HAL
open science

Large-time asymptotics in deep learning

Carlos Esteve-Yagüe, Borjan Geshkovski, Dario Pighin, Enrique Zuazua

To cite this version:

Carlos Esteve-Yagüe, Borjan Geshkovski, Dario Pighin, Enrique Zuazua. Large-time asymptotics in deep learning. 2021. hal-02912516v2

HAL Id: hal-02912516

https://hal.science/hal-02912516v2

Preprint submitted on 29 Mar 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

LARGE-TIME ASYMPTOTICS IN DEEP LEARNING

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

Abstract. We consider the neural ODE perspective of supervised learning and study the impact of the final time T (which may indicate the depth of a corresponding ResNet) in training. For the classical L^2–regularized empirical risk minimization problem, whenever the neural ODE dynamics are homogeneous with respect to the parameters, we show that the training error is at most of the order $\mathcal{O}\left(\frac{1}{T}\right)$. Furthermore, if the loss inducing the empirical risk attains its minimum, the optimal parameters converge to minimal L^2–norm parameters which interpolate the dataset. By a natural scaling between T and the regularization hyperparameter λ we obtain the same results when $\lambda \searrow 0$ and T is fixed. This allows us to stipulate generalization properties in the overparametrized regime, now seen from the large depth, neural ODE perspective. To enhance the polynomial decay, inspired by turnpike theory in optimal control, we propose a learning problem with an additional integral regularization term of the neural ODE trajectory over $[0, T]$. In the setting of ℓ^p–distance losses, we prove that both the training error and the optimal parameters are at most of the order $\mathcal{O}\left(e^{-\mu t}\right)$ in any $t \in [0, T]$. The aforementioned stability estimates are also shown for continuous space-time neural networks, taking the form of nonlinear integro-differential equations. By using a time-dependent moving grid for discretizing the spatial variable, we demonstrate that these equations provide a framework for addressing ResNets with variable widths.

Contents

1. Introduction 2
2. Roadmap to learning via neural ODEs 8
3. Empirical risk minimization 11
4. Augmented empirical risk minimization 19
5. The interpolation regime 28
6. Continuous space-time neural networks 33
7. Proofs 39
8. Concluding remarks 54
Appendix A. Auxiliary proofs 57
References 61

Keywords. Deep learning, ResNets, neural ODEs, regularization path, optimal control, exponential stability, approximation, turnpike theory.
AMS Subject Classification. 49J15; 49M15; 49J20; 49K20; 93C20; 49N05.

Date: March 29, 2021.

1. Introduction

Modern supervised learning addresses the problem of predicting from data, which roughly consists in approximating an unknown function $f : \mathcal{X} \to \mathcal{Y}$ from N known but possibly noisy samples $\{\vec{x}_i, \vec{y}_i\}_{i=1}^N \subset \mathcal{X} \times \mathcal{Y}$. Depending on the nature of the labels $\vec{y}_i$, one distinguishes two types of supervised learning tasks, namely that of *classification* (labels take values in a finite set of m classes, e.g. $\mathcal{Y} := \{1, \dots, m\}$) and *regression* (labels take continuous values in $\mathcal{Y} \subset \mathbb{R}^m$). In many applications, the dimension d of each sample $\vec{x}_i \in \mathcal{X} \subset \mathbb{R}^d$ may be big compared to the number/dimension m of the labels – in image classification for instance, a sample of the ImageNet dataset [Krizhevsky et al., 2012], which has $m = 1000$ classes, is an element of $\mathbb{R}^{65536}$.

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 parametric 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 [Goodfellow et al., 2016] for a comprehensive presentation of these topics.

Deep neural networks are parametrized computational architectures which propagate each individual sample of the input data $\{\vec{x}_i\}_{i=1}^N \subset \mathcal{X} \subset \mathbb{R}^d$ across a sequence of linear parametric operators and simple nonlinearities. The so-called *residual neural networks* (ResNets, [He et al., 2016]) may, in the simplest case, be cast as schemes of the mould

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

for all $i \in [N]$, where we set $[N] := \{1, \dots, N\}$. The unknowns are the states $\mathbf{x}_i^k \in \mathbb{R}^d$ for any $i \in [N]$, σ is an explicit scalar, Lipschitz continuous nonlinear function defined component-wise in (1.1), $\left\{w^k, b^k\right\}_{k=0}^{N_{\text{layers}}-1}$ are optimizable parameters (controls) with $w^k \in \mathbb{R}^{d \times d}$ – called weights, and $b^k \in \mathbb{R}^d$ – called biases, and $N_{\text{layers}} \geqslant 1$ designates the number of layers referred to as the depth. The training process consists in finding optimal parameters steering all of the network outputs $\mathbf{x}_i^{N_{\text{layers}}}$ as close as possible to the corresponding labels $\vec{y}_i$ by solving

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

whilst guaranteeing reliable performance on unseen data (ensuring *generalization*). Here $\text{loss}(\cdot, \cdot)$ is a given continuous and nonnegative function which differs depending on the task in hand – for instance $\text{loss}(x, y) := \|x - y\|_{\ell^p}^p$ for $p \in \{1, 2\}$ is commonly used for regression tasks, while $\text{loss}(x, y) = \log(1 + \exp(-yx))$ may be used for binary classification, namely when $\vec{y}_i \in \{-1, 1\}$ (we refer to Section 2.3 for more general settings). On the other hand, $P : \mathbb{R}^d \to \mathbb{R}^m$ is an affine map whose coefficients in practice are part of the optimizable parameters. In our work, we shall assume that P is given and specified on a case-by-case basis.

Due to the inherent dynamical systems nature of ResNets, several recent works have aimed at studying an associated continuous-time formulation in some detail, a trend started with the works [Weinan, 2017; Haber and Ruthotto, 2017] (see also [Lu et al.,

2018; Ruthotto and Haber, 2019]). This perspective is motivated by the simple observation that for any $i \in [N]$ and for $T > 0$, (1.1) is roughly the forward Euler scheme for the neural ordinary differential equation (neural ODE)

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

The continuous-time, neural ODE formalism of deep learning has been used to great effect in applications – for instance, by using adaptive ODE solvers [Chen et al., 2018; Dupont et al., 2019; Queiruga et al., 2020], symplectic and multigrid methods [Celledoni et al., 2020; Gunther et al., 2020], or indirect training algorithms based on the Pontryagin Maximum Principle [Li et al., 2017; Benning et al., 2019] –, and also for generative modeling through normalizing flows [Grathwohl et al., 2018; Chen et al., 2019]. We emphasize that the origins of continuous-time supervised learning go back to the 1980s – the neural network model proposed in [Hopfield, 1982] is a differential equation, whereas in [LeCun et al., 1988] back-propagation is connected to the adjoint method arising in optimal control. Related works include studies on identification of the weights from data [Albertini and Sontag, 1993; Albertini et al., 1993] and controllability of continuous-time recurrent networks [Sontag and Sussmann, 1997; Sontag and Qiao, 1999].

The role of the final time horizon $T > 0$, which plays a key role in the control of dynamical systems, has not been analyzed in the context of supervised learning problems via models such as (1.2). Now note once again that in the ResNet (1.1) the time-step $\Delta t = {}^{T}/_{N_{\text{layers}}}$ is fixed (equal to 1), and each time instance of a forward Euler discretization to (1.2) would represent a different layer of (1.1). Hence, whenever the time-step $\Delta t = {}^{T}/_{N_{\text{layers}}}$ is fixed (or goes to zero when T is increased), the time horizon T in (1.2) serves as an indicator of the number of layers N_{layers} in the ResNet (1.1). Thus, a good knowledge of the behavior of the learning problem and the neural ODE flow over longer time horizons is desirable in view of understanding approximation and generalization properties. In this work, we aim to bridge this gap by proposing several insights stemming from an analysis of the role of the time horizon T.

1.1. **Our contributions.** We shall focus this presentation on the neural ODE (1.2), but our results also hold for other systems, as seen in the respective statements.

1. We first consider the classical regularized empirical risk minimization problem

$$\inf_{\substack{[w,b]\in\mathcal{H}(0,T;\mathbb{R}^{d_u}) \\ \mathbf{x}_i(\cdot) \text{ solves } (1.2)}} \underbrace{\frac{1}{N}\sum_{i=1}^{N} \text{loss}\big(P\mathbf{x}_i(T),\, \vec{y}_i\big)}_{:=\mathcal{E}(\mathbf{x}(T))} + \underbrace{\lambda \Big\| [w,b] \Big\|^2_{\mathcal{H}(0,T;\mathbb{R}^{d_u})}}_{\text{regularization}} \tag{1.3}$$

where[1] $\mathcal{H}$ is either L^2 or H^1. In Theorem 3.1, we show that when $\text{loss}(\cdot,\cdot)$ and the affine map $P : \mathbb{R}^d \to \mathbb{R}^m$ are such that the minimum of $\mathcal{E}$ (equal to zero) is attained and when the activation function σ in (1.2) is 1–homogeneous, the training error $\mathcal{E}(\mathbf{x}_T(T))$ of the collection $\mathbf{x}_T = \{\mathbf{x}_{T,i}\}_{i\in[N]}$ of solutions to (1.2) corresponding to any solution $[w_T, b_T]$ to the minimization problem (1.3), is at

[1]Here $H^1(0,T;\mathbb{R}^{d_u})$ denotes the Sobolev space of square integrable functions from $(0,T)$ to $\mathbb{R}^{d_u}$ with square integrable weak derivatives (see Section 1.4). We consider H^1–regularization in the setting of (1.2) to ensure the existence of minimizers, see Remark 1.

most of the order $\mathcal{O}\left(\frac{1}{T}\right)$, whilst the optimal parameters $[w_T, b_T]$ converge, on a suitable time-scale, to a solution $[w^*, b^*]$ of

$$\inf_{\substack{[w,b]\in\mathcal{H}(0,1;\mathbb{R}^{d_u}) \\ \mathbf{x}_i(\cdot) \text{ solves (1.2) in } [0,1] \\ \text{and} \\ \mathcal{E}(\mathbf{x}(1))=0}} \left\|[w,b]\right\|^2_{\mathcal{H}(0,1;\mathbb{R}^{d_u})} \tag{1.4}$$

when $T \longrightarrow \infty$.

Let us put the above result into context. For neural ODEs for which L^2–regularization suffices, we remark that $T \longrightarrow \infty$ is equivalent to $\lambda \searrow 0$. The latter is the regularization path limit, studied in the literature for linear models and multi-layer perceptrons (but not for more compound neural ODE models), where the solutions obtained in the limit can be shown to satisfy desirable generalization properties (see Section 1.2).

Using similar arguments as when $T \longrightarrow \infty$, in Theorem 3.2 we obtain the same conclusions when $\lambda \searrow 0$ and T is fixed. Consequently, Theorem 3.1 stipulates generalization properties – namely, optimizing with $T \gg 1$, which may be interpreted as a larger depth for ResNets, has the practically desirable effect of making the training error close to zero, but by means of parameters with the smallest amplitude and thus trajectories with the least oscillations.

2. In the setting of losses for which $\mathcal{E}$ does not attain its minimum equal to zero, occurring in many classification contexts (for instance $\text{loss}(Px, y) = \log\left(1 + e^{-yPx}\right)$ when $\vec{y}_i \in \mathcal{Y} = \{-1, 1\}$ or multi-label tasks where $\vec{y}_i \in [m]$ for $m \geqslant 2$ via cross-entropy loss), we show that the training error $\mathcal{E}(\mathbf{x}_T(T))$ is at most of the order $\mathcal{O}\left(\frac{1}{T^\alpha}\right)$ for all $\alpha \in (0, 1)$ (see Theorem 3.3). Analog results hold when $T > 0$ is fixed and $\lambda \searrow 0$ (see Theorem 3.4).

3. To enhance the polynomial stability of the training error $\mathcal{E}(\mathbf{x}_T(T))$ with respect to T, we also consider the augmented empirical risk minimization problem[2]

$$\inf_{\substack{[w,b]\in\mathcal{H}(0,T;\mathbb{R}^{d_u}) \\ \mathbf{x}_i(\cdot) \text{ solves (1.2)}}} \mathcal{E}(\mathbf{x}(T)) + \frac{1}{N}\sum_{i=1}^{N}\int_0^T \|\mathbf{x}_i(t) - \overline{\mathbf{x}}_i\|^2 \, \mathrm{d}t + \lambda\left\|[w,b]\right\|^2_{\mathcal{H}(0,T;\mathbb{R}^{d_u})}, \tag{1.5}$$

where $P : \mathbb{R}^d \to \mathbb{R}^m$ appearing in $\mathcal{E}$ is Lipschitz and surjective, loss is an ℓ^p–distance, while the targets $\overline{\mathbf{x}}_i \in P^{-1}(\{\vec{y}_i\})$ for all $i \in [N]$ are arbitrary but given. Under a specific controllability assumption but without any differentiability assumptions on the activation function σ or smallness assumptions on the dataset, in Theorem 4.1 (see also Theorem 4.2) we show that optimal parameters $[w_T(t), b_T(t)]$ and the training error $\mathcal{E}(\mathbf{x}_T(t))$ of the corresponding vector $\mathbf{x}_T(t)$ of solutions to (1.1) are at most of the order $\mathcal{O}(e^{-\mu t})$ for any $t \in [0, T]$ and some $\mu > 0$ independent of T.

This result is in line with Theorem 3.1 and Theorem 3.3, but with an improved estimate of the time horizon needed to be ε–close to the interpolation regime for any given $\varepsilon > 0$. Due to the exponentially small global minimizers, numerical experiments show that the learned flow is simple, which could stipulate possible generalization properties. Theorem 4.1 is a manifestation of the

[2]As for (1.3), we consider $\mathcal{H} = L^2$ when existence of minimizers can be ensured; however, unlike for (1.3), in the setting of (1.2), we will sometimes consider a BV regularization (see Section 1.4 for a definition) rather than the stronger H^1 regularization for technical reasons (see Remark 6).

so-called *turnpike property*[3], well-known in optimal control theory ([Porretta and Zuazua, 2013; Trélat and Zuazua, 2015; Faulwasser and Grüne, 2020]).

Problem (1.5) is motivated by the natural training problem which aims at minimizing the error over every time/layer:

$$\inf_{\substack{[w,b]\in\mathcal{H}(0,T;\mathbb{R}^{d_u})\\ \mathbf{x}_i(\cdot)\text{ solves }(1.2)}} \int_0^T \mathcal{E}(\mathbf{x}(t))\,\mathrm{d}t + \lambda\Big\|[w,b]\Big\|^2_{\mathcal{H}(0,T;\mathbb{R}^{d_u})}, \tag{1.6}$$

where $\mathrm{loss}(\cdot,\cdot)$ appearing in $\mathcal{E}$ is continuous and nonnegative, but otherwise arbitrary. Whilst left without proof, numerical experiments stipulate that a similar decay for the training error, and, combined with Theorem 4.1, motivate the usage of (1.6) in practice (see Section 4.1).

4. In Theorem 5.1 we show that the controllability assumption needed for Theorem 4.1 is satisfied for a subclass of neural ODEs with C^1–regular activation functions σ. We also illustrate the relation between the amplitude of the weights needed to reach the interpolation regime, and the dispersion of the input data (see Proposition 5.1).

5. To address variable width architectures motivated by multi-layer perceptrons and convolutional neural networks, in Section 6 we study a continuous space-time neural network formulation taking the form of a scalar integro-differential equation proposed in [Liu and Markowich, 2020]. We show that, by making use of a time-dependent moving grid for discretizing the spatial variable, these equations provide a framework for addressing ResNets with variable widths. Furthermore, in Theorem 6.1 (resp. Theorem 6.2), we show that some of our finite-dimensional conclusions from Theorem 3.1 (resp. Theorem 4.1) transfer, under similar assumptions, to the continuous space-time neural networks.

1.2. **Related work.** We note that the impact of the time horizon for the regularized empirical risk minimization problem has been studied from a computational perspective in [Effland et al., 2020]. We also refer the reader to the recent work [Faulwasser et al., 2021] for a stability analysis for the augmented supervised learning problem (1.5) in the ResNet setting, and to [Yagüe and Geshkovski, 2021] for a rigorous analysis of the problem (1.6) with an L^1–regularization of the parameters, where sparsity patterns of the parameters are shown.

In [Thorpe and van Gennip, 2018] (see also [Avelin and Nyström, 2020]), 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 step-size Δt goes 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 results are of different nature. Rather than proving 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 indicates the number of layers for the associated time-discretization when the time-step Δt is fixed, and aim to characterize the impact of T on the optimal parameters and on the training error.

[3]The turnpike property is a staple of optimal control problems such as the linear quadratic regulator (LQR) over large but finite time horizons; it refers to the fact that optimal controls (parameters) and trajectories, in long time intervals, remain close to the optimal stationary controls and trajectories. In the setting of (1.5), due to the structure of the neural ODE, the optimal stationary parameters are zero, and thus the optimal stationary state is the collection $\{\overline{\mathbf{x}}_i\}_{i\in[N]}$, as reflected in our result.

Our results are also related to several questions studied in existing literature.

Universal approximation. On a first note, the asymptotic results presented herein may (heuristically) be interpreted as approximation results in the sense of the universal approximation theory. These are density results for neural networks, 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 in Lebesgue spaces. The first result in this direction is the seminal work [Cybenko, 1989], which indicates that shallow neural networks with increasing width, i.e. a superposition of sufficiently many dilated and translated sigmoids, may approximate any continuous function on compact sets. We also refer to [Hornik et al., 1989; Pinkus, 1999] for an extension to multi-layer neural networks. Our results are somewhat dual to [Cybenko, 1989] – therein, to increase the approximation accuracy, the width is allowed to grow, whilst we fix the width and allow the depth to increase. We do note however that we prove approximation properties for the optimal parameters, and for a fixed dataset, unlike what is commonly done in universal approximation theorem, where the parameters are not known explicitly. We refer to the thesis [Müller, 2019] for results and a comprehensive review of universal approximation results for ResNets, and to the recent works [Li et al., 2019; Teshima et al., 2020; Zhang et al., 2020] and [Ruiz-Balet and Zuazua, 2021], for universal approximation results for neural ODE and for an analysis on the latter's working mechanisms.

Regularization path limit: $\lambda \searrow 0$. The regularization path limit $\lambda \searrow 0$ for minimizers $[w_\lambda, b_\lambda]$ has also been addressed in the machine learning literature, albeit for more classical models. For instance in [Rosset et al., 2004, 2003], the authors study linear logistic regression, and show convergence of the margin to the max-margin as $\lambda \searrow 0$, assuming linearly separable data. The max-margin, support vector machine solution (see Remark 4 for details) is a special example among all solutions that fit the training data – another example includes minimal ℓ^2–norm solution for linear regression (or generally supervised learning tasks in which the loss may attain its minimum). Both these solutions can be shown to ensure generalization by virtue of explicit generalization error estimates [Bartlett and Mendelson, 2002; Kakade et al., 2009] for linear models or multi-layer perceptrons. This insight could stipulate a likely generalization capacity of our asymptotic limits as $T \longrightarrow \infty$ or $\lambda \searrow 0$.

The results of [Rosset et al., 2004, 2003] have subsequently been extended in [Wei et al., 2019] (and some of the references therein) to perceptrons with ReLU activations, where the intrinsic homogeneity of the network is used – the authors prove an analog result to Theorem 3.2 when $\lambda \searrow 0$ for regression tasks (the loss is a distance) and two-layer perceptrons. Our results can be seen as an extension of the aforementioned insights to the neural ODE context.

Implicit regularization of gradient methods. When training without explicit regularization (i.e. $\lambda = 0$), a common approach in the literature is to resort to algorithm-dependent generalization analysis, where the limit solutions are akin to the limit solutions when $\lambda \searrow 0$. Whilst the former is not a question we address in this work, where generally estimates are provided in terms of the number of algorithm iterations rather than the number of layers, we provide a brief overview of the literature for completeness. We refer the reader to the recent review [Bartlett et al., 2021] for a comprehensive presentation of these artifacts.

The implicit bias of gradient descent ([Soudry et al., 2018; Gunasekar et al., 2018]) indicates that in the overparametrized regime, after training a linear model or perceptron with gradient-based methods until zero training error, without requiring any explicit parameter regularization, among the many predictors which overfit on the training dataset, the algorithm selects the one which performs best on the test dataset (e.g. minimal ℓ^2–norm solution or max-margin solution). Thus, even-though the approach is significantly different from that when $\lambda \searrow 0$, the asymptotic limits oftentimes coincide.

Recent works have shown that gradient descent can allow overparametrized multi-layer networks to attain arbitrarily low training error on fairly general datasets ([Du et al., 2019; Allen-Zhu et al., 2019a,b]), and find minimum-norm/maximum-margin solutions that fit the data in the settings of logistic regression, deep linear networks, and symmetric matrix factorization ([Gunasekar et al., 2018; Soudry et al., 2018; Ji and Telgarsky, 2018; Ma et al., 2020]). In [Chizat and Bach, 2018, 2020] overparametrization is approached from the point of view of the width of the neural network, unlike our depth-inspired perspective. The authors consider a two-layer perceptron with ReLU activation, exhibit a mean-field, Wasserstein gradient flow formulation of the descent scheme yielding the optimal parameters, and prove that these parameters approach global minimizers of the cost functional, with the global minimizer being characterized as a max-margin classifier in a certain non-Hilbert space of functions. We refer the reader to [Sirignano and Spiliopoulos, 2020; Javanmard et al., 2020; Mei et al., 2018; Nitanda and Suzuki, 2017; Chizat et al., 2018] for related works in this direction, and [Nguyen and Pham, 2020] for an extension of the aforementioned convergence results to multi-layer perceptrons.

1.3. **Outline.** The remainder of the paper is organized as follows. In Section 2, we give a comprehensive presentation of the neural ODE perspective of deep supervised learning. In Section 3, we present our main results in the context of regularized empirical risk minimization (Theorem 3.1 and Theorem 3.3). In Section 4, we present our main result for the augmented empirical risk minimization problem, namely exponential decay of the training error with exponentially small parameters (Theorem 4.1 and also Theorem 4.2), as well as several numerical experiments stimulating the validity of our conjectures. Finally in Section 6, we present the continuous space-time analog of residual neural networks with variable widths, depict some possible approaches for passing from the continuous to the discrete case, and present extensions of Theorem 3.1 and Theorem 4.1 in this context. The proofs of all results may be found in Section 7.

1.4. **Notation, conventions, assumptions.** We set $\dot{x}(t) := \frac{\mathrm{d}x}{\mathrm{d}t}(t)$ and $[n] := \{1, \dots, n\}$. Given $a \in \mathbb{R}^n$, we denote by $a^\top$ its transpose. We use the notation x_T to display the dependence of a vector $x \in \mathbb{R}^n$ on the time horizon T. We denote by $\|a\|$ the standard euclidean norm when $a \in \mathbb{R}^n$ is a vector, and the entry-wise euclidean norm (Frobenius norm) when $a \in \mathbb{R}^{n\times m}$ is a matrix – we recall

$$\|a\| := \left(\sum_{j=1}^{n} \sum_{k=1}^{m} |a_{j,k}|^2 \right)^{1/2}.$$

We denote by $\mathrm{Lip}(\mathbb{R})$ the set of functions $f : \mathbb{R} \longrightarrow \mathbb{R}$ 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 $f : (0,T) \longrightarrow \mathbb{R}^n$ which are square integrable (resp. square integrable and with a square integrable weak derivative) – recall that $H^1(0,T;\mathbb{R}^n)$ is

endowed with the norm $\|f\|^2_{H^1(0,T;\mathbb{R}^n)} := \|f\|^2_{L^2(0,T;\mathbb{R}^n)} + \|\dot{f}\|^2_{L^2(0,T;\mathbb{R}^n)}$. We will use the convention $H^0 = L^2$. We recall (see [Ambrosio et al., 2000, p. 117]) the definition of the space of *bounded variation functions* $\mathrm{BV}(0,T;\mathbb{R}^n)$ as the space of integrable functions whose weak derivative is a finite Radon measure:

$$\mathrm{BV}(0,T;\mathbb{R}^n) := \Big\{ f \in L^1(0,T;\mathbb{R}^n) \colon \exists\, \mathrm{D}f \in \mathcal{M}(0,T;\mathbb{R}^n) \text{ such that } \sum_{j=1}^n \int_0^T f_j(t)\varphi_j'(t)\,\mathrm{d}t = -\sum_{j=1}^n \int_0^T \varphi_j(t)\,\mathrm{d}\mathrm{D}f_j(t), \quad \forall \varphi \in C_c^1(0,T;\mathbb{R}^n) \Big\}.$$

Here, $\mathcal{M}(0,T;\mathbb{R}^n)$ denotes the set of finite Radon measures on $(0,T)$ with values in $\mathbb{R}^n$. The space $\mathrm{BV}(0,T;\mathbb{R}^n)$ is endowed with the norm

$$\|f\|_{\mathrm{BV}(0,T;\mathbb{R}^n)} := \|f\|_{L^1(0,T;\mathbb{R}^n)} + \sum_{j=1}^n |\mathrm{D}f_j|(0,T).$$

These definitions apply to matrix valued functions by simply considering the vectorized form of the matrix. Finally, we say that a function $f : \mathbb{R} \to \mathbb{R}$ is α–homogeneous, for $\alpha > 0$, if $f(cx) = c^\alpha x$ for $c \in \mathbb{R}$ and $x \in \mathbb{R}$.

Throughout the remainder of this work, we will work under the following couple of assumptions, which are universal in the context of machine learning.

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

$$\{\vec{x}_i, \vec{y}_i\}_{i\in[N]} \subset \mathcal{X} \times \mathcal{Y},$$

where $\mathcal{X} \subset \mathbb{R}^d$ and $\vec{x}_i \neq \vec{x}_j$ for $i \neq j$. We suppose that $\mathcal{Y} \neq \varnothing$, with $\mathcal{Y} \subset \mathbb{R}^m$ for regression tasks, $\mathcal{Y} = \{-1, 1\}$ for binary classification and $\mathcal{Y} = [m]$ for mutli-label classification tasks.

We henceforth denote

$$d_u := d \times (d+1), \qquad d_x := dN.$$

We will, most importantly, make use of the following abuse of notation: given the input data $\{\vec{x}_i\}_{i\in[N]}$ with $\vec{x}_i \in \mathbb{R}^d$, we define the stacked vector $\mathbf{x}^0 \in \mathbb{R}^{d_x}$ by

$$\mathbf{x}^0 := \begin{bmatrix} \vec{x}_1 \\ \vdots \\ \vec{x}_N \end{bmatrix} \qquad \text{with} \qquad \mathbf{x}_i^0 = \vec{x}_i \quad \text{for} \quad i \in [N]. \tag{1.7}$$

We shall apply convention (1.7) to other vectors generally denoted by $\mathbf{x} \in \mathbb{R}^{d_x}$ and defined by stacking given sub-vectors $\{\mathbf{x}_i\}_{i\in[N]}$ with $\mathbf{x}_i \in \mathbb{R}^d$:

$$\mathbf{x} := \begin{bmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_N \end{bmatrix}.$$

Thus, when we write $\mathbf{x}_i$ for $i \in [N]$, we insinuate that $\mathbf{x}_i \in \mathbb{R}^d$, rather than considering solely the i-th row of $\mathbf{x} \in \mathbb{R}^{d_x}$ (which is a scalar).

The following assumption is satisfied by most of the commonly used activation functions, including sigmoids such as $\sigma(x) = \tanh(x)$, and rectifiers: $\sigma(x) = \max\{\alpha x, x\}$ for $\alpha \in [0, 1)$.

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

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

2. Roadmap to learning via neural ODEs

2.1. **Feed-forward neural networks.** The canonical example of a feed-forward neural network is the *multi-layer perceptron* (MLP), which generally takes the form

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

for $i \in [N]$; here $N_{\text{layers}} \geqslant 1$ is the depth of (2.1), each k is a layer, the vector $\mathbf{x}_i^k \in \mathbb{R}^{d_k}$ designates the state at the layer k, d_k is referred to as the width of the layer k, while $w^k \in \mathbb{R}^{d_{k+1} \times d_k}$ and $b^k \in \mathbb{R}^{d_k}$ are the optimizable weight and bias parameters of (2.1). Finally, $\sigma \in \mathrm{Lip}(\mathbb{R})$ is a fixed nonlinear activation function – by abuse of notation, we define the vector-valued analog of σ component-wise, namely, $\sigma : \mathbb{R}^d \longrightarrow \mathbb{R}^d$ is defined by $\sigma(\mathbf{x})_j := \sigma(\mathbf{x}_j)$ for $j \in [d]$. 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 $0 \leqslant a < 1$. In practice, the activation σ is generally selected using cross-validation. It can readily be seen that the formulation (2.1) coincides with the more conventional formulation of neural networks as compositional structures of parametric affine operators and nonlinearities, as namely $\mathbf{x}_i^{N_{\text{layers}}} = (\sigma \circ \Lambda^k \circ \ldots \circ \sigma \circ \Lambda^0)(\vec{x}_i)$, 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 affine maps and the nonlinearity σ, to the effect of considering the almost equivalent, but somewhat simpler system

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

We will henceforth concentrate on residual neural networks (ResNets). Contrary to the multi-layer perceptrons (2.1) – (2.2), when considering ResNets 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

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

for $i \in [N]$, where $\mathbf{x}_i^k \in \mathbb{R}^d$ for any i, k, $u^k := [w^k, b^k] \in \mathbb{R}^{d \times (d+1)}$ and $\mathfrak{f}$ as the right hand side in (2.1) or (2.2). As explained in [Lu et al., 2018], other network architectures (including convolutional neural networks) can be fit into the residual network framework.

2.2. **Neural ODEs.** It is readily seen that (2.3) corresponds, modulo a scaling factor $\Delta t = {}^{T}/_{N_{\text{layers}}} = 1$, to the forward Euler discretization of

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

for $i \in [N]$. Here $T > 0$ is a fixed time horizon, and $u(t) := [w(t), b(t)] \in \mathbb{R}^{d\times(d+1)}$. As per what precedes, the nonlinearity $\mathfrak{f}$ in (2.4) may take the form

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

or

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

for $i \in [N]$. We will address both cases in our analytical study, and emphasize the stark differences between the two. 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 different dimensions:

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

where now $w_1(t) \in \mathbb{R}^{d_{\text{hid}}\times d}$, $w_2(t) \in \mathbb{R}^{d\times d_{\text{hid}}}$, $b_1(t) \in \mathbb{R}^{d_{\text{hid}}}$ and $b_2(t) \in \mathbb{R}^d$.

2.3. **Training.** For an input sample $\vec{x}_i \in \mathbb{R}^d$, the output of the neural ODE (2.4), which is used for comparison with the corresponding label $\vec{y}_i$, is a projection of the form $P\mathbf{x}_i(T) \in \mathbb{R}^m$ for some affine map $P : \mathbb{R}^d \to \mathbb{R}^m$:

$$Px := p_1 x + p_2,$$

where $p_1 \in \mathbb{R}^{m\times d}$ and $p_2 \in \mathbb{R}^m$. In the context of binary classification, namely $m = 1$ with $\vec{y}_i = \pm 1$, one may also use $Px := \tanh(p_1 x + p_2)$, for instance. The parameters p_1, p_2 are usually part of the trainable variables, but, as mentioned in the introduction, we shall assume that they are given (but otherwise arbitrary, e.g., p_1 and p_2 may be sampled as a random matrix and random vector respectively) for technical reasons.

In supervised learning, one seeks to tune the parameters $[w, b]$ so that $P\mathbf{x}_i(T)$ approaches $\vec{y}_i$ for $i \in [N]$ with respect to some loss function. To this end, the Tikhonov-regularized, empirical risk minimization problem

$$\inf_{\substack{[w,b] \\ \mathbf{x}_i(\cdot) \text{ solves } (2.4)}} \frac{1}{N}\sum_{i=1}^{N} \text{loss}\big(P\mathbf{x}_i(T), \vec{y}_i\big) + \lambda \Big\| [w, b] \Big\|^2_{H^k(0,T;\mathbb{R}^{d_u})} \tag{2.8}$$

is considered, where $\lambda > 0$ is fixed, and $\mathbf{x}_i$ solves (2.4) with $\mathfrak{f}$ as in (2.5) or (2.6) (although, more general cases such as (2.7) can also be considered). Here

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

is a given continuous function, which in our work we will choose on a case-by-case basis. Common examples include $\text{loss}(x, y) = \|x - y\|^p_{\ell^p}$ with $p = 1, 2$ for regression tasks ($\mathcal{Y} \subset \mathbb{R}^m$), and the cross-entropy loss

$$\text{loss}(x, y) = -\log\left(\frac{e^{x_y}}{\sum_{j=1}^m x_j}\right) \qquad x \in \mathbb{R}^m,\ y \in [m]$$

for classification tasks. Note that (2.8) is the empirical and regularized approximation of the expected risk minimization problem:

$$\inf_{\substack{[w,b] \\ \mathbf{x}_i(\cdot) \text{ solves } (2.4)}} \mathbb{E}\Big[\text{loss}\left(P\mathbf{x}_\cdot(T), \cdot\right)\Big],$$

where $\mathbb{E}[f(\cdot, \cdot)] := \int_{\mathbb{R}^d\times\mathbb{R}^m} f(x, y)\, \mathrm{d}\rho(x, y)$, 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 joint probability distribution,

from which one samples the training dataset $\{\vec{x}_i, \vec{y}_i\}_{i\in[N]}$. We shall solely focus on the empirical problem in this work.

By virtue of the direct method in the calculus of variations, the existence[4] of minimizers for the learning problems we consider herein can readily be shown when the appropriate parameter regularization is used (see Lemma 7.2). We put emphasis on the following remark (see also Remark 6).

Remark 1 (Sobolev regularization)**.** *We stress the consideration of a Sobolev, H^1–regularization in the case of (3.2) as this is sufficient to guarantee the existence of a minimizer. Indeed, an issue could arise if solely an L^2–regularization is used due to the specific nonlinear form of the neural ODE (3.2), which could (but might not) be an impediment for passing to the limit in the equation using only weak convergences (recall $\{\sin(nx)\}_{n=1}^{\infty}$ and $\sigma(x) = \max\{x, 0\}$). This issue is specific to the continuous-time setting, as in the discrete-time thus finite dimensional optimization setting, weak and strong convergences coincide. To our knowledge, a proof of existence of a minimizer in the context of L^2–regularized problems under System (3.2) is not known.*

3. Empirical risk minimization

Throughout the paper we will focus on neural ODEs given by (2.4) with $\mathfrak{f}$ as in (2.5) or (2.6). We comment on further extensions on a case-by-case basis. It will be rather convenient to work with the full stacked state trajectory

$$\mathbf{x}(t) = \begin{bmatrix} \mathbf{x}_1(t) \\ \vdots \\ \mathbf{x}_N(t) \end{bmatrix},$$

and we recall that we make use of the convention (1.7), and we recall that $d_u := d\times(d+1)$ and $d_x := dN$. 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

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

In this section, we consider the problem of regularized empirical risk minimization. For simplicity of notation, we henceforth denote the empirical risk by

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

[4]Uniqueness ought not to be expected (as in most deep learning tasks) due to the inherent lack of convexity. Notwithstanding, there exist cases (e.g. two-layer perceptrons) where uniqueness can be ensured by lifting the training problem as a minimization problem over measures, which may render the problem convex (see [Chizat and Bach, 2018]).

for $\mathbf{x} \in \mathbb{R}^{d_x}$, where $P : \mathbb{R}^d \to \mathbb{R}^m$ and $\mathrm{loss}(\cdot,\cdot) \in C^0(\mathbb{R}^m \times \mathcal{Y}; \mathbb{R}_+)$ are given – both will change with respect to the task in question (regression, classification), as discussed in Section 2.3.

For fixed $\lambda > 0$, in this section we will study the behavior when $T \gg 1$ of global minimizers $[w_T, b_T]$ to the functional

$$J_{\lambda,T}(w,b) = \mathcal{E}(\mathbf{x}(T)) + \lambda \Big\| [w,b] \Big\|^2_{H^k(0,T;\mathbb{R}^{d_u})} \tag{3.5}$$

where $\mathbf{x} \in C^0([0,T];\mathbb{R}^{d_x})$ is the unique solution to either (3.3) ($k = 0$) or (3.2) ($k = 1$) corresponding to the parameters $[w,b] \in H^k(0,T;\mathbb{R}^{d_u})$, noting (3.1).

3.1. **Non-negative losses.** We begin by considering the case wherein the function $(x,y) \mapsto \mathrm{loss}(Px,y)$ defining the empirical risk in (3.4) may attain its minimum 0. Namely, unless and until stated otherwise, we shall suppose

Assumption 3. *We suppose that $P : \mathbb{R}^d \to \mathbb{R}^m$ and $\mathrm{loss} \in C^0(\mathbb{R}^m \times \mathcal{Y}; \mathbb{R}_+)$ are such that $\mathcal{E}$ defined in (3.4) may attain its minimum 0.*

This is the case for several losses used in practice, including

- ℓ^p–losses: $\mathrm{loss}(x,y) = \|x-y\|^p_{\ell^p}$, $p \in \{1,2\}$, with $P : \mathbb{R}^d \to \mathbb{R}^m$ an affine map, or, more generally, losses which are radial functions with respect to $x - y$ (such as those induced by a distance inferred by a norm). Such modeling assumptions are typically made in the context of regression tasks ($\vec{y}_i \in \mathcal{Y} \subset \mathbb{R}^m$), wherein by minimizing the empirical risk one looks to interpolate the training data by means of the projected neural ODE flow.
- The *multi-class hinge loss*

$$\mathrm{loss}(x,y) = \sum_{\substack{j=1 \\ j \neq y}}^{m} \max\{0, 1 - x_y + x_j\} \qquad x \in \mathbb{R}^m,\ y \in [m],$$

  with $P : \mathbb{R}^d \to \mathbb{R}^m$ being a matrix, which may be used for classification tasks.

With regard to Assumption 3, we define the following notion of interpolation of the dataset $\{\vec{x}_i, \vec{y}_i\}_{i\in[N]}$.

Definition 3.1 (Interpolation)**.** *Let $P : \mathbb{R}^d \to \mathbb{R}^m$ be any non-zero map and let $\mathrm{loss} \in C^0(\mathbb{R}^m \times \mathcal{Y}; \mathbb{R}_+)$ be such that Assumption 3 is satisfied. We say that (3.3) (resp. (3.2)) interpolates the dataset $\{\vec{x}_i, \vec{y}_i\}_{i\in[N]}$ in some time $T > 0$ if there exists a time $T > 0$ and some parameters $[w,b] \in L^2(0,T;\mathbb{R}^{d_u})$ (resp. in $H^1(0,T;\mathbb{R}^{d_u})$) such that the unique solution $\mathbf{x}(\cdot)$ to (3.3) (resp. (3.2)), noting (3.1), satisfies*

$$\mathcal{E}(\mathbf{x}(T)) = 0.$$

Note that this is a slight abuse of terminology, as precisely interpolation would rather refer to having

$$P\mathbf{x}_i(T) = \vec{y}_i \qquad \text{for all} \quad i \in [N],$$

which is equivalent to the stated definition in the context of losses induced by a distance, but not in the context of the hinge loss. We shall, however, make use of the terminology for cohesion.

Let us also note that by means of an elementary time-scaling, if (3.3) or (3.2) interpolates the dataset in some time $T > 0$, it interpolates it in any time, in particular, in

time 1. We will make use of this observation to simplify the subsequent presentation and analysis by assuming interpolation in time 1 without loss of generality.

We may state our main result in this context.

Theorem 3.1. *Fix $\lambda > 0$. Suppose that $P : \mathbb{R}^d \to \mathbb{R}^m$ is any non-zero affine map, and suppose that* loss $\in C^0(\mathbb{R}^m \times \mathcal{Y}; \mathbb{R}_+)$ *is such that Assumption 3 is satisfied. Assume that (3.3) (resp. (3.2) with σ 1–homogeneous) interpolates the dataset $\{\vec{x}_i, \vec{y}_i\}_{i\in[N]}$ in time 1 in the sense of Definition 3.1. For any $T \geqslant 1$ let $[w_T, b_T] \in L^2(0, T; \mathbb{R}^{d_u})$ (resp. in $H^1(0, T; \mathbb{R}^{d_u})$) be any pair of global minimizers to $J_{\lambda,T}$ defined in (3.5), and let $\mathbf{x}_T(\cdot)$ be the unique associated solution to (3.3) (resp. (3.2)), noting (3.1). The following properties then hold.*

(i) There exists a constant $\mathfrak{C} = \mathfrak{C}\left(\{\vec{x}_i, \vec{y}_i\}_{i\in[N]}, \lambda\right) > 0$ independent of T such that

$$\mathcal{E}(\mathbf{x}_T(T)) \leqslant \frac{\mathfrak{C}}{T}.$$

(ii) There exists a sequence $\{T_n\}_{n=1}^{\infty}$, with $T_n > 0$ and $T_n \xrightarrow[n\longrightarrow\infty]{} \infty$, and some $\mathbf{x}_\circ \in \mathbb{R}^{d_x}$ with $\mathcal{E}(\mathbf{x}_\circ) = 0$ such that, along a subsequence,

$$\mathbf{x}_{T_n}(T_n) \xrightarrow[n\longrightarrow\infty]{} \mathbf{x}_\circ. \tag{3.6}$$

(iii) For any $n \geqslant 1$, set

$$\begin{aligned} w_n(t) &:= T_n\, w_{T_n}\,(t\, T_n) && \text{for } t \in [0,1], \\ b_n(t) &:= T_n\, b_{T_n}\,(t\, T_n) && \text{for } t \in [0,1]. \end{aligned}$$

Then along a subsequence,

$$\Big\| [w_n, b_n] - [w^*, b^*] \Big\|_{H^k(0,1;\mathbb{R}^{d_u})} \xrightarrow[n\longrightarrow\infty]{} 0,$$

where $[w^*, b^*] \in H^k(0, 1; \mathbb{R}^{d_u})$ is a solution to the minimization problem

$$\inf_{\substack{[w,b]\in H^k(0,1;\mathbb{R}^{d_u}) \\ \mathbf{x}(\cdot) \text{ solves (3.2) (resp. (3.3)) in } [0,1] \\ \text{and} \\ \mathcal{E}(\mathbf{x}(1)) = 0}} \Big\| [w, b] \Big\|^2_{H^k(0,1;\mathbb{R}^{d_u})}.$$

Idea of proof. The proof of Theorem 3.1 may be found in Section 7.1. Let us motivate the main underlying idea.

Under the above assumptions, both (3.2) and (3.3) will be 1–homogeneous with respect to the parameters $[w(t), b(t)]$. Namely, both (3.2) and (3.3) (noting (3.1)) can be written as

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathfrak{f}\left([w(t), b(t)], \mathbf{x}(t)\right) & \text{in } (0, T) \\ \mathbf{x}(0) = \mathbf{x}^0, \end{cases} \tag{3.7}$$

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

A simple computation (see Lemma 7.1) then leads to noting that, given some parameters $u^1 := [w^1, b^1]$ and the solution $\mathbf{x}^1$ to

$$\begin{cases} \dot{\mathbf{x}}^1(t) = \mathfrak{f}\left(\left[w^1(t), b^1(t)\right], \mathbf{x}^1(t)\right) & \text{in } (0, 1) \\ \mathbf{x}^1(0) = \mathbf{x}^0, \end{cases} \tag{3.8}$$

then $u^T(t) := \frac{1}{T}u^1(\frac{t}{T})$ for $t \in [0,T]$ is such that $\mathbf{x}^T(t) := \mathbf{x}^1(\frac{t}{T})$ solves (3.7). Under the interpolation assumption, we may find $u^1 \in H^k(0,T_0;\mathbb{R}^{d_u})$ such that the corresponding solution $\mathbf{x}^1$ satisfies $\mathcal{E}(\mathbf{x}^1(1)) = 0$, and then use the above scaling and the optimality of u_T to deduce

$$J_{\lambda,T}(u_T) \leqslant \mathcal{E}\left(\mathbf{x}^1(1)\right) + \frac{\lambda}{T}\left\|u^1\right\|^2_{H^k(0,1;\mathbb{R}^{d_u})} = \frac{\lambda}{T}\left\|u^1\right\|^2_{H^k(0,1;\mathbb{R}^{d_u})}$$

for $T \geqslant 1$. This will imply the decay estimate of $\mathcal{E}$, and combined with some more technical compactness arguments, will yield the remaining convergence results as well.

On another hand, considering the case of (3.3) and thus $k = 0$, we see that

$$\begin{aligned}
&\inf_{\substack{u_T=[w_T,b_T]\in L^2(0,T;\mathbb{R}^{d_u})\\ \mathbf{x}_T(\cdot)\text{ solves (3.7)}}} \mathcal{E}(\mathbf{x}_T(T)) + \lambda\int_0^T \|u_T(t)\|^2\,\mathrm{d}t\\
&= \inf_{\substack{u_T=[w_T,b_T]\in L^2(0,T;\mathbb{R}^{d_u})\\ \mathbf{x}_T(\cdot)\text{ solves (3.7)}}} \mathcal{E}(\mathbf{x}_T(T)) + \frac{\lambda}{T}\int_0^1 \|Tu_T(sT)\|^2\,\mathrm{d}s\\
&= \inf_{\substack{u^1=[w^1,b^1]\in L^2(0,1;\mathbb{R}^{d_u})\\ \mathbf{x}^1(\cdot)\text{ solves (3.8)}}} \mathcal{E}(\mathbf{x}^1(1)) + \frac{\lambda}{T}\int_0^1 \left\|u^1(s)\right\|^2\,\mathrm{d}s.
\end{aligned}$$

This computation indicates that one may consider the behavior when $T \longrightarrow \infty$ for fixed $\lambda > 0$ and that when $\lambda \searrow 0$ for fixed $T > 0$ in the same fashion. Although this scaling is specific to the L^2–regularization setting, it motivates completing Theorem 3.1 with the following result.

Theorem 3.2. *Under the assumptions of Theorem 3.1, fix $T > 0$, and for any $\lambda > 0$, let $[w_\lambda, b_\lambda] \in L^2(0,T;\mathbb{R}^{d_u})$ (resp. $H^1(0,T;\mathbb{R}^{d_u})$) be any pair of global minimizers to $J_{\lambda,T}$ defined in (3.5), and let $\mathbf{x}_\lambda$ be the unique associated solution to (3.3) (resp. (3.2)), noting (3.1). The following properties then hold.*

(i) There exists a constant $\mathfrak{C} = \mathfrak{C}\left(\{\vec{x}_i, \vec{y}_i\}_{i\in[N]}, T\right) > 0$ independent of $\lambda > 0$ such that

$$\mathcal{E}(\mathbf{x}_\lambda(T)) \leqslant \mathfrak{C}\,\lambda.$$

(ii) There exists a sequence $\{\lambda_n\}_{n=1}^\infty$, with $\lambda_n > 0$ and $\lambda_n \xrightarrow[n\to\infty]{} 0$, and some $\mathbf{x}_\circ \in \mathbb{R}^{d_x}$ with $\mathcal{E}(\mathbf{x}_\circ) = 0$ such that, along a subsequence

$$\mathbf{x}_{\lambda_n}(T) \xrightarrow[n\longrightarrow\infty]{} \mathbf{x}_\circ.$$

(iii) Moreover, along a subsequence,

$$\left\| [w_{\lambda_n}, b_{\lambda_n}] - [w^*, b^*] \right\|_{H^k(0,T;\mathbb{R}^{d_u})} \xrightarrow[n\longrightarrow\infty]{} 0,$$

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

$$\inf_{\substack{[w,b]\in H^k(0,T;\mathbb{R}^{d_u})\\ \mathbf{x}(\cdot)\text{ solves (3.2) (resp. (3.3))}\\ \text{and}\\ \mathcal{E}(\mathbf{x}(T))=0}} \left\|[w,b]\right\|^2_{H^k(0,T;\mathbb{R}^{d_u})}.$$

Remark 2 (Homogeneity). • *The impediment that appears in the results presented above when considering neural ODEs of the form*

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{w}^1(t)\sigma\left(\mathbf{w}^2(t)\mathbf{x}(t) + \mathbf{b}^2(t)\right) + \mathbf{b}^1(t) & \text{in } (0,T) \\ \mathbf{x}(0) = \mathbf{x}^0, \end{cases} \tag{3.9}$$

where $w^1(t) \in \mathbb{R}^{d\times d_{\text{hid}}}$ and $w^2 \in \mathbb{R}^{d_{\text{hid}}\times d}$, with $\mathbf{w}^1(t)$ and $\mathbf{w}^2(t)$ defined after stacking the states $\{\mathbf{x}_i(t)\}_{i\in[N]}$ within $\mathbf{x}(t)$ following convention (1.7), is the lack of homogeneity (and thus scaling) of the dynamics with respect to the parameters. Let us elaborate by focusing on $\mathbf{b}^2 \equiv \mathbf{b}^1 \equiv 0$ for simplicity. Let $\mathbf{x}_T$ denote the solution to (3.9) with parameters $[\mathbf{w}^1_T, \mathbf{w}^2_T]$. We set

$$\mathbf{x}_1(t) := \mathbf{x}_T(tT) \qquad \text{for } t \in [0,1],$$

and we see that, assuming σ is α–homogeneous with $\alpha > 0$, $\mathbf{x}_1$ solves (3.9) on $[0,1]$ with parameters

$$\left[w^1_1, w^2_1\right] := \left[T^p w^1_T(tT), T^{q/\alpha} w^2_T(tT)\right] \qquad \text{for } t \in (0,1),$$

with $p + q = 1$, noting (3.1). Solely looking at the squared L^2–norms of the parameters, we see that

$$\begin{aligned}\int_0^T \|w^1_T(t)\|^2 \,\mathrm{d}t + \int_0^T \|w^2_T(t)\|^2 \,\mathrm{d}t &= T\int_0^1 \left\|w^1_T(sT)\right\|^2 \mathrm{d}s + T\int_0^1 \left\|w^2_T(sT)\right\|^2 \mathrm{d}s \\ &= T^{1-2p}\int_0^1 \left\|w^1_1(s)\right\|^2 \mathrm{d}s + T^{1-2q/\alpha}\int_0^1 \left\|w^2_1(s)\right\|^2 \mathrm{d}s.\end{aligned}$$

We see that, in order to have a scaling factor which decays when $T \longrightarrow \infty$, a property which is a cornerstone of our proof, we would simultaneously need $1 - 2p < 0$ and $1 - \frac{2q}{\alpha} < 0$, i.e. $p > \frac{1}{2}$ and $q > \frac{\alpha}{2}$. Due to the constraint $p + q = 1$, this would entail $\alpha < 1$, which means that σ cannot be globally Lipschitz continuous.

• *Let us, in line with what is discussed just above, check what is the needed degree of homogeneity of σ in the case of System (3.2) to ensure that the results of Theorem 3.1 hold. Let us thus assume that σ is α–homogeneous with $\alpha > 0$. Then, setting $\mathbf{x}_1(t) := \mathbf{x}_T(tT)$ for $t \in [0,1]$ where $\mathbf{x}_T$ solves (3.2) with parameters $[w_T, b_T]$, we see that $\mathbf{x}_1(t)$ solves (3.2) on $[0,1]$ with parameters $[w_1(t), b_1(t)] = \left[T^{1/\alpha} w_T(tT), T^{1/\alpha} b_T(tT)\right]$ for $t \in (0,1)$. Then, looking at the L^2–norms of the parameters, we see that*

$$\begin{aligned}\int_0^T \|w_T(t)\|^2 \,\mathrm{d}t + \int_0^T \|b_T(t)\|^2 \,\mathrm{d}t &= T\int_0^1 \|w_T(sT)\|^2 \,\mathrm{d}s + T\int_0^T \|b_T(sT)\|^2 \,\mathrm{d}s \\ &= T^{1-2/\alpha}\int_0^1 \|w_1(s)\|^2 \,\mathrm{d}s + T^{1-2/\alpha}\int_0^1 \|b_1(s)\|^2 \,\mathrm{d}s,\end{aligned}$$

and thus, our methodology really needs $\alpha \in [1,2)$. Note that this brief computation also indicates the precise proximity of $\mathcal{E}(\mathbf{x}_T(T))$ to 0 for σ is α–homogeneous, which is of the order of $\mathcal{O}\left(T^{1-2/\alpha}\right)$.

Remark 3 (The output layer P). *We note that the output layer parameters given by the affine map P are fixed, but in general arbitrary. This is due to the fact that if we were to optimize P as well, we would have to ensure that the optimal P is bounded with respect to the limiting hyper-parameter (λ or T). This in turn could perhaps be ensured*

if we were to regularize the output layer as well, but would, in turn, be an impediment to the scaling arguments we use in all proofs since then the parameter regularization norm would not scale polynomially with T.

3.2. **Positive losses.** We henceforth consider the standard setting of classification tasks, wherein the labels $\vec{y}_i$ take values in a set of $m \geqslant 2$ classes – unless stated otherwise, we consider $\vec{y}_i \in [m]$ for all $i \in [N]$. We will focus on the cross-entropy loss in (3.4), which we recall, when evaluated along the neural ODE output, reads

$$\mathrm{loss}(P\mathbf{x}_i(T), \vec{y}_i) := -\log\left(\frac{e^{P\mathbf{x}_i(T)_{\vec{y}_i}}}{\sum_{j=1}^m e^{P\mathbf{x}_i(T)_j}}\right). \tag{3.10}$$

Here $P : \mathbb{R}^d \longrightarrow \mathbb{R}^m$ is an affine map, with more assumptions made precise later on.

An important feature of the cross-entropy loss is the fact that it is not coercive with respect to the first variable – namely, as $P\mathbf{x}_i(T)_{\vec{y}_i}$ goes to infinity, the loss goes to zero. In particular, the infimum 0 is never attained. This is in line with intuition regarding the classification task, as one looks to separate the features with respect to their individual class in the label space $\mathbb{R}^m$.

The problem consisting of classifying a given dataset is closely tied to the following rather intuitive notion of separability, which we will need in subsequent results.

Definition 3.2 (Separability)**.** *Let $P : \mathbb{R}^d \longrightarrow \mathbb{R}^m$ be any non-zero affine map. We say that (3.3) (resp. (3.2)) separates the dataset $\{\vec{x}_i, \vec{y}_i\}_{i\in[N]}$ with respect to P if there exists a time $T > 0$ and some parameters $[w, b] \in L^2(0, T; \mathbb{R}^{d_u})$ (resp. in $H^1(0, T; \mathbb{R}^{d_u})$) such that the unique solution $\mathbf{x}(\cdot)$ to (3.3) (resp. (3.2)) satisfies*

$$P\mathbf{x}_i(T)_{\vec{y}_i} - \max_{\substack{j\in[N]\\ j\neq\vec{y}_i}} P\mathbf{x}_i(T)_j > 0 \qquad \text{for all } i \in [N].$$

In other words, a parametrized neural ODE flow separates the given dataset if the corresponding *margin* $\gamma_{[w,b]}$, defined as

$$\gamma_{[w,b]} := \min_{i\in[N]}\left(P\mathbf{x}_i(T)_{\vec{y}_i} - \max_{\substack{j\in[N]\\ j\neq\vec{y}_i}} P\mathbf{x}_i(T)_j\right) \tag{3.11}$$

is positive. We may now state our main result in the classification context, which entails a quantitative rate of decay as $T \longrightarrow \infty$ of the training error with cross-entropy loss for ReLU activated neural ODEs.

Theorem 3.3. *Let $\{\vec{x}_i, \vec{y}_i\}_{i\in[N]}$ be a given dataset with $\vec{x}_i \in \mathbb{R}^d$ and $\vec{y}_i \in [m]$. Let $\lambda > 0$ be fixed, and let $\mathfrak{Q} : \mathbb{R}^{d_x} \longrightarrow \mathbb{R}^d$ be a non-zero affine map such that $\mathfrak{Q}\vec{x}_i \geqslant 0$ for $i \in [N]$. Set (recall convention (1.7))*

$$\mathbf{x}_i^0 := \mathfrak{Q}\vec{x}_i \qquad \text{for } \ i \in [N],$$

and let $P \in \mathbb{R}^{m\times d}$ be any non-zero matrix such that System (3.2), with $\sigma(x) = \max\{x, 0\}$, separates the dataset $\left\{\mathbf{x}_i^0, \vec{y}_i\right\}_{i\in[N]}$ with respect to P in some time $T_0 > 0$ as per Definition 3.2, and let γ denote the associated margin as defined in (3.11). For any $T \geqslant T_0$, let $[w_T, b_T] \in H^1(0, T; \mathbb{R}^{d_u})$ be any pair of global minimizers to $J_{\lambda,T}$ defined in (3.5)–(3.10), and let $\mathbf{x}_T(\cdot)$ be the associated unique solution to (3.2) with $\sigma(x) = \max\{x, 0\}$.

Then, there exists a constant $\mathfrak{C} = \mathfrak{C}\left(\{\vec{x}_i, \vec{y}_i\}_{i\in[N]}, \lambda\right) > 0$ *independent of* $T > 0$ *such that*

$$\mathcal{E}(\mathbf{x}_T(T)) \leqslant \log\left(1 + (m-1)e^{-\gamma\, e^{\frac{T^\alpha}{2}}}\right) + \mathfrak{C}\, T^{2\alpha-1} \tag{3.12}$$

holds for any $\alpha \in \left(0, \frac{1}{2}\right)$.

By using a Taylor expansion, one sees that the first term in the upper bound is negligible when $T \gg 1$. Thus, the training error $\mathcal{E}(\mathbf{x}_T(T))$ is at most of the order $\mathcal{O}\left(T^{2\alpha-1}\right)$.

We note that the above theorem is very specific to neural ODEs of the form (3.2) with ReLU activations, and the specific form of the cross-entropy loss, from which the first term in the estimate (3.12) is derived. This is due to the proof strategy, which relies on using the positivity of the right hand side to, in some sense, obtain a linear equation for the projected output features for some auxiliary parameters constructed within the proof, and thus have an explicit solution for these parameters of the form $\sim e^t$. This stimulates the appearance of the second exponential within the log in (3.12).

Much like what we observed in the setting of losses which attain their minimum, we can expect to link the limit as T goes to infinity with the convergence of the regularization path, namely the limit as $\lambda \searrow 0$. This is depicted in the following theorem.

Theorem 3.4. *Under the assumptions of Theorem 3.3, fix* $T \geqslant T_0$ *and for any* $\lambda > 0$ *let* $[w_\lambda, b_\lambda] \in H^1(0, T; \mathbb{R}^{d_u})$ *be any pair of global minimizers to* $J_{\lambda,T}$ *defined in* (3.5)–(3.10), *and let* $\mathbf{x}_\lambda(\cdot)$ *be the associated unique solution to* (3.2) *with* $\sigma(x) = \max\{x, 0\}$. *Then, there exists a constant* $\mathfrak{C} = \mathfrak{C}\left(\{\vec{x}_i, \vec{y}_i\}_{i\in[N]}, T\right) > 0$ *independent of* $\lambda > 0$ *such that*

$$\mathcal{E}(\mathbf{x}_\lambda(T)) \leqslant \log\left(1 + (m-1)e^{-\gamma\, e^{\frac{\lambda^{-\alpha}}{2}}}\right) + \mathfrak{C}\, \lambda^{-2\alpha+1}$$

holds for any $\alpha \in \left(0, \frac{1}{2}\right)$.

Remark 4 (Absence of margin convergence)**.** *We note that, unlike Theorem 3.1 (and Theorem 3.2), in Theorem 3.3 (and Theorem 3.4) we do not provide any result regarding the behavior of the optimal parameters* $[w_T, b_T]$ *as* $T \longrightarrow \infty$ *(resp.* $[w_\lambda, b_\lambda]$ *as* $\lambda \searrow 0$*). Let us elaborate on this absence in the case* $\lambda \searrow 0$.

- *In [Wei et al., 2019] (see also [Rosset et al., 2003]), the authors show that for the perceptron without bias*[5] $\Phi(x, u) := w^2\sigma(w^1 x)$, *where* σ *is, say,* 1*–homogeneous,* $u = \left[w^1, w^2\right]$ *with* $w^2 \in \mathbb{R}^{m\times d_{hid}}$ *and* $w^1 \in \mathbb{R}^{d_{hid}\times d}$, *any global minimizer* u_λ *to*

$$J_\lambda(u) := \frac{1}{N}\sum_{i=1}^{N} -\log\left(\frac{e^{\Phi(\vec{x}_i,u)_{\vec{y}_i}}}{\sum_{j=1}^{m} e^{\Phi(\vec{x}_i,u)_j}}\right) + \lambda\|u\|^2$$

is such that

$$\gamma_{\overline{u}_\lambda} \xrightarrow[\lambda\searrow 0]{} \gamma^*,$$

where $\overline{u}_\lambda = {}^{u_\lambda}/_{\|u_\lambda\|}$, $\gamma_{\overline{u}_\lambda}$ *is the margin of* $\Phi(\cdot, \overline{u}_\lambda)$ *as defined in* (3.11), *and* $\gamma^* > 0$ *is the max-margin:*

$$\gamma^* := \max_{\|u\|\leqslant 1} \gamma_u.$$

[5]In fact, the result holds for a multi-layer perceptron of arbitrary but fixed depth – we concentrate on this simpler scenario for presentational purposes.

A set of parameters u^* maximizing the margin are then called a max-margin *solution (here we focus on ℓ^2–margins; the ℓ^1–case has also been studied in the machine literature). The proof of this fact relies crucially on the specific form of the perceptron, as the map $u \longmapsto \Phi(x,u)$ is positively homogeneous (of degree 2) for any $x \in \mathbb{R}^d$. By using the exponential scaling of the cross entropy, $J_\lambda(u_\lambda)$ can be lower bounded roughly by $e^{-\|u_\lambda\|^2\gamma_\lambda}$, and also has an upper bound that scales with $e^{-\|u_\lambda\|^2\gamma^*}$. Again making use of the fact that $u \longmapsto \Phi(x,u)$ is positively homogeneous, one can ensure that $\|u_\lambda\| \longrightarrow \infty$ as $\lambda \searrow 0$, and then take $\|u_\lambda\|$ large enough so that $\gamma^* - \gamma_\lambda$ vanishes.*

In the context of neural ODEs such as (3.2) (and the corresponding ResNet, for that matter), to replicate such ideas, one would need to ensure that the map $[w,b] \longmapsto P\mathbf{x}(T)$, where $\mathbf{x}$ solves (3.2), is positively homogeneous (of some degree). In fact, let us see under which conditions the following relaxed problem, can be solved. We seek a non-decreasing continuous function $f : (0,\infty) \to (0,\infty)$ such that

$$\mathbf{x}_\alpha = f(\alpha)\mathbf{x} \qquad \text{for all } \alpha > 0,$$

where $\mathbf{x}$ solves (3.2) with parameters $[\mathbf{w}, 0]$ (noting (3.1)) and $\mathbf{x}_\alpha$ solves (3.2) with parameters $[\alpha\mathbf{w}, 0]$. If such an f were to exist, then by differentiating one sees that

$$\dot{\mathbf{x}}_\alpha(t) = f(\alpha)\dot{\mathbf{x}}(t) = f(\alpha)\sigma(\mathbf{w}(t)\mathbf{x}(t)).$$

Using the fact that $\dot{\mathbf{x}}_\alpha(t)$ solves (3.2) with parameters $[\alpha\mathbf{w}, 0]$, we moreover see that

$$\sigma(\alpha\mathbf{w}(t)f(\alpha)\mathbf{x}(t)) = f(\alpha)\sigma(\mathbf{w}(t)\mathbf{x}(t)).$$

Now we see that the above relation translates to having

$$\sigma(\alpha f(\alpha)s) = f(\alpha)\sigma(s) \qquad \text{for all } \ \alpha > 0,\ s \in \mathbb{R}. \tag{3.13}$$

This implies $\sigma(0) = 0$, and would also be an obstruction to the Lipschitz continuity of σ at $x = 0$ (indeed, fix $s \in \mathbb{R}$ such that $\sigma(s) \neq 0$, and let $\alpha \searrow 0$). Thus, the only increasing function $f : (0,\infty) \to (0,\infty)$ such that (3.13) holds and $\sigma \in \mathrm{Lip}(\mathbb{R})$ is the zero function.

- *The issue presented just above does not occur when the only trainable parameter is the additive bias $\mathbf{b}(t)$. Indeed, if for instance $\mathbf{w}(t) = \mathbf{w}_\circ = \mathrm{diag}(w_\circ)$ is fixed, $\mathbf{x}(t)$ solves (3.2) with parameters $[\mathbf{w}_\circ, \mathbf{b}(t)]$ and $\mathbf{x}_\alpha(t)$ solves (3.2) with parameters $[\mathbf{w}_\circ, \alpha\mathbf{b}(t)]$, then one readily sees that $P\mathbf{x}_{\alpha,i}(t) = \alpha P\mathbf{x}_i(t)$ for all $\alpha > 0$ whenever $P\vec{x}_i = 0$ for $i \in [N]$ (this is the case when augmenting the data $\vec{x}_i$ and projecting orthogonally onto the added euclidean space via P). Hence, one may show, by adapting the techniques of [Wei et al., 2019], that the margin $\gamma_{[w_\circ,\bar{b}_T]}$ corresponding to normalized global minimizers $\bar{b}_T = b_T/\|b_T\|$, converges to the max-margin*

$$\gamma^* := \sup_{\|b\|_{H^1(0,1;\mathbb{R}^{d_u})} \leqslant 1} \gamma_{[w_\circ,\bar{b}]}$$

as $T \longrightarrow \infty$ (with an analog result when $\lambda \searrow 0$ and T is fixed). We omit the statement and proof of this result due to the possible limitations of not optimizing the weights in practical contexts.

In view of this discussion, one sees that in order to deduce what the limit for the margin in the fully trained neural ODE (and ResNet) setting would be, one would likely have to conceive a different strategy to that of [Rosset et al., 2003; Wei et al., 2019] which relies on the homogeneity of the parameter to output map.

Remark 5 (Different losses)**.** *We note that in the context of binary classification, one could consider $\vec{y}_i \in \{-1, 1\}$ and train with the logistic loss in (3.4)*

$$\mathrm{loss}(P\mathbf{x}_i(T), \vec{y}_i) := \log\left(1 + e^{-\vec{y}_i P\mathbf{x}_i(T)}\right),$$

where $P : \mathbb{R}^d \to \mathbb{R}$ is an affine map. The statements of Theorem 3.3 and Theorem 3.4 also hold in the context of the logistic loss defined above by a straightforward repetition of our proofs.

Another class of losses when $\vec{y}_i \in [m]$ is the purely exponential loss

$$\mathrm{loss}(P\mathbf{x}_i(T), \vec{y}_i) := e^{-\vec{y}_i P\mathbf{x}_i(T)}.$$

Note that, again, by a slight adaptation of our techniques, one can show an analog variant of Theorem 3.3 (and Theorem 3.4) with an estimate of the form

$$\mathcal{E}(\mathbf{x}_T(T)) \leqslant e^{-\gamma e^{\frac{T^\alpha}{2}}} + \mathfrak{C}\, T^{2\alpha-1}.$$

4. Augmented empirical risk minimization

We are now interested in seeing whether one can obtain better quantitative estimates for the decay of the training error $\mathcal{E}$ to 0 with respect to the time horizon ($\sim$ number of layers) $T > 0$ – namely, improve the $\mathcal{O}\left(\frac{1}{T}\right)$–rate of convergence of the training error to 0 manifested in Theorem 3.1 and Theorem 3.3. We provide a proof of exponential stability in the setting of an augmented supervised learning problem, concentrating solely on L^2–parameter regularization (and hence we shall assume the existence of a minimizer in the setting of (3.2)). We shall also consider a BV–parameter regularization, which suffices to ensure the existence of minimizers, but only provide a polynomial convergence of the averages of the training error and optimal parameters to zero as $T \longrightarrow \infty$, whilst ensuring a uniform bound on the total variation of the optimal parameters.

Unless stated otherwise, we will henceforth solely concentrate on empirical risks

$$\mathcal{E}(\mathbf{x}) := \frac{1}{N}\sum_{i=1}^{N} \mathrm{loss}(P\mathbf{x}_i, \vec{y}_i) \qquad \text{for } \ \mathbf{x} \in \mathbb{R}^{d_x}, \tag{4.1}$$

induced by a loss satisfying the following condition.

Assumption 4. *We assume that $\mathcal{Y} \subset \mathbb{R}^m$ and $\mathrm{loss} \in C^0(\mathbb{R}^m \times \mathcal{Y}; \mathbb{R}_+)$ is such that there exist constants $c > 0$ (possibly depending on m) and $\alpha > 0$ such that*

$$\mathrm{loss}(x, y) \leqslant c\, \|x - y\|_{\ell^2}^{\alpha} \qquad \textit{for } \ x, y \in \mathbb{R}^m.$$

Typical examples of such losses are $\mathrm{loss}(x, y) = \|x - y\|_{\ell^p}^p$ for $p \geqslant 1$ and $\mathcal{Y} \subset \mathbb{R}^m$; note however that this assumption excludes the log-loss and the cross-entropy loss. In (4.1), $P \in \mathrm{Lip}(\mathbb{R}^d; \mathbb{R}^m)$ is any given surjective and non-zero map, which, in the context of regression tasks, is additionally an affine map, while in the context of binary classification, e.g. $\vec{y}_i \in \mathcal{Y} = \{-1, 1\}$, may be an affine map composed with a thresholding nonlinearity with range $[-1, 1]$.

Inspired from insights in optimal control theory, for fixed $\lambda > 0$, we will study the behavior when $T \gg 1$ of global minimizers to the functional

$$J_T(w,b) := \mathcal{E}(\mathbf{x}(T)) + \frac{1}{N}\int_0^T \|\mathbf{x}(t) - \overline{\mathbf{x}}\|^2 \,\mathrm{d}t + \lambda \Big\|[w,b]\Big\|^2_{L^2(0,T;\mathbb{R}^{d_u})}, \tag{4.2}$$

with $\mathcal{E}$ as in (4.1)–Assumption 4, and where $\overline{\mathbf{x}}_i \in P^{-1}(\{\vec{y}_i\})$ for all $i \in [N]$ are arbitrary but fixed.

We note that, contrary to the case where we minimize the training error at the final time T, here the same scaling does not appear, which allows us to deduce an equivalence with $\lambda \longrightarrow 0$. Hence, we will solely be interested in the behavior when $T \gg 1$. In fact, we ought to expect a result of slightly different nature to Theorem 3.1. This is because the trajectory tracking term introduces a stronger time-scale in the behavior of the optimization problem as $T \gg 1$. To see this, consider the neural ODE (3.3) for simplicity, and note that

$$\begin{aligned}
&\inf_{\substack{u_T \in L^2(0,T;\mathbb{R}^{d_u}) \\ \mathbf{x}_T \text{ solves } (3.7)}} \mathcal{E}(\mathbf{x}_T(T)) + \frac{1}{N}\int_0^T \|\mathbf{x}_T(t) - \overline{\mathbf{x}}\|^2 \,\mathrm{d}t + \lambda \int_0^T \|u_T(t)\|^2 \,\mathrm{d}t \\
&= \inf_{\substack{u_T \in L^2(0,T;\mathbb{R}^{d_u}) \\ \mathbf{x}_T \text{ solves } (3.7)}} \mathcal{E}(\mathbf{x}_T(T)) + \frac{T}{N}\int_0^1 \left\|\mathbf{x}_T\left(\frac{s}{T}\right) - \overline{\mathbf{x}}\right\|^2 \mathrm{d}s + \frac{\lambda}{T}\int_0^1 \|T u_T(sT)\|^2 \,\mathrm{d}s \\
&= \inf_{\substack{u^1 \in L^2(0,1;\mathbb{R}^{d_u}) \\ \mathbf{x}^1 \text{ solves } (3.8)}} \mathcal{E}\left(\mathbf{x}^1(1)\right) + \frac{T}{N}\int_0^1 \|\mathbf{x}^1(s) - \overline{\mathbf{x}}\|^2 \,\mathrm{d}s + \frac{\lambda}{T}\int_0^1 \left\|u^1(s)\right\|^2 \mathrm{d}s.
\end{aligned} \tag{4.3}$$

We see that, unlike Theorem 3.1, the trajectory tracking term in (4.3) carries significance when $T \gg 1$.

We will require the following *controllability* notion, which is rather natural in the context of the result that follows.

Definition 4.1 (Controllability with linear cost)**.** *We say that*

- *System* (3.3) *is* controllable with linear cost *if for any $T > 0$ and $r > 0$ there exists a constant $\mathfrak{C}(T,r) > 0$ such that for any $\mathbf{x}^0 \in \mathbb{R}^{d_x}$ and $\mathbf{x}^1 \in \mathbb{R}^{d_x}$ satisfying $\left\|\mathbf{x}^0 - \mathbf{x}^1\right\| \leqslant r$, there exists a pair of parameters $[w,b] \in L^2(0,T;\mathbb{R}^{d_u})$ for which the corresponding unique solution $\mathbf{x}(\cdot)$ to* (3.3)*, noting* (3.1)*, satisfies $\mathbf{x}(T) = \mathbf{x}^1$, and*

$$\Big\|[w,b]\Big\|_{L^2(0,T;\mathbb{R}^{d_u})} \leqslant \mathfrak{C}(T,r)\left\|\mathbf{x}^0 - \mathbf{x}^1\right\| \tag{4.4}$$

  holds.

- *System* (3.2) *is* controllable with linear cost *if for any $T > 0$ and $r > 0$ there exists a constant $\mathfrak{C}(T,r) > 0$ such that for any $\mathbf{x}^0 \in \mathbb{R}^{d_x}$ and $\mathbf{x}^1 \in \mathbb{R}^{d_x}$ satisfying $\left\|\mathbf{x}^0 - \mathbf{x}^1\right\| \leqslant r$, there exists a pair of parameters $[w,b] \in C^0([0,T];\mathbb{R}^{d_u}) \cap \mathrm{BV}(0,T;\mathbb{R}^{d_u})$ for which the corresponding unique solution $\mathbf{x}(\cdot)$ to* (3.2)*, noting* (3.1)*, satisfies $\mathbf{x}(T) = \mathbf{x}^1$, and*

$$\Big\|[w,b]\Big\|_{L^\infty(0,T;\mathbb{R}^{d_u})} + \Big\|[w,b]\Big\|_{\mathrm{BV}(0,T;\mathbb{R}^{d_u})} \leqslant \mathfrak{C}(T,r)\left\|\mathbf{x}^0 - \mathbf{x}^1\right\| \tag{4.5}$$

  holds.

We refer to Theorem 5.1 for further analysis and comments regarding Definition 4.1, in particular regarding the amplitude estimates (4.4) – (4.5). Note that (4.5) is a weaker condition than assuming an $H^1(0,T;\mathbb{R}^{d_u})$–estimate due to the Sobolev embedding. On another hand, due to the homogeneity of the dynamics, one readily sees (as per Lemma 7.1) that the controllability assumption holds in any time $T > 0$ if it holds in some time $T_0 > 0$.

We refer to (4.4) – (4.5) as *linear estimates on the cost* of $[w, b]$ as, in addition to the linear growth with respect to $\|\mathbf{x}^0 - \mathbf{x}^1\|$ such estimates are a hallmark of linear controlled dynamical systems of the form $\dot{x}(t) = Ax(t) + Bu(t)$, where $u(t) \in \mathbb{R}^m$ is the control, $x(t) \in \mathbb{R}^n$ is the state, with $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times m}$ being given matrices (see [Zuazua, 2007]).

We are in a position to state our main result in the context of the augmented supervised learning problem consisting of minimizing (4.2).

Theorem 4.1 (Exponential stability)**.** *Fix $\lambda > 0$, let $P \in \mathrm{Lip}(\mathbb{R}^d;\mathbb{R}^m)$ be any given non-zero surjective map, and let $\overline{\mathbf{x}} \in \mathbb{R}^{d_x}$ with $\overline{\mathbf{x}}_i \in P^{-1}(\{\vec{y}_i\})$ for $i \in [N]$ be arbitrary but fixed. Suppose that system (3.3) (resp. (3.2) with σ 1–homogeneous) is controllable with linear cost in the sense of Definition 4.1. Then, there exists $T^* > 0$ and constants $\mathfrak{C} = \mathfrak{C}\left(\{\vec{x}_i, \vec{y}_i\}_{i\in[N]}, \lambda, N, \alpha, P\right) > 0$ and $\mu = \mu\left(\{\vec{x}_i, \vec{y}_i\}_{i\in[N]}, \lambda, N, \alpha\right) > 0$ such that for any $T \geqslant T^*$, any pair of parameters $[w_T, b_T] \in L^2(0,T;\mathbb{R}^{d_u})$ minimizing J_T defined in (4.2), and the corresponding unique solution $\mathbf{x}_T(\cdot)$ to (3.3) (resp. (3.2)) satisfy*

$$\|w_T(t)\| + \|b_T(t)\| \leqslant \mathfrak{C}\, e^{-\mu t}$$

for a.e. $t \in [0,T]$ and

$$\mathcal{E}(\mathbf{x}_T(t)) + \|\mathbf{x}_T(t) - \overline{\mathbf{x}}\| \leqslant \mathfrak{C}\, e^{-\mu t}$$

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

The convergence/stability rate entailed by Theorem 4.1 is not only noticeably stronger than those in Theorem 3.1 and Theorem 3.3, but the estimate holds in any time $t \in [0,T]$ (i.e., at every layer when viewed from the ResNet perspective) and not only for the output features.

Theorem 4.1 can be proven by following the framework presented[6] in [Esteve et al., 2020]. We provide most of the arguments in Section 7.4 due to technical changes in the underlying model.

Remark 6 (Existence of minimizers, BV–regularization)**.** *In Theorem 4.1 we have subjacently assumed the existence of a minimizer of (4.2) for the neural ODE (3.2). By solely regularizing the L^2–norm of the parameters, we are, a priori, not aware how to ensure the strong convergence in L^1 of a minimizing sequence of parameters, which would suffice for applying the direct method in the calculus of variations. On the other hand, using a Sobolev H^1–regularization would entail that optimizable parameters are continuous (by the Sobolev embedding) and thus render our specific proof strategy incompatible, as we rely on constructing discontinuous suboptimal parameters by zero extensions*

[6]As discussed in the introduction, Theorem 4.1 is a specific manifestation of the so-called *turnpike property*, a paradigm dating back to the works of John von Neumann ([von Neumann, 1945]), and works in economics by Paul Samuelson et al. ([Dorfman et al., 1958]). A turnpike theory under smallness assumptions on the inputs and/or targets, combining the Pontryagin Maximum Principle, linearization arguments and precise estimates on Riccati equations, and covering a wide variety of nonlinear optimal control problems is developed in [Trélat and Zuazua, 2015] – with extensions to Lipschitz nonlinearities and avoiding smallness conditions found in [Esteve et al., 2020].

In view of this, in the setting of (3.2) *we may also make use of a* BV*–regularization rather than* H^1*, as while either of these regularizations suffice to guarantee compactness of the flow, which is sufficient to ensure the existence of a minimizer, the* BV*–regularization allows for discontinuous parameters: one would then rather consider*

$$J_T(w,b) := \mathcal{E}(\mathbf{x}(T)) + \frac{1}{N}\int_0^T \|\mathbf{x}(t)-\overline{\mathbf{x}}\|^2\,\mathrm{d}t + \lambda\Big\|[w,b]\Big\|^2_{\mathrm{BV}(0,T;\mathbb{R}^{d_u})}. \tag{4.6}$$

Taking the above remark into consideration, we are able to prove the following partial stability and convergence result in the setting of BV–regularized parameters. We focus on (3.2), since (3.3) can be covered by means of L^2–regularization.

Theorem 4.2 (Convergence of averages, stability)**.** *Fix* $\lambda > 0$*, let* $P \in \mathrm{Lip}(\mathbb{R}^d;\mathbb{R}^m)$ *be any given non-zero surjective map, and let* $\overline{\mathbf{x}} \in \mathbb{R}^{d_x}$ *with* $\overline{\mathbf{x}}_i \in P^{-1}(\{\vec{y}_i\})$ *for* $i \in [N]$ *be arbitrary but fixed. Suppose that* (3.2)*, with* σ 1*–homogeneous, is controllable with linear cost in the sense of Definition 4.1. The following facts then hold.*

(i) (Stability). There exists a constant $\mathfrak{C} = \mathfrak{C}\left(\{\vec{x}_i,\vec{y}_i\}_{i\in[N]}, \left\|\mathbf{x}^0-\overline{\mathbf{x}}\right\|, \lambda, N, P\right) > 0$ *such that for any* $T \geqslant 1$*, any pair of parameters* $[w_T, b_T] \in \mathrm{BV}(0,T;\mathbb{R}^{d_u})$ *minimizing* J_T *defined in* (4.6)*, and the corresponding unique solution* $\mathbf{x}_T(\cdot)$ *to* (3.2) *satisfy*

$$\mathcal{E}(\mathbf{x}_T(t)) + \|\mathbf{x}_T(t)-\overline{\mathbf{x}}\| + \Big|\mathrm{D}\,[w_T,b_T]\,\Big|(0,T) \leqslant \mathfrak{C}$$

for all $t \in [0,T]$*, where* $|\mathrm{D}\,[w_T,b_T]|\,(0,T)$ *denotes the total variation of* $[w_T,b_T]$*.*

(ii) (Convergence of averages). Any pair of parameters $[w_T,b_T] \in \mathrm{BV}(0,T;\mathbb{R}^{d_u})$ *minimizing* J_T *defined in* (4.6)*, and the corresponding unique solution* $\mathbf{x}_T(\cdot)$ *to* (3.2) *also satisfy*

$$\frac{1}{T}\int_0^T \mathcal{E}(\mathbf{x}_T(t))\,\mathrm{d}t + \frac{1}{T}\int_0^T \|\mathbf{x}(t)-\overline{\mathbf{x}}\|\,\mathrm{d}t + \frac{1}{T}\int_0^T \Big\|[w_T(t),b_T(t)\Big\|\,\mathrm{d}t \xrightarrow[T\longrightarrow\infty]{} 0.$$

The convergence of averages are very much related to the so-called *integral turnpike* property, which has been studied in some optimal control contexts (see e.g. [Trélat and Zhang, 2018]). Note that, in addition to these convergences, the stability estimates of item *(i)* entail that the oscillations of the training error may be controlled, in any time t, uniformly with respect to the time horizon T. Moreover, the total variation of the optimal parameters is also uniformly bounded with respect to T.

We are unable to provide exponential stability estimates as in Theorem 4.1 due to some constructions specific to the proof of Theorem 4.1, wherein suboptimal parameters with jumps at time instances are constructed. While the L^2–norm does not see these jump singularities, the BV–norm does and renders our strategy incompatible.

Remark 7 (Extensions)**.** *We only stated Theorem 4.1 for neural ODEs of the form* (3.3)*, or* (3.2) *with* σ 1*–homogeneous. This is solely to guarantee the exponential stability estimate of the optimal parameters, for which our proof requires using the underlying scaling endowed by the homogeneity of the dynamics. But in fact, the exponential stability estimate of the training error* $\mathcal{E}(\mathbf{x}_T(T))$ *and a uniform-in-*T *bound of the optimal parameters can be shown for more complicated neural ODE dynamics such as* (3.9)*, solely by a small adaptation of the proof.*

Remark 8 (Dependence on N)**.** *All of the results presented in what precedes hold for a fixed but otherwise arbitrary number of data samples N. In Theorem 4.1, one also notes an explicit dependence of the constants $\mathfrak{C} > 0$ and $\mu > 0$ on N – in fact, both constants might have a tendency to depend in an exponential manner with respect to N due to the subjacent application of a Grönwall inequality for the stacked neural ODE system, for which the parameters are pasted N times. The dependence on N provided in our proof might not be sufficient or sharp for studying a possible large data limit (e.g., via a law of large numbers argument of some kind) – we leave this issue open.*

Example 4.1 (A numerical experiment)**.** *In Figure 1 – Figure 2, we depict*[7] *a manifestation of the exponential stability insinuated by Theorem 4.1 on a toy binary classification task (namely $\vec{y}_i \in \{-1, 1\}$) with $N = 2400$ training samples and 600 test samples, where $P(\cdot) = \tanh(p_1 \ \cdot + p_2)$ with $p_1 \in \mathbb{R}^{1\times 2}$ and $p_2 \in \mathbb{R}$ randomly sampled from a uniform distribution on $[0, 10]$. Note that, while in theory $\tanh : \mathbb{R} \to [-1, 1]$ is not surjective (it is only bijective onto $(-1, 1)$), we use it as a thresholding nonlinearity for simplicity, as of course numerically the lack of surjectivity does not appear due to floating point accuracy.*

To discretize the full continuous-time optimization problem, we use direct shooting, which is a first discretize then optimize *approach. We consider the neural ODE (3.3) with $\sigma(x) = \tanh(x)$ (we use the ResNet (2.3)), with $T = 15$ (and thus 15 layers) and $\lambda = 0.01$. Finally, we discretize the integrals using an elementary trapezoidal quadrature. We note that the learned flow has a distinctly simple variation in Figure 2, and, albeit on a toy task, we observe satisfactory generalization properties in Figure 3.*

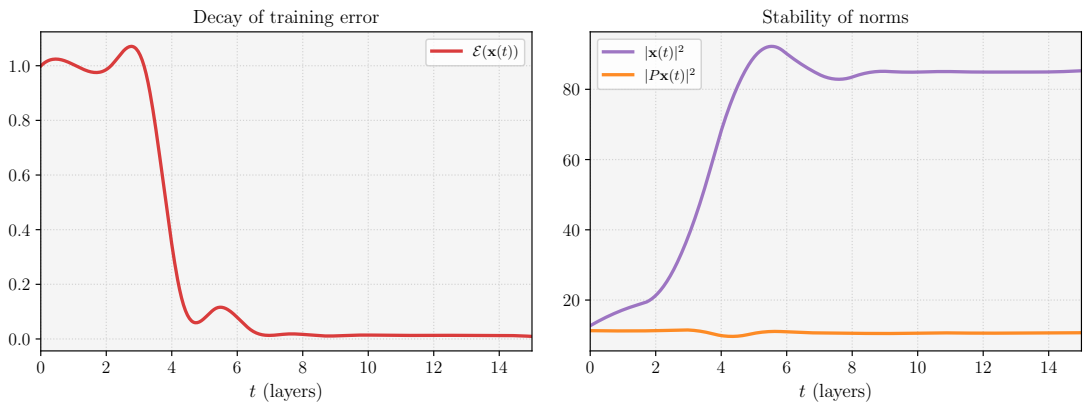


FIGURE 1. **Example 4.1:** We depict a manifestation of the stability results of Theorem 4.1 for the state $\mathbf{x}_T(t)$ (*right*) and the training error $\mathcal{E}(\mathbf{x}_T(t))$ (*left*) over $t \in [0, T]$. We observe that, after a finite time, the training error and trajectory remain at a steady configuration, so further times/layers could be discarded from training.

[7]Software experiments were done using `PyTorch` [Paszke et al., 2017] (and may be found at `https://github.com/borjanG/dynamical.systems`), using the Adam optimizer [Kingma and Ba, 2014] with learning rate equal to 10^{-3}. 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).

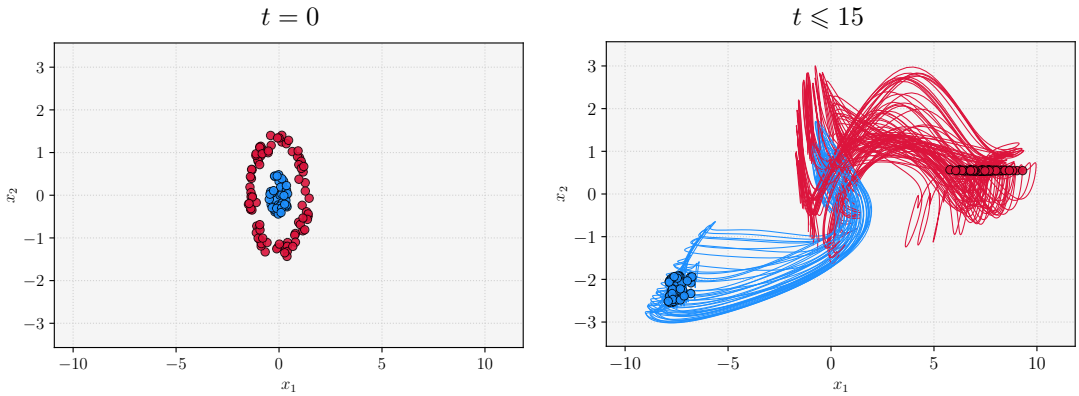
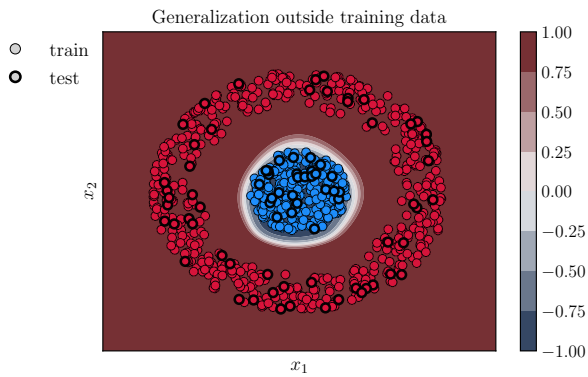


FIGURE 2. **Example 4.1:** A batch of training data (*left*) and the evolution of the corresponding trained trajectories $\mathbf{x}_{T,i}(t)$ (*right*) in the phase plane. The learned flow is simple, with moderate variations, due to the exponentially small parameters. We refer to https://github.com/borjanG/dynamical.systems/blob/master/videos/example4-1.mp4 for a movie of the evolution of the trajectories, where the stability phenomenon depicted in Figure 1 can be visualized.

FIGURE 3. **Example 4.1:** The trained classifier on $[-2, 2]^2$ and evaluated on a batch of the test set. The simplicity of the learned flow ensures relatively satisfactory generalization as the shape of the dataset is learned adequately, and the test set is correctly classified.



4.1. **The motivating problem.** Due to the specific nature of the proof of Theorem 4.1 we have restricted our study to a trajectory tracking term consisting of the squared $L^2(0, T; \mathbb{R}^{d_x})$–norm, even-though the final cost $\mathcal{E}(\mathbf{x}_T(T))$ allows us to address both classification and regression tasks. However, having to look for targets $\overline{\mathbf{x}}$ in the preimage of the labels $\vec{y}_i$ by P for any general task may not scale computationally and might bias the prediction.

Interestingly enough, at least numerically (any analytical result remains an open problem), we observe that the stabilization phenomenon persists (although, in theory, perhaps not with the same rate) when the term $\|\mathbf{x}(t) - \overline{\mathbf{x}}\|^2$ is replaced by the training error $\mathcal{E}(\mathbf{x}(t))$ with a general and possibly non-coercive loss, for instance, the cross-entropy loss on a multi-label classification tasks as seen in Figure 7 & Figure 13. In fact, one could stipulate that this stabilization phenomenon (be it exponential or not, but in any $t \in [0, T]$ rather than just for the output features at time T) holds for global minimizers

of functionals of the form

$$J_T(w,b) := \int_0^T \mathcal{E}(\mathbf{x}(t))\,\mathrm{d}t + \lambda \Big\| [w,b] \Big\|^2_{\mathcal{H}(0,T;\mathbb{R}^{d_u})}, \tag{4.7}$$

with $\mathcal{E}$ as in (3.4) and loss being continuous and nonnegative, but otherwise arbitrary; $\mathcal{H}$ is for instance L^2 or BV. We perform several numerical experiments to justify[8] this claim (see Example 4.3 – Example 4.6).

Note that in these experiments, even-though the loss is taken as cross-entropy and is thus not coercive, in addition to a stability property for $\mathcal{E}(\mathbf{x}(t))$ to 0 we also see (e.g. in Figure 7) that the trajectories $\mathbf{x}(t)$ and features $\{P\mathbf{x}_i(t)\}_{i\in[N]}$ stabilize towards some targets in sufficiently large time. However, due to the fact that $\mathcal{E}$ does not attain its minimizer, it is a priori not clear how one may characterize the targets to which $\mathbf{x}(t)$ and features $\{P\mathbf{x}_i(t)\}_{i\in[N]}$ stabilize. We refer to [Yagüe and Geshkovski, 2021] for a result illustrating a similar stability phenomenon in the setting of (4.7) with L^1–parameter regularization.

Example 4.2 (Concentric spheres)**.** *For comparison purposes, we consider an identical dataset setting to that of Figure 1 – Figure 3. We consider the neural ODE* (3.3) *with* $\sigma \equiv \tanh$*, cross-entropy loss and* $\lambda = 0.01$*, with the output layer having the form* $Px = p_1 x + p_2$*, with* $p_1 \in \mathbb{R}^{2\times 2}, p_2 \in \mathbb{R}^2$ *both being part of the trainable parameters. We visualize the output of the experiments in Figure 4 – Figure 6 below.*

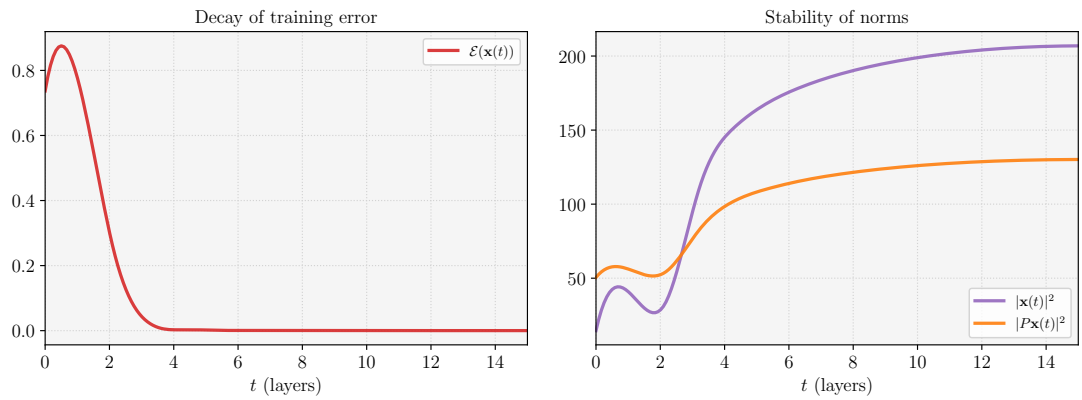


FIGURE 4. **Example 4.2:** The decay of the training error (*left*) and stabilization of the trained trajectories $\mathbf{x}(t)$ and $\{P\mathbf{x}_i(t)\}_{i\in[N]}$ (*right*).

Example 4.3 (Annuli in three colors)**.** *We consider a toy classification task with three labels, namely* $\vec{y}_i \in [3]$*, with each label corresponding to a different color. The dataset consists of* $N = 3200$ *training samples and* 800 *test samples. We consider the cross-entropy loss* (3.10) *in the training error in* (4.7) *and* $\lambda = 0.01$*, and we consider the neural ODE* (3.3) *with* $T = 15$ *(we use a forward Euler scheme to obtain a corresponding ResNet with fixed time-step equal to* 1*), where* $\sigma \equiv \tanh$*. No augmentation of the initial*

[8]We do not insinuate that our numerical experiments are comparable with state of the art configurations, as we only look to motivate and illustrate the mathematical phenomena studied in what precedes. Indeed, we generally make use of a forward Euler scheme compared to more advanced adaptive schemes, as used for instance in [Chen et al., 2018]. Our experiments should rather be seen as proof of concept.

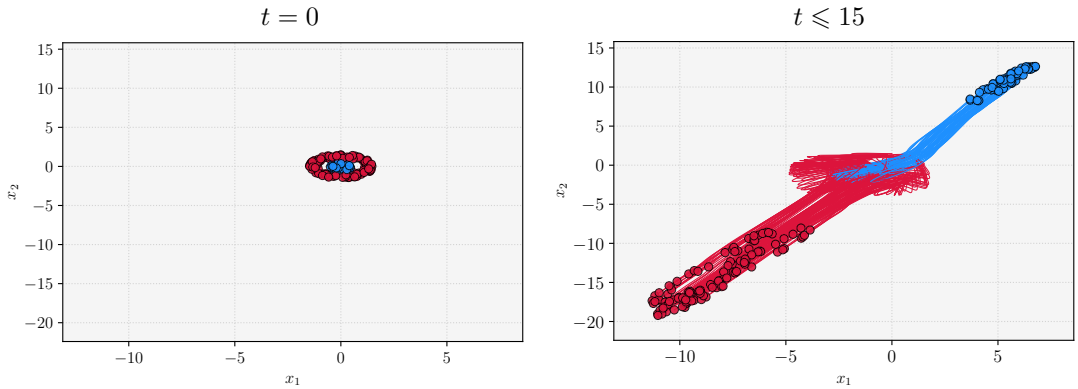


FIGURE 5. **Example 4.2:** A batch of training data (*left*) and the evolution of the corresponding trained trajectories $\mathbf{x}_{T,i}(t)$ (*right*). See https://github.com/borjanG/dynamical.systems/blob/master/videos/exampleA-1.mp4 for a movie of the evolution of the trajectories, where the stability phenomenon depicted in Figure 4 can be visualized.

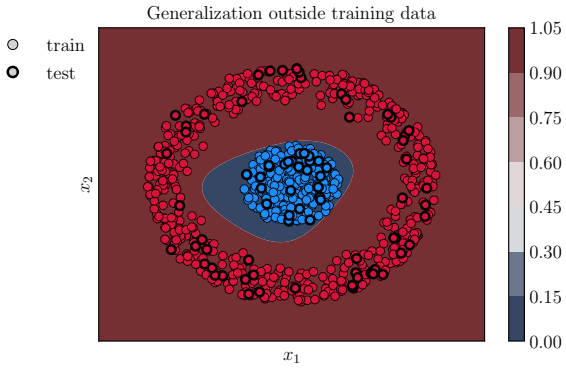


FIGURE 6. **Example 4.2:** The trained classifier on $[-2, 2]^2$ and its evaluation on a batch of the test data. The shape of the data is captured accurately.

data is used, and the trajectories evolve in the ambient dimension $d = 2$. The output layer is parametrized by $Px = p_1 x + p_2$, where $p_1 \in \mathbb{R}^{3\times 2}$, $p_2 \in \mathbb{R}^3$ are part of the trainable parameters. We display the results of the experiments in Figure 7 – Figure 9.

Example 4.4 (XOR)**.** *We consider a canonical binary classification task ($\vec{y}_i \in [2]$) – the XOR dataset (Figure 11) consisting of $N = 3200$ training samples and 800 test samples. To further illustrate the genericity of the stability phenomenon, we now consider the neural ODE (3.9), with $T = 15$ (we use a forward Euler scheme to obtain a corresponding ResNet with fixed time-step equal to 1), where $\sigma \equiv \tanh$ and $d_{hid} = 3$. We again consider cross-entropy loss in the empirical risk $\mathcal{E}$ defined in (4.7) and $\lambda = 0.01$, and we focus solely on L^2–parameter regularization (for simplicity). No augmentation of the initial data is used, and the trajectories evolve in the ambient dimension $d = 2$. The output layer is defined as $Px = p_1 x + p_2$, with $p_1 \in \mathbb{R}^{2\times 2}$, $p_2 \in \mathbb{R}^2$ both being part of the trainable parameters. We display the results in Figure 10 – Figure 12.*

Example 4.5 (MNIST)**.** *We now show that the stabilization phenomenon may also be observed on more realistic datasets such as MNIST [LeCun et al., 2010]. MNIST is a dataset consisting of handwritten digits ranging from 0 to 9, with a training set consisting of 60000 samples, and a test set consisting of 10000 samples. Each input sample $\vec{x}_i$ is a*

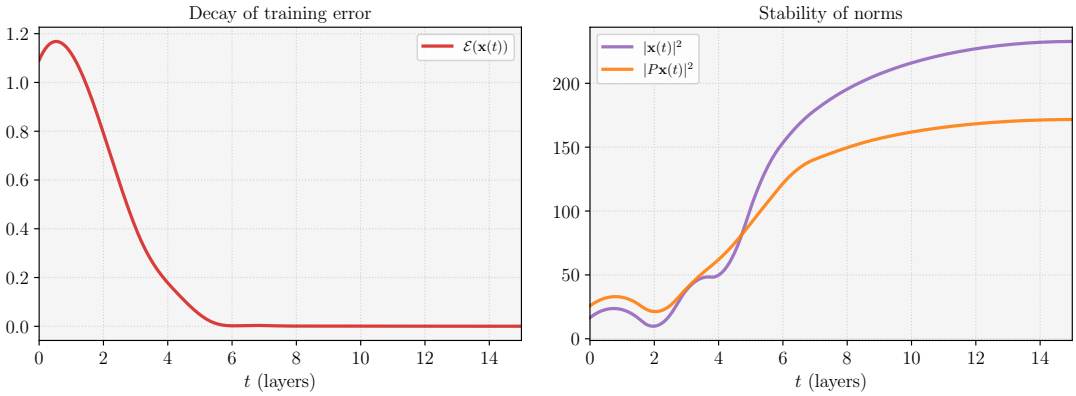


FIGURE 7. **Example 4.3:** The decay of the training error (*left*) and stabilization of the trained trajectories $\mathbf{x}(t)$ and $\{P\mathbf{x}_i(t)\}_{i\in[N]}$ (*right*).

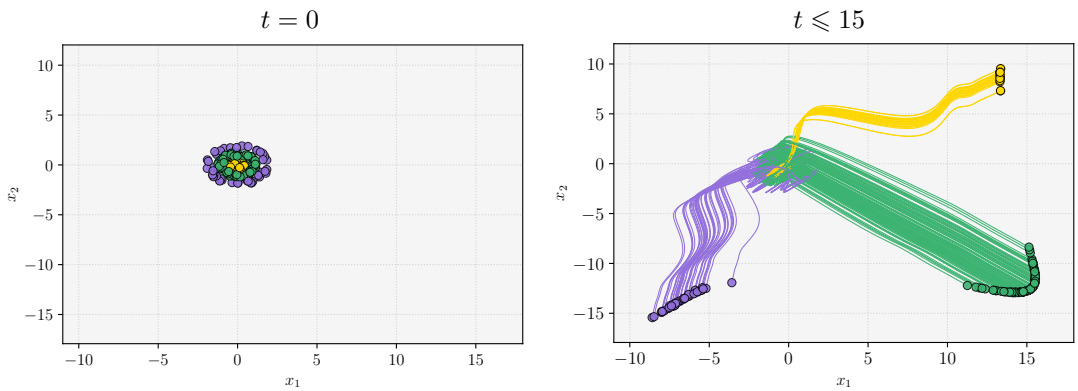


FIGURE 8. **Example 4.3:** A batch of training data (*left*) and the evolution of the corresponding trained trajectories $\mathbf{x}_{T,i}(t)$ (*right*). See https://github.com/borjanG/dynamical.systems/blob/master/videos/example4-2.mp4 for a movie of the evolution of the trajectories, where the stability phenomenon depicted in Figure 7 can be visualized.

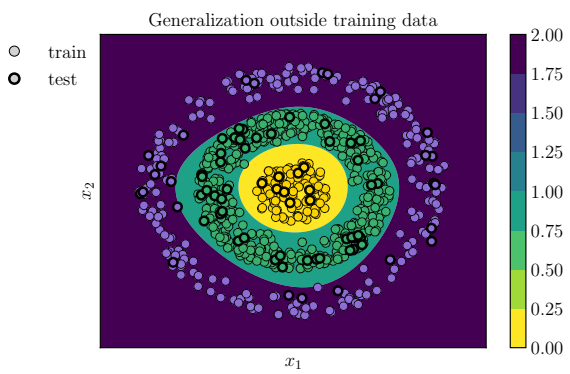


FIGURE 9. **Example 4.3:** The trained classifier on $[-2.5, 2.5]^2$ and its evaluation on a batch of the test data. The shape of the dataset is captured accurately.

grayscale, 28×28 *image of a handwritten digit, and thus an element of* $\mathbb{R}^{784}$*; the dataset has* 10 *labels:* $\vec{y}_i \in [10]$*. We consider a similar setup as in Example 4.4 – the model is*

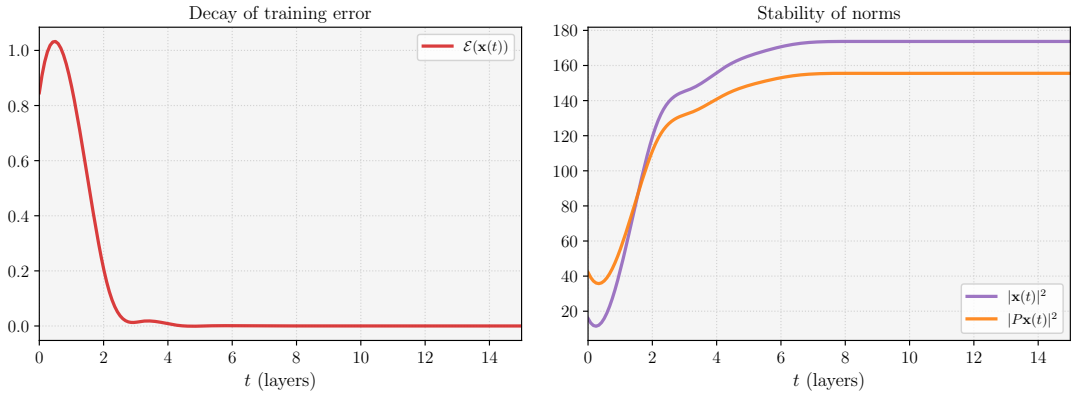


FIGURE 10. **Example 4.4:** The decay of the training error (*left*) and stabilization of the trained trajectories $\mathbf{x}(t)$ and $\{P\mathbf{x}_i(t)\}_{i\in[N]}$ (*right*).

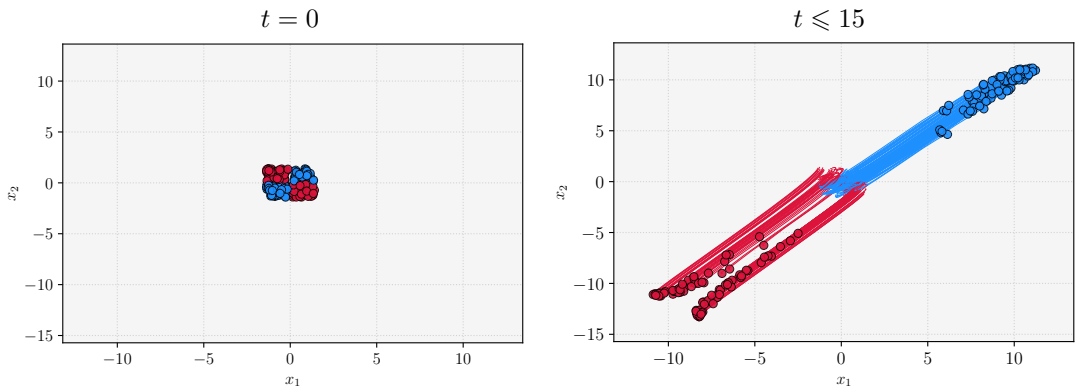


FIGURE 11. **Example 4.4:** A batch of training data (*left*) and the evolution of the corresponding trained trajectories $\mathbf{x}_{T,i}(t)$ (*right*). See https://github.com/borjanG/dynamical.systems/blob/master/videos/example4-3.mp4 for a movie of the evolution of the trajectories, where the stability phenomenon depicted in Figure 10 can be visualized.

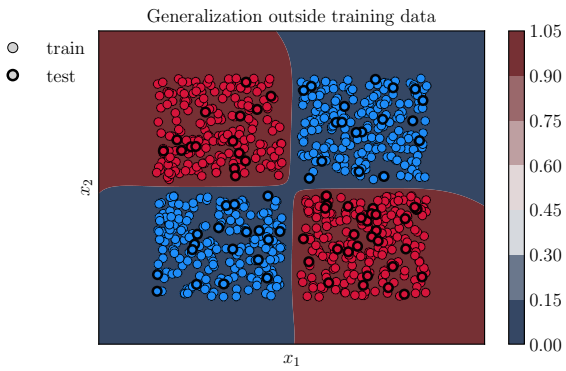


FIGURE 12. **Example 4.4:** The trained classifier on $[-2.5, 2.5]^2$ and its evaluation on a batch of the test data. The shape of the dataset is captured accurately.

parametrized as (2.4) – (2.7) *(we use a forward Euler scheme to obtain a corresponding ResNet with fixed time-step) where* $d_{\text{hid}} = 32$ *and* $\sigma \equiv \tanh$*, we consider cross-entropy*

loss in the training error in (4.7) *and only* L^2*–regularization of the parameters, with* $T = 20$*. We emphasize that we do not use any convolutional layers nor other commonly used operations in image classification (e.g. batch normalization, max-pooling) in the underlying ResNet architecture, and we solely concentrate on basic matrix weights. The output layer is parametrized by* $Px = p_1 x + p_2$*, where* $p_1 \in \mathbb{R}^{10\times 784}$*,* $p_2 \in \mathbb{R}^{10}$ *are part of the trainable parameters. We show the results of the experiments in Figure 13 – Figure 14.*

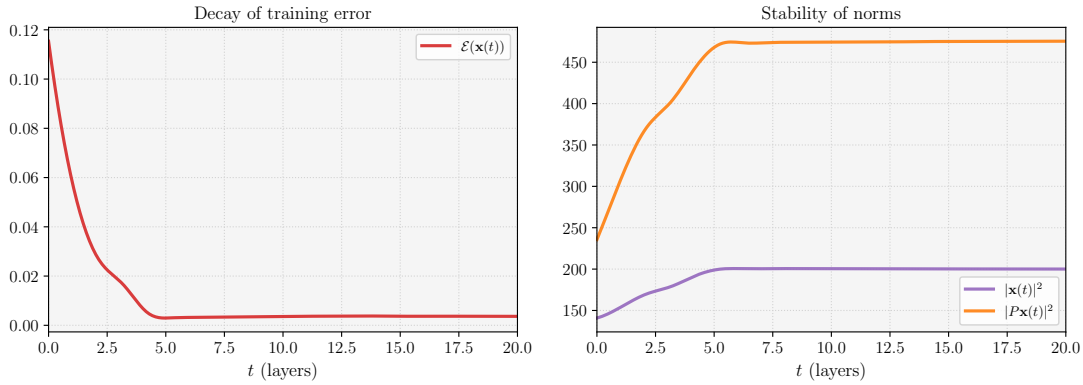


FIGURE 13. **Example 4.5:** The decay of the training error (*left*) and stabilization of the trained trajectories $\mathbf{x}(t)$ and $\{P\mathbf{x}_i(t)\}_{i\in[N]}$ (*right*).

Example 4.6 (Fashion MNIST)**.** *Fashion-MNIST is intended to serve as a direct drop-in replacement for the original MNIST dataset for benchmarking machine learning algorithms. It shares the same image size and structure of training and testing splits. We consider the same setup as in Example 4.5, and we show the results of the experiments in Figure 16 – Figure 18.*

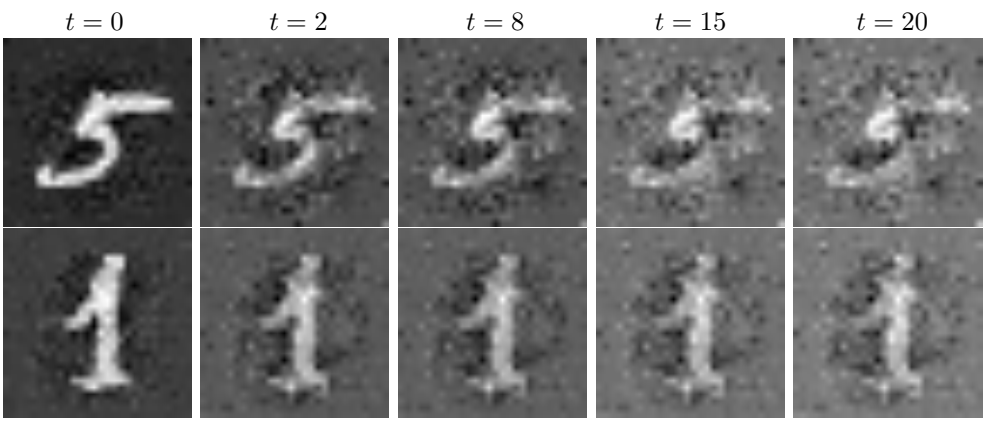


FIGURE 14. **Example 4.5:** We depict the evolution of two individual samples $\mathbf{x}_i(t) \in \mathbb{R}^{784}$ mapped onto a 28×28 grid. We see that each trajectory stabilizes to some stationary configuration. The trained model tends to "diffuse" (in a caloric sense) the input signal ahead of classifying via the softmax applied to $P\mathbf{x}_i(t) \in \mathbb{R}^{10}$.

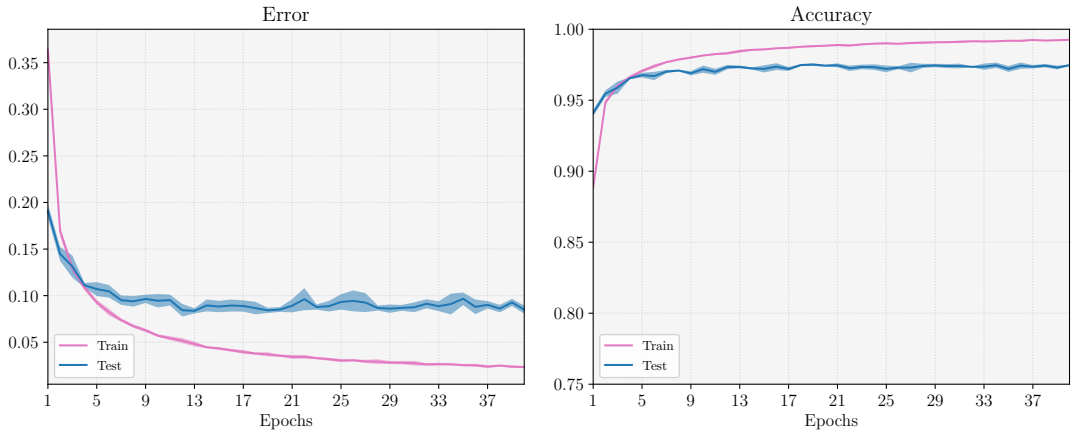


FIGURE 15. **Example 4.5:** The validation error and accuracy over training epochs (experiments repeated 10 times); in this simplified dataset setting, generalization is not necessarily compromised due to the introduction of an integrated empirical risk.

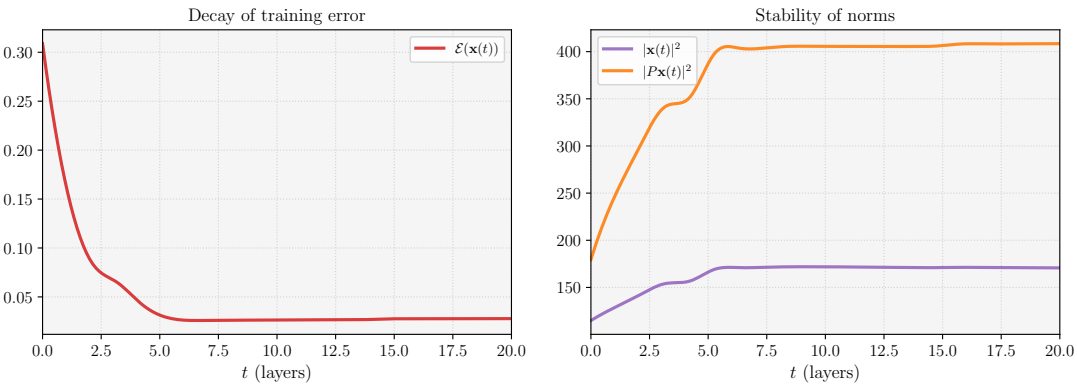


FIGURE 16. **Example 4.6:** The decay of the training error (*left*) and stabilization of the trained trajectories $\mathbf{x}(t)$ and $\{P\mathbf{x}_i(t)\}_{i\in[N]}$ (*right*).

5. THE INTERPOLATION REGIME

The majority of our results stated in the preceding sections stipulate whether and how the output $\mathbf{x}(T)$ of the neural ODE trajectory approaches the so-called interpolation regime ($\mathcal{E}(\mathbf{x}(T)) = 0$ with $\mathcal{E}$ given in (3.4)) when T increases. It is thus of interest to also illuminate some of the properties of the parameters which allow the trajectory to reach a minimizer of the empirical risk $\mathcal{E}$, and to see whether such parameters indeed exist.

By means of an elementary Grönwall argument, we can show the following illustrative result, which stipulates a lower bound for the amplitude of the weights w in terms of the dispersion or concentration of the input data.

Proposition 5.1. *Let $P : \mathbb{R}^d \to \mathbb{R}^m$ be surjective, and let $T > 0$. Assume that for some parameters $[w, b] \in L^1(0, T; \mathbb{R}^{d_u})$ the solution $\mathbf{x} \in C^0([0, T]; \mathbb{R}^{d_x})$ to either (3.3)*

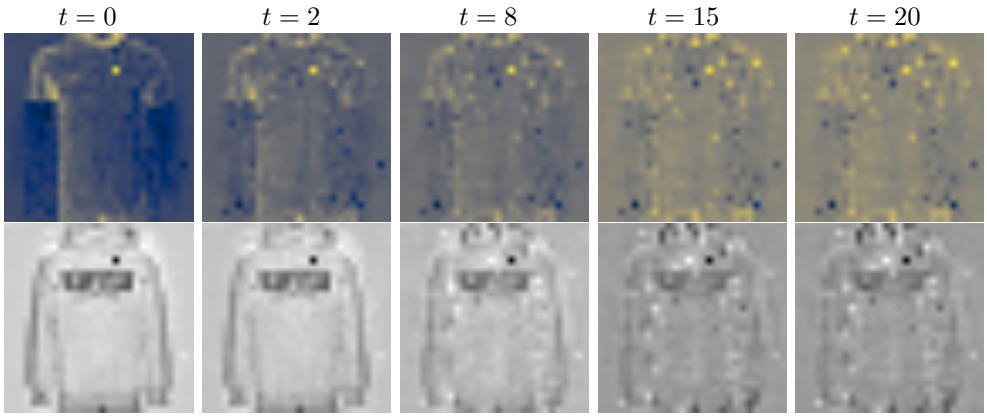


FIGURE 17. **Example 4.6:** We depict the evolution of two individual samples $\mathbf{x}_i(t) \in \mathbb{R}^{784}$ mapped onto a 28×28 grid (both sets of images are grayscale, but a different colormap is used to enhance visibility).

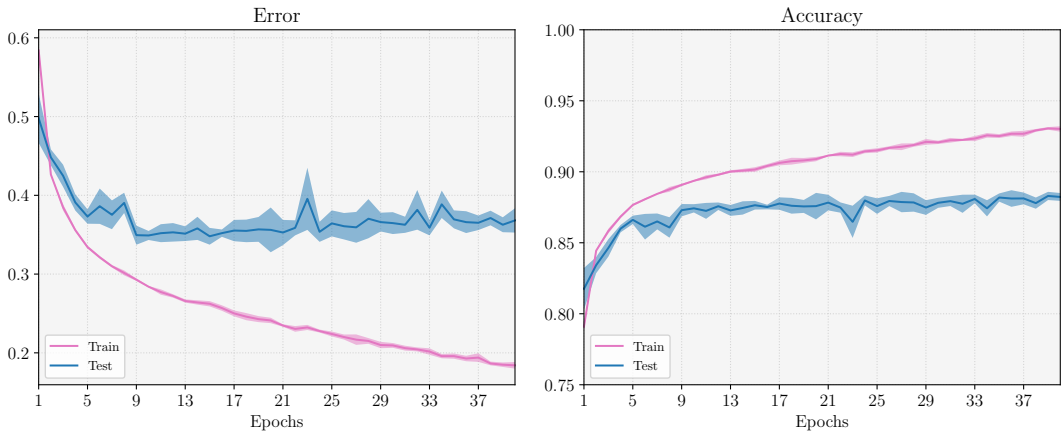


FIGURE 18. **Example 4.6:** The validation error and accuracy over training epochs (experiments repeated 10 times); as anticipated, generalization is not as good as for the simpler MNIST dataset. The lower accuracy with respect to state of the art configurations could also be due to the fact that we do not make use of convolutional layers.

or (3.2) *satisfies*

$$P\mathbf{x}_i(T) = \vec{y}_i \qquad \textit{for all } i \in [N]. \tag{5.1}$$

Then

$$\|w\|_{L^1(0,T;\mathbb{R}^{d_u})} \geqslant \mathfrak{C}(\sigma) \max_{\substack{(i,j)\in[N]^2 \\ i\neq j}} \inf_{\substack{\mathbf{x}_i^1 \in P^{-1}(\{\vec{y}_i\}) \\ \mathbf{x}_j^1 \in P^{-1}(\{\vec{y}_j\})}} \log\left(\frac{\left\|\mathbf{x}_i^1 - \mathbf{x}_j^1\right\|}{\left\|\mathbf{x}_i^0 - \mathbf{x}_j^0\right\|}\right), \tag{5.2}$$

where $\mathfrak{C}(\sigma) > 0$ *is the Lipschitz constant of* $\sigma \in \mathrm{Lip}(\mathbb{R})$.

By virtue of Cauchy-Schwarz, (5.2) clearly implies

$$\|w\|_{L^2(0,T;\mathbb{R}^{d_u})} \geqslant \frac{\mathfrak{C}(\sigma)}{\sqrt{T}} \max_{\substack{(i,j)\in[N]^2 \\ i\neq j}} \inf_{\substack{\mathbf{x}_i^1\in P^{-1}(\{\vec{y}_i\}) \\ \mathbf{x}_j^1\in P^{-1}(\{\vec{y}_j\})}} \log\left(\frac{\left\|\mathbf{x}_i^1-\mathbf{x}_j^1\right\|}{\left\|\mathbf{x}_i^0-\mathbf{x}_j^0\right\|}\right).$$

This observation leads us to stipulate the interest of L^2–regularization and increasing T: should the training data contains inputs which are very concentrated in the ambient space, parameters in the interpolation regime ((5.1) is equivalent to $\mathcal{E}(\mathbf{x}(T)) = 0$ whenever $\mathcal{E}$ attains its minimum) will dissipate in L^2–norm over longer time horizons.

On another hand, the difference quotient

$$\kappa\left(\mathbf{x}_i^0,\mathbf{x}_i^1\right) := \frac{\left\|\mathbf{x}_i^1-\mathbf{x}_j^1\right\|}{\left\|\mathbf{x}_i^0-\mathbf{x}_j^0\right\|}$$

can roughly be seen as an indicator of the variations of the flow, and thus the weight matrix would control the latter's simplicity. Indeed, denoting by $\Phi_T : \mathbb{R}^d \to \mathbb{R}^d$ the map defined by $\Phi_T(\mathbf{x}_i^0) = \mathbf{x}_i(T)$, where $\mathbf{x}_i(t)$ solves either (3.3) or (3.2), one can roughly stipulate that κ is an approximation $\nabla\Phi_T$ (evaluated at some intermediate point – here $\nabla\Phi_T$ denotes the Jacobian matrix of the flow map Φ_T). This can be argued already by using, for instance, the Cauchy mean-value theorem. Let us provide further detail and focus on (3.3) for simplicity, and assume $\sigma \in C^1(\mathbb{R}) \cap \mathrm{Lip}(\mathbb{R})$. We may linearize (3.3) with respect to the initial datum $\mathbf{x}_i^0$ to obtain

$$\begin{cases} \dot{\mathbf{z}}_i(t) = w(t)\begin{bmatrix} \sigma'\left(\mathbf{x}_{i,1}^0\right)\mathbf{z}_{i,1}(t) \\ \vdots \\ \sigma'\left(\mathbf{x}_{i,d}^0\right)\mathbf{z}_{i,d}(t) \end{bmatrix} := \widehat{w}(t)\mathbf{z}_i(t) \quad \text{for } \ t\in(0,T) \\ \mathbf{z}_i(0) = \mathbf{z}_i^0 \end{cases}$$

for $i \in [N]$. One then sees that

$$\exp\left(\int_0^T \widehat{w}(t)\,\mathrm{d}t\right) = \nabla\Phi_T(\mathbf{x}_i^0)$$

for $i \in [N]$. One may thus see $\int_0^T \widehat{w}(t)\,\mathrm{d}t$ as a "multi-dimensional" logarithm of $\nabla\Phi_T(\mathbf{x}_i^0)$. Thus, the Jacobian of the flow, which is an indicator of its variations and thus of its simplicity, can be measured by the weight matrix.

5.1. **On Definition 4.1.** To complete this section, we state the following interpolation result, which includes an estimate on the parameters with respect to the distance of the target and the initial datum, which somewhat enhances the validity of the assumption we make in Theorem 3.1. While such an estimate is standard in the setting of linear models, it is not provided by sufficient controllability conditions for nonlinear systems such as the Chow-Rashevski theorem [Coron, 2007, Chapter 3, Section 3.3].

Theorem 5.1. *Let $T > 0$ and assume that $N \leqslant d$. Let $\mathbf{x}^1 \in \mathbb{R}^{d_x}$ be given, and assume that the activation function $\sigma \in C^1(\mathbb{R}) \cap \mathrm{Lip}(\mathbb{R})$ is such that*

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

is a system of linearly independent vectors in $\mathbb{R}^d$*. Then, there exist universal constants* $r > 0$ *and* $\mathfrak{C} > 0$ *such that for any datum* $\mathbf{x}^0 \in \mathbb{R}^{d_x}$ *satisfying* $\left\|\mathbf{x}^0 - \mathbf{x}^1\right\| \leqslant r$*, there exists a weight matrix* $w \in L^\infty(0,T;\mathbb{R}^{d\times d})$ *such that the unique solution* $\mathbf{x}(\cdot)$ *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 following estimate holds

$$\|w\|_{L^\infty(0,T;\mathbb{R}^{d\times d})} \leqslant \frac{\mathfrak{C}}{T}\left\|\mathbf{x}^0 - \mathbf{x}^1\right\|.$$

Remark 9.
- *For simplicity of presentation, we have not exhibited the bias parameter* $b(t)$*. One can readily check that, in the presence of this additional parameter, the assumption* $N \leqslant d$ *can be relaxed to* $N \leqslant d+1$*.*
- *The case* $N > d+1$ *may be treated by, for instance, appending additional features (e.g. zeros as in [Dupont et al., 2019]) to the input data* $\mathbf{x}_i^0$ *for* $i \in [N]$ *as to guarantee that the augmented datum is of dimension* $d_{\text{aug}} \geqslant N$*.*

In the discrete-time context of neural networks such as (2.1) or (2.3), the property analog to Definition 3.1 is also well explored in the literature, and is commonly called *finite sample expressivity* ([Zhang et al., 2016]). An additional interest is that of estimating the number of parameters – referred to as *the memorization capacity* – needed to manifest this property. For further results in this direction we refer the reader to [Yun et al., 2019; Bubeck et al., 2020a,b] and the references therein.

In the ODE context, the property of finite sample expressivity finds its analog in the *ensemble* or *simultaneous controllability*, wherein one requires only 1 pair of parameters/controls to steer N trajectories of the same system to N prescribed targets – this is the property we show in Theorem 5.1. The literature on such controllability results of neural ODEs, mostly relying on geometrical techniques such as Lie brackets techniques (see [Coron, 2007, Chapter 3, Section 3.3]), under specific constraints on the activations function, is vast (see e.g. [Cuchiero et al., 2020; Agrachev and Sarychev, 2020; Tabuada and Gharesifard, 2020]). We refer to [Ruiz-Balet and Zuazua, 2021] for further results in this direction.

6. Continuous space-time neural networks

We now come back to the scheme (2.3) defining a ResNet with $N_{\text{layers}} \geqslant 2$ layers. Whilst such networks are widely used in practice, 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])

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

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 $\mathbf{x}^k \in \mathbb{R}^{d_k}$ 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 $\Pi^k \in \mathbb{R}^{d_{k+1}\times d_k}$ is a projection/embedding operator which serves to match dimensions. Much like in the fixed width case, we may also write the variable-width ResNet when $\mathfrak{f}$ is parametrized as in (2.6) or otherwise.

The continuous space-time network. It is not immediately obvious how one can see (6.1) 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 $i \in [N]$ we consider the scalar integro-differential equation

$$\begin{cases} \partial_t \mathbf{x}_i(t,x) = \sigma\left(\displaystyle\int_\Omega w(t,x,\xi)\mathbf{x}_i(t,\xi)\,\mathrm{d}\xi + b(t,x)\right) & \text{for } (t,x)\in(0,T)\times\Omega \\ \mathbf{x}_i(0,x) = \mathbf{x}_i^{\text{in}}(x) & \text{for } x\in\Omega. \end{cases} \tag{6.2}$$

Here $\Omega \subset \mathbb{R}^{d_\Omega}$ is a bounded domain, where $d_\Omega \geqslant 1$. We emphasize that $\mathbf{x}_i(t,x) \in \mathbb{R}$ 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 $\mathbf{x}_i^{\text{in}} \in C^0(\overline{\Omega})$ is such that there exist $\{x_j\}_{j=1}^d \subset \Omega$ such that $\mathbf{x}_i^{\text{in}}(x_j) = (\vec{x}_i)_j$. Such a datum can always be found (e.g. by interpolation). The continuum model (6.2) is proposed in [Liu and Markowich, 2020] where well-posedness is established, and is also suggested in [Weinan, 2017] albeit in a slightly different context. We distinguish two typical cases for choosing the shape of Ω as well as d_Ω.

- **Variable-width ResNets.** If in the discretized level, we seek to simply obtain a variable-width residual network such as (6.1) (or even the standard ResNet analog (2.3)), it suffices to consider $\Omega = (0,1)$, thus $d_\Omega = 1$. We give more detail on possible possible discretizations in Section 6.1 and Remark 10.
- **Convolutional Neural Networks.** The situation is slightly more delicate in the case of CNNs[9], which are typically used in tasks arising in computer vision. We provide a proposal covering the continuous-time analog of CNNs with partial generality.

  Assume that the dataset $\{\vec{x}_i\}_{i\in[N]}$ 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 (6.2) 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

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

  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,

[9]The mathematical theory of structural properties of CNNs in feed-forward form (without skip-connections) is well-established – see for instance [Mallat, 2016] and the references therein.

the number of extracted features (namely the number of filters). The bias parameter b can be omitted in this case, if desired.

One possible way to discretize the above continuous-time model and obtain a CNN-ResNet as in [He et al., 2016] is to follow the arguments in Section 6.1, 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 beyond the scope of this work.

Remark 10. *Observe that the continuous space-time model (6.2) (resp. (6.3)) 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 $i \in [N]$, we consider the initial datum*

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

We write the weight w as

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

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

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

yielding the vector $[b_j(t)]_{1 \leqslant j \leqslant d}$ of biases at time t. As $\mathbf{x}_i^{\text{in}}$, w and b are all linear combinations of Dirac masses, by plugging them in (6.2), we rewrite the integrals as sums, and setting, for any $i \in [N]$,

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

for $j \in [d]$, we see that $(\mathbf{x}_i)_j$ solves

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

This is just the j–th equation of the (2.4) – (2.5) for $i \in [N]$.

Correspondingly for $i \in [N]$ we may consider

$$\begin{cases} \partial_t \mathbf{x}_i(t,x) = \displaystyle\int_0^1 w(t,x,\xi) \sigma(\mathbf{x}_i(t,\xi)) \, \mathrm{d}\xi + b(t,x) & \text{in } (0,T) \times \Omega \\ \mathbf{x}_i(0,x) = \mathbf{x}_i^{\text{in}}(x) & \text{in } \Omega. \end{cases} \tag{6.3}$$

All of the above discussions also apply for this system.

6.1. **From continuous to discrete.** The passage from (6.2) to a discrete-time scheme such as (6.1) is not immediately obvious, and to our knowledge has not been presented in the literature. To proceed, it is important to observe the inherent link between the layer k and the width d_k in (6.1). This motivates discretizing (6.2) 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 (6.2) which reads[10] (we omit the dependence on i for notational simplicity)

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

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

$$\begin{cases} \mathbf{x}^{k+1} = \Pi^k \mathbf{x}^k + \sigma\left(w^k \mathbf{x}^k + b^k\right) & \text{for } k \in \{0,\dots,N_{\mathrm{layers}}-1\} \\ \mathbf{x}^0 = \vec{x}. \end{cases} \tag{6.4}$$

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

The derivation below is purely for illustrative purposes – an adaptive solver ought to perform better than an adaptation of an Euler scheme as (6.4). Moreover, the subsequent arguments will of course also apply to (6.3).

Let

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

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 $\Delta t = \frac{T}{N_{\mathrm{layers}}}$, but more general time-adaptive sequences can be considered. For any $k \in \{0,\dots,N_{\mathrm{layers}}\}$, 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 a 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

$$\partial_t \mathbf{x}(t^{k+1}, x_j(t^{k+1})) = \sigma\left(\int_0^1 w\left(t^{k+1}, x_j(t^{k+1}), \xi\right)\mathbf{x}(t^k,\xi)\,\mathrm{d}\xi + b\left(t^{k+1}, x_j(t^{k+1})\right)\right) \tag{6.5}$$

for $k \in \{0,\dots,N_{\mathrm{layers}}-1\}$ and $j \in \{1,\dots,d_{k+1}\}$. Hence, in view of the preceding discussion, some kind of interpolation may be needed to justify a backward Euler discretization of the time derivative $\partial_t \mathbf{x}$ appearing in (6.5) at the grid nodes.

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

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

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

Following through the above discussion, the main issue in writing down a forward difference discretization to $\partial_t \mathbf{x}(t^{k+1}, x_j(t^{k+1}))$ 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 $\mathbf{x}(t^{k+1}, x_j(t^{k+1})) - \mathbf{x}(t^k, x_j(t^k))$ 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

$$\widehat{\mathbf{x}}_j^k := \mathbf{x}_\iota^k + \frac{\mathbf{x}_\iota^k - \mathbf{x}_{\iota-1}^k}{x_\iota^k - x_{\iota-1}^k} \left(x_j^{k+1} - x_\iota^k \right). \tag{6.6}$$

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

$$\partial_t \mathbf{x}(t^{k+1}, x_j(t^{k+1})) \approx \frac{\mathbf{x}_j^{k+1} - \widehat{\mathbf{x}}_j^k}{\Delta t} \tag{6.7}$$

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 (6.5): for $j \in \{1, \dots, d_{k+1}\}$, we write

$$\int_0^1 w\left(t^{k+1}, x_j(t^{k+1}), \xi\right) \mathbf{x}(t^k, \xi) \, \mathrm{d}\xi \approx \sum_{\iota=1}^{d_k} \alpha_\iota w\left(t^{k+1}, x_j(t^{k+1}), x_\iota(t^k)\right) \mathbf{x}(t^k, x_\iota(t^k)). \tag{6.8}$$

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

Let us now define

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

and

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

The above definitions, as well as (6.7) and (6.8) applied to (6.5), lead us to (6.4), where Δt has been "omitted" as a factor of the nonlinearity. In view of (6.6), the operator $\Pi^k \in \mathbb{R}^{d_{k+1}} \times \mathbb{R}^{d_k}$ takes the explicit form

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

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 Π^k only has 2 non-zero elements at every row $j \in \{1, \dots, d_{k+1}\}$. This concludes our derivation.

Remark 11 (Generating moving grids)**.** *Whilst we have assumed a very simple given time-dependent grid, one may certainly generate more sophisticated moving grids – we refer to [Budd et al., 2009, Section 3] for a comprehensive overview on the existing methods, which have found extensive use in the discretization of partial differential equations manifesting shock waves and/or free boundaries.*

6.2. **The supervised learning problem.** Given a training dataset $\{\vec{x}_i, \vec{y}_i\}_{i\in[N]}$ with $\vec{x}_i \in \mathbb{R}^d$ for any $i \in [N]$, just as in the finite dimensional context, we begin by writing the equation satisfied by the stacked vector of states $\mathbf{x} := [\mathbf{x}_1, \dots, \mathbf{x}_N]^\top$ corresponding to the stacked vector of data $\mathbf{x}^{\text{in}} := [\mathbf{x}_1^{\text{in}}, \dots, \mathbf{x}_N^{\text{in}}]^\top$, where each $\mathbf{x}_i$ is the solution to either (6.2) or (6.3) corresponding to the datum $\mathbf{x}_i^{\text{in}}$, and control parameters $[w, b]$ which are the same for all i. The stacked continuous space-time neural networks we consider are thus either

$$\begin{cases} \partial_t \mathbf{x}(t,x) = \sigma\left(\int_\Omega \mathbf{w}(t,x,\xi)\mathbf{x}(t,\xi)\,\mathrm{d}\xi + \mathbf{b}(t,x)\right) & \text{in } (0,T)\times\Omega \\ \mathbf{x}(0,x) = \mathbf{x}^{\text{in}}(x) & \text{in } \Omega \end{cases} \tag{6.9}$$

or

$$\begin{cases} \partial_t \mathbf{x}(t,x) = \int_\Omega \mathbf{w}(t,x,\xi)\sigma(\mathbf{x}(t,\xi))\,\mathrm{d}\xi + \mathbf{b}(t,x) & \text{in } (0,T)\times\Omega \\ \mathbf{x}(0,x) = \mathbf{x}^{\text{in}}(x) & \text{in } \Omega. \end{cases} \tag{6.10}$$

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

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

6.2.1. *Empirical risk minimization.* As before, we first consider the regularized empirical risk minimization problem

$$\inf_{\substack{[w,b]\in H^k(0,T;\mathfrak{U}) \\ \text{subject to (6.9) (resp.(6.10))}}} \mathcal{E}(\mathbf{x}(T)) + \lambda \left\| [w,b] \right\|^2_{H^k(0,T;\mathfrak{U})}, \tag{6.11}$$

where $\alpha > 0$ is fixed, $k = 0$ for (6.10) and $k = 1$ for (6.9), $\mathfrak{U} := L^2(\Omega\times\Omega)\times L^2(\Omega)$, and we define the training error as

$$\mathcal{E}(\mathbf{x}(T)) := \frac{1}{N}\sum_{i=1}^N \text{loss}(P\mathbf{x}_i(T), \vec{y}_i), \tag{6.12}$$

where loss $\in C^0(\mathbb{R}^m \times \mathcal{Y}; \mathbb{R}_+)$ and $P : L^2(\Omega) \to \mathbb{R}^m$ is given. The optimization problem (6.11) admits a solution by the direct method in the calculus of variations.

In view of the rather universal nature of the proof to Theorem 3.1 and Theorem 4.1 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 $\mathfrak{U}$ respectively. Whence, we state the infinite-dimensional (partial) analog to Theorem 3.1.

Theorem 6.1. *Let $\lambda > 0$ be fixed, and let $\mathbf{x}^{\text{in}} \in \left(C^0(\overline{\Omega})\right)^N$ be such that $\mathbf{x}_i^{\text{in}}(x_j) = (\vec{x}_i)_j$. Suppose that $P : L^2(\Omega) \to \mathbb{R}^m$ is any non-zero affine map, and suppose that* loss $\in C^0(\mathbb{R}^m \times \mathcal{Y}; \mathbb{R}_+)$ *is such that Assumption 3 is satisfied. Assume that (6.9) (resp. (6.10) with σ positively homogeneous of degree 1) interpolates the dataset $\{\mathbf{x}_i^{\text{in}}, \vec{y}_i\}_{i\in[N]}$ in time 1 in the sense of Definition 3.1. For any $T \geqslant 1$, let $\mathbf{x}_T \in C^0\big([0,T]; L^2(\Omega)^N\big)$ be the unique solution to (6.9) (resp. (6.10)), associated to any global minimizer $[w_T, b_T] \in H^k(0,T;\mathfrak{U})$ of the functional in (6.11), where $k = 0$ in the case of (6.10) and $k = 1$ in the case of (6.9). The following properties then hold.*

(i) There exists a constant $\mathfrak{C} = \mathfrak{C}\left(\{\vec{x}_i, \vec{y}_i\}_{i\in[N]}, \lambda\right) > 0$ independent of T such that

$$\mathcal{E}(\mathbf{x}_T(T)) \leqslant \frac{\mathfrak{C}}{T}.$$

(ii) There exists a sequence $\{T_n\}_{n=1}^{\infty}$, with $T_n > 0$ and $T_n \xrightarrow[n\longrightarrow\infty]{} \infty$, and some $\mathbf{x}_\circ \in L^2(\Omega)^N$ with $\mathcal{E}(\mathbf{x}_\circ) = 0$ such that, along a subsequence,

$$\mathcal{E}\left(\mathbf{x}_{T_n}(T_n)\right) \xrightarrow[n\longrightarrow\infty]{} 0$$

and

$$\mathbf{x}_{T_n}(T_n) \xrightarrow[n\longrightarrow\infty]{\rightharpoonup} \mathbf{x}_\circ \qquad \text{weakly in } L^2(\Omega)^N.$$

For the sake of completeness, we give a sketch of the proof – by indicating the only changes with respect to that of Theorem 3.1.

Proof of Theorem 6.1. We note that the infinite-dimensional analog of Lemma 7.1 may easily be shown to hold, and one may readily repeat precisely the same arguments as in the proof of Theorem 3.1, replacing $\mathbb{R}^{d_u}$ and $\mathbb{R}^{d_x}$ by $\mathfrak{U}$ and $L^2(\Omega)^N$ respectively throughout. The only difference occurs in regarding the arguments on strong L^2–convergence of the sequence of controls in the case $k = 1$ – in the infinite dimensional case, we may exhibit the Aubin-Lions compactness lemma instead of Rellich-Kondrachov to conclude. □

6.2.2. *Augmented empirical risk minimization.* We similarly consider the augmented supervised learning problem

$$\inf_{\substack{[w,b]\in L^2(0,T;\mathfrak{U}) \\ \text{subject to (6.10)}}} \mathcal{E}(\mathbf{x}(T)) + \frac{1}{N}\int_0^T \|\mathbf{x}(t) - \overline{\mathbf{x}}\|^2_{L^2(\Omega)}\,\mathrm{d}t + \lambda\Big\|[w,b]\Big\|^2_{L^2(0,T;\mathfrak{U})}, \tag{6.13}$$

where $\mathcal{E}$ is as in (6.12) and loss$(\cdot,\cdot)$ satisfies Assumption 4. We solely concentrate on System (6.10) to avoid a possibly abundance of technical details in the context of $\mathrm{BV}_t L^2_x$ analysis. Again, $P : L^2(\Omega) \longrightarrow \mathbb{R}^m$ is a surjective map, and $\mathbf{x}_i \in P^{-1}(\{\vec{y}_i\}) \subset L^2(\Omega)$ for $i \in [N]$ are arbitrary, but fixed. As expected, the analog exponential decay result holds for (6.13).

Theorem 6.2 (Exponential stability)**.** *Fix $\lambda > 0$, let $P \in \mathrm{Lip}(L^2(\Omega); \mathbb{R}^m)$ be any given surjective map, and let $\overline{\mathbf{x}} \in L^2(\Omega)^N$ with $\overline{\mathbf{x}}_i \in P^{-1}(\{\vec{y}_i\})$ for $i \in [N]$ be arbitrary but fixed. Suppose that (6.10) is controllable with linear cost in some time $T_0 > 0$ in the sense of Definition 4.1. Then, there exists $T^* > 0$ and constants $\mathfrak{C} = \mathfrak{C}\left(\{\vec{x}_i, \vec{y}_i\}_{i\in[N]}, \lambda, N, \alpha, P\right) > 0$ and $\mu = \mu\left(\{\vec{x}_i, \vec{y}_i\}_{i\in[N]}, \lambda, N, \alpha\right) > 0$ such that for any $T \geqslant T^*$, any pair of parameters $[w_T, b_T] \in L^2(0,T;\mathfrak{U})$ solving the minimization problem (6.13), and the corresponding unique solution $\mathbf{x}_T(\cdot)$ to (6.10) satisfy*

$$\big\|w_T(t)\big\| + \big\|b_T(t)\big\| \leqslant \mathfrak{C}$$

for a.e. $t \in [0,T]$ *and*

$$\mathcal{E}(\mathbf{x}_T(t)) + \|\mathbf{x}_T(t) - \overline{\mathbf{x}}\|_{L^2(\Omega)} \leqslant \mathfrak{C}\, e^{-\mu t}$$

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

The proof is omitted and left to the reader, as it follows precisely the same arguments as that of Theorem 4.1.

7. Proofs

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

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathfrak{f}([w(t), b(t)], \mathbf{x}(t)) & \text{in } (0,T) \\ \mathbf{x}(0) = \mathbf{x}^0 \in \mathbb{R}^{d_x}, \end{cases} \tag{7.1}$$

with

$$\mathfrak{f}([0,0], \mathbf{x}) = 0, \qquad \mathfrak{f}([\alpha w, \alpha b], \mathbf{x}) = \alpha \mathfrak{f}([w,b], \mathbf{x}) \quad \text{for } \alpha > 0. \tag{7.2}$$

We will refer to $u := [w,b]$ as *the control* of the ODE system, in accordance with control theory vocabulary. We begin with following short but key lemma.

Lemma 7.1. *Let* $T_0 > 0$ *and* $[w_{T_0}, b_{T_0}] \in L^1(0,T_0;\mathbb{R}^{d_u})$ *be given, and let* $\mathbf{x}_{T_0}(\cdot)$ *be the unique solution to*

$$\begin{cases} \dot{\mathbf{x}}_{T_0}(t) = \mathfrak{f}\left([w_{T_0}(t), b_{T_0}(t)], \mathbf{x}_{T_0}(t)\right) & \text{in } (0,T_0) \\ \mathbf{x}_{T_0}(0) = \mathbf{x}^0 \in \mathbb{R}^{d_x}, \end{cases} \tag{7.3}$$

(i.e. (7.1) *on* $(0,T_0)$*) with* $\mathfrak{f}$ *as in either* (3.3) *or* (3.2)*, thus satisfying* (7.2)*. Let* $T > 0$*, and define*

$$w_T(t) := \frac{T_0}{T} w_{T_0}\left(t\frac{T_0}{T}\right), \qquad b_T(t) := \frac{T_0}{T} b_{T_0}\left(t\frac{T_0}{T}\right) \qquad \text{for } t \in [0,T],$$

and

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

Then $\mathbf{x}_T(\cdot)$ *is the unique solution to* (7.1) *(with the same* $\mathfrak{f}$ *as in* (7.3)*) associated to* $[w_T, b_T]$.

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 of Lemma 7.1. Since $\mathbf{x}_{T_0}$ is the solution to (7.3), the change of variable $\tau = s\frac{T}{T_0}$ as well as (7.2), we have

$$\begin{aligned} \mathbf{x}_T(t) := \mathbf{x}_{T_0}\left(t\frac{T_0}{T}\right) &= \mathbf{x}^0 + \int_0^{t\frac{T_0}{T}} \mathfrak{f}\left([w_{T_0}(s), b_{T_0}(s)], \mathbf{x}_{T_0}(s)\right) \mathrm{d}s \\ &= \mathbf{x}^0 + \int_0^t \frac{T_0}{T} \mathfrak{f}\left(\left[w_{T_0}\left(\tau\frac{T_0}{T}\right), b_{T_0}\left(\tau\frac{T_0}{T}\right)\right], \mathbf{x}_{T_0}\left(\tau\frac{T_0}{T}\right)\right) \mathrm{d}\tau \\ &= \mathbf{x}^0 + \int_0^t \mathfrak{f}\left([w_T(\tau), b_T(\tau)], \mathbf{x}_T(\tau)\right) \mathrm{d}\tau. \end{aligned}$$

It follows that $\mathbf{x}_T$ solves (7.1), and we conclude by uniqueness. □

We will also need the following lemma.

Lemma 7.2 (Compactness of the flow)**.** *Let $T > 0$ be fixed. The maps*

(i) $\Phi_T : [w, b] \longmapsto \mathbf{x}(\cdot)$ *mapping* $L^2(0, T; \mathbb{R}^{d_u})$ *to* $C^0([0, T]; \mathbb{R}^{d_x})$ *where* $\mathbf{x}(\cdot)$ *solves* (3.3)*,*

(ii) $\Phi_T : [w, b] \longmapsto \mathbf{x}(\cdot)$ *mapping* $L^2(0, T; \mathbb{R}^{d_u}) \cap \mathrm{BV}(0, T; \mathbb{R}^{d_u})$ *to* $C^0([0, T]; \mathbb{R}^{d_x})$ *where* $\mathbf{x}(\cdot)$ *solves* (3.2)*,*

(iii) $\Phi_T : [w, b] \longmapsto \mathbf{x}(\cdot)$ *mapping* $H^1(0, T; \mathbb{R}^{d_u})$ *to* $C^0([0, T]; \mathbb{R}^{d_x})$ *where* $\mathbf{x}(\cdot)$ *solves* (3.2)*,*

are all compact.

We postpone the proof to the appendix. We are now in a position to prove the main result.

Proof of Theorem 3.1. We will henceforth, for notational convenience, extensively make use of the notation $u := [w, b]$. 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 showing

$$\mathcal{E}\left(\mathbf{x}_T(T)\right) \lesssim T^{-1} \tag{7.4}$$

uniformly in T. By the interpolation assumption, there exists some $u^1 \in L^2(0, 1; \mathbb{R}^{d_u})$ such that the associated solution $\mathbf{x}^1$ to (3.3) on $[0, 1]$ satisfies $\mathcal{E}(\mathbf{x}^1(1)) = 0$. Using the optimality of u_T and the scaling relations from Lemma 7.1, we obtain

$$\begin{aligned} J_{\lambda,T}\left(u_T\right) &= \mathcal{E}\left(\mathbf{x}_T(T)\right) + \lambda \left\|u_T\right\|^2_{L^2(0,T;\mathbb{R}^{d_u})} \\ &\leqslant \mathcal{E}(\mathbf{x}^1(1)) + \frac{\lambda}{T}\left\|u^1\right\|^2_{L^2(0,1;\mathbb{R}^{d_u})} \end{aligned}$$

for all $T > 0$. Since $\mathcal{E}(\mathbf{x}^1(1)) = 0$ by the interpolation assumption, the above inequality implies

$$0 \leqslant \mathcal{E}\left(\mathbf{x}_T(T)\right) \leqslant \frac{\lambda}{T}\left\|u^1\right\|^2_{L^2(0,1;\mathbb{R}^{d_u})} \tag{7.5}$$

for all $T > 0$. Estimate (7.5) clearly implies (7.4).

Part 2. We now look to prove (3.6). To this end, we will look to 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 $\{\mathcal{E} = 0\}$.

For any $T > 0$, set

$$u^{\mathrm{aux}}(t) := \frac{1}{T} u^1\left(\frac{t}{T}\right) \qquad \text{for } t \in [0, T].$$

We argue similarly as in Part 1. Making use of Lemma 7.1 once again, and since $\mathcal{E}(\mathbf{x}^1(1)) = 0$, we see that

$$\begin{aligned} J_{\lambda,T}\left(u^{\mathrm{aux}}\right) &= \mathcal{E}\left(\mathbf{x}^1(1)\right) + \frac{\lambda}{T}\left\|u^1\right\|^2_{L^2(0,1;\mathbb{R}^{d_u})} \\ &= \frac{\lambda}{T}\left\|u^1\right\|^2_{L^2(0,1;\mathbb{R}^{d_u})}. \end{aligned} \tag{7.6}$$

Using the optimality of u_T, one sees that

$$J_{\lambda,T}\left(u^{\mathrm{aux}}\right) \geqslant J_{\lambda,T}\left(u_T\right) \geqslant \lambda \left\|u_T\right\|^2_{L^2(0,T;\mathbb{R}^{d_u})}. \tag{7.7}$$

Combining (7.7) and (7.6), we deduce that

$$\|u_T\|^2_{L^2(0,T;\mathbb{R}^{d_u})} \leqslant \frac{1}{T}\left\|u^1\right\|^2_{L^2(0,1;\mathbb{R}^{d_u})} \tag{7.8}$$

for any $T>0$. Now by integrating (3.3), and using the fact that σ is globally Lipschitz continuous with constant $C(\sigma)>0$ and satisfies $\sigma(0)=0$, for any $t\in[0,T]$ we have

$$\left\|\mathbf{x}_T(t)-\mathbf{x}^0\right\| \leqslant N^{1/2}C(\sigma)\int_0^t \|w_T(s)\|\,\|\mathbf{x}_T(s)\|\,\mathrm{d}s + N^{1/2}\,\|b_T\|_{L^1(0,T;\mathbb{R}^d)}\,.$$

By using the Grönwall inequality, we obtain

$$\left\|\mathbf{x}_T(T)-\mathbf{x}^0\right\| \leqslant N^{1/2}\,\|b_T\|_{L^1(0,T;\mathbb{R}^d)}\exp\left(N^{1/2}C(\sigma)\int_0^T \|w_T(s)\|\,\mathrm{d}s\right),$$

whereas by Cauchy-Schwarz, it follows that

$$\left\|\mathbf{x}_T(T)-\mathbf{x}^0\right\| \leqslant T^{1/2}N^{1/2}\,\|b_T\|_{L^2(0,T;\mathbb{R}^d)}\exp\left(T^{1/2}N^{1/2}C(\sigma)\,\|w_T\|_{L^2(0,T;\mathbb{R}^{d\times d})}\right).$$

At this point, employing (7.8), we deduce

$$\left\|\mathbf{x}_T(T)-\mathbf{x}^0\right\| \leqslant N^{1/2}\left\|u^1\right\|_{L^2(0,1;\mathbb{R}^{d_u})}\exp\left(N^{1/2}C(\sigma)\left\|u^1\right\|_{L^2(0,1;\mathbb{R}^{d_u})}\right).$$

Since u^1 is independent of T, we conclude that the set $\{\mathbf{x}_T(T)\}_{T>0}$ is bounded. Whence, there exists a sequence $\{T_n\}_{n=1}^\infty$ with $T_n>0$ and $T_n\longrightarrow\infty$ as $n\longrightarrow\infty$ and some $\mathbf{x}_\circ\in\mathbb{R}^{d_x}$ such that

$$\mathbf{x}_{T_n}(T_n)\longrightarrow\mathbf{x}_\circ \qquad \text{as } n\longrightarrow\infty. \tag{7.9}$$

Since $\mathcal{E}\left(\mathbf{x}_{T_n}(T_n)\right)\longrightarrow 0$ as $n\longrightarrow\infty$ by (7.4), by continuity of $\mathcal{E}$, we have $\mathcal{E}(\mathbf{x}_\circ)=0$. This concludes the proof of (3.6).

Part 3. We now address the third 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 $L^2(0,1;\mathbb{R}^{d_u})$.

Let $u^\dagger\in L^2(0,1;\mathbb{R}^{d_u})$ be any solution to

$$\inf_{\substack{u\in L^2(0,1;\mathbb{R}^{d_u})\\ \mathbf{x}(\cdot)\text{ solves (3.3)}\\ \text{and}\\ \mathcal{E}(\mathbf{x}(1))=0}}\int_0^1 \|u(t)\|^2\,\mathrm{d}t. \tag{7.10}$$

Denote by $\mathbf{x}^\dagger$ the corresponding solution to (3.3) on $[0,1]$. We claim that

$$\|u_n\|_{L^2(0,1;\mathbb{R}^{d_u})} \leqslant \left\|u^\dagger\right\|_{L^2(0,1;\mathbb{R}^{d_u})}, \qquad \text{for all } n\geqslant 1. \tag{7.11}$$

We prove this claim by contradiction. Indeed, assume that we had

$$\left\|u^\dagger\right\|_{L^2(0,1;\mathbb{R}^{d_u})} < \|u_n\|_{L^2(0,1;\mathbb{R}^{d_u})} \qquad \text{for some } n\geqslant 1.$$

We consider

$$u_n^\dagger(t) := \frac{1}{T_n}u^\dagger\left(\frac{t}{T_n}\right) \qquad \text{for } t\in[0,T_n],$$

whose corresponding state $\mathbf{x}_n^\dagger$, solution to (3.3) on $[0,T_n]$, satisfies $\mathbf{x}_n^\dagger(T_n)=\mathbf{x}^\dagger(1)$ by Lemma 7.1. On another hand, by assumption we have $\mathcal{E}(\mathbf{x}^\dagger(1))=0$. It then follows

that

$$J_{\lambda,T_n}\left(u_n^\dagger\right) = \frac{\lambda}{T_n}\left\|u^\dagger\right\|^2_{L^2(0,1;\mathbb{R}^{d_u})} < \mathcal{E}\left(\mathbf{x}_{T_n}(T_n)\right) + \frac{\lambda}{T_n}\|u_n\|^2_{L^2(0,1;\mathbb{R}^{d_u})} = J_{T_n}\left(u_{T_n}\right),$$

which contradicts the fact that u_{T_n} minimizes J_{T_n}. Hence, (7.11) holds, and $\{u_n\}_{n=1}^\infty$ is bounded in $L^2(0,1;\mathbb{R}^{d_u})$. Consequently, by the Banach-Alaoglu theorem, there exists $u^* = [w^*, b^*] \in L^2(0,1;\mathbb{R}^{d_u})$ such that

$$u_n \rightharpoonup u^* \qquad \text{weakly in } L^2(0,1;\mathbb{R}^{d_u}),$$

along some subsequence as $n \longrightarrow \infty$. Moreover, using the properties of equation (3.3) (Lemma 7.2), we deduce that the trajectory $\mathbf{x}_n$ associated to u_n satisfies

$$\mathbf{x}_n \longrightarrow \mathbf{x}^* \qquad \text{strongly in } C^0([0,1];\mathbb{R}^{d_x}) \tag{7.12}$$

as $n \longrightarrow \infty$, where $\mathbf{x}^*$ is the solution to (3.3) on $[0,1]$, associated to u^*. On another hand, note that by Lemma 7.1, $\mathbf{x}_{T_n}(t) = \mathbf{x}_n(\frac{t}{T_n})$ for $t \in [0,T_n]$, whence $\mathbf{x}_{T_n}(T_n) = \mathbf{x}_n(1)$ and thus, combining (7.12) and (7.9), we see that $\mathbf{x}^*(1) = \mathbf{x}_\circ$. Consequently, u^* is a control such that $\mathcal{E}(\mathbf{x}^*(1)) = \mathcal{E}(\mathbf{x}_\circ) = 0$, thus satisfying the constraint in (7.10). In view of this, we may also use (7.11) and the weak lower semicontinuity of the L^2–norm to write

$$\begin{aligned}\left\|u^\dagger\right\|_{L^2(0,1;\mathbb{R}^{d_u})} \leqslant \|u^*\|_{L^2(0,1;\mathbb{R}^{d_u})} &\leqslant \liminf_{n\longrightarrow\infty}\|u_n\|_{L^2(0,1;\mathbb{R}^{d_u})}\\ &\leqslant \lim_{n\longrightarrow\infty}\|u_n\|_{L^2(0,1;\mathbb{R}^{d_u})}\\ &\leqslant \limsup_{n\longrightarrow\infty}\|u_n\|_{L^2(0,1;\mathbb{R}^{d_u})}\\ &\leqslant \left\|u^\dagger\right\|_{L^2(0,1;\mathbb{R}^{d_u})},\end{aligned} \tag{7.13}$$

clearly implying that

$$\lim_{n\longrightarrow\infty}\|u_n\|_{L^2(0,1;\mathbb{R}^{d_u})} = \|u^*\|_{L^2(0,1;\mathbb{R}^{d_u})}.$$

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

$$u_n \longrightarrow u^* \qquad \text{strongly in } L^2(0,1;\mathbb{R}^{d_u})$$

along some subsequence as $n \longrightarrow \infty$. Moreover, from (7.13) we deduce that, since $u^\dagger$ is a solution to (7.10) and since u^* satisfies the constraints therein, u^* is a solution to (7.10) 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 (7.6), one has

$$\begin{aligned}J_{\lambda,T}(u^{\text{aux}}) &= \mathcal{E}(\mathbf{x}^1(1)) + \frac{\lambda}{T}\left\|u^1\right\|^2_{L^2(0,1;\mathbb{R}^{d_u})} + \frac{\lambda}{T^3}\left\|\dot{u}^1\right\|^2_{L^2(0,1;\mathbb{R}^{d_u})}\\ &= \frac{\lambda}{T}\left\|u_{T_0}\right\|^2_{L^2(0,1;\mathbb{R}^{d_u})} + \frac{\lambda}{T^3}\left\|\dot{u}^1\right\|^2_{L^2(0,1;\mathbb{R}^{d_u})}.\end{aligned}$$

This is not an impediment to (7.7), which remains true, and one can clearly deduce that $\{\mathbf{x}_T(T)\}_{T>0}$ is bounded as well. Similarly, (7.5) holds with a bound of the form

$$0 \leqslant \mathcal{E}(\mathbf{x}_T(T)) \leqslant \frac{\lambda}{T} \left\|u^1\right\|^2_{L^2(0,1;\mathbb{R}^{d_u})} + \frac{\lambda}{T^3} \left\|\dot{u}^1\right\|^2_{L^2(0,1;\mathbb{R}^{d_u})}.$$

Whence the remainder of parts 1 and 2 hold in this context as well.

In Part 3, we emphasize 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 (7.12) 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 (7.12) by arguing just as in the proof of Lemma 7.2.

This concludes the proof. □

7.2. **Proof of Theorem 3.2.** The proof closely follows the lines of that just above. Let us consider $k = 1$, since the case $k = 0$ is equivalent to Theorem 3.2. We present minimal details for completeness.

Proof of Theorem 3.2. We again make use of the notation $u := [w, b]$. We first show

$$\mathcal{E}(\mathbf{x}_\lambda(T)) \lesssim \lambda \tag{7.14}$$

uniformly in $\lambda > 0$ – we argue as in the proof of Theorem 3.1 just above, exhibiting, by the interpolation assumption, parameters $u^1 \in L^2(0,1;\mathbb{R}^{d_u})$ such that $\mathcal{E}(\mathbf{x}^1(1)) = 0$. We may obtain an estimate like (7.5) and conclude. Now, the same arguments as in Part 2 of the proof of Theorem 3.1 may be used to deduce that $\{\mathbf{x}_\lambda(T)\}_{\lambda>0}$ is a bounded subset of $\mathbb{R}^d$, and hence there exists a sequence $\{\lambda_n\}_{n=1}^\infty$ of positive numbers with $\lambda_n \searrow 0$ as $n \longrightarrow \infty$ and some $\mathbf{x}_\circ \in \mathbb{R}^{d_x}$ such that

$$\mathbf{x}_{\lambda_n}(T) \xrightarrow[n\longrightarrow\infty]{} \mathbf{x}_\circ.$$

Using (7.14) we deduce that $\mathcal{E}(\mathbf{x}_\circ) = 0$. Finally, the proof of the last fact is identical to that done for Theorem 3.1, so we omit it. □

7.3. **Proof of Theorem 3.3.** We now provide a proof of our main result in the context of classification tasks.

Proof of Theorem 3.3. We recall that, by assumption, $\mathbf{x}_i^0 := \mathfrak{Q}\vec{x}_i \geqslant 0$ for $i \in [N]$. Now let $\left[\widehat{w}, \widehat{b}\right] \in H^k(0, T_0; \mathbb{R}^{d_u})$ be a pair of parameters which separates the training dataset $\left\{\mathbf{x}_i^0, \vec{y}_i\right\}_{i\in[N]}$ with respect to P in time $T_0 > 0$, i.e., such that the solution $\widehat{\mathbf{x}} = [\widehat{\mathbf{x}}_1, \ldots, \widehat{\mathbf{x}}_N]$ to (3.2), with initial condition $\mathbf{x}^0 = [\mathbf{x}_1^0, \ldots, \mathbf{x}_N^0]$, corresponding to $\left[\widehat{w}, \widehat{b}\right]$, satisfies

$$\min_{i\in[N]} \left\{ P\,\widehat{\mathbf{x}}_i(T_0)_{\vec{y}_i} - \max_{\substack{j\in[N]\\ j\neq\vec{y}_i}} P\,\widehat{\mathbf{x}}_i(T_0)_j \right\} =: \gamma > 0. \tag{7.15}$$

Now fix an arbitrary $\alpha \in \left(0, \frac{1}{2}\right)$, and, for any $T > 0$, define

$$\left[w^\dagger(t), b^\dagger(t)\right] := \begin{cases} \dfrac{2T_0}{T}\left[\widehat{w}\left(t\dfrac{2T_0}{T}\right), \widehat{b}\left(t\dfrac{2T_0}{T}\right)\right] & \text{for } t \in \left[0, \dfrac{T}{2}\right] \\ T^{\alpha-1}\left[\mathrm{Id}_d, 0_d\right] & \text{for } t \in \left(\dfrac{T}{2}, T\right], \end{cases}$$

where Id_d is the identity matrix in $\mathbb{R}^{d\times d}$ and 0_d is the zero vector in $\mathbb{R}^d$. By virtue of the scaling in Lemma 7.1, for $t \in \left[\frac{T}{2}, T\right]$, the trajectories $\mathbf{x}^\dagger = \left[\mathbf{x}_1^\dagger, \dots, \mathbf{x}_N^\dagger\right]$ associated to $\left[w^\dagger, b^\dagger\right]$ are given by the solution to

$$\begin{cases} \dot{\mathbf{x}}_i^\dagger(t) = \sigma\left(T^{\alpha-1}\mathbf{x}_i^\dagger(t)\right) & \text{for } t \in \left[\dfrac{T}{2}, T\right] \\ \mathbf{x}_i^\dagger\left(\dfrac{T}{2}\right) = \widehat{\mathbf{x}}_i\left(T_0\right). \end{cases} \tag{7.16}$$

Moreover, since $\sigma(x) = \max\{x, 0\} \geqslant 0$, the right hand side in (3.2) is nonnegative. Using the assumption that the initial conditions are of the form $\mathbf{x}_i^0 = \mathfrak{Q}\vec{x}_i \geqslant 0$, it follows that $\widehat{\mathbf{x}}_i(T_0) \geqslant 0$ for all $i \in [N]$. We can therefore drop σ in (7.17) and deduce that $P\mathbf{x}_i^\dagger(t)$ solves

$$\begin{cases} \dfrac{\mathrm{d}}{\mathrm{d}t} P\mathbf{x}_i^\dagger(t) = T^{\alpha-1} P\mathbf{x}_i^\dagger(t) & \text{for } t \in \left[\dfrac{T}{2}, T\right] \\ P\mathbf{x}_i^\dagger\left(\dfrac{T}{2}\right) = P\widehat{\mathbf{x}}_i\left(T_0\right). \end{cases} \tag{7.17}$$

Hence, we have

$$P\mathbf{x}_i^\dagger(t) = P\widehat{\mathbf{x}}_i(T_0) e^{T^{\alpha-1}(t-T/2)}, \qquad \text{for all } t \in \left[\frac{T}{2}, T\right].$$

Now, using the definition of the cross-entropy loss and the margin γ in (7.15), we compute, for any $i \in [N]$,

$$\begin{aligned} \operatorname{loss}\left(\mathbf{x}_i^\dagger(T), \vec{y}_i\right) &= -\log\left(\frac{e^{P\widehat{\mathbf{x}}_i(T_0)_{\vec{y}_i} e^{\frac{T^\alpha}{2}}}}{\sum_{j=1}^m e^{P\widehat{\mathbf{x}}_i(T_0)_j e^{\frac{T^\alpha}{2}}}}\right) \\ &= \log\left(1 + \sum_{j \neq \vec{y}_i} e^{\left(P\widehat{\mathbf{x}}_i(T_0)_j e^{\frac{T^\alpha}{2}}\right) - \left(P\widehat{\mathbf{x}}_i(T_0)_{\vec{y}_i} e^{\frac{T^\alpha}{2}}\right)}\right) \\ &\leqslant \log\left(1 + (m-1)\exp\left(-\gamma \exp\left(\frac{T^\alpha}{2}\right)\right)\right). \end{aligned}$$

Then, we can estimate

$$\mathcal{E}\left(\mathbf{x}^\dagger(T)\right) \leqslant \log\left(1 + (m-1)\exp\left(-\gamma \exp\left(\frac{T^\alpha}{2}\right)\right)\right). \tag{7.18}$$

On the other hand, using the definition of $\left[w^\dagger, b^\dagger\right]$, we deduce

$$\begin{aligned} \left\|\left[w^\dagger, b^\dagger\right]\right\|^2_{H^1(0,T;\mathbb{R}^{d_u})} &= \left\|\left[w^\dagger, b^\dagger\right]\right\|^2_{H^1\left(0,\frac{T}{2};\mathbb{R}^{d_u}\right)} + \left\|\left[w^\dagger, b^\dagger\right]\right\|^2_{H^1\left(\frac{T}{2},T\right)} \\ &\leqslant \frac{C_1}{T} + C_2 T^{2(\alpha-1)} T, \end{aligned}$$

for some constants $C_1, C_2 > 0$ depending only on λ, T_0 and $\left[\widehat{w}, \widehat{b}\right]$. From this estimate, together with (7.18), we obtain, for $T > T_0$,

$$J_{\lambda,T}\left(w^\dagger, b^\dagger\right) \leqslant \log\left(1 + (m-1)\exp\left(-\gamma \exp\left(\frac{T^\alpha}{2}\right)\right)\right) + CT^{2\alpha-1},$$

for some constant $C > 0$ depending on λ, T_0, $\left[\widehat{w}, \widehat{b}\right]$, but independent of T. Using the above estimate, we may conclude from the optimality of $[w_T, b_T]$, as

$$\begin{aligned}\mathcal{E}\left(\mathbf{x}_T(T)\right) \leqslant J_{\lambda,T}\left(w_T, b_T\right) &\leqslant J_{\lambda,T}\left(w^\dagger, b^\dagger\right) \\ &\leqslant \log\left(1 + (m-1)\exp\left(-\gamma\,\exp\left(\frac{T^\alpha}{2}\right)\right)\right) + CT^{2\alpha-1}.\end{aligned}$$

□

7.4. **Proof of Theorem 4.1.** The proof of Theorem 4.1 is rather involved – we will first require the following lemmas. We will focus on System (3.2), but with small adaptations the proofs (except the exponential stability estimate for the parameters) can be shown for systems of the form (3.9). System (3.3) is addressed in [Esteve et al., 2020].

Lemma 7.3 (Grönwall-like estimate)**.** *Let $T > 0$ be given, and let $\overline{\mathbf{x}} \in \mathbb{R}^{d_x}$. For any $[w, b] \in L^1(0, T; \mathbb{R}^{d_u})$ and $\mathbf{x}^0 \in \mathbb{R}^{d_x}$, let $\mathbf{x} \in C^0([0, T]; \mathbb{R}^{d_x})$ denote the unique solution to (3.2), noting (3.1). The following facts then hold.*

(i) There exist a couple of constants $C_1 = C_1\left(\sigma, \|\overline{\mathbf{x}}\|, \max\left\{1, \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\|\right\}, N\right) > 0$ and $C_2 = C_2(\sigma, \|\overline{\mathbf{x}}\|, N) > 0$ independent of T such that

$$\|\mathbf{x}(t) - \overline{\mathbf{x}}\| \leqslant \mathfrak{C}\left(\left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\| + \left\|[w, b]\right\|_{L^1(0,T;\mathbb{R}^{d_u})} + \|\mathbf{x} - \overline{\mathbf{x}}\|_{L^2(0,T;\mathbb{R}^{d_x})}\right)$$

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

$$\mathfrak{C} := C_1 \max\left\{1, \left\|[w, b]\right\|_{L^1(0,T;\mathbb{R}^{d_u})}\right\} \exp\left(C_2\left\|w\right\|_{L^1(0,T;\mathbb{R}^{d\times d})}\right).$$

(ii) If moreover $[w, b] \in L^2(0, T; \mathbb{R}^{d_u})$, then

$$\|\mathbf{x}(t) - \overline{\mathbf{x}}\| \leqslant \mathfrak{C}\left(\left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\| + \left\|[w, b]\right\|_{L^2(0,T;\mathbb{R}^{d_u})} + \|\mathbf{x} - \overline{\mathbf{x}}\|_{L^2(0,T;\mathbb{R}^{d_x})}\right)$$

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

$$\mathfrak{C} := C_1 \max\left\{1, \left\|[w, b]\right\|_{L^2(0,T;\mathbb{R}^{d_u})}\right\} \exp\left(C_2\left\|w\right\|_{L^2(0,T;\mathbb{R}^{d\times d})}\right).$$

The proof of Lemma 7.3 may be found in the Appendix. Now suppose $\mathbf{x}^{\tau_\circ} \in \mathbb{R}^{d_x}$ is given. Let $T > 0$ and $\tau_\circ \in [0, T)$ be fixed, and consider the cost functional

$$J_{\tau_\circ,T}(w, b) := \frac{1}{N}\int_{\tau_\circ}^T \|\mathbf{x}(t) - \overline{\mathbf{x}}\|^2\,\mathrm{d}t + \lambda\left\|[w, b]\right\|^2_{L^2(\tau_\circ,T;\mathbb{R}^{d_u})}, \tag{7.19}$$

with $\mathbf{x}(\cdot)$ being the solution to

$$\begin{cases}\dot{\mathbf{x}}(t) = \mathfrak{f}\left([w(t), b(t)], \mathbf{x}(t)\right) & \text{in } (\tau_\circ, T) \\ \mathbf{x}(\tau_\circ) = \mathbf{x}^{\tau_\circ},\end{cases} \tag{7.20}$$

with $\mathfrak{f}$ as the dynamics in (3.3) or (3.2). We then have the following result.

Lemma 7.4 (Uniform estimate of optimal trajectories)**.** *Fix $\lambda > 0$, let $P \in \mathrm{Lip}(\mathbb{R}^d; \mathbb{R}^m)$ be any given surjective map, and let $\overline{\mathbf{x}} \in \mathbb{R}^{d_x}$ with $\overline{\mathbf{x}}_i \in P^{-1}(\{\vec{y}_i\})$ for $i \in [N]$ be arbitrary but fixed. Let $r > 0$ be fixed. Suppose that (3.3) (resp. (3.2) with σ 1–homogeneous) is controllable with linear cost in the sense of Definition 4.1. Then, there exists a constant*

$\mathfrak{C} = \mathfrak{C}(\sigma, \|\overline{\mathbf{x}}\|, \lambda, N, d, r) > 0$ *such that for all* $T > 0$, $\tau_\circ \in [0, T)$ *and* $\mathbf{x}^{\tau_\circ} \in \mathbb{R}^{d_x}$ *such that*

$$\|\mathbf{x}^{\tau_\circ} - \overline{\mathbf{x}}\| \leqslant r,$$

any pair of parameters $[w_T, b_T] \in L^2(\tau_\circ, T; \mathbb{R}^{d_u})$ *minimizing* $J_{\tau_\circ,T}$ *and corresponding solution* $\mathbf{x}_T(\cdot)$ *to* (7.20)*, with* $\mathfrak{f}$ *as in* (3.3) *(resp* (3.2) *with* σ *1–homogeneous), are such that*

$$\left\| [w_T, b_T] \right\|_{L^2(\tau_\circ,T;\mathbb{R}^{d_u})} + \|\mathbf{x}_T - \overline{\mathbf{x}}\|_{L^2(\tau_\circ,T;\mathbb{R}^{d_x})} + \|\mathbf{x}_T(t) - \overline{\mathbf{x}}\| \leqslant \mathfrak{C} \, \|\mathbf{x}^{\tau_\circ} - \overline{\mathbf{x}}\|$$

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

The proof of Lemma 7.4 follows precisely the arguments given in the first step of the proof of Theorem 4.2, in which one solely needs to change the initial time (0 by $\tau_\circ$) and datum ($\mathbf{x}^0$ by $\mathbf{x}^{\tau_\circ}$), the BV–norms by L^2, and note the precise form of the constant $\mathfrak{C}$ in (7.30). We give a brief sketch in the appendix for the sake of clarity.

We will also make use of the following short observation.

Lemma 7.5. *Let* $T > 0$ *and let* $[w_T, b_T] \in L^2(0, T; \mathbb{R}^{d_u})$ *be a pair of minimizers to* J_T *defined in* (4.2)*, and denote by* $\mathbf{x}_T$ *the corresponding solution to* (7.20) *on* $[0, T]$ *with* $\mathbf{x}_T(0) = \mathbf{x}^0$*. Let* $\tau_\circ \in [0, T)$ *be given. Then* $[w_T, b_T]|_{[\tau_\circ,T]}$ *minimize* $J_{\tau_\circ,T}$ *defined in* (7.19) *for System* (7.20) *with initial data* $\mathbf{x}^{\tau_\circ} = \mathbf{x}_T(\tau_\circ)$.

We refer to the appendix for a proof.

Remark 12. *We note that our proof is specific to* L^2*–regularization and does not transfer to* BV*–regularization a priori, due to the fact that the* BV *norm may see the singularities in discontinuous parameters. Therein lies the main impediment to ensuring the validity of Theorem 4.1 for* BV*–regularized problems by means of our strategy.*

Before concluding this section with a proof of Theorem 4.1, we also state the following key lemma.

Lemma 7.6. *Let* X *be a Banach space,* $T > a \geqslant 0$ *and* $f \in C^0([a, T]; X)$*. For any* $\tau \leqslant T - a$*, there exists* $t_1 \in [a, a + \tau)$ *such that*

$$\|f(t_1)\|_X \leqslant \frac{\|f\|_{L^2(a,T;X)}}{\sqrt{\tau}}.$$

The proof may be found in the appendix. We may now provide the proof of Theorem 4.1.

Proof of Theorem 4.1. We shall concentrate on the neural ODE (3.2). The proof of the full result for (3.3) is identical to that presented in [Esteve et al., 2020]. We split the proof in two parts.

Part 1: *Stability estimates for* $\mathcal{E}(\mathbf{x}_T(t)) + \|\mathbf{x}_T(t) - \overline{\mathbf{x}}\|$. For

$$r := \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\|,$$

denote by $\mathfrak{C}_1 = \mathfrak{C}_1(\sigma, \|\overline{\mathbf{x}}\|, \lambda, N, d, r) > 0$ the universal constant given by Lemma 7.4. Let

$$\tau > \max\left\{\mathfrak{C}_1^4, \mathfrak{C}_1^2\right\}$$

be fixed and let

$$T \geqslant \tau + 1.$$

First, note that by Lemma 7.4 (with $\tau_\circ = 0$, $r := \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\|$ and $\mathbf{x}^{\tau_\circ} = \mathbf{x}^0$), we have

$$\|\mathbf{x}_T - \overline{\mathbf{x}}\|_{L^2(0,T;\mathbb{R}^{d_x})} + \|\mathbf{x}_T(t) - \overline{\mathbf{x}}\| \leqslant \mathfrak{C}_1 \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\| \qquad \text{for } t \in [0, T]. \tag{7.21}$$

Now note that for $t \in [0, \tau+1]$, the desired exponential stability estimate can easily be obtained since the length of the time interval is independent of T. Indeed, from (7.21) we get

$$\begin{aligned}\|\mathbf{x}_T(t) - \overline{\mathbf{x}}\| &\leqslant \mathfrak{C}_1 \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\| \exp(t)\exp(-t)\\ &\leqslant \mathfrak{C}_1 \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\| \exp(\tau+1)\exp(-t),\end{aligned}$$

and, since $\overline{\mathbf{x}}_i \in P^{-1}(\{\vec{y}_i\})$ for $i \in [N]$, using this estimate we also find

$$\begin{aligned}\mathcal{E}(\mathbf{x}_T(t)) \leqslant \frac{c}{N}\sum_{i=1}^N \left\|P\mathbf{x}_{T,i}(t) - \vec{y}_i\right\|^\alpha &\leqslant \frac{c}{N}\|P\|^\alpha \sum_{i=1}^N \|\mathbf{x}_{T,i}(t) - \overline{\mathbf{x}}_i\|^\alpha\\ &\lesssim \mathfrak{C}_1^\alpha \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\|^\alpha \exp(\alpha(\tau+1))\exp(-\alpha t).\end{aligned}$$

Thus, only the case $t \in [\tau+1, T]$ remains. To do so, we proceed in three steps.

Step 1). **Preparation.** Since $\tau \leqslant T$, using Lemma 7.6 and then Lemma 7.4 (with $\tau_\circ = 0$, $r := \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\|$ and $\mathbf{x}^{\tau_\circ} = \mathbf{x}^0$) we see that there exists $\tau_\circ \in [0, \tau)$ such that

$$\|\mathbf{x}_T(\tau_\circ) - \overline{\mathbf{x}}\| \leqslant \frac{\|\mathbf{x}_T - \overline{\mathbf{x}}\|_{L^2(0,T;\mathbb{R}^{d_x})}}{\sqrt{\tau}} \leqslant \frac{\mathfrak{C}_1}{\sqrt{\tau}}\left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\|. \tag{7.22}$$

By Lemma 7.5, the parameters $[w_T, b_T]|_{[\tau_\circ,T]}$ minimize the functional $J_{\tau_\circ,T}$ for System (7.20) with initial data $\mathbf{x}^{\tau_\circ} = \mathbf{x}_T(\tau_\circ)$, to which the solution is precisely $\mathbf{x}_T|_{[\tau_\circ,T]}$. Now, applying Lemma 7.4 (this time with with $\tau_\circ$ as in (7.22), $\mathbf{x}^{\tau_\circ} = \mathbf{x}_T(\tau_\circ)$ and with $r = \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\|$, as $\mathfrak{C}_1/\sqrt{\tau} < 1$) in combination with (7.22), yields

$$\|\mathbf{x}_T(t) - \overline{\mathbf{x}}\| \leqslant \mathfrak{C}_1 \|\mathbf{x}_T(\tau_\circ) - \overline{\mathbf{x}}\| \leqslant \frac{\mathfrak{C}_1^2}{\sqrt{\tau}}\left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\|, \tag{7.23}$$

which holds for all $t \in [\tau_\circ, T]$. Since $\tau_\circ < \tau$, (7.23) also holds for $t \in [\tau, T]$.

Step 2). **Bootstrap.** We iterate (7.23) and show that for any $n \in \mathbb{N}$ satisfying $n \leqslant T/\tau$, the following estimate holds:

$$\|\mathbf{x}_T(t) - \overline{\mathbf{x}}\| \leqslant \left(\frac{\mathfrak{C}_1^2}{\sqrt{\tau}}\right)^n \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\| \qquad \text{for } t \in [n\tau, T]. \tag{7.24}$$

We proceed by induction – the case $n = 1$ holds by (7.23). Thus suppose that (7.24) holds at some stage $n \geqslant 2$ and suppose that $n+1 \leqslant T/\tau$. Now the parameters $[w_T, b_T]|_{[n\tau,T]}$ minimize $J_{n\tau,T}$ by Lemma 7.5, and so we can apply Lemma 7.4 (with $\tau_\circ = n\tau$, $\mathbf{x}^{\tau_\circ} = \mathbf{x}_T(n\tau)$, and with $r = \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\|$ as $\mathfrak{C}_1^2/\sqrt{\tau} < 1$ and (7.24) is assumed to hold) in combination with Lemma 7.6 (as $n+1 \leqslant T/\tau$ clearly implies that $\tau \leqslant T - n\tau$) to deduce that there exists some time $t_1 \in [n\tau, (n+1)\tau)$ such that

$$\|\mathbf{x}_T(t_1) - \overline{\mathbf{x}}\| \leqslant \frac{\|\mathbf{x}_T - \overline{\mathbf{x}}\|_{L^2(n\tau,T;\mathbb{R}^{d_x})}}{\sqrt{\tau}} \leqslant \frac{\mathfrak{C}_1}{\sqrt{\tau}}\|\mathbf{x}_T(n\tau) - \overline{\mathbf{x}}\|.$$

We once again apply (7.24) in the inequality just above to deduce

$$\|\mathbf{x}_T(t_1) - \overline{\mathbf{x}}\| \leqslant \frac{\mathfrak{C}_1}{\sqrt{\tau}}\left(\frac{\mathfrak{C}_1^2}{\sqrt{\tau}}\right)^n \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\|. \tag{7.25}$$

We then apply Lemma 7.4 for a second time (with $\tau_\circ = t_1$, $\mathbf{x}^{\tau_\circ} = \mathbf{x}_T(t_1)$ and with $r = \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\|$, as we have (7.25) with $\max\{\mathfrak{C}_1, \mathfrak{C}_1^2\}/\sqrt{\tau} < 1$) and use (7.25) to find

$$\|\mathbf{x}_T(t) - \overline{\mathbf{x}}\| \leqslant \mathfrak{C}_1 \|\mathbf{x}_T(t_1) - \overline{\mathbf{x}}\| \leqslant \left(\frac{\mathfrak{C}_1^2}{\sqrt{\tau}}\right)^{n+1} \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\|,$$

which holds for all $t \in [t_1, T]$. Clearly, as $t_1 < (n+1)\tau$, the above estimate also holds for all $t \in [(n+1)\tau, T]$. This completes the proof of (7.24).

Step 3). **Conclusion.** Suppose $t \in [\tau + 1, T]$ is arbitrary and fixed. Set $n(t) := \left\lfloor \frac{t}{\tau+1} \right\rfloor$; note that $n(t) \geqslant 1$, $t \geqslant n(t)\tau$ and $n(t) \leqslant T/\tau$. We may then apply (7.24) to find that

$$\|\mathbf{x}_T(t) - \overline{\mathbf{x}}\| \leqslant \left(\frac{\mathfrak{C}_1^2}{\sqrt{\tau}}\right)^{n(t)} \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\|.$$

Since $\tau > \mathfrak{C}_1^4$ and $n(t) \geqslant \frac{t}{\tau+1} - 1$, from the above inequality we infer

$$\begin{aligned}\|\mathbf{x}_T(t) - \overline{\mathbf{x}}\| &\leqslant \exp\left(-n(t)\log\left(\frac{\sqrt{\tau}}{\mathfrak{C}_1^2}\right)\right) \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\| \\ &\leqslant \frac{\sqrt{\tau}}{\mathfrak{C}_1^2} \exp\left(-\frac{\log\left(\frac{\sqrt{\tau}}{\mathfrak{C}_1^2}\right)}{\tau+1} t\right) \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\|.\end{aligned}$$

The desired exponential stability estimate for $\|\mathbf{x}_T(t) - \overline{\mathbf{x}}\|$ thus also holds for $t \in [\tau+1, T]$, with $\mu := \frac{\log\left(\frac{\sqrt{\tau}}{\mathfrak{C}_1^2}\right)}{\tau+1} > 0$ and $\mathfrak{C} := \frac{\sqrt{\tau}}{\mathfrak{C}_1^2} \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\|$. Note that, arguing as in the previous case,

$$\mathcal{E}(\mathbf{x}_T(t)) \lesssim \frac{1}{N} \sum_{i=1}^{N} \left\|P\mathbf{x}_{T,i}(t) - \vec{y}_i\right\|^\alpha \lesssim \mathfrak{C}^\alpha \exp(-\alpha\mu t).$$

This concludes the proof of the first part.

Part 2: *Stability estimate for the parameters.* The stability estimate for the parameters in the setting of (3.2) closely follows the proof presented in [Esteve et al., 2020] – we give a sketch of the main ideas. We henceforth interchange between the notation $u := [w, b]$ and $[w, b]$ to ease the reading.

Fix an arbitrary $t \in [0, T)$ and $0 < h \ll 1$, so that $t + 2h^2 + 2h \in [0, T]$, and set

$$u^{\text{aux}}(s) := \begin{cases} u_T(s) & \text{for } s \in [0, t] \\ \frac{1}{2} u_T\left(t + \frac{s-t}{2}\right) & \text{for } s \in (t, t + 2h^2] \\ \frac{h+2}{2} u_T\left(\left(\frac{h+2}{2}\right) s - \frac{h+2}{2}(t + 2h^2) + t + h^2\right) & \text{for } s \in (t + 2h^2, t + 2h^2 + 2h] \\ u_T(s) & \text{for } s \in (t + 2h^2 + 2h, T]. \end{cases}$$

From Lemma 7.1 that the solution $\mathbf{x}^{\text{aux}}$ to (3.2) associated to the above pair is precisely

$$\mathbf{x}^{\text{aux}}(s) := \begin{cases} \mathbf{x}_T(s) & \text{for } \ s \in [0,t] \\ \mathbf{x}_T\left(t + \dfrac{s-t}{2}\right) & \text{for } \ s \in (t, t+2h^2] \\ \mathbf{x}_T\left(\left(\dfrac{h+2}{2}\right)s - \dfrac{h+2}{2}(t+2h^2) + t + h^2\right) & \text{for } \ s \in (t+2h^2, \\ & \qquad\qquad t+2h^2+2h] \\ \mathbf{x}_T(s) & \text{for } s \in (t+2h^2+2h, T]. \end{cases}$$

This specific construction is in particular done as to ensure that $\mathbf{x}^{\text{aux}}(T) = \mathbf{x}_T(T)$ and so $\mathcal{E}(\mathbf{x}^{\text{aux}}(T)) = \mathcal{E}(\mathbf{x}_T(T))$. Now by several straightforward computations (which may be found in [Esteve et al., 2020]) we deduce that

$$\begin{aligned} J_T(u^{\text{aux}}) &= \mathcal{E}(\mathbf{x}_T(T)) + \frac{1}{N}\int_0^T \|\mathbf{x}_T(s) - \overline{\mathbf{x}}\|^2 \,\mathrm{d}s + \frac{1}{N}\int_t^{t+h^2} \|\mathbf{x}_T(s) - \overline{\mathbf{x}}\|^2 \,\mathrm{d}s \\ &\quad + \frac{1}{N}\left(\frac{2}{h+2} - 1\right)\int_{t+h^2}^{t+2h^2+2h} \|\mathbf{x}_T(s) - \overline{\mathbf{x}}\|^2 \,\mathrm{d}s \\ &\quad + \lambda\int_0^T \|u_T(s)\|^2 \,\mathrm{d}s - \frac{\lambda}{2}\int_t^{t+h} \|u_T(s)\|^2 \,\mathrm{d}s + \lambda\frac{h}{2}\int_{t+h^2}^{t+h^2+2h} \|u_T(s)\|^2 \,\mathrm{d}s \\ &\leqslant \mathcal{E}(\mathbf{x}_T(T)) + \frac{1}{N}\int_0^T \|\mathbf{x}_T(s) - \overline{\mathbf{x}}\|^2 \,\mathrm{d}s + \frac{1}{N}\int_t^{t+h} \|\mathbf{x}_T(s) - \overline{\mathbf{x}}\|^2 \,\mathrm{d}s \\ &\quad + \lambda\int_0^T \|u_T(s)\|^2 \,\mathrm{d}s - \frac{\lambda}{2}\int_t^{t+h} \|u_T(s)\|^2 \,\mathrm{d}s + \lambda\frac{h}{2}\int_{t+h^2}^{t+h^2+2h} \|u_T(s)\|^2 \,\mathrm{d}s. \end{aligned}$$

The above identity combined with the optimality inequality $J_T(u_T) \leqslant J_T(u^{\text{aux}})$ leads us to

$$\frac{\lambda}{2}\int_t^{t+h} \|u_T(s)\|^2 \,\mathrm{d}s \leqslant \frac{1}{N}\int_t^{t+h} \|\mathbf{x}_T(s) - \overline{\mathbf{x}}\|^2 \,\mathrm{d}s + \lambda\frac{h}{2}\int_{t+h^2}^{t+h^2+2h} \|u_T(s)\|^2 \,\mathrm{d}s.$$

Using the exponential stability estimate for $\|\mathbf{x}_T(\cdot) - \overline{\mathbf{x}}\|$, we may find

$$\begin{aligned} \frac{\lambda}{2}\frac{1}{h}\int_t^{t+h} \|u_T(s)\|^2 \,\mathrm{d}s &\leqslant \frac{1}{N}\frac{1}{h}\int_t^{t+h} \|\mathbf{x}_T(s) - \overline{\mathbf{x}}\|^2 \,\mathrm{d}s + \frac{\lambda}{2}\int_{t+h^2}^{t+h^2+2h} \|u_T(s)\|^2 \,\mathrm{d}s \\ &\leqslant \frac{\mathfrak{C}}{N}\frac{1}{h}\int_t^{t+h} e^{-2\mu s} \,\mathrm{d}s + \frac{\lambda}{2}\int_{t+h^2}^{t+h^2+2h} \|u_T(s)\|^2 \,\mathrm{d}s \\ &\leqslant \frac{\mathfrak{C}}{N}e^{-2\mu t} + \frac{\lambda}{2}\int_{t+h^2}^{t+h^2+2h} \|u_T(s)\|^2 \,\mathrm{d}s. \end{aligned}$$

By using the Lebesgue dominated convergence theorem (applied to the integrable function $s \mapsto \|u_T(s)\|^2 \mathbf{1}_{(t+h^2, t+h^2+2h)}(s)$) the second integral in the estimate just above goes to 0 when $h \searrow 0$. Hence, by applying the Lebesgue differentiation theorem in the estimate just above, we deduce that

$$\left\|[w_T(t), b_T(t)]\right\|^2 = \|u_T(t)\|^2 = \lim_{h\searrow 0} \frac{1}{h}\int_t^{t+h} \|u_T(s)\|^2 \,\mathrm{d}s \leqslant \frac{\mathfrak{C}}{2N\lambda}e^{-2\mu t}$$

for a.e. $t \in [0,T]$, as desired. □

7.5. **Proof of Theorem 4.2.**

Proof of Theorem 4.2. We will interchange the notations $[w, b]$ and $u := [w, b]$ for simplicity, and we split the proof in two parts. We henceforth set $r := \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\|$.

Part 1: *Uniform estimates.* We shall first establish uniform-in-T estimates – we find some $\mathfrak{C} > 0$ independent of T such that

$$\|\mathbf{x}_T(t) - \overline{\mathbf{x}}\| + \|\mathbf{x}_T - \overline{\mathbf{x}}\|_{L^2(0,T;\mathbb{R}^{d_x})} + \left\|[w_T, b_T]\right\|_{\mathrm{BV}(0,T;\mathbb{R}^{d_u})} \leqslant \mathfrak{C} \tag{7.26}$$

whenever $T \geqslant 1$ and for $t \in [0, T]$. By virtue of the controllability assumption, there exist parameters $u^\dagger \in C^0([0, 1]; \mathbb{R}^{d_u}) \cap \mathrm{BV}([0, 1]; \mathbb{R}^{d_u})$ such that the corresponding solution $\mathbf{x}^\dagger(\cdot)$ to (3.2) on $(0, 1)$ satisfies $\mathbf{x}^\dagger(1) = \overline{\mathbf{x}}$. By integrating the equation satisfied by $\mathbf{x}^\dagger(\cdot)$, we see that for $t \in [0, 1]$,

$$\begin{aligned}\left\|\mathbf{x}^\dagger(t) - \overline{\mathbf{x}}\right\| \leqslant \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\| + c(\sigma)\Big(\int_0^t \left\|\mathbf{w}^\dagger(s)\right\| \left\|\mathbf{x}^\dagger(s) - \overline{\mathbf{x}}\right\| \,\mathrm{d}s \\ + \|\overline{\mathbf{x}}\| \int_0^t \left\|\mathbf{w}^\dagger(s)\right\| \,\mathrm{d}s + \int_0^t \left\|\mathbf{b}^\dagger(s)\right\| \,\mathrm{d}s\Big)\end{aligned}$$

where $c(\sigma) > 0$ is the Lipschitz constant of σ. By virtue of Grönwall's inequality and (4.5), we deduce that

$$N^{-1/2} \left\|\mathbf{x}^\dagger(t) - \overline{\mathbf{x}}\right\| \leqslant C_1 \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\| \exp\Big(C_1 \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\|\Big) \tag{7.27}$$

for $t \in [0, 1]$ and for some constant $C_1 = C_1(\sigma, \|\overline{\mathbf{x}}\|, r) > 0$ independent of T (as well as and N, and only depends on $\mathbf{x}^0$ via r). We now set (recall that $T \geqslant 1$)

$$u^{\mathrm{aux}}(t) := \begin{cases} u^\dagger(t) & \text{in } (0, 1) \\ 0 & \text{in } (1, T), \end{cases}$$

and we denote by $\mathbf{x}^{\mathrm{aux}}(\cdot)$ the corresponding solution to (3.2). One notes that $u^{\mathrm{aux}} \in L^2(0, T; \mathbb{R}^{d_u}) \cap \mathrm{BV}(0, T; \mathbb{R}^{d_u})$, as the jump at $t = 1$ only accounts to a Dirac mass. Moreover, $\mathbf{x}^{\mathrm{aux}}(t) = \overline{\mathbf{x}}$ for $t \in [1, T]$ and thus also $P\mathbf{x}_i^{\mathrm{aux}}(T) = \vec{y}_i$ for $i \in [N]$, which implies $\mathcal{E}(\mathbf{x}^{\mathrm{aux}}(T)) = 0$. By using the optimality inequality $J_T(u_T) \leqslant J_T(u^{\mathrm{aux}})$ and the fact that $\mathcal{E}(\mathbf{x}^{\mathrm{aux}}(T)) = 0$ and $\mathcal{E} \geqslant 0$, we thus deduce that

$$\begin{aligned} &N^{-1}\left\|\mathbf{x}_T - \overline{\mathbf{x}}\right\|^2_{L^2(0,T;\mathbb{R}^{d_x})} + \lambda\left\|u_T\right\|^2_{\mathrm{BV}(0,T;\mathbb{R}^{d_u})} \\ &\leqslant N^{-1}\left\|\mathbf{x}^{\mathrm{aux}} - \overline{\mathbf{x}}\right\|^2_{L^2(0,T;\mathbb{R}^{d_x})} + \lambda\left\|u^{\mathrm{aux}}\right\|^2_{\mathrm{BV}(0,T;\mathbb{R}^{d_u})} \\ &\leqslant N^{-1}\left\|\mathbf{x}^\dagger - \overline{\mathbf{x}}\right\|^2_{L^2(0,1;\mathbb{R}^{d_x})} + 2\lambda\left\|u^\dagger\right\|^2_{\mathrm{BV}(0,1;\mathbb{R}^{d_u})} + 2\lambda(d^2 + d)\sum_{j=1}^{d^2+d}\left|u_j^\dagger(1)\right|^2, \end{aligned} \tag{7.28}$$

where we used the vectorized form of $u^\dagger$ in the second component of the BV–norm whilst maintaining the same notation (recall that $d_u := d \times (d + 1)$). Using (7.27) and

(4.5), we find

$$N^{-1}\left\|\mathbf{x}^\dagger-\overline{\mathbf{x}}\right\|^2_{L^2(0,1;\mathbb{R}^{d_x})}+2\lambda\left\|u^\dagger\right\|^2_{\mathrm{BV}(0,1;\mathbb{R}^{d_u})}+2\lambda(d^2+d)\sum_{j=1}^{d^2+d}\left|u_j^\dagger(1)\right|^2 \leqslant \underbrace{C_2\left\|\mathbf{x}^0-\overline{\mathbf{x}}\right\|^2\exp\left(2C_1\left\|\mathbf{x}^0-\overline{\mathbf{x}}\right\|\right)}_{:=\mathfrak{C}_0^2} \tag{7.29}$$

for some constant $C_2=C_2(\sigma,\|\overline{\mathbf{x}}\|,r,d,\max\{1,\lambda\})>0$. Combining (7.28) and (7.29), and recalling Lemma 7.3, we may conclude that (7.26) holds with

$$\mathfrak{C}:=C_3\max\left\{1,\frac{\mathfrak{C}_0}{\lambda}\right\}\exp\left(C_4\frac{\mathfrak{C}_0}{\lambda}\right)\left\|\mathbf{x}^0-\overline{\mathbf{x}}\right\| \tag{7.30}$$

for some constants $C_3=C_3(\sigma,\|\overline{\mathbf{x}}\|,r,N,\lambda)>0$ and $C_4=C_4(\sigma,\|\overline{\mathbf{x}}\|,N)>0$.

Part 2: *Conclusion.* We note that the desired stability estimates thus follow from (7.26), and, since $\overline{\mathbf{x}}_i\in P^{-1}(\{\vec{y}_i\})$ for $i\in[N]$, we also have

$$\mathcal{E}(\mathbf{x}_T(t))\leqslant N^{-1}\sum_{i=1}^N\left\|P\mathbf{x}_{T,i}(t)-\vec{y}_i\right\|^\alpha\leqslant N^{-1}\|\mathbf{x}_T(t)-\overline{\mathbf{x}}\|^\alpha.$$

We conclude the proof by noting that the convergence of averages follows by dividing both sides in (7.26) by T, using the estimate just above for the training error term, and letting $T\longrightarrow\infty$. □

7.6. **Proof of Proposition 5.1.** The proof of Proposition 5.1 is a straightforward Grönwall argument. We sketch it for completeness.

Proof of Proposition 5.1. For simplicity of presentation but without any loss of generality, we will henceforth concentrate on system (3.3). For any $t\in[0,T]$, $i\in[N]$ and $j\in[N]$, 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

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

We apply the Grönwall inequality with the effect of

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

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

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

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

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

By taking the log on both sides we obtain (5.2). □

7.7. **Proof of Theorem 5.1.** The following short functional analysis lemma will be of use in the proof of Theorem 5.1. We omit the proof, which follows by using the open mapping theorem (see e.g. [Brezis, 2010, Theorem 2.6, pp. 35]).

Lemma 7.7. *Let $\mathcal{H}_1$ and $\mathcal{H}_2$ be two real Hilbert spaces. Let*

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

be a linear, bounded and surjective operator. Then

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

is linear and bounded.

Proof of Theorem 5.1. Inspired by the techniques in [Coron and Trélat, 2004; Pighin and Zuazua, 2018], 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, there exists an $\eta > 0$, such that whenever $\left\|\mathbf{x}^1 - \mathbf{x}^0\right\| \leqslant \eta$,

$$\left\{\sigma\left(\gamma_1(s)\right), \ldots, \sigma\left(\gamma_i(s)\right), \ldots, \sigma\left(\gamma_N(s)\right)\right\} \tag{7.31}$$

is also a system of linearly independent vectors in $\mathbb{R}^d$ for any $s \in [0,1]$. Following the framework of Lemma 7.7, for any $s \in [0,1]$, define

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

By the linear independence of the system of vectors (7.31), Λ_s is surjective for any $s \in [0,1]$. Hence, using Lemma 7.7, we see that

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

is a linear and bounded operator for any $s \in [0,1]$, and, since (7.31) is independent and the arc γ is continuous, $\{\Gamma_s\}_{s\in[0,1]}$ is uniformly bounded in operator norm:

$$\|\Gamma_s\|_{\mathcal{L}\left(\mathbb{R}^{d_x};\mathbb{R}^{d\times d}\right)} \leqslant C \tag{7.32}$$

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

$$w(t) := \Gamma_{s_t}\left(\frac{\mathbf{x}^1 - \mathbf{x}^0}{T}\right), \tag{7.33}$$

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

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

where

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

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 $i \in [N]$. 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 conclude by showing that w satisfies the stated estimate. By the definition of w in (7.33) as well as (7.32), for any $t \in [0, T]$ we have

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

as desired. □

Remark 13 (Removing the smallness assumption)**.** *One could perhaps adapt the argument in the proof of Theorem 5.1 (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 Proposition 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 $i \neq j$.

8. Concluding remarks

In this work, we have addressed the impact of the final time horizon T in general learning problems for neural ODEs.

- In the empirical risk minimization problem with a Tikhonov (L^2 or H^1) parameter regularization, we concluded via Theorem 3.1 – Theorem 3.3 that when T is large enough, the obtained optimal/trained parameters for neural ODEs are such that the corresponding trajectories reach zero training error with a quantitative rate (thus, stipulate an approximation property of the trained model with respect to T), whilst doing so with the least oscillations possible. 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. In more practical terms, to ensure that the global minimizer is near zero training error, while training, one could systematically increase the time horizon T

whilst keeping the regularization parameter $\lambda > 0$ fixed. Moreover, roughly the same conclusions hold when T is fixed and λ is rendered small, thereby linking our insights with the literature on the regularization path limit for various machine learning models.

- To obtain better quantitative estimates on the time horizon (and thus, number of layers when the time-step is fixed, e.g. in ResNets) required to be ε–close to the zero training error regime, for a given tolerance $\varepsilon > 0$, we introduced a minimization problem wherein we added a tracking term which regularizes the state trajectories over the entire time horizon. In Theorem 4.1, we show that the training error and the optimal parameters are at most of the order $\mathcal{O}\left(e^{-\mu t}\right)$ for all $t \in [0, T]$. This result, along with numerical experiments, demonstrates a strong approximation rate of the trained neural ODE flow (which ought to be compared with universal approximation results, in which, a key caveat is that there is no scalable method to compute the theoretically guaranteed parameters), with parameters which are exponentially small, and could thus stipulate that the flow would tend to oscillate little. Moreover, the exponential decay estimate also ensures that T need not be chosen too large to render the training error small.

8.1. **Outlook.** We present a list of questions and topics which would be complementary to our work.

Generalization bounds. To complement our analytical study on the long time horizon/large layer regime, it would be of interest to provide generalization error bounds for the limiting, least L^2–norm parameters in the interpolation regime obtained in Theorem 3.1, via, for instance, commonly used metrics such as the VC dimension [Vapnik, 2013] or Rademacher complexity [Bartlett and Mendelson, 2002]. Such studies are, to the best of our knowledge, not done in the context of models such as neural ODEs.

Stability estimate for (4.7). We provided a proof of the exponential stability estimate of the training error and optimal parameters in the context of ℓ^2–like losses, and without regularizing the output $P\mathbf{x}_i(t)$ but rather the features $\mathbf{x}_i(t)$ over all time/layer $t \in [0, T]$. We could stipulate that, whenever P is Lipschitz (and possibly real analytic) and such that the training error attains its minimum (e.g. when P is a matrix, or a matrix composed with a sigmoid truncated by some cut-off function), the exponential stability result could hold by making use of the Łojasiewicz inequality: for a compact set $K \subset \mathbb{R}^d$ and two continuous, sub-analytic functions $\mathfrak{g}, \mathfrak{h} : K \longrightarrow \mathbb{R}$, if $\mathfrak{g}^{-1}(\{0\}) \subset \mathfrak{h}^{-1}(\{0\})$ then

$$\|\mathfrak{h}(x)\|^{\alpha} \leqslant c\|\mathfrak{g}(x)\| \qquad \text{for all } x \in K,$$

holds for some $c > 0$ and $\alpha \in \mathbb{N}$ (see [Lojasiewicz, 1959] and also [Bierstone and Milman, 1988, Theorem 6.4]). This however remains an open problem. On the other hand, addressing analytically the (exponential) stability stipulated by the numerical experiments presented herein for non ℓ^2–losses such as cross-entropy also remains open.

A pre-training algorithm. Note that the insight from Theorem 4.1 and the experiments performed for (4.7) could motivate the following algorithm, which trains a shallower network (i.e., a neural ODE with a shorter time horizon) at first, and then successively increases this time horizon in an iterative manner.

Algorithm 1: A pre-training algorithm.

Result: T, $[w_T, b_T]$
fix $\varepsilon > 0$, $T_\circ > 0$;
$j \leftarrow 1$;
Find $[w_{jT_\circ}, b_{jT_\circ}]$ and $\mathbf{x}_{jT_\circ}$ by solving

$$\inf_{[w,b]\in H^k((j-1)T_\circ, jT_\circ;\mathbb{R}^{d_u})} \int_{(j-1)T_\circ}^{jT_\circ} \mathcal{E}(\mathbf{x}(t))\,\mathrm{d}t + \Big\|[w,b]\Big\|^2_{H^k((j-1)T_\circ, jT_\circ;\mathbb{R}^{d_u})} \tag{8.1}$$

subject to

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathfrak{f}\Big([w(t), b(t)], \mathbf{x}(t)\Big) & \text{in } \ ((j-1)T_\circ, jT_\circ) \\ \mathbf{x}((j-1)T_\circ) = \mathbf{x}^{(j-1)T_\circ}. \end{cases} \tag{8.2}$$

$[w_T, b_T]_{|((j-1)T_\circ, jT_\circ)} \leftarrow [w_{jT_\circ}, b_{jT_\circ}]$;
$\mathbf{x}^j \leftarrow \mathbf{x}_{jT_\circ}(jT_\circ)$;
$T \leftarrow jT_\circ$;
while $\mathcal{E}(\mathbf{x}(jT_\circ)) > \varepsilon$ **do**
  $j \leftarrow j + 1$;
  find $[w_{jT_\circ}, b_{jT_\circ}]$ and $\mathbf{x}_{jT_\circ}$ by solving (8.1) – (8.2);
  $[w_T, b_T]_{|((j-1)T_\circ, jT_\circ)} \leftarrow [w_{jT_\circ}, b_{jT_\circ}]$;
  $\mathbf{x}^{jT_\circ} \leftarrow \mathbf{x}_{jT_\circ}(jT_\circ)$;
  $T \leftarrow jT_\circ$
end

One of the most distinguished characteristics of (4.2) 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 ResNets, 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 artifact, and represents a *greedy algorithm* which uses only the number of layers strictly needed, thus avoiding unnecessary ones. We leave open the possible numerical analysis of the above algorithm (see [Goodfellow et al., 2016, Chapter 15] for some related literature on pre-training algorithms).

More general models. It would be of interest to directly investigate the appearance of the phenomena presented herein to neural ODEs with state of the art configurations, including convolutional, batch-normalization and max-pooling layers – we have solely focused our theoretical analysis to the basic settings. Further extensions could include the study of ResNet variants such as MomentumNets [Sander et al., 2021], which are second order ODEs wherein $\dot{\mathbf{x}}$ is seen as a damping term, and mean field variants [Weinan et al., 2019].

Unsupervised learning. As discussed in the introduction, the neural ODE representation of deep supervised learning, due to the invertibility of the neural ODE flow, has seen fruitful applications in generative modeling via continuous normalizing flows (see [Ruthotto and Haber, 2021]). In generative modeling and unsupervised learning, one aims to infer the probability distribution of the inputs $\{\vec{x}_i\}_{i\in[N]}$ rather than, for instance, draw a decision boundary as in classification tasks via supervised learning. It would be of interest, in view of the existing applications, to investigate the potential use of the results presented in this work to the context of unsupervised learning.

Acknowledgments. B.G. acknowledges Daniel Tenbrinck and Lukas Pflug (FAU Erlangen-Nürnberg) for discussions on the foundations of neural networks and non-local equations.

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.

Appendix A. Auxiliary proofs

Proof of Lemma 7.2. The proof is mostly identical in for all of the three items, the difference being an underlying compact embedding.

Proof of *(i)*. Let $\{[w_n, b_n]\}_{n=1}^{\infty} \subset L^2(0,T;\mathbb{R}^{d_u})$ be a bounded sequence in $L^2(0,T;\mathbb{R}^{d_u})$. By the Banach-Alaoglu theorem, there exists a pair $\left[w^\dagger, b^\dagger\right] \in L^2(0,T;\mathbb{R}^{d_u})$ such that, along some subsequence,

$$[w_n, b_n] \xrightarrow[n\longrightarrow\infty]{} \left[w^\dagger, b^\dagger\right] \qquad \text{weakly in } L^2(0,T;\mathbb{R}^{d_u}).$$

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 $\mathbf{b}_n := [b_n, \dots, b_n]$ to $\mathbf{b}^\dagger := [b^\dagger, \dots, b^\dagger]$. Let $\mathbf{x}^\dagger \in C^0([0,T];\mathbb{R}^{d_x})$ be the unique solution to (3.3) associated to $[w^\dagger, b^\dagger]$ and the initial datum $\mathbf{x}^0$. Let us prove that

$$\mathbf{x}_n \xrightarrow[n\longrightarrow\infty]{} \mathbf{x}^\dagger \qquad \text{strongly in } C^0([0,T];\mathbb{R}^{d_x}) \tag{A.1}$$

along the aforementioned subsequence. Take an arbitrary $t \in [0,T]$. Note that

$$\begin{aligned}
\mathbf{x}_n(t) - \mathbf{x}^\dagger(t) &= \int_0^t \Big[\mathbf{w}_n(\tau)\sigma(\mathbf{x}_n(\tau)) + \mathbf{b}_n(\tau)\Big]\,\mathrm{d}\tau - \int_0^t \Big[\mathbf{w}^\dagger(\tau)\sigma\left(\mathbf{x}^\dagger(\tau)\right) + \mathbf{b}^\dagger(\tau)\Big]\,\mathrm{d}\tau \\
&= \int_0^t \Big[\mathbf{w}_n(\tau)\sigma(\mathbf{x}_n(\tau)) - \mathbf{w}_n(\tau)\sigma\left(\mathbf{x}^\dagger(\tau)\right)\Big]\,\mathrm{d}\tau \\
&\quad + \int_0^t \Big[\mathbf{w}_n(\tau)\sigma\left(\mathbf{x}^\dagger(\tau)\right) - \mathbf{w}^\dagger(\tau)\sigma\left(\mathbf{x}^\dagger(\tau)\right)\Big]\,\mathrm{d}\tau \\
&\quad + \int_0^t \Big[\mathbf{b}_n(\tau) - \mathbf{b}^\dagger(\tau)\Big]\,\mathrm{d}\tau.
\end{aligned}$$

Hence, using the fact that σ is globally Lipschitz with constant $c(\sigma) > 0$,

$$\begin{aligned}
\left\|\mathbf{x}_n(t) - \mathbf{x}^\dagger(t)\right\| &\leqslant \int_0^t \|\mathbf{w}_n(\tau)\| \left\|\sigma\Big(\mathbf{x}_n(\tau)\Big) - \sigma\left(\mathbf{x}^\dagger(\tau)\right)\right\|\,\mathrm{d}\tau \\
&\quad + \left\|\int_0^t \sigma\left(\mathbf{x}^\dagger(\tau)\right)\Big[\mathbf{w}_n(\tau) - \mathbf{w}^\dagger(\tau)\Big]\,\mathrm{d}\tau\right\| \\
&\quad + \left\|\int_0^t \Big[\mathbf{b}_n(\tau) - \mathbf{b}^\dagger(\tau)\Big]\,\mathrm{d}\tau\right\| \\
&\leqslant c(\sigma)\int_0^t \|\mathbf{w}_n(\tau)\| \left\|\mathbf{x}_n(\tau) - \mathbf{x}^\dagger(\tau)\right\|\,\mathrm{d}\tau + c_n,
\end{aligned}$$

with

$$c_n := \left\| \int_0^t \sigma\left(\mathbf{x}^\dagger(\tau)\right)\left[\mathbf{w}_n(\tau) - \mathbf{w}^\dagger(\tau)\right] \mathrm{d}\tau \right\| + \left\| \int_0^t \left[\mathbf{b}_n(\tau) - \mathbf{b}^\dagger(\tau)\right] \mathrm{d}\tau \right\|.$$

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

$$\begin{aligned}\left\|\mathbf{x}_n(t) - \mathbf{x}^\dagger(t)\right\| &\leqslant c_n \exp\left(c(\sigma)\int_0^t \|\mathbf{w}_n(\tau)\| \,\mathrm{d}\tau\right)\\ &\leqslant c_n \exp\left(c(\sigma)\sqrt{T}\,\|\mathbf{w}_n\|_{L^2(0,T;\mathbb{R}^{d\times d\times N})}\right)\\ &\leqslant c_n \exp\left(c(\sigma)\sqrt{T}M\right).\end{aligned}$$

As $c_n \longrightarrow 0$ 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). Hence, Φ_T sends bounded sequences in $L^2(0,T;\mathbb{R}^{d_u})$ into strongly convergent (sub)sequences in $C^0([0,T];\mathbb{R}^{d_x})$ and is thus compact.

Proof of *(ii)*. Let $\{[w_n, b_n]\}_{n=1}^\infty \subset L^2(0,T;\mathbb{R}^{d_u}) \cap \mathrm{BV}(0,T;\mathbb{R}^{d_u})$ be a bounded sequence in $L^2(0,T;\mathbb{R}^{d_u}) \cap \mathrm{BV}(0,T;\mathbb{R}^{d_u})$. By the compactness of the embedding

$$\mathrm{BV}(0,T;\mathbb{R}^{d_u}) \hookrightarrow L^1(0,T;\mathbb{R}^{d_u})$$

(see [Ambrosio et al., 2000, Theorem 3.23]), there exists a pair $\left[w^\dagger, b^\dagger\right] \in \mathrm{BV}(0,T;\mathbb{R}^{d_u})$ such that, along some subsequence,

$$[w_n, b_n] \xrightarrow[n\longrightarrow\infty]{} \left[w^\dagger, b^\dagger\right] \qquad \text{strongly in } \ L^1(0,T;\mathbb{R}^{d_u}).$$

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 $\mathbf{b}_n := [b_n, \ldots, b_n]$ to $\mathbf{b}^\dagger := [b^\dagger, \ldots, b^\dagger]$. Let $\mathbf{x}^\dagger \in C^0([0,T];\mathbb{R}^{d_x})$ be the unique solution to (3.2) associated to $[w^\dagger, b^\dagger]$ and the initial datum $\mathbf{x}^0$. Let us prove (A.1). Arguing as above, we see that

$$\begin{aligned}\left\|\mathbf{x}_n(t) - \mathbf{x}^\dagger(t)\right\| &\leqslant c(\sigma)\int_0^t \left\|\mathbf{w}_n(\tau) - \mathbf{w}^\dagger(\tau)\right\| \|\mathbf{x}_n(\tau)\| \,\mathrm{d}\tau\\ &\quad + c(\sigma)\int_0^t \left\|\mathbf{w}^\dagger(\tau)\right\| \left\|\mathbf{x}_n(\tau) - \mathbf{x}^\dagger(\tau)\right\| \,\mathrm{d}\tau\\ &\quad + c(\sigma)\int_0^t \left\|\mathbf{b}_n(\tau) - \mathbf{b}^\dagger(\tau)\right\| \,\mathrm{d}\tau,\end{aligned}$$

where $c(\sigma) > 0$ is the Lipschitz constant of σ, and thus, using Grönwall's inequality, we obtain

$$\left\|\mathbf{x}_n(t) - \mathbf{x}^\dagger(t)\right\| \leqslant c(\sigma)c_n \exp\left(c(\sigma)\int_0^t \left\|\mathbf{w}^\dagger(\tau)\right\| \,\mathrm{d}\tau\right),$$

where

$$c_n := \int_0^t \left\|\mathbf{w}_n(\tau) - \mathbf{w}^\dagger(\tau)\right\| \|\mathbf{x}_n(\tau)\| \,\mathrm{d}\tau + \int_0^t \left\|\mathbf{b}_n(\tau) - \mathbf{b}^\dagger(\tau)\right\| \,\mathrm{d}\tau.$$

Using the fact that $\mathbf{x}_n$ is bounded uniformly in n in $C^0([0,T];\mathbb{R}^{d_x})$ (this follows by applying a Grönwall argument) and the strong L^1–convergences of the parameters, we conclude that $c_n \longrightarrow 0$, whence (A.1) follows.

The proof of *(iii)* follows by arguing as in *(i)* and *(ii)*, with an intermediate use of the Rellich-Kondrachov compactness theorem to ensure strong convergence of the parameters in L^2 – we omit the proof. This concludes the proof. □

Proof of Lemma 7.3. We focus on proving *(i)* and split the proof in two steps. The proof of *(ii)* will follow by simply applying a Cauchy-Schwarz inequality at the conclusion of each of the two steps.

Step 1. Let us first suppose that $t \leqslant 1$. By integrating (3.2) on the interval $[0,t] \subset [0,1]$ and using the fact that $\sigma \in \mathrm{Lip}(\mathbb{R})$ and $\sigma(0) = 0$, it may be seen that

$$\|\mathbf{x}(t)\| \leqslant \left\|\mathbf{x}^0\right\| + c(\sigma) \int_0^t \|\mathbf{w}(s)\| \|\mathbf{x}(s)\| \,\mathrm{d}s + c(\sigma) \int_0^t \|\mathbf{b}(s)\| \,\mathrm{d}s$$

for $t \in [0,1]$, where $c(\sigma) > 0$ denotes the Lipschitz constant of σ. By the Grönwall inequality, we then have

$$\|\mathbf{x}(t)\| \leqslant \underbrace{\left(\left\|\mathbf{x}^0\right\| + c(\sigma) N^{1/2} \|b\|_{L^1(0,1;\mathbb{R}^d)}\right) \exp\left(c(\sigma) N^{1/2} \|w\|_{L^1(0,1;\mathbb{R}^{d\times d})}\right)}_{:= C\left(\left\|[w,b]\right\|_{L^1(0,1;\mathbb{R}^{d_u})}\right)} \tag{A.2}$$

for $t \in [0,1]$. Using (A.2), it may be seen that

$$\begin{aligned}\|\mathbf{x}(t) - \overline{\mathbf{x}}\| &\leqslant \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\| + c(\sigma) N^{1/2} \left(\|\mathbf{x}(t)\| \|w\|_{L^1(0,1;\mathbb{R}^{d\times d})} + \|b\|_{L^1(0,1;\mathbb{R}^d)}\right) \\ &\leqslant \left\|\mathbf{x}^0 - \overline{\mathbf{x}}\right\| + c(\sigma) N^{1/2} \max\left\{1, C\left(\left\|[w,b]\right\|_{L^1(0,1;\mathbb{R}^{d_u})}\right)\right\} \left\|[w,b]\right\|_{L^1(0,1;\mathbb{R}^{d_u})}.\end{aligned}$$

Step 2. Now suppose that $t \in (1,T]$. We first show that there exists a $t^* \in (t-1,t]$ such that

$$\|\mathbf{x}(t^*) - \overline{\mathbf{x}}\| \leqslant \|\mathbf{x} - \overline{\mathbf{x}}\|_{L^2(0,T;\mathbb{R}^{d_x})}. \tag{A.3}$$

This follows by a contradiction argument – indeed, suppose that

$$\|\mathbf{x}(t^*) - \overline{\mathbf{x}}\| > \|\mathbf{x} - \overline{\mathbf{x}}\|_{L^2(0,T;\mathbb{R}^{d_x})}.$$

for all $t^* \in (t-1,t]$, then

$$\int_0^T \|\mathbf{x}(s) - \overline{\mathbf{x}}\|^2 \,\mathrm{d}s \geqslant \int_{t-1}^t \|\mathbf{x}(s) - \overline{\mathbf{x}}\|^2 \,\mathrm{d}s > \int_0^T \|\mathbf{x}(s) - \overline{\mathbf{x}}\|^2 \,\mathrm{d}s,$$

which is a contradiction. Consequently, we know that there exists a $t^* \in (t-1,t]$ such that (A.3) holds. By integrating (3.2) in $[t^*,t]$ and using the fact that $\sigma \in \mathrm{Lip}(\mathbb{R})$ and $\sigma(0) = 0$, we see that

$$\|\mathbf{x}(t) - \overline{\mathbf{x}}\| \leqslant \|\mathbf{x}(t^*) - \overline{\mathbf{x}}\| + c(\sigma) \int_{t^*}^t \Big(\|\mathbf{w}(s)\| \, \|\mathbf{x}(s) - \overline{\mathbf{x}}\| + \|\overline{\mathbf{x}}\| \, \|\mathbf{w}(s)\| + \|\mathbf{b}(s)\| \Big) \,\mathrm{d}s.$$

By virtue of Grönwall's inequality, it can then be seen that

$$N^{-1/2} \|\mathbf{x}(t) - \overline{\mathbf{x}}\| \leqslant C \left(\|\mathbf{x}(t^*) - \overline{\mathbf{x}}\| + \left\|[w,b]\right\|_{L^1(0,T;\mathbb{R}^{d_u})} \right) \exp\left(C \left\|w\right\|_{L^1(0,T;\mathbb{R}^{d\times d})}\right)$$

for some $C = C(\sigma, \|\overline{\mathbf{x}}\|) > 0$. Using (A.3), we conclude the proof. □

Proof of Lemma 7.4. We will make use of the notation $u := [w,b]$ for simplicity.

Let us first suppose that $T \geqslant 1 + \tau_\circ$. In this case, the arguments follow precisely those presented in the proof of Theorem 4.2. We just give a brief sketch of the main differences. Due to the controllability assumption, there exist $u^\dagger \in L^2(\tau_\circ, 1+\tau_\circ; \mathbb{R}^{d_u})$

such that the corresponding solution $\mathbf{x}^\dagger(\cdot)$ to (7.20) on $(\tau_\circ, 1+\tau_\circ)$ satisfies $\mathbf{x}^\dagger(1+\tau_\circ) = \overline{\mathbf{x}}$. By a Grönwall argument we see that for $t \in [\tau_\circ, 1+\tau_\circ]$,

$$\left\|\mathbf{x}^\dagger(t) - \overline{\mathbf{x}}\right\| \leqslant C_1 N^{1/2} \left\|\mathbf{x}^{\tau_\circ} - \overline{\mathbf{x}}\right\| \exp\left(C_1 N^{1/2} \left\|\mathbf{x}^{\tau_\circ} - \overline{\mathbf{x}}\right\|\right) \tag{A.4}$$

holds for some constant $C_1 = C_1(\sigma, \|\overline{\mathbf{x}}\|, r) > 0$ independent of $\tau_\circ, T, \mathbf{x}^{\tau_\circ}, N$. We now set

$$u^{\text{aux}}(t) := \begin{cases} u^\dagger(t) & \text{in } (\tau_\circ, 1+\tau_\circ) \\ 0 & \text{in } (1+\tau_\circ, T), \end{cases}$$

and we denote by $\mathbf{x}^{\text{aux}}(\cdot)$ the associated solution to (7.20). We note that $\mathbf{x}^{\text{aux}}(t) = \overline{\mathbf{x}}$ for $t \in [1+\tau_\circ, T]$ and thus $\mathcal{E}(\mathbf{x}^{\text{aux}}(T)) = 0$. Just as for the proof of Theorem 4.2, by using $J_{\tau_\circ,T}(u_T) \leqslant J_{\tau_\circ,T}(u^{\text{aux}})$, the fact that $\mathcal{E}(\mathbf{x}^{\text{aux}}(T)) = 0$, $\mathcal{E} \geqslant 0$, (A.4), (4.4), and recalling Lemma 7.3, we may conclude the proof when $T \geqslant 1+\tau_\circ$.

Now suppose that $T \leqslant 1+\tau_\circ$. By making use of $J_{\tau_\circ,T}(u_T) \leqslant J_{\tau_\circ,T}(u_{1+\tau_\circ})$, where $u_{1+\tau_\circ} \in L^2(\tau_\circ, 1+\tau_\circ; \mathbb{R}^{d_u})$ is a minimizer of $J_{\tau_\circ,1+\tau_\circ}$, and since by what precedes we have

$$J_{\tau_\circ,T}(u_{1+\tau_\circ}) \leqslant J_{\tau_\circ,1+\tau_\circ}(u_{1+\tau_\circ}) \leqslant C_2 N \left\|\mathbf{x}^{\tau_\circ} - \overline{\mathbf{x}}\right\|^2 \exp\left(2C_1 N^{1/2} r\right),$$

we may combine this inequality with Lemma 7.3 to conclude the proof. □

Proof of Lemma 7.5. We argue by contradiction. Suppose that there exist parameters $\left[w^\dagger, b^\dagger\right] \in L^2(\tau_\circ, T; \mathbb{R}^{d_u})$ such that

$$J_{\tau_\circ,T}\left(w^\dagger, b^\dagger\right) < J_{\tau_\circ,T}(w_T, b_T).$$

Set

$$[w^{\text{aux}}(t), b^{\text{aux}}(t)] := \begin{cases} [w_T(t), b_T(t)] & \text{for } t \in [0, \tau_\circ) \\ \left[w^\dagger(t), b^\dagger(t)\right] & \text{for } t \in [\tau_\circ, T]. \end{cases}$$

Denoting by $\mathbf{x}^{\text{aux}}$ the associated neural ODE trajectory on $[0,T]$ with $\mathbf{x}^{\text{aux}}(0) = \mathbf{x}^0$, we see that $\mathbf{x}^{\text{aux}}(T) = \mathbf{x}^\dagger(T)$ and so $\mathcal{E}(\mathbf{x}^{\text{aux}}(T)) = \mathcal{E}(\mathbf{x}^\dagger(T))$. Consequently,

$$\begin{aligned} J_T(w^{\text{aux}}, b^{\text{aux}}) &= \int_0^{\tau_\circ} \|\mathbf{x}_T(t) - \overline{\mathbf{x}}\|^2 \,\mathrm{d}t + \lambda \int_0^T \left\|[w_T(t), b_T(t)]\right\|^2 \mathrm{d}t + J_{\tau_\circ,T}\left(w^\dagger, b^\dagger\right) \\ &< J_T(w_T, b_T), \end{aligned}$$

which contradicts the optimality of $[w_T, b_T]$. This concludes the proof. □

Proof of Lemma 7.6. Set $\eta(\tau) := \frac{\|f\|_{L^2(a,T;X)}}{\sqrt{\tau}}$. We argue by contradiction. Suppose that

$$\|f(t)\|_X > \eta(\tau) \qquad \text{for all } t \in [a, a+\tau).$$

Then

$$\int_a^T \|f(t)\|_X^2 \,\mathrm{d}t \geqslant \int_a^{a+\tau} \|f(t)\|_X^2 \,\mathrm{d}t > \tau\eta(\tau)^2.$$

Whence

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

which is a contradiction. □

References

Agrachev, A. and Sarychev, A. (2020). Control on the manifolds of mappings as a setting for deep learning. *arXiv preprint arXiv:2008.12702*.

Albertini, F. and Sontag, E. D. (1993). For neural networks, function determines form. *Neural Networks*, 6(7):975–990.

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.

Allen-Zhu, Z., Li, Y., and Liang, Y. (2019a). Learning and generalization in overparameterized neural networks, going beyond two layers. In *Advances in Neural Information Processing Systems*, pages 6158–6169.

Allen-Zhu, Z., Li, Y., and Song, Z. (2019b). A convergence theory for deep learning via over-parameterization. In *International Conference on Machine Learning*, pages 242–252. PMLR.

Ambrosio, L., Fusco, N., and Pallara, D. (2000). *Functions of bounded variation and free discontinuity problems*. Courier Corporation.

Avelin, B. and Nyström, K. (2020). Neural ODEs as the deep limit of ResNets with constant weights. *Anal. Appl.*, pages 1–41.

Bartlett, P. L. and Mendelson, S. (2002). Rademacher and Gaussian complexities: Risk bounds and structural results. *J. Mach. Learn. Res.*, 3(Nov):463–482.

Bartlett, P. L., Montanari, A., and Rakhlin, A. (2021). Deep learning: a statistical viewpoint.

Benning, M., Celledoni, E., Ehrhardt, M. J., Owren, B., and Schönlieb, C.-B. (2019). Deep learning as optimal control problems: Models and numerical methods. *J. Comput. Dyn.*, 6(2):171.

Bierstone, E. and Milman, P. D. (1988). Semianalytic and subanalytic sets. *Publ. Math. Inst. Hautes Études Sci.*, 67:5–42.

Brezis, H. (2010). *Functional analysis, Sobolev spaces and partial differential equations*. Springer Science & Business Media.

Bubeck, S., Eldan, R., Lee, Y. T., and Mikulincer, D. (2020a). Network size and weights size for memorization with two-layers neural networks. *arXiv preprint arXiv:2006.02855*.

Bubeck, S., Li, Y., and Nagaraj, D. (2020b). A law of robustness for two-layers neural networks. *arXiv preprint arXiv:2009.14444*.

Budd, C. J., Huang, W., and Russell, R. D. (2009). Adaptivity with moving grids. *Acta Numer.*, 18:111–241.

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, 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, 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, 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, L. and Bach, F. (2020). Implicit bias of gradient descent for wide two-layer neural networks trained with the logistic loss. In *Proceedings of Thirty Third Conference on Learning Theory, PMLR*, volume 125, pages 1305–1338.

Chizat, L., Oyallon, E., and Bach, F. (2018). On lazy training in differentiable programming. *arXiv preprint arXiv:1812.07956*.

Coron, J.-M. (2007). *Control and nonlinearity*. Number 136. American Mathematical Soc.

Coron, J.-M. and Trélat, E. (2004). Global steady-state controllability of one-dimensional semilinear heat equations. *SIAM J. Control Optim.*, 43(2):549–569.

Cuchiero, C., Larsson, M., and Teichmann, J. (2020). Deep neural networks, generic universal interpolation, and controlled ODEs. *SIAM J. Math. Data Sci.*, 2(3):901–919.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Math. Control Signals Systems*, pages 303–314.

Dorfman, R., Samuelson, P., and Solow, R. (1958). *Linear Programming and Economic Analysis*. Dover Books on Advanced Mathematics. Dover Publications.

Du, S., Lee, J., Li, H., Wang, L., and Zhai, X. (2019). Gradient descent finds global minima of deep neural networks. In *International Conference on Machine Learning*, pages 1675–1685. PMLR.

Dupont, E., Doucet, A., and Teh, Y. W. (2019). Augmented Neural ODEs. In *Advances in Neural Information Processing Systems*, pages 3134–3144.

Effland, A., Kobler, E., Kunisch, K., and Pock, T. (2020). Variational networks: An optimal control approach to early stopping variational methods for image restoration. *J. Math. Imaging Vision*, pages 1–21.

Esteve, C., Geshkovski, B., Pighin, D., and Zuazua, E. (2020). Turnpike in Lipschitz-nonlinear optimal control. *arXiv preprint arXiv:2011.11091*.

Faulwasser, T. and Grüne, L. (2020). Turnpike properties in optimal control: An overview of discrete-time and continuous-time results. *arXiv preprint arXiv:2011.13670*.

Faulwasser, T., Hempel, A.-J., and Streif, S. (2021). On the turnpike to design of deep neural nets: Explicit depth bounds. *arXiv preprint arXiv:2101.03000*.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*.

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*.

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.

Gunther, S., Ruthotto, L., Schroder, J. B., Cyr, E. C., and Gauger, N. R. (2020). Layer-parallel training of deep residual neural networks. *SIAM J. Math. Data Sci.*, 2(1):1–23.

Haber, E. and Ruthotto, L. (2017). Stable architectures for deep neural networks. *Inverse Problems*, 34(1):014004.

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.

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. U.S.A.*, 79(8):2554–2558.

Hornik, K., Stinchcombe, M., and White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural networks*, pages 359–366.

Javanmard, A., Mondelli, M., Montanari, A., et al. (2020). Analysis of a two-layer neural network via displacement convexity. *Annals of Statistics*, 48(6):3619–3642.

Ji, Z. and Telgarsky, M. (2018). Risk and parameter convergence of logistic regression. *arXiv preprint arXiv:1803.07300*.

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.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

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, Y., Cortes, C., and Burges, C. (2010). Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2.

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.

Li, Q., Chen, L., Tai, C., and Weinan, E. (2017). Maximum principle based algorithms for deep learning. *J. Mach. Learn. Res.*, 18(1):5998–6026.

Li, Q., Lin, T., and Shen, Z. (2019). Deep learning via dynamical systems: An approximation perspective. *arXiv preprint arXiv:1912.10382*.

Liu, H. and Markowich, P. (2020). Selection dynamics for deep neural networks. *J. Differ. Eq.*, 269(12):11540–11574.

Lojasiewicz, S. (1959). Sur le problème de la division. *Studia Math*, 18:87–136.

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, C., Wang, K., Chi, Y., and Chen, Y. (2020). Implicit regularization in nonconvex statistical estimation: Gradient descent converges linearly for phase retrieval, matrix completion, and blind deconvolution. *Found Comput Math*, 20:451–632.

Mallat, S. (2016). Understanding deep convolutional networks. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374(2065):20150203.

Mei, S., Montanari, A., and Nguyen, P.-M. (2018). A mean field view of the landscape of two-layer neural networks. *Proc. Natl. Acad. Sci. U.S.A.*, 115(33):E7665–E7671.

Müller, J. (2019). Universal flow approximation with deep residual networks. *arXiv preprint arXiv:1910.09599*.

Nguyen, P.-M. and Pham, H. T. (2020). A rigorous framework for the mean field limit of multilayer neural networks. *arXiv preprint arXiv:2001.11443*.

Nitanda, A. and Suzuki, T. (2017). Stochastic particle gradient descent for infinite ensembles. *arXiv preprint arXiv:1712.05438*.

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, D. and Zuazua, E. (2018). Controllability under positivity constraints of semilinear heat equations. *Math. Control Relat. Fields*, 8(3&4):935.

Pinkus, A. (1999). Approximation theory of the MLP model in neural networks. *Acta Numer.*, 8(1):143–195.

Porretta, A. and Zuazua, E. (2013). Long time versus steady state optimal control. *SIAM J. Control Optim.*, 51(6):4242–4273.

Queiruga, A. F., Erichson, N. B., Taylor, D., and Mahoney, M. W. (2020). Continuous-in-depth neural networks. *arXiv preprint arXiv:2008.02389*.

Rosset, S., Zhu, J., and Hastie, T. (2003). Margin maximizing loss functions. *Advances in Neural Information Processing Systems*, 16:1237–1244.

Rosset, S., Zhu, J., and Hastie, T. (2004). Boosting as a regularized path to a maximum margin classifier. *J. Mach. Learn. Res.*, 5(Aug):941–973.

Ruiz-Balet, D. and Zuazua, E. (2021). Neural ODE control for classification, approximation and transport. *In preparation*.

Ruthotto, L. and Haber, E. (2019). Deep neural networks motivated by partial differential equations. *Journal of Mathematical Imaging and Vision*, pages 1–13.

Ruthotto, L. and Haber, E. (2021). An introduction to deep generative modeling. *arXiv preprint arXiv:2103.05180*.

Sander, M. E., Ablin, P., Blondel, M., and Peyré, G. (2021). Momentum residual neural networks. *arXiv preprint arXiv:2102.07870*.

Sirignano, J. and Spiliopoulos, K. (2020). Mean field analysis of neural networks: A law of large numbers. *SIAM J. Appl. Math.*, 80(2):725–752.

Sontag, E. and Sussmann, H. (1997). Complete controllability of continuous-time recurrent neural networks. *Systems Control Lett.*, 30(4):177–183.

Sontag, E. D. and Qiao, Y. (1999). Further results on controllability of recurrent neural networks. *Systems Control Lett.*, 36(2):121–129.

Soudry, D., Hoffer, E., Nacson, M. S., Gunasekar, S., and Srebro, N. (2018). The implicit bias of gradient descent on separable data. *J. Mach. Learn. Res.*, 19(1):2822–2878.

Tabuada, P. and Gharesifard, B. (2020). Universal approximation power of deep neural networks via nonlinear control theory. *arXiv preprint arXiv:2007.06007*.

Teshima, T., Tojo, K., Ikeda, M., Ishikawa, I., and Oono, K. (2020). Universal approximation property of neural ordinary differential equations. *arXiv preprint arXiv:2012.02414*.

Thorpe, M. and van Gennip, Y. (2018). Deep limits of residual neural networks. *arXiv preprint arXiv:1810.11741*.

Trélat, E. and Zhang, C. (2018). Integral and measure-turnpike properties for infinite-dimensional optimal control systems. *Mathematics of Control, Signals, and Systems*, 30(1):1–34.

Trélat, E. and Zuazua, E. (2015). The turnpike property in finite-dimensional nonlinear optimal control. *J. Differ. Equ.*, 258(1):81–114.

Vapnik, V. (2013). *The nature of statistical learning theory*. Springer Science & Business Media.

von Neumann, J. (1945). A model of general economic equilibrium. *Review of Economic Studies*, 13(1):1–9.

Wei, C., Lee, J. D., Liu, Q., and Ma, T. (2019). Regularization matters: Generalization and optimization of neural nets vs their induced kernel. In *Advances in Neural Information Processing Systems*, pages 9712–9724.

Weinan, E. (2017). A proposal on machine learning via dynamical systems. *Commun. Math. Stat.*, 5(1):1–11.

Weinan, E., Han, J., and Li, Q. (2019). A mean-field optimal control formulation of deep learning. *Research in the Mathematical Sciences*, 6(1):1–41.

Yagüe, C. E. and Geshkovski, B. (2021). Sparse approximation in learning via neural ODEs. *arXiv preprint arXiv:2102.13566*.

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.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. (2016). Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*.

Zhang, H., Gao, X., Unterman, J., and Arodz, T. (2020). Approximation capabilities of neural ODEs and invertible residual networks. In *International Conference on Machine Learning*, pages 11086–11095. PMLR.

Zuazua, E. (2007). Controllability and observability of partial differential equations: some results and open problems. In *Handbook of differential equations: evolutionary equations*, volume 3, pages 527–621. Elsevier.

Carlos Esteve-Yagüe, 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