarrow \min_{L^2(\Omega)^N} \phi$$

and

$$\mathbf{z}^{T_n}(T_n) \rightharpoonup \mathbf{z}^\dagger \qquad \text{weakly in} \quad L^2(\Omega)^N$$

as $n \to +\infty$.

Adopting the notation from above, we similarly propose the supervised learning problem with a tracking term:

$$\inf_{\substack{[w,b]^\top \in H^k(0,T;\mathcal{U}) \\ \text{subject to (6.6) (resp. (6.7))}}} \phi(\mathbf{z}(T,\cdot)) + \frac{\alpha}{2}\left\|[w,b]^\top\right\|^2_{H^k(0,T;\mathcal{U})}\,\mathrm{d}t + \frac{\beta}{2}\int_0^T \|\mathbf{z}(t,\cdot) - \mathbf{z}_d(\cdot)\|^2_{L^2(\Omega)^N}\,\mathrm{d}t, \tag{6.10}$$

where $\alpha > 0$, $\beta > 0$ and $\mathbf{z}_d \in L^2(\Omega)^N$ are given. We also redefine the minimal cost as per Definition 3.2: given $T > 0$, $\mathbf{z}^0, \mathbf{z}^1 \in L^2(\Omega)^N$, we set

$$\kappa_T(\mathbf{z}^0, \mathbf{z}^1) := \inf_{\substack{[w,b]^\top \in H^k(0,T;\mathcal{U}) \\ \text{subject to (6.6) (resp. (6.7))} \\ \text{and} \\ \mathbf{z}(0)=\mathbf{z}^0, \mathbf{z}(T)=\mathbf{z}^1}} \left\|[w,b]^\top\right\|^2_{H^k(0,T;\mathcal{U})}.$$

As expected, the analog turnpike result holds for (6.10).

Theorem 6.2. *Let $\mathbf{z}^{\text{in}} \in C^0(\overline{\Omega})^N$, and let $\mathbf{z}_d \in \mathcal{R}_{T_0}(\mathbf{z}^{\text{in}})$ for some $T > 0$ be given. Assume that there exist $L_N > 0$ and $r > 0$ such that*

$$\kappa_{T_0}(\mathbf{x}, \mathbf{z}_d) \leq L_N^2 \|\mathbf{x} - \mathbf{z}_d\|^2_{L^2(\Omega)^N} \quad \text{and} \quad \kappa_{T_0}(\mathbf{z}_d, \mathbf{x}) \leq L_N^2 \|\mathbf{x} - \mathbf{z}_d\|^2_{L^2(\Omega)^N}$$

for all $\mathbf{x} \in \{\mathbf{x} \in L^2(\Omega)^N : \|\mathbf{x} - \mathbf{z}_d\|_{L^2(\Omega)^N} \leq r\}$. Let $T \geq 2T_0$ be fixed, and let $\mathbf{z}^T \in C^0([0,T]; L^2(\Omega)^N)$ be the unique solution to (6.7) (resp. (6.6)) associated to control parameters $u^T := \left[w^T, b^T\right]^\top \in H^k(0,T;\mathcal{U})$ solving the minimization problem (6.10), where $k = 0$ in the case of (6.7) and $k = 1$ in the case of (6.6).

Then there exist $C = C(\alpha, \beta, \mathbf{z}_d, \mathbf{z}^{\text{in}}, N) > 0$, $\gamma = \gamma(\alpha, \beta, \mathbf{z}_d, \mathbf{z}^{\text{in}}, N) > 0$ *and* $\mu = \mu(\alpha, \beta, N) > 0$ *such that*

$$\left\| \left[w^T, b^T \right]^\top \right\|_{H^k(0,T;\mathcal{U})} \leq C \left(\|\mathbf{z}_d - \mathbf{z}^{\text{in}}\|_{L^2(\Omega)^N} + \sqrt{\phi(\mathbf{z}_d)} \right)$$

and

$$\left\| \mathbf{z}^T(t, \cdot) - \mathbf{z}_d(\cdot) \right\|_{L^2(\Omega)^N} \leq \gamma \left(e^{-\mu t} + e^{-\mu(T-t)} \right)$$

hold for all $t \in [0, T]$.

6.3. **From continuous to discrete.** The passage from (6.4) (resp. (6.5)) to a discrete-time scheme such as (6.2) (resp. (6.3)) is not immediately obvious, and to our knowledge has not been explained in the literature. To proceed, it is important to observe the inherent link between the layer k and the width d_k in (6.2). This motivates discretizing (6.4) in the spatial variable $x \in (0, 1)$ by using a *time-dependent grid*, which has a different number of nodes d_k at each time-step. We give more detail on this in what follows.

Let us demonstrate that (6.4) which reads (without loss of generality, we omit the dependence on i for clarity)

$$\begin{cases} \partial_t \mathbf{z}(t, x) = \sigma \left(\displaystyle\int_0^1 w(t, x, \xi) \mathbf{z}(t, \xi) \, \mathrm{d}\xi + b(t, x) \right) & \text{in } (0, T) \times (0, 1) \\ \mathbf{z}(0, x) = \mathbf{z}^{\text{in}}(x) & \text{in } (0, 1), \end{cases}$$

where $\mathbf{z}^{\text{in}}$ is such that $\mathbf{z}^{\text{in}}(x_j) = \vec{x}_{,j}$ for some $\{x_j\}_{j=1}^d \subset [0, 1]$, can be discretized to read exactly as

$$\begin{cases} \mathbf{z}^{k+1} = P^k \mathbf{z}^k + \sigma \left(w^k \mathbf{z}^k + b^k \right) & \text{for } k \in \{0, \dots, N_{\text{layers}} - 1\} \\ \mathbf{z}^0 = \vec{x}. \end{cases} \tag{6.11}$$

Here $\mathbf{z}^k \in \mathbb{R}^{d_k}$, $w^k \in \mathbb{R}^{d_{k+1} \times d_k}$ and $b^k \in \mathbb{R}^{d_{k+1}}$, with $d_0 := d$ and $\{d_k\}_{k=1}^{N_{\text{layers}}}$ given positive integers, and $P^k \in \mathbb{R}^{d_{k+1} \times d_k}$.

The arguments below will clearly also apply for passing from (6.5) to (6.3).

Our demonstration below is purely for illustrative purposes, as, in view of the preceding theoretical and numerical results, we stipulate that an adaptive solver ought to perform better than an adaptation of an Euler scheme as (6.11).

Remark 7. The choice of the spatial interval $[0, 1]$ is completely arbitrary – one may of course consider any bounded interval of $\mathbb{R}$.

Let

$$\left\{ t^0, \dots, t^{N_{\text{layers}}} \right\}, \qquad \text{with } \; t^0 := 0 \text{ and } \; t^{N_{\text{layers}}} := T,$$

be a given, non-decreasing sequence of time-steps. For simplicity of presentation, let us assume that the time-steps are uniform, namely $t^k = k\Delta t$ with $\Delta t = \frac{T}{N_{\text{layers}}}$, but of course more general time-adaptive sequences can be considered.

For any $k \in \{0, \dots, N_{\text{layers}}\}$, let us assume that we are given a grid

$$\left\{ x_j \left(t^k \right) \right\}_{j=1}^{d_k} \subset [0, 1]$$

which is ordered and uniformly distributed. For simplicity of presentation, in our discussion we will assume that $x_1(t^k) = 0$ and $x_{d_k}(t^k) = 1$ for any k. However

by means of an elementary time-step-dependent dilation, this restriction may be removed. We note that, not only there might be no overlap of grid nodes over different time-steps, but moreover, the number of grid nodes changes at each time-step k.

We will seek for an appropriate discretization of

$$\partial_t \mathbf{z}(t^{k+1}, x_j(t^{k+1})) = \sigma\left(\int_0^1 w\left(t^{k+1}, x_j(t^{k+1}), \xi\right)\mathbf{z}(t^k,\xi)\ \mathrm{d}\xi + b\left(t^{k+1}, x_j(t^{k+1})\right)\right) \tag{6.12}$$

for $k \in \{0,\dots,N_{\text{layers}}-1\}$ and $j \in \{1,\dots,d_{k+1}\}$. Hence, in view of the preceding discussion, some kind of interpolation may needed to justify a backward Euler discretization of the time derivative $\partial_t \mathbf{z}$ appearing in (6.12) at the grid nodes.

For any given $k \in \{0,\dots,N_{\text{layers}}-1\}$ and $j \in \{1,\dots,d_k\}$, we shall henceforth denote

$$x_j^k := x_j(t^k), \qquad \mathbf{z}_j^k := \mathbf{z}(t^k, x_j^k).$$

Following through the above discussion, the main issue in writing down a forward difference discretization to $\partial_t \mathbf{z}(t^{k+1}, x_j(t^{k+1}))$ appears whenever for a given k one has $d_k \neq d_{k+1}$, as it is a priori not possible to make sense of the expression $\mathbf{z}(t^{k+1}, x_j(t^{k+1})) - \mathbf{z}(t^k, x_j(t^k))$ for $j \neq 1$. Indeed, all $\iota \in \{2,\dots,d_k\}$ are such that $x_\iota(t^k) \notin \{x_j(t^{k+1})\}_{j=1}^{d_{k+1}}$, due to the uniformity of the grid.

Let us give an elementary argument for addressing this issue. Given k and given any $j \in \{1,\dots,d_{k+1}\}$, there clearly exists $\iota \in \{2,\dots,d_k\}$ such that $x_j^{k+1} \in [x_{\iota-1}^k, x_\iota^k]$. For such indices, we may thus define the linear interpolant

$$\widehat{\mathbf{z}}_j^k := \mathbf{z}_\iota^k + \frac{\mathbf{z}_\iota^k - \mathbf{z}_{\iota-1}^k}{x_\iota^k - x_{\iota-1}^k}\left(x_j^{k+1} - x_\iota^k\right). \tag{6.13}$$

This is nothing but an approximation of the first order Taylor expansion of $\mathbf{z}(t^{k+1}, x_j(t^{k+1}))$ with respect to the second variable. Using this interpolant, we may consider the simple forward difference

$$\partial_t \mathbf{z}(t^{k+1}, x_j(t^{k+1})) \approx \frac{\mathbf{z}_j^{k+1} - \widehat{\mathbf{z}}_j^k}{\Delta t} \tag{6.14}$$

for any $k \in \{0,\dots,N_{\text{layers}}-1\}$ and any $j \in \{1,\dots,d_{k+1}\}$.

We may now use any Newton-Cotes formula to discretize the integral term in (6.12): for $j \in \{1,\dots,d_{k+1}\}$, we write

$$\int_0^1 w\left(t^{k+1}, x_j(t^{k+1}), \xi\right)\mathbf{z}(t^k,\xi)\,\mathrm{d}\xi \approx \sum_{\iota=1}^{d_k} \alpha_\iota w\left(t^{k+1}, x_j(t^{k+1}), x_\iota(t^k)\right)\mathbf{z}(t^k, x_\iota(t^k)). \tag{6.15}$$

Here the $\alpha_\iota > 0$ are the corresponding weights of the chosen Newton-Cotes formula.

Let us now define

$$\mathbf{z}^k := \begin{bmatrix} \mathbf{z}(t^k, x_1(t^k)) \\ \vdots \\ \mathbf{z}(t^k, x_{d_k}(t^k)) \end{bmatrix} \in \mathbb{R}^{d_k}, \qquad b^k := \begin{bmatrix} b(t^{k+1}, x_1(t^{k+1})) \\ \vdots \\ b(t^{k+1}, x_{d_{k+1}}(t^{k+1})) \end{bmatrix} \in \mathbb{R}^{d_{k+1}}$$

and

$$w^k := \left[\alpha_\iota w(t^{k+1}, x_j(t^{k+1}), x_\iota(t^k))\right]_{1\le j\le d_{k+1} 1\le \iota\le d_k} \in \mathbb{R}^{d_{k+1}\times d_k}.$$

The above definitions, as well as (6.14) and (6.15) applied to (6.12), lead us to (6.11), where Δt has been "omitted" as a factor of the nonlinearity. In view of (6.13), the operator $P^k \in \mathbb{R}^{d_{k+1}} \times \mathbb{R}^{d_k}$ takes the explicit

$$P^k = \sum_{j=1}^{d_{k+1}} \left(\left\{ 1 + \frac{x_j^{k+1} - x_{\iota(j)}^k}{x_{\iota(j)}^k - x_{\iota(j)-1}^k} \right\} \overline{e}_j e_{\iota(j)}^\top - \frac{x_j^{k+1} - x_{\iota(j)}^k}{x_{\iota(j)}^k - x_{\iota(j)-1}^k} \overline{e}_j e_{\iota(j)-1}^\top \right),$$

where $\iota(j) \in \{2, \dots, d_k\}$ is such that $x_j^{k+1} \in [x_{\iota(j)-1}^k, x_{\iota(j)}^k]$, while $\{\overline{e}_j\}_{j=1}^{d_{k+1}}$ and $\{e_j\}_{j=1}^{d_k}$ denote the canonical bases of $\mathbb{R}^{d_{k+1}}$ and $\mathbb{R}^{d_k}$ respectively. We notice that the matrix P^k only has 2 non-zero elements at every row $j \in \{1, \dots, d_{k+1}\}$.

Several remarks are in order.

Remark 8. We first note that if the number of spatial grid nodes is a fixed constant $d_k = d$, by the above arguments, we recover (again modulo time-step constants) the original fixed-width ResNet model (6.1). In fact, (2.4) with g as in (2.5) (resp. (2.6)) corresponds to a spatial discretization of (6.4) (resp. (6.5)) on a fixed grid. Hence, (6.4) and (6.5) may be viewed as the infinite width & depth versions of (2.4).

Remark 9. We have taken the most simple discretization schemes for reducing our continuous model, namely a forward Euler scheme and a linear interpolant in (6.13), followed by a simple Newton-Cotes quadrature formula. Just as in the fixed-width neural networks, it is to be expected that using an adaptive solver for the continuous problem, provided some rule for generating the spatial mesh, would lead to significantly better convergence and performance.

Remark 10 (Generating moving grids)**.** Whilst we have assumed a very simple given time-dependent grid, one may most certainly generate more sophisticated grids by means of a variety of available methods. We refer to [Budd et al., 2009] for more detail on the existing methods for generating moving grids, which have found extensive use in the discretization of PDEs manifesting *shock waves* and/or *free boundaries*.

7. Concluding remarks and outlook

In this work, we have addressed the behavior when the time horizon goes to infinity, of general but widely used learning problems for continuous-time neural networks (neural ODEs). We have, more generally, sought to give a rigorous theoretical framework for the treatment of deep supervised learning by means of control theoretical and numerical analysis techniques with a PDE flavor, which could lead to a more fundamental understanding of the former topic, and to the development of improved algorithms and methods.

- In the classical empirical risk minimization problem with a Tikhonov control regularization term, we concluded via Theorem 3.1 – Theorem 6.1 that in large time horizons, the obtained optimal (trained) parameters for neural ODEs are such that the corresponding trajectories reach zero training error, whilst doing so with the least oscillations possible, as the parameters retain minimal norm. In the associated discrete-time, residual neural network setting, this result indicates that adding more layers before training would guarantee the optimal trajectories approach the zero training error

regime, but do so without overfitting. This long time horizon property is independent of the optimization algorithm used to minimize the functional.

- To obtain quantitative estimates on the time horizon (and thus, number of layers) required to be ε–close to the zero training error regime, for a given tolerance $\varepsilon > 0$, we also considered a minimization problem wherein we added a tracking term which regularizes the state trajectories over the entire time horizon. Using novel nonlinear techniques, in Theorem 4.1 – Theorem 4.2 – Theorem 6.2 we established the turnpike property, and consequently Corollary 4.1, which roughly stipulates that the optimal neural ODE output is $\mathcal{O}\left(e^{-\mu T}\right)$–close to the zero training error regime.

  Heuristically, the turnpike property indicates an intrinsic notion of distance between the different layers of a neural network: it indicates a way to choose where to localize the different time-steps (i.e. layers), whence, the layers near $t = 0$ and $t = T$ carry, in some sense, more relevance than those in the middle. This is a priori not clear if one considers a discrete neural network such as (1.1).
- We discovered in Section 5 that the optimal parameters operating in the zero training error regime cannot be arbitrarily small – they namely strongly depend on how the input data is spread out, as points which are in a neighborhood but have different labels are significantly more difficult to separate than others.

7.1. **Outlook.** We present a non-exhaustive list of questions and topics, related to our work, which ought to be explored in prospective studies and would be complementary to our work.

7.1.1. *Machine learning.*

1. **Turnpike property for** (4.9) Due to the rather involved technical nature of the proof of Theorem 4.1, we have not proven a turnpike phenomenon of the form (4.10) in the case of the optimization problem for (4.9). A global result might a priori not be obvious due to the presence of the (possibly nonlinear) normalization layer φ (see [Pighin and Sakamoto, 2020] for a related study). In view of the encouraging numerical simulations however, we nonetheless believe that such a property should hold in this context as well, assuming similar reachability conditions.

2. **Generalization bounds and regularization.** To complement our analytical study on the long time horizon/large layer regime, it would be of interest to provide strong generalization error bounds in this context via commonly used metrics such as the VC dimension [Vapnik, 2013] or Rademacher complexity [Bartlett and Mendelson, 2002]. On a related note, complementary to the L^2 and H^1–regularization of the control parameters we considered in this work, it would be of interest to see if our results can be extended to other relevant settings, e.g. L^1–regularization for neural ODEs such (3.2), or whether the H^1–regularization can be weakened to, say, TV-regularization.

3. **Sophisticated algorithms and datasets.** Motivated by Theorem 3.1 and Theorem 4.1, we have set forth a couple of greedy pre-training algorithms

in Section 3.1 and Section 4.2 respectively. It would definitely be of interest to have a numerical implementation of these algorithms in the neural ODE context to empirically demonstrate their performance.

Furthermore, the continuous space-time neural networks of Section 6 may be used in a variety of different ways: one may choose to use adaptive ODE solvers once the spatial discretization has been performed (using, for instance, a Monte-Carlo method), thus yielding more degrees of freedom in the choice of a neural network. A detailed study of this topics, both from a theoretical and numerical perspective, should be conducted. To further our numerical study, tests on state of the art realistic datasets such as CIFAR-10 [Krizhevsky et al., 2009] are also envisaged.

4. **Improved solvers.** From the computational perspective, implementing a more sophisticated direct collocation scheme [Kelly, 2017] rather than a direct shooting method, could be of interest. Moreover, whereas neural ODEs are of relative difference to interacting particle models arising in collective behavior, reduced order model strategies such as Random Batch Methods [Jin et al., 2020] ought to be experimented with in this context.

5. **Long-time behavior in Reinforcement Learning.** An effort has been made in recent years in effectively hybridizing the fields of optimal control and reinforcement learning, see e.g. [Recht, 2019, Bertsekas, 2019]. Roughly speaking, whereas supervised learning may be viewed as open-loop control, reinforcement learning may be viewed as adaptive feedback control, albeit in a stochastic setting, requiring elements such as Markov decision processes. Turnpike, perhaps in expectation and taking stochastic disturbances into consideration, could be of significant relevance in this context.

6. **Neural networks with priors.** All of our results, as well as a concise mathematical theory, could be applicable to other non-standard neural network architectures such as *Hamiltonian* or *Lagrangian* neural networks, recently proposed in the computer science literature [Greydanus et al., 2019, Cranmer et al., 2020]. These architectures, somewhat like CNNs, take advantage of the specific nature of the considered data, and have been used to obtain impressive empirical results, for instance, in the context of regression tasks such as learning the solution to the d'Alembert wave equation $(\partial_t^2 - \Delta)u = 0$ from data. As the wave equation is a typical problem which manifests high frequency phenomena which are not easily taken care of using regression and model reduction techniques [Zuazua, 2005], a full mathematical understanding of the works cited above is of interest.

7.1.2. *Control theory.*

1. **Exact-controllability.** For both Theorem 3.1 and Theorem 4.1, the assumption that the underlying finite or infinite-dimensional dynamics are controllable could be rather useful. We do emphasize that, rather than controlling a single trajectory associated to one single initial datum by means of one single controls, in this machine learning context, we aim to control $N \gg 1$ trajectories – associated to N different initial data – of the same dynamical system by means of one single control $u = [w, b]^\top$.

In the context of (3.3), the system is of a particular control-affine form, for which several Lie bracket techniques (e.g. the Chow-Rashevski theorem) may perhaps be applied, complementary to our independent result Theorem 5.2, but a full global theory is lacking.

2. **A weaker notion of controllability.** From both the theoretical results and numerical experiments, one may notice that, in the particular context of binary classification, the main effort in the learning process consists in steering the blue and red trajectories to regions wherein they are separated by a hyperplane in the space wherein the dynamics evolves. This problem may in fact be addressed from a more fundamental and theoretical point of view.

   For the sake of presentation, for any $i \in \{1, \dots, N\}$ we consider

$$\begin{cases} \dot{\mathbf{x}}_i(t) = w(t)\sigma(\mathbf{x}_i(t)) + b(t) & \text{in } (0,T) \\ \mathbf{x}_i(0) = \vec{x}_i \in \mathbb{R}^d. \end{cases}$$

   In the context of binary classification, we take labels $\vec{y}_i \in \{-1, 1\}$ for any $i \in \{1, \dots, N\}$. The goal would thus consist in finding $[w, b]^\top \in L^2(0, T; \mathbb{R}^{d_u})$ and a vector $\mathbf{e} \in \mathbb{R}^d$ such that

$$\vec{y}_i \langle \mathbf{x}_i(T), \mathbf{e} \rangle > 0 \qquad \text{for all } \ i \in \{1, \dots, N\}.$$

   This is a much weaker notion than the one of simultaneous exact controllability; it is rather somewhat a *separation* condition, for which a Hahn-Banach type argument might be of use. Proving this separation property could thus be of interest in view of understanding the fundamental procedures behind the task of binary classification.

Acknowledgments. B.G. acknowledges Daniel Tenbrinck and Lukas Pflug (FAU Erlangen-Nürnberg) for valuable discussions on the foundations of neural networks and nonlocal equations respectively, and Emilien Dupont (Oxford) for helpful remarks regarding the numerics of neural ODEs.

Appendix A. Auxiliary results

A.1. **Existence of minimizers.** For the sake of completeness, and usage of the arguments in some of the other proofs, we sketch a proof of the existence of minimizers via the classical direct method.

Proof of Proposition 2.1. We shall concentrate solely on the case $k = 0$, as modulo an application of the Rellich-Kondrachov compactness theorem, the arguments are exactly the same in the case $k = 1$.

Let $\left\{[w_n, b_n]^\top\right\}_{n=1}^{+\infty} \subset L^2(0, T; \mathbb{R}^{d_u})$ be a minimizing sequence, namely a sequence satisfying

$$\lim_{n \to +\infty} \mathcal{J}_T(w_n, b_n) = \inf_{[w,b]^\top \in L^2(0,T;\mathbb{R}^{d_u})} \mathcal{J}_T(w, b).$$

For any $n \geq 1$, denote by $\mathbf{x}_n \in C^0([0, T]; \mathbb{R}^{d_x})$ the unique solution to (3.3) – (3.1) associated to $[w_n, b_n]^\top$ and the initial datum $\mathbf{x}^0$. Note that

$$\mathcal{J}_T(w, b) \geq \frac{\alpha}{2} \int_0^T \left\| [w(t), b(t)]^\top \right\|^2 \, \mathrm{d}t,$$

whence $\mathcal{J}_T$ is *coercive*, in the sense that $\mathcal{J}_T(u) \to +\infty$ when $\|u\|_{L^2} \to +\infty$. Since $\mathcal{J}_T$ is coercive, it follows that $\{[w_n, b_n]^\top\}_{n=1}^{+\infty}$ is bounded in $L^2(0,T;\mathbb{R}^{d_u})$. Whence, there exists $[w^\dagger, b^\dagger]^\top \in L^2(0,T;\mathbb{R}^{d_u})$ such that

$$\begin{aligned} w_n &\rightharpoonup w^\dagger \qquad \text{weakly in } \ L^2(0,T;\mathbb{R}^{d\times d}) \\ b_n &\rightharpoonup b^\dagger \qquad \text{weakly in } \ L^2(0,T;\mathbb{R}^{d}) \end{aligned}$$

along a subsequence as $n \to +\infty$. Of course, the same convergences thence hold for $\mathbf{w}_n := \mathrm{diag}_N(w_n)$ to $\mathbf{w}^\dagger := \mathrm{diag}_N(w^\dagger)$, as well as $\mathbf{b}_n := [b_n, \dots, b_n]^\top$ to $\mathbf{b}^\dagger := [b^\dagger, \dots, b^\dagger]^\top$. Let $\mathbf{x}^\dagger \in C^0([0,T];\mathbb{R}^{d_x})$ be the unique solution to (3.3) associated to $[w^\dagger, b^\dagger]^\top$ and the initial datum $\mathbf{x}^0$. Let us prove that

$$\mathbf{x}_n \longrightarrow \mathbf{x}^\dagger \qquad \text{strongly in } \ C^0([0,T];\mathbb{R}^{d_x}) \tag{A.1}$$

along the aforementioned subsequence as $n \to +\infty$. Take an arbitrary $t \in [0,T]$. Note that

$$\begin{aligned} \mathbf{x}_n(t) - \mathbf{x}^\dagger(t) &= \int_0^t \Big[\mathbf{w}_n(\tau)\sigma(\mathbf{x}_n(\tau)) + \mathbf{b}_n(\tau)\Big]\,\mathrm{d}\tau - \int_0^t \left[\mathbf{w}^\dagger(\tau)\sigma\left(\mathbf{x}^\dagger(\tau)\right) + \mathbf{b}^\dagger(\tau)\right]\,\mathrm{d}\tau \\ &= \int_0^t \Big[\mathbf{w}_n(\tau)\sigma(\mathbf{x}_n(\tau)) - \mathbf{w}_n(\tau)\sigma\left(\mathbf{x}^\dagger(\tau)\right)\Big]\,\mathrm{d}\tau \\ &\quad + \int_0^t \Big[\mathbf{w}_n(\tau)\sigma\left(\mathbf{x}^\dagger(\tau)\right) - \mathbf{w}^\dagger(\tau)\sigma\left(\mathbf{x}^\dagger(\tau)\right)\Big]\,\mathrm{d}\tau \\ &\quad + \int_0^t \Big[\mathbf{b}_n(\tau) - \mathbf{b}^\dagger(\tau)\Big]\,\mathrm{d}\tau. \end{aligned}$$

Hence, using the fact that σ is globally Lipschitz with constant $L_\sigma > 0$,

$$\begin{aligned} \left\|\mathbf{x}_n(t) - \mathbf{x}^\dagger(t)\right\| &\le \int_0^t \|\mathbf{w}_n(\tau)\| \left\|\sigma\Big(\mathbf{x}_n(\tau)\Big) - \sigma\left(\mathbf{x}^\dagger(\tau)\right)\right\|\,\mathrm{d}\tau \\ &\quad + \left\|\int_0^t \sigma\left(\mathbf{x}^\dagger(\tau)\right)\left[\mathbf{w}_n(\tau) - \mathbf{w}^\dagger(\tau)\right]\,\mathrm{d}\tau\right\| \\ &\quad + \left\|\int_0^t \Big[\mathbf{b}_n(\tau) - \mathbf{b}^\dagger(\tau)\Big]\,\mathrm{d}\tau\right\| \\ &\le L_\sigma \int_0^t \|\mathbf{w}_n(\tau)\| \left\|\mathbf{x}_n(\tau) - \mathbf{x}^\dagger(\tau)\right\|\,\mathrm{d}\tau + c_n, \end{aligned}$$

with

$$c_n := \left\|\int_0^t \sigma\left(\mathbf{x}^\dagger(\tau)\right)\left[\mathbf{w}_n(\tau) - \mathbf{w}^\dagger(\tau)\right]\,\mathrm{d}\tau\right\| + \left\|\int_0^t \Big[\mathbf{b}_n(\tau) - \mathbf{b}^\dagger(\tau)\Big]\,\mathrm{d}\tau\right\|.$$

Using Grönwall's inequality, Cauchy-Schwarz, and the boundedness of the L^2–norm of $\{\mathbf{w}_n\}_{n=1}^{+\infty}$ by some constant $M > 0$ independent of t, we thence obtain

$$\begin{aligned} \left\|\mathbf{x}_n(t) - \mathbf{x}^\dagger(t)\right\| &\le c_n \exp\left(L_\sigma \int_0^t \|\mathbf{w}_n(\tau)\|\,\mathrm{d}\tau\right) \\ &\le c_n \exp\left(L_\sigma \sqrt{T}\,\|\mathbf{w}_n\|_{L^2(0,T;\mathbb{R}^{d\times d\times N})}\right) \\ &\le c_n \exp\left(L_\sigma \sqrt{T}\,M\right). \end{aligned}$$

As $c_n \to 0$ along any subsequence as $n \to +\infty$ by virtue of the weak convergences of $\{\mathbf{w}_n\}_{n=1}^{+\infty}$ to $\mathbf{w}^\dagger$ and $\{\mathbf{b}_n\}_{n=1}^{+\infty}$ to $\mathbf{b}^\dagger$, we deduce (A.1). Now using the weak lower semicontinuity of the squared $L^2(0,T;\mathbb{R}^{d_u})$–norm, the continuity of ϕ, (A.1) and – if $\beta > 0$ – the Lebesgue dominated convergence theorem, we deduce

$$\begin{aligned}\inf_{[w,b]^\top \in L^2(0,T;\mathbb{R}^{d_u})} \mathcal{J}_T(w,b) &= \lim_{n\to+\infty} \mathcal{J}_T(w_n,b_n)\\ &\geq \liminf_{n\to+\infty}\ \mathcal{J}_T(w_n,b_n)\\ &\geq \mathcal{J}_T(w^\dagger,b^\dagger).\end{aligned}$$

Whence $[w^\dagger,b^\dagger]^\top$ is a minimizer. This concludes the proof. □

A.2. **Results on ODEs.**

Lemma A.1. *Let $T>0$ and $\mathbf{x}_d \in \mathbb{R}^{d_x}$ be given. For any $u := [w,b]^\top \in L^2(0,T;\mathbb{R}^{d_u})$ and $\mathbf{x}^0 \in \mathbb{R}^{d_x}$, let $\mathbf{x} \in C^0([0,T];\mathbb{R}^{d_x})$ be the solution to either (3.2) or (3.3). Then there exist $C_1 = C_1(\sigma,\mathbf{x}_d,N) > 0$ and $C_2 = C_2(\sigma,N)$ such that*

$$\|\mathbf{x}(t)-\mathbf{x}_d\| \leq C\left(\|\mathbf{x}^0-\mathbf{x}_d\| + \|u\|_{L^2(0,T;\mathbb{R}^{d_u})} + \|\mathbf{x}-\mathbf{x}_d\|_{L^2(0,T;\mathbb{R}^{d_x})}\right)$$

holds for all $t\in[0,T]$, where

$$C := C_1 \exp\left(C_2\|w\|_{L^2(0,T;\mathbb{R}^{d\times d})}\right).$$

Proof. We will first show that for any $t\in(1,T]$, there exists a $t^*\in(t-1,t]$ such that

$$\|\mathbf{x}(t^*)-\mathbf{x}_d\| \leq \|\mathbf{x}-\mathbf{x}_d\|_{L^2(0,T;\mathbb{R}^{d_x})}. \tag{A.2}$$

To this end, we argue by contradiction. Assume that

$$\|\mathbf{x}(t^*)-\mathbf{x}_d\| \ > \|\mathbf{x}-\mathbf{x}_d\|_{L^2(0,T;\mathbb{R}^{d_x})}$$

for all $t^*\in(t-1,t]$. Whence

$$\|\mathbf{x}-\mathbf{x}_d\|^2_{L^2(0,T;\mathbb{R}^{d_x})} = \int_0^T \|\mathbf{x}(t)-\mathbf{x}_d\|^2\,\mathrm{d}t \geq \int_{t-1}^t \|\mathbf{x}(\tau)-\mathbf{x}_d\|^2\,\mathrm{d}\tau > \|\mathbf{x}-\mathbf{x}_d\|^2_{L^2(0,T;\mathbb{R}^{d_x})},$$

which contradicts the hypothesis. Thus (A.2) holds.

Now, let $t\in[0,T]$. By integrating the equation satisfied by $\mathbf{x}$, we first see that

$$\mathbf{x}(t)-\mathbf{x}_d = \mathbf{x}^0-\mathbf{x}_d + \int_0^t \sigma(\mathbf{w}(\tau)\mathbf{x}(\tau)+\mathbf{b}(\tau))\,\mathrm{d}\tau,$$

in the case of (3.2), and

$$\mathbf{x}(t)-\mathbf{x}_d = \mathbf{x}^0-\mathbf{x}_d + \int_0^t (\mathbf{w}(\tau)\sigma(\mathbf{x}(\tau))+b(\tau))\,\mathrm{d}\tau,$$

in the case of (3.3). As $\sigma\in\mathrm{Lip}(\mathbb{R})$ and $\sigma(0)=0$, if $t\leq 1$ we clearly have

$$\|\mathbf{x}(t)-\mathbf{x}_d\| \leq C_0\left(\|\mathbf{x}^0-\mathbf{x}_d\| + \|u\|_{L^2(0,T;\mathbb{R}^{d_u})}\right)$$

for some $C_0 = C_0(\sigma) > 0$ by Cauchy-Schwarz.

On the other hand, when $t>1$, we know that there exists $t^*\in(t-1,t]$ such that (A.2) holds. Let us concentrate on estimating the case of (3.3), as the one of

(3.2) follows similar arguments. Now writing the definition of a solution $\mathbf{x}$ to (3.2) in $[t^*, t]$, namely

$$\mathbf{x}(t)-\mathbf{x}_d = \mathbf{x}(t^*)-\mathbf{x}_d+\int_{t^*}^t \mathbf{w}(\tau)\Big(\sigma(\mathbf{x}(\tau))-\sigma(\mathbf{x}_d)\Big)\,\mathrm{d}\tau+\sigma(\mathbf{x}_d)\int_{t^*}^t \mathbf{w}(\tau)\,\mathrm{d}\tau+\int_{t^*}^t \mathbf{b}(\tau)\,\mathrm{d}\tau,$$

we see that, by using the Lipschitz character of σ,

$$\begin{aligned}\|\mathbf{x}(t)-\mathbf{x}_d\| \ &\leq \|\mathbf{x}(t^*)-\mathbf{x}_d\| + L_\sigma\int_{t^*}^t \|\mathbf{w}(\tau)\|\|\mathbf{x}(\tau)-\mathbf{x}_d\|\,\mathrm{d}\tau\\ &\quad + \|\sigma(\mathbf{x}_d)\|\int_{t^*}^t \|\mathbf{w}(\tau)\|\,\mathrm{d}\tau + \int_{t^*}^t \|\mathbf{b}(\tau)\|\,\mathrm{d}\tau.\end{aligned}$$

Now combining Cauchy-Schwarz, the fact that $t-t^*\leq 1$, (A.2), and the Grönwall inequality, we obtain

$$\|\mathbf{x}(t)-\mathbf{x}_d\| \leq C_1\exp\left(L_\sigma\sqrt{\int_{t^*}^t \|\mathbf{w}(\tau)\|^2\,\mathrm{d}\tau}\right)\left(\|\mathbf{x}-\mathbf{x}_d\|_{L^2(0,T;\mathbb{R}^{d_x})}+\int_{t^*}^t \left\|[\mathbf{w}(\tau),\mathbf{b}(\tau)]^\top\right\|\,\mathrm{d}\tau\right),$$

for some $C_1 = C_1(\sigma, \mathbf{x}_d, N) > 0$, from which the desired statement readily follows. □

Lemma A.2. *Let $T > 0$. For any $u := [w, b]^\top \in L^2(0,T;\mathbb{R}^{d_u})$ and $\mathbf{x}^0 \in \mathbb{R}^d$, let $\mathbf{x} \in C^0([0,T];\mathbb{R}^{d_x})$ be the solution to either (3.2) or (3.3). Then there exist $C = C(N,\sigma) > 0$ such that*

$$\|\mathbf{x}(t)\| \leq C\left(\left\|\mathbf{x}^0\right\| + \sqrt{T}\,\|b\|_{L^2(0,T)}\right)\exp\left(\sqrt{T}\,\|w\|_{L^2(0,T)}\right)$$

holds for all $t \in [0,T]$.

Proof. The proof is again based on the Grönwall inequality.

Let $t \in [0,T]$. We first note that

$$\mathbf{x}(t) = \mathbf{x}^0 + \int_0^t \sigma(\mathbf{w}(\tau)\mathbf{x}(\tau)+\mathbf{b}(\tau))\,\mathrm{d}\tau,$$

in the case of (3.2), and

$$\mathbf{x}(t) = \mathbf{x}^0 + \int_0^t (\mathbf{w}(\tau)\sigma(\mathbf{x}(\tau))+b(\tau))\,\mathrm{d}\tau,$$

in the case of (3.3). As $\sigma \in \mathrm{Lip}(\mathbb{R})$ and $\sigma(0) = 0$, we have

$$\|\mathbf{x}(t)\| \leq C_0\left(\left\|\mathbf{x}^0\right\| + \int_0^T \|b(\tau)\|\,\mathrm{d}\tau\right) + C_0\int_0^t \|\mathbf{w}(\tau)\|\,\|\mathbf{x}(\tau)\|\,\mathrm{d}\tau$$

for some $C_0 = C_0(N,\sigma) > 0$.

Then, by Grönwall and Cauchy-Schwarz, for any $t \in [0,T]$,

$$\begin{aligned}\|\mathbf{x}(t)\| \ &\leq C\left(\left\|\mathbf{x}^0\right\| + \int_0^t \|b(\tau)\|\,\mathrm{d}\tau\right)\exp\left(\int_0^t \|w(\tau)\|\,\mathrm{d}\tau\right)\\ &\leq C\left(\left\|\mathbf{x}^0\right\| + \int_0^T \|b(\tau)\|\,\mathrm{d}\tau\right)\exp\left(\int_0^T \|w(\tau)\|\,\mathrm{d}\tau\right)\\ &\leq C\left(\left\|\mathbf{x}^0\right\| + \sqrt{T}\,\|b\|_{L^2(0,T;\mathbb{R}^d)}\right)\exp\left(\sqrt{T}\,\|w\|_{L^2(0,T;\mathbb{R}^{d\times d})}\right),\end{aligned}$$

for some $C = C(N, \sigma) > 0$, as desired. □

Appendix B. Numerical methods

In the displayed figures, we concentrated mainly on simple classification tasks, but of course, our theoretical results do not make any prior assumption on the task and apply to complex classification and regression tasks as well. All experiments were coded using `PyTorch` [Paszke et al., 2017]. Our code may be found at `https://github.com/borjanG/dynamical.systems`, and is adapted from the code presented in [Dupont et al., 2019]. Experiments were conducted on a personal MacBook Pro laptop (2.4 GHz Quad-Core Intel Core i5, 16GB RAM, Intel Iris Plus Graphics 1536 MB).

For visualizing state trajectories over time, we considered the concentric spheres dataset consisting of blue (labeled -1) and red (labeled $+1$) points $\{\vec{x}_i\}_{i=1}^N$, mainly set in $\mathbb{R}$ or $\mathbb{R}^2$, with a possible 1–dimensional augmentation of the data by zero-concatenation for technical purposes (see Remark 1), sampled from the function

$$f(x) = \begin{cases} -1 & \text{if } |x| \leq r_1 \\ +1 & \text{if } r_2 \leq |x| \leq r_3. \end{cases}$$

Here $r_1, r_2, r_3 > 0$ are all given.

Discretization: Direct method. To discretize the full continuous-time optimization problem, we use *direct shooting* (see [Trélat, 2005, Chapter 9]), which is a *first discretize then optimize* approach. We are given a grid $\{t^0, \dots, t^{N_{\text{layers}}}\}$ and use a simple explicit Euler ODE solver, with $t^0 = 0$ and $t^{N_{\text{layers}}} = T$. The direct shooting approach consists in simply optimizing over the values of the weights w and biases b at the nodes t^k, i.e.

$$w(t^k) = w^k, \qquad b(t^k) = b^k.$$

This leads to a nonlinear programming problem, which is commonly solved in machine learning via a stochastic gradient descent variant and backpropagation (see e.g. [Bottou et al., 2018]). We will use the Adam algorithm [Kingma and Ba, 2014], with learning rate equal to 10^{-3}.

We concentrated on the simplest case where the states $\mathbf{x}_i = \mathbf{x}_i(t)$ are given by the neural network

$$\begin{cases} \dot{\mathbf{x}}_i(t) = w(t) \tanh(\mathbf{x}_i(t)) + b(t) & \text{in } (0, T) \\ \mathbf{x}_i(0) = \vec{x}_i, \end{cases}$$

with possible dimension augmentation of the initial data $\mathbf{x}_i(0)$ (see Remark 1). We considered a parametrized projection $\varphi(x) := \tanh(\theta_1 x + \theta_2) \in \mathbb{R}$ for $x \in \mathbb{R}^d$ where $\theta_1 \in \mathbb{R}^{1\times d}$ and $\theta_2 \in \mathbb{R}$ are learned parameters. Regarding the loss function, we will generally consider the classical least squares $\text{loss}(x, y) := \frac{1}{2m} \sum_{i=1}^m |x_j - y_j|^2$. The weight decay $0 < \alpha \ll 1$ is specified in a case by case scenario. Generally, we train with $N = 3000$ training samples, and display only a batch of 128 subsamples. Finally, we made use of a simple trapezoidal quadrature rule for discretizing the $L^2(0, T; \mathbb{R}^{d_u})$–norm of the weight-bias pairs $[w(t), b(t)]^\top$ and the tracking term appearing in turnpike results.

References

[Albertini and Sontag, 1993a] Albertini, F. and Sontag, E. D. (1993a). Discrete-time transitivity and accessibility: analytic systems. *SIAM Journal on Control and Optimization*, 31(6):1599–1622.

[Albertini and Sontag, 1993b] Albertini, F. and Sontag, E. D. (1993b). For neural networks, function determines form. *Neural Networks*, 6(7):975–990.

[Albertini et al., 1993] Albertini, F., Sontag, E. D., and Maillot, V. (1993). Uniqueness of weights for neural networks. *Artificial Neural Networks for Speech and Vision*, pages 115–125.

[Anderson and Kokotovic, 1987] Anderson, B. D. and Kokotovic, P. V. (1987). Optimal control problems over large time intervals. *Automatica*, 23(3):355–363.

[Avelin and Nyström, 2020] Avelin, B. and Nyström, K. (2020). Neural ODEs as the deep limit of ResNets with constant weights. *Analysis and Applications*, pages 1–41.

[Barron et al., 2008] Barron, A. R., Cohen, A., Dahmen, W., and DeVore, R. A. (2008). Approximation and learning by greedy algorithms. *The Annals of Statistics*, 36(1):64–94.

[Bartlett and Mendelson, 2002] Bartlett, P. L. and Mendelson, S. (2002). Rademacher and Gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3(Nov):463–482.

[Bengio et al., 2007] Bengio, Y., Lamblin, P., Popovici, D., and Larochelle, H. (2007). Greedy layer-wise training of deep networks. pages 153–160.

[Benning et al., 2019] Benning, M., Celledoni, E., Ehrhardt, M. J., Owren, B., and Schönlieb, C.-B. (2019). Deep learning as optimal control problems: Models and numerical methods. *Journal of Computational Dynamics*, 6(2):171.

[Bertsekas, 1995] Bertsekas, D. P. (1995). *Dynamic programming and optimal control*, volume 1. Athena scientific Belmont, MA.

[Bertsekas, 2019] Bertsekas, D. P. (2019). *Reinforcement learning and optimal control*. Athena Scientific Belmont, MA.

[Binev et al., 2011] Binev, P., Cohen, A., Dahmen, W., DeVore, R., Petrova, G., and Wojtaszczyk, P. (2011). Convergence rates for greedy algorithms in reduced basis methods. *SIAM Journal on Mathematical Analysis*, 43(3):1457–1472.

[Bölcskei et al., 2019] Bölcskei, H., Grohs, P., Kutyniok, G., and Petersen, P. (2019). Optimal approximation with sparsely connected deep neural networks. *SIAM Journal on Mathematics of Data Science*, 1(1):8–45.

[Bottou et al., 2018] Bottou, L., Curtis, F. E., and Nocedal, J. (2018). Optimization methods for large-scale machine learning. *Siam Review*, 60(2):223–311.

[Bousquet and Elisseeff, 2002] Bousquet, O. and Elisseeff, A. (2002). Stability and generalization. *Journal of machine learning research*, 2(Mar):499–526.

[Brezis, 2010] Brezis, H. (2010). *Functional analysis, Sobolev spaces and partial differential equations*. Springer Science & Business Media.

[Bruna and Mallat, 2013] Bruna, J. and Mallat, S. (2013). Invariant scattering convolution networks. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1872–1886.

[Budd et al., 2009] Budd, C. J., Huang, W., and Russell, R. D. (2009). Adaptivity with moving grids. *Acta Numerica*, 18:111–241.

[Burger and Neubauer, 2001] Burger, M. and Neubauer, A. (2001). Error bounds for approximation with neural networks. *Journal of Approximation Theory*, 112(2):235–250.

[Cardaliaguet et al., 2012] Cardaliaguet, P., Lasry, J.-M., Lions, P.-L., and Porretta, A. (2012). Long time average of mean field games. *Networks & Heterogeneous Media*, 7(2).

[Cardaliaguet et al., 2013] Cardaliaguet, P., Lasry, J.-M., Lions, P.-L., and Porretta, A. (2013). Long time average of mean field games with a nonlocal coupling. *SIAM Journal on Control and Optimization*, 51(5):3558–3591.

[Cardaliaguet and Porretta, 2019] Cardaliaguet, P. and Porretta, A. (2019). Long time behavior of the master equation in mean field game theory. *Analysis & PDE*, 12(6):1397–1453.

[Carlson et al., 2012] Carlson, D., Haurie, A., and Leizarowitz, A. (2012). *Infinite Horizon Optimal Control: Deterministic and Stochastic Systems*. Springer Berlin Heidelberg.

[Celledoni et al., 2020] Celledoni, E., Ehrhardt, M. J., Etmann, C., McLachlan, R. I., Owren, B., Schönlieb, C.-B., and Sherry, F. (2020). Structure preserving deep learning. *arXiv preprint arXiv:2006.03364*.

[Chen et al., 2019] Chen, R. T., Behrmann, J., Duvenaud, D. K., and Jacobsen, J.-H. (2019). Residual flows for invertible generative modeling. In *Advances in Neural Information Processing Systems*, pages 9916–9926.

[Chen et al., 2018] Chen, T. Q., Rubanova, Y., Bettencourt, J., and Duvenaud, D. K. (2018). Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, pages 6571–6583.

[Chizat and Bach, 2018] Chizat, L. and Bach, F. (2018). On the global convergence of gradient descent for over-parameterized models using optimal transport. In *Advances in neural information processing systems*, pages 3036–3046.

[Chizat and Bach, 2020] Chizat, L. and Bach, F. (2020). Implicit bias of gradient descent for wide two-layer neural networks trained with the logistic loss. *arXiv preprint arXiv:2002.04486*.

[Cohen and DeVore, 2015] Cohen, A. and DeVore, R. (2015). Approximation of high-dimensional parametric pdes. *Acta Numerica*, 24:1–159.

[Conforti et al., 2020] Conforti, G., Kazeykina, A., and Ren, Z. (2020). Game on random environement, mean-field Langevin system and neural networks. *arXiv preprint arXiv:2004.02457*.

[Coron, 2007] Coron, J.-M. (2007). *Control and nonlinearity*. Number 136. American Mathematical Soc.

[Coron and Trélat, 2004] Coron, J.-M. and Trélat, E. (2004). Global steady-state controllability of one-dimensional semilinear heat equations. *SIAM journal on control and optimization*, 43(2):549–569.

[Cranmer et al., 2020] Cranmer, M., Greydanus, S., Hoyer, S., Battaglia, P., Spergel, D., and Ho, S. (2020). Lagrangian neural networks. *arXiv preprint arXiv:2003.04630*.

[Cuchiero et al., 2019] Cuchiero, C., Larsson, M., and Teichmann, J. (2019). Deep neural networks, generic universal interpolation, and controlled ODEs. *arXiv preprint arXiv:1908.07838*.

[Cybenko, 1989] Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, pages 303–314.

[Dáger and Zuazua, 2006] Dáger, R. and Zuazua, E. (2006). *Wave propagation, observation and control in 1-d flexible multi-structures*, volume 50. Springer Science & Business Media.

[Daubechies et al., 2019] Daubechies, I., DeVore, R., Foucart, S., Hanin, B., and Petrova, G. (2019). Nonlinear approximation and (Deep) ReLU networks. *arXiv preprint arXiv:1905.02199*.

[Dorfman et al., 1958] Dorfman, R., Samuelson, P., and Solow, R. (1958). *Linear Programming and Economic Analysis*. Dover Books on Advanced Mathematics. Dover Publications.

[Dupont et al., 2019] Dupont, E., Doucet, A., and Teh, Y. W. (2019). Augmented Neural ODEs. In *Advances in Neural Information Processing Systems*, pages 3134–3144.

[E, 2017] E, W. (2017). A proposal on machine learning via dynamical systems. *Communications in Mathematics and Statistics*, 5(1):1–11.

[E et al., 2019] E, W., Han, J., and Li, Q. (2019). A mean-field optimal control formulation of deep learning. *Research in the Mathematical Sciences*, 6(1):10.

[Esteve et al., 2020] Esteve, C., Kouhkouh, H., Pighin, D., and Zuazua, E. (2020). The turnpike property and the long-time behavior of the Hamilton-Jacobi equation. *arXiv preprint arXiv:2006.10430*.

[Fan et al., 2019] Fan, J., Cong, M., and Yiqiao, Z. (2019). A selective overview of deep learning. *arXiv preprint arXiv:1904.05526*.

[Goodfellow et al., 2016] Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*.

[Goodfellow et al., 2014] Goodfellow, I. J., Shlens, J., and Szegedy, C. (2014). Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

[Grathwohl et al., 2018] Grathwohl, W., Chen, R. T., Bettencourt, J., Sutskever, I., and Duvenaud, D. (2018). Ffjord: Free-form continuous dynamics for scalable reversible generative models. *arXiv preprint arXiv:1810.01367*.

[Greydanus et al., 2019] Greydanus, S., Dzamba, M., and Yosinski, J. (2019). Hamiltonian neural networks. In *Advances in Neural Information Processing Systems*, pages 15353–15363.

[Grüne and Guglielmi, 2018] Grüne, L. and Guglielmi, R. (2018). Turnpike properties and strict dissipativity for discrete time linear quadratic optimal control problems. *SIAM Journal on Control and Optimization*, 56(2):1282–1302.

[Grüne and Müller, 2016] Grüne, L. and Müller, M. A. (2016). On the relation between strict dissipativity and turnpike properties. *Systems & Control Letters*, 90:45–53.

[Grüne et al., 2019] Grüne, L., Schaller, M., and Schiela, A. (2019). Exponential sensitivity and turnpike analysis for linear quadratic optimal control of general evolution equations. *Journal of Differential Equations*.

[Grüne et al., 2020] Grüne, L., Schaller, M., and Schiela, A. (2020). Exponential sensitivity and turnpike analysis for linear quadratic optimal control of general evolution equations. *Journal of Differential Equations*, 268(12):7311–7341.

[Gunasekar et al., 2018] Gunasekar, S., Lee, J. D., Soudry, D., and Srebro, N. (2018). Implicit bias of gradient descent on linear convolutional networks. In *Advances in Neural Information Processing Systems*, pages 9461–9471.

[Haber and Ruthotto, 2017] Haber, E. and Ruthotto, L. (2017). Stable architectures for deep neural networks. *Inverse Problems*, 34(1):014004.

[Haurie, 1976] Haurie, A. (1976). Optimal control on an infinite time horizon: the turnpike approach. *Journal of Mathematical Economics*, 3(1):81–102.

[He et al., 2016] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

[Hinton et al., 2012] Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N., et al. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal processing magazine*, 29(6):82–97.

[Hirsch, 1989] Hirsch, M. W. (1989). Convergent activation dynamics in continuous time networks. *Neural networks*, 2(5):331–349.

[Hopfield, 1982] Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554–2558.

[Hornik et al., 1989] Hornik, K., Stinchcombe, M., and White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural networks*, pages 359–366.

[Hu et al., 2019] Hu, K., Kazeykina, A., and Ren, Z. (2019). Mean-field Langevin system, optimal control and deep neural networks. *arXiv preprint arXiv:1909.07278*.

[Jabir et al., 2019] Jabir, J.-F., Šiška, D., and Szpruch, Ł. (2019). Mean-field neural ODEs via relaxed optimal control. *arXiv preprint arXiv:1912.05475*.

[Jin et al., 2020] Jin, S., Li, L., and Liu, J.-G. (2020). Random batch methods (rbm) for interacting particle systems. *Journal of Computational Physics*, 400:108877.

[Kakade et al., 2009] Kakade, S. M., Sridharan, K., and Tewari, A. (2009). On the complexity of linear prediction: Risk bounds, margin bounds, and regularization. In *Advances in Neural Information Processing Systems*, pages 793–800.

[Kelly, 2017] Kelly, M. (2017). An introduction to trajectory optimization: How to do your own direct collocation. *SIAM Review*, 59(4):849–904.

[Kidger and Lyons, 2020] Kidger, P. and Lyons, T. (2020). Universal approximation with deep narrow networks. *Conference on Learning Theory*.

[Kingma and Ba, 2014] Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

[Krizhevsky et al., 2009] Krizhevsky, A., Hinton, G., et al. (2009). Learning multiple layers of features from tiny images.

[Krizhevsky et al., 2012] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105.

[LeCun et al., 2015] LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444.

[LeCun et al., 1990] LeCun, Y., Boser, B. E., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W. E., and Jackel, L. D. (1990). Handwritten digit recognition with a back-propagation network. In *Advances in Neural Information Processing Systems*, pages 396–404.

[LeCun et al., 1988] LeCun, Y., Touresky, D., Hinton, G., and Sejnowski, T. (1988). A theoretical framework for back-propagation. In *Proceedings of the 1988 connectionist models summer school*, volume 1, pages 21–28. CMU, Pittsburgh, Pa: Morgan Kaufmann.

[Leung et al., 2014] Leung, M. K., Xiong, H. Y., Lee, L. J., and Frey, B. J. (2014). Deep learning of the tissue-regulated splicing code. *Bioinformatics*, 30(12):i121–i129.

[Li et al., 2017] Li, Q., Chen, L., Tai, C., and E, W. (2017). Maximum principle based algorithms for deep learning. *The Journal of Machine Learning Research*, 18(1):5998–6026.

[Lions, 1988] Lions, J.-L. (1988). Exact controllability, stabilization and perturbations for distributed systems. *SIAM Review*, 30(1):1–68.

[Liu and Markowich, 2019] Liu, H. and Markowich, P. (2019). Selection dynamics for deep neural networks. *arXiv preprint arXiv:1905.09076*.

[Livni et al., 2014] Livni, R., Shalev-Shwartz, S., and Shamir, O. (2014). On the computational efficiency of training neural networks. In *Advances in Neural Information Processing Systems*, pages 855–863.

[Lohéac and Zuazua, 2016] Lohéac, J. and Zuazua, E. (2016). From averaged to simultaneous controllability. In *Annales de la Faculté des sciences de Toulouse: Mathématiques*, volume 25, pages 785–828.

[Lu et al., 2020] Lu, Y., Ma, C., Lu, Y., Lu, J., and Ying, L. (2020). A mean-field analysis of deep ResNet and beyond: Towards provable optimization via overparameterization from depth. *arXiv preprint arXiv:2003.05508*.

[Lu et al., 2018] Lu, Y., Zhong, A., Li, Q., and Dong, B. (2018). Beyond finite layer neural networks: Bridging deep architectures and numerical differential equations. In *International Conference on Machine Learning*, pages 3276–3285.

[Ma et al., 2019] Ma, C., Wu, L., et al. (2019). Machine learning from a continuous viewpoint. *arXiv preprint arXiv:1912.12777*.

[Mallat, 2012] Mallat, S. (2012). Group invariant scattering. *Communications on Pure and Applied Mathematics*, 65(10):1331–1398.

[Mallat, 2016] Mallat, S. (2016). Understanding deep convolutional networks. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374(2065):20150203.

[McKenzie, 1963] McKenzie, L. W. (1963). Turnpike theorems for a generalized Leontief model. *Econometrica: Journal of the Econometric Society*, pages 165–180.

[McKenzie, 1976] McKenzie, L. W. (1976). Turnpike theory. *Econometrica: Journal of the Econometric Society*, pages 841–865.

[Michel et al., 1989] Michel, A. N., Farrell, J. A., and Porod, W. (1989). Qualitative analysis of neural networks. *IEEE Transactions on Circuits and Systems*, 36(2):229–243.

[Montanari and Zhong, 2020] Montanari, A. and Zhong, Y. (2020). The interpolation phase transition in neural networks: Memorization and generalization under lazy training. *arXiv preprint arXiv:2007.12826*.

[Mukherjee et al., 2006] Mukherjee, S., Niyogi, P., Poggio, T., and Rifkin, R. (2006). Learning theory: stability is sufficient for generalization and necessary and sufficient for consistency of empirical risk minimization. *Advances in Computational Mathematics*, 25:161–193.

[Ng, 2004] Ng, A. Y. (2004). Feature selection, L^1 vs. L^2 regularization, and rotational invariance. In *Proceedings of the twenty-first international conference on Machine learning*, page 78.

[Paszke et al., 2017] Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. (2017). Automatic differentiation in PyTorch.

[Pighin, 2020] Pighin, D. (2020). The turnpike property in semilinear control. *arXiv:2004.03269*.

[Pighin and Sakamoto, 2020] Pighin, D. and Sakamoto, N. (2020). The turnpike with lack of observability. *arXiv preprint arXiv:2007.14081*.

[Pighin and Zuazua, 2018] Pighin, D. and Zuazua, E. (2018). Controllability under positivity constraints of semilinear heat equations. *Mathematical Control & Related Fields*, 8:935–964.

[Pineda, 1987] Pineda, F. J. (1987). Generalization of back-propagation to recurrent neural networks. *Physical Review Letters*, 59(19):2229.

[Pinkus, 1999] Pinkus, A. (1999). Approximation theory of the mlp model in neural networks. *Acta numerica*, 8(1):143–195.

[Poggio et al., 2004] Poggio, T., Rifkin, R., Mukherjee, S., and Niyogi, P. (2004). General conditions for predictivity in learning theory. *Nature*, 428(6981):419–422.

[Porretta and Zuazua, 2013] Porretta, A. and Zuazua, E. (2013). Long time versus steady state optimal control. *SIAM Journal on Control and Optimization*, 51(6):4242–4273.

[Porretta and Zuazua, 2016] Porretta, A. and Zuazua, E. (2016). Remarks on long time versus steady state optimal control. In *Mathematical Paradigms of Climate Science*, pages 67–89. Springer.

[Ranzato et al., 2007] Ranzato, M., Poultney, C., Chopra, S., and Cun, Y. L. (2007). Efficient learning of sparse representations with an energy-based model. In *Advances in Neural Information Processing Systems*, pages 1137–1144.

[Rapaport and Cartigny, 2004] Rapaport, A. and Cartigny, P. (2004). Turnpike theorems by a value function approach. *ESAIM: Control, Optimisation and Calculus of Variations*, 10(1):123–141.

[Rapaport and Cartigny, 2005] Rapaport, A. and Cartigny, P. (2005). Competition between most rapid approach paths: necessary and sufficient conditions. *Journal of Optimization Theory and Applications*, 124(1):1–27.

[Recht, 2019] Recht, B. (2019). A tour of reinforcement learning: The view from continuous control. *Annual Review of Control, Robotics, and Autonomous Systems*, 2:253–279.

[Rockafellar, 1973] Rockafellar, R. (1973). Saddle points of Hamiltonian systems in convex problems of Lagrange. *Journal of Optimization Theory and Applications*, 12(4):367–390.

[Rubanova et al., 2019] Rubanova, Y., Chen, R. T., and Duvenaud, D. K. (2019). Latent ordinary differential equations for irregularly-sampled time series. In *Advances in Neural Information Processing Systems*, pages 5320–5330.

[Ruiz-Balet and Zuazua, 2019] Ruiz-Balet, D. and Zuazua, E. (2019). Control under constraints for multi-dimensional reaction-diffusion monostable and bistable equations. *arXiv preprint arXiv:1912.13066*.

[Ruthotto and Haber, 2019] Ruthotto, L. and Haber, E. (2019). Deep neural networks motivated by partial differential equations. *Journal of Mathematical Imaging and Vision*, pages 1–13.

[Ruthotto et al., 2020] Ruthotto, L., Osher, S. J., Li, W., Nurbekyan, L., and Fung, S. W. (2020). A machine learning framework for solving high-dimensional mean field game and mean field control problems. *Proceedings of the National Academy of Sciences*, 117(17):9183–9193.

[Samuelson, 1972] Samuelson, P. A. (1972). The general saddle point property of optimal-control motions. *Journal of Economic Theory*, 5(1):102 – 120.

[Simonyan and Zisserman, 2014] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[Sontag and Sussmann, 1997] Sontag, E. and Sussmann, H. (1997). Complete controllability of continuous-time recurrent neural networks. *Systems & Control letters*, 30(4):177–183.

[Sontag and Qiao, 1999] Sontag, E. D. and Qiao, Y. (1999). Further results on controllability of recurrent neural networks. *Systems & Control letters*, 36(2):121–129.

[Soudry et al., 2018] Soudry, D., Hoffer, E., Nacson, M. S., Gunasekar, S., and Srebro, N. (2018). The implicit bias of gradient descent on separable data. *The Journal of Machine Learning Research*, 19(1):2822–2878.

[Sutskever et al., 2014] Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, pages 3104–3112.

[Tabuada and Gharesifard, 2020] Tabuada, P. and Gharesifard, B. (2020). Universal approximation power of deep neural networks via nonlinear control theory. *arXiv preprint arXiv:2007.06007*.

[Thorpe and van Gennip, 2018] Thorpe, M. and van Gennip, Y. (2018). Deep limits of residual neural networks. *arXiv preprint arXiv:1810.11741*.

[Trélat, 2005] Trélat, E. (2005). *Contrôle optimal: théorie & applications*.

[Trélat et al., 2018] Trélat, E., Zhang, C., and Zuazua, E. (2018). Steady-state and periodic exponential turnpike property for optimal control problems in Hilbert spaces. *SIAM Journal on Control and Optimization*, 56(2):1222–1252.

[Trélat and Zuazua, 2015] Trélat, E. and Zuazua, E. (2015). The turnpike property in finite-dimensional nonlinear optimal control. *Journal of Differential Equations*, 258(1):81–114.

[Tzen and Raginsky, 2019] Tzen, B. and Raginsky, M. (2019). Neural stochastic differential equations: Deep latent Gaussian models in the diffusion limit. *arXiv preprint arXiv:1905.09883*.

[Vapnik, 2013] Vapnik, V. (2013). *The nature of statistical learning theory*. Springer Science & Business Media.

[von Neumann, 1945] von Neumann, J. (1945). A model of general economic equilibrium. *Review of Economic Studies*, 13(1):1–9.

[Wilde and Kokotovic, 1972] Wilde, R. and Kokotovic, P. (1972). A dichotomy in linear control theory. *IEEE Transactions on Automatic control*, 17(3):382–383.

[Yu et al., 2010] Yu, D., Shizhen, W., and Li, D. (2010). Sequential labeling using deep-structured conditional random fields. *IEEE Journal of Selected Topics in Signal Processing*, 4(6):965–973.

[Yun et al., 2019] Yun, C., Sra, S., and Jadbabaie, A. (2019). Small relu networks are powerful memorizers: a tight analysis of memorization capacity. In *Advances in Neural Information Processing Systems*, pages 15558–15569.

[Zaslavski, 2006] Zaslavski, A. J. (2006). *Turnpike properties in the calculus of variations and optimal control*, volume 80. Springer Science & Business Media.

[Zhang et al., 2016] Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. (2016). Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*.

[Zhang and Schaeffer, 2019] Zhang, L. and Schaeffer, H. (2019). Forward stability of resnet and its variants. *Journal of Mathematical Imaging and Vision*, pages 1–24.

[Zuazua, 2005] Zuazua, E. (2005). Propagation, observation, and control of waves approximated by finite difference methods. *SIAM Review*, 47(2):197–243.

Carlos Esteve, Borjan Geshkovski, Dario Pighin
Departamento de Matemáticas
Universidad Autónoma de Madrid
28049 Madrid, Spain
and
Chair of Computational Mathematics
Fundación Deusto
Av. de las Universidades, 24
48007 Bilbao, Basque Country, Spain
Email address: {carlos.esteve, borjan.geshkovski, dario.pighin}@uam.es

Enrique Zuazua
Chair in Applied Analysis, Alexander von Humboldt-Professorship
Department of Mathematics
Friedrich-Alexander-Universität Erlangen-Nürnberg
91058 Erlangen, Germany
and
Chair of Computational Mathematics
Fundación Deusto
Av. de las Universidades, 24
48007 Bilbao, Basque Country, Spain
and
Departamento de Matemáticas
Universidad Autónoma de Madrid
28049 Madrid, Spain
Email address: enrique.zuazua@fau.de