Interplay between depth and width for interpolation in neural ODEs

Antonio Álvarez-López[a,*], Arselane Hadj Slimane[b], Enrique Zuazua[a,c,d]

[a]*Universidad Autónoma de Madrid, C. Francisco Tomás y Valiente, 7, Madrid, 28049, Spain*
[b]*ENS Paris Saclay, Avenue des science, 4, Gif-sur-Yvette, 91190, France*
[c]*Chair for Dynamics, Control, Machine Learning, and Numerics, Alexander von Humboldt-Professorship, Department of Mathematics, Friedrich-Alexander-Universität Erlangen-Nürnberg, Cauerstraße, 11, Erlangen, 91058, Germany*
[d]*Fundación Deusto, Av. de Universidades, 24, Bilbao, 48007, Spain*

Abstract

Neural ordinary differential equations (neural ODEs) have emerged as a natural tool for supervised learning from a control perspective, yet a complete understanding of their optimal architecture remains elusive. In this work, we examine the interplay between their width p and number of layer transitions L (effectively the depth $L + 1$). Specifically, we assess the model expressivity in terms of its capacity to interpolate either a finite dataset $\mathcal{D}$ comprising N pairs of points or two probability measures in $\mathbb{R}^d$ within a Wasserstein error margin $\varepsilon > 0$. Our findings reveal a balancing trade-off between p and L, with L scaling as $O(1 + N/p)$ for dataset interpolation, and $L = O\left(1 + (p\varepsilon^d)^{-1}\right)$ for measure interpolation.

In the autonomous case, where $L = 0$, a separate study is required, which we undertake focusing on dataset interpolation. We address the relaxed problem of ε-approximate controllability and establish an error decay of $\varepsilon \sim O(\log(p)p^{-1/d})$. This decay rate is a consequence of applying a universal approximation theorem to a custom-built Lipschitz vector field that interpolates $\mathcal{D}$. In the high-dimensional setting, we further demonstrate that $p = O(N)$ neurons are likely sufficient to achieve exact control.

Keywords: Neural ODEs, Depth, Width, Simultaneous controllability, Transport equation, Wasserstein distance
2000 MSC: 34H05, 35Q49, 68T07, 93B05

1. Introduction

Residual neural networks (ResNets) are formally defined as the family of discrete systems

$$\begin{cases} \mathbf{x}_{k+1} & = \mathbf{x}_k + W_k\boldsymbol{\sigma}\left(A_k\mathbf{x}_k + \mathbf{b}_k\right), \\ \mathbf{x}_0 & \in \mathbb{R}^d, \end{cases} \quad (1.1)$$

where $k = 0, \ldots, L$, $W_k \in \mathbb{R}^{d \times p}$, $A_k \in \mathbb{R}^{p \times d}$ and $\mathbf{b}_k \in \mathbb{R}^p$, for some $d \geq 1$, $L \geq 0$ and $p \geq 1$. Each time step k identifies a layer of the network. The number of layers $L + 1$ is the *depth* of (1.1). The parameter p is the *width* of (1.1), identifying the number of neurons per layer. The activation function $\boldsymbol{\sigma} : \mathbb{R}^p \to \mathbb{R}^p$ is defined as the column vector $\boldsymbol{\sigma}(\mathbf{y}) = \left(\sigma(y^{(1)}), \ldots, \sigma(y^{(p)})\right)^\top$ from a chosen non-linear function $\sigma : \mathbb{R} \to \mathbb{R}$. We consider the Rectified Linear Unit (ReLU), given by $\sigma(z) = \max\{z, 0\}$, $z \in \mathbb{R}$.

It has been noted [11, 18, 6, 5, 30] that (1.1) can be identified with the forward Euler discretization scheme for the class of continuous models known as neural ordinary differential equations (neural ODEs),

$$\begin{cases} \dot{\mathbf{x}}(t) & = W(t)\boldsymbol{\sigma}\left(A(t)\mathbf{x}(t) + \mathbf{b}(t)\right), \\ \mathbf{x}(0) & = \mathbf{x}_0 \in \mathbb{R}^d, \end{cases} \quad (1.2)$$

where $(W, A, \mathbf{b}) \in L^\infty\left((0, T), \mathbb{R}^{d \times p} \times \mathbb{R}^{p \times d} \times \mathbb{R}^p\right)$ for some $T > 0$. Here, $t \in (0, T)$ parameterizes the evolution of the states through a continuous range of layers. As discussed in [28, 2], it is common to assume that $(W, A, \mathbf{b})$ is a step function over $(0, T)$, to align closer with the dynamics of (1.1). Then, since σ is Lipschitz, existence and uniqueness of solutions hold for any $(W, A, \mathbf{b})$ and initial condition $\mathbf{x}_0$. Equation (1.2) can be equivalently written as

$$\dot{\mathbf{x}} = \sum_{i=1}^p \mathbf{w}_i(t)\sigma(\mathbf{a}_i(t) \cdot \mathbf{x} + b_i(t)), \quad (1.3)$$

where $\mathbf{w}_i$ and $\mathbf{a}_i$ are respectively the p columns of W and the p rows of A, both seen as column vectors in $\mathbb{R}^d$, while b_i is the i-th coordinate of $\mathbf{b}$, for $i = 1, \ldots, p$. In this work, we use formulation (1.3), although, for simplicity, we represent $(\mathbf{w}_i, \mathbf{a}_i, b_i)_{i=1}^p$ in their matrix form $(W, A, \mathbf{b})$, which corresponds to the equivalent system (1.2). Equation (1.3) can be naturally extended to handle probability distributions, rather than points in $\mathbb{R}^d$, by interpreting its right-hand side as the advection field that drives the evolution of a measure ρ. This extension gives rise to the neural

---
[*]Corresponding author
Email addresses: `antonio.alvarezl@uam.es` (Antonio Álvarez-López), `arselane.hadj_slimane@ens-paris-saclay.fr` (Arselane Hadj Slimane), `enrique.zuazua@fau.de` (Enrique Zuazua)
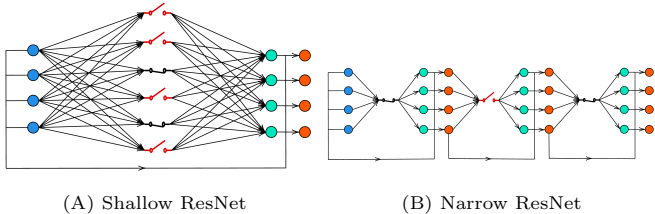
(A) Shallow ResNet   (B) Narrow ResNet

Figure 1: Qualitative representation of models (1.5) and (1.6) as discrete systems. Blue circles represent the input $\mathbf{x}$; switches depict ReLU functions; green circles indicate the result of $W\boldsymbol{\sigma}(A\mathbf{x} + \mathbf{b})$; orange circles represent the output after residual connections.

transport equation [29]:

$$\partial_t \rho + \mathrm{div}_{\mathbf{x}}\Big(\sum_{i=1}^{p} \mathbf{w}_i\,\sigma(\mathbf{a}_i \cdot \mathbf{x} + b_i)\rho\Big) = 0. \qquad (1.4)$$

Prior research indicates that control theory offers significant potential for examining the properties of neural ODEs, for instance, via optimal control [11, 14, 15] or geometric control techniques [32, 1, 31]. A fundamental problem still open is to develop a comprehensive understanding of the roles played by depth and width with respect to the expressive power of the model, see [19, 24, 16]. This property is often evaluated by its capacity to interpolate either a finite set of point pairs or two given probability measures.

The first, commonly referred to as finite-sample expressivity [35], is associated with the approximation power of the model, see [22]. It essentially amounts to a simultaneous control problem, where the aim is to find a control function $(W, A, \mathbf{b})$ such that the associated input-output map, given by the flow of (1.3), maps N specified points to their N corresponding target points in $\mathbb{R}^d$. Throughout this work, figures represent each input point as a colored solid circle and its corresponding target point as an empty circle of the same color.

In the second scenario, we aim to control the transport dynamics described by (1.4) in order to transform a given initial density ρ_0 into another density ρ_T. This task is highly relevant for probabilistic modeling or the generation of synthetic data via normalizing flows [21, 26, 17]. We approach it as an approximate control problem in the Wasserstein-q metric space for $q \geq 1$, extending prior work focused on W_2, see [12], or in W_1 with $p = 1$, see [28].

Our main objective is the development of a comprehensive theory of interpolation for the family of models described by (1.3), linking the error to the specific architecture given by p and L. Both numerical and theoretical studies [20, 13, 25] suggest that networks with greater depth usually achieve better performance. This tendency is particularly noticeable in training [36, 35], which reinforces the intuition that a deeper network should possess greater expressivity, i.e., an enhanced ability to learn more complex non-linear functions. Understanding the balance between width and depth is thus vital for the optimal design of networks. We tackle this significant question using the continuous framework of neural ODEs, where depth is

expressed as $L + 1$, L being the number of time discontinuities of the control $(W, A, \mathbf{b})$. As we vary L and p, two limiting models emerge.

Shallow neural ODEs. $L = 0$ is fixed, while the width p can be as large as required:

$$\dot{\mathbf{x}} = \sum_{i=1}^{p} \mathbf{w}_i \sigma\left(\mathbf{a}_i \cdot \mathbf{x} + b_i\right), \qquad (1.5)$$

where $\{(\mathbf{w}_i, \mathbf{a}_i, b_i)\}_{i=1}^{p} \subset \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}$ are constant controls, making the equation autonomous. The field on the right-hand side of (1.5) corresponds to a one hidden layer neural network with d components. The approximation capacity of this class of functions has been extensively studied (see [7, 27, 8]). The discrete version of (1.5) can be identified with a one hidden layer ResNet (see fig. 1A).

Narrow neural ODEs. $p = 1$ is fixed, while the depth $L + 1$ can be as large as required:

$$\dot{\mathbf{x}} = \mathbf{w}(t)\sigma\left(\mathbf{a}(t) \cdot \mathbf{x} + b(t)\right), \qquad (1.6)$$

where $(\mathbf{w}, \mathbf{a}, b) \in L^\infty\left((0, T), \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}\right)$. The ability of this model to interpolate data and approximate functions has been explored in [28, 23]. It offers the advantage of easier construction of explicit controls compared to (1.5), owing to its simplified dynamics, albeit at the expense of increased depth, which scales with the cardinal N of the dataset. The discrete version of (1.6) corresponds to a deep ResNet with one neuron per hidden layer, so it alternates layers of dimension 1 and d (see fig. 1A).

Developing a unified theory that bridges shallow and narrow neural ODEs would combine the vast work done for (1.5) with the intuitive dynamics of (1.6). Moreover, it would facilitate the optimal design of a neural ODE through the strategic choice of depth and width. This entails optimizing the complexity κ, defined as the total number of parameters in (1.3):

$$\kappa := (L + 1) \times p \times (2d + 1). \qquad (1.7)$$

Indeed, on each of the $L + 1$ hidden layers, p neurons of dimension $2d + 1$ need to be determined.

1.1. Roadmap

In section 2, we present the main results of our work in two parts. First, in section 2.1, we study the problem of interpolating a finite dataset in (1.3), which is recast as the property of simultaneous control for neural ODEs. Second, in section 2.2, we approach approximate controllability of probability measures using the dynamics provided by the neural transport equation (1.4). In section 3, we discuss the main implications of our work and pose some open questions. In section 4, we prove the main results and provide the necessary tools as lemmas.

- We use subscripts to identify the particular elements from a dataset and superscripts for the coordinates of a vector. In addition, (column) vectors are denoted with bold letters and matrices with capital letters.

- We denote by $\mathbf{x} \cdot \mathbf{y}$ the scalar product of $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$.

- We denote by $\lceil z \rceil$ the lowest integer greater than or equal to $z \in \mathbb{R}$, and by $\lfloor z \rfloor$ the highest integer lower than or equal to z.

- We denote by $\mathbb{S}^{d-1}$ the $(d-1)$-dimensional sphere in $\mathbb{R}^d$.

- We denote by $\mathrm{Lip}\left(\mathbb{R}^d, \mathbb{R}^d\right)$ the space of Lipschitz-continuous vector fields in the usual norm, and by L_V the Lipschitz constant of each $\mathbf{V} \in \mathrm{Lip}\left(\mathbb{R}^d, \mathbb{R}^d\right)$.

- Given any Borel measure μ in $\mathbb{R}^d$ and any measurable function $f : \mathbb{R}^d \to \mathbb{R}^d$, we denote by $f_{\#}\mu$ the pushforward measure, defined for every Borel subset $A \subset \mathbb{R}^d$ by

$$f_{\#}\mu(A) = \mu(f^{-1}(A)).$$

2. Main results

2.1. Simultaneous control

Let $N \geq 1$, $d \geq 2$, and consider a dataset

$$\mathcal{D} = \{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^N \subset \mathbb{R}^d \times \mathbb{R}^d \qquad (2.1)$$

with $\mathbf{x}_n \neq \mathbf{x}_m$ and $\mathbf{y}_n \neq \mathbf{y}_m$ for all $n \neq m$. First, we study the finite-sample expressivity of the general model (1.3), recast as a problem of simultaneous control.

Problem-Definition. For any fixed time horizon $T > 0$, find controls

$$\{(\mathbf{w}_i, \mathbf{a}_i, b_i)\}_{i=1}^p \subset L^\infty\left((0,T); \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}\right),$$

for some $p \geq 1$, such that the flow $\Phi_T(\cdot; W, A, \mathbf{b})$ of (1.2) *interpolates* the dataset $\mathcal{D}$, i.e., it simultaneously drives each data point from its initial position $\mathbf{x}_n$ to its target $\mathbf{y}_n$. This is fulfilled when

$$\Phi_T(\mathbf{x}_n; W, A, \mathbf{b}) = \mathbf{y}_n \qquad \text{for all } n = 1, \dots, N,$$

where $W, A, \mathbf{b}$ are respectively the matrix with columns $\mathbf{w}_i$, the matrix with rows $\mathbf{a}_i$ and the vector with components b_i, for $i = 1, \dots, p$.

Our first result provides a relationship between L and p that ensures interpolation of $\mathcal{D}$:

Theorem 1. *Let $N \geq 1$, $d \geq 2$ and $T > 0$ be fixed. Consider the dataset $\mathcal{D}$ as defined in (2.1). For any $p \geq 1$, there exists a piecewise constant control*

$$(W, A, \mathbf{b}) \in L^\infty\left((0,T); \mathbb{R}^{p \times d} \times \mathbb{R}^{p \times d} \times \mathbb{R}^p\right)$$

such that the flow $\Phi_T(\cdot; W, A, \mathbf{b})$ generated by (1.3) interpolates the dataset $\mathcal{D}$, i.e.,

$$\Phi_T(\mathbf{x}_n; W, A, \mathbf{b}) = \mathbf{y}_n, \qquad \text{for all } n = 1, \dots, N.$$

Furthermore, the number of discontinuities of $(W, A, \mathbf{b})$ is

$$L = 2\lceil N/p \rceil - 1. \qquad (2.2)$$

Remark 1. If the target points $\{\mathbf{y}_n\}_{n=1}^N$ in (2.1) are not distinct, interpolation is not achievable due to the uniqueness of solutions in the system (1.3). In such cases, we relax the statement from exact to approximate controllability by applying theorem 1 to an ε-perturbation of the targets, for some $\varepsilon > 0$.

Let us briefly describe the algorithm. First, we pivot around the $x^{(1)}$-coordinate and control the remaining $d-1$ coordinates. Consequently, the trajectory of each data point $\mathbf{x}_n$ is confined within the hyperplane defined by the equation $x_n^{(1)} = x_n^{(1)}$. Then, we pivot using the controlled coordinates to adjust $x^{(1)}$. This algorithm requires a depth of $2\lceil N/p \rceil$ layers, which is independent of the dimension d, since a constant control suffices to simultaneously steer $d-1$ coordinates in the first step, assuming that $p > N$.

Remark 2. Our approach is broadly applicable to any activation function, provided it meets the following three conditions:

1. σ loc. Lipschitz;   2. $\sigma(z)|_{z \leq 0} = 0$;   3. $\sigma(z)|_{z > 0} > 0$.

This generalization guarantees the extension of theorem 1 to more general activation functions such as the ReLU powers $\sigma^k(z) = \max\{z, 0\}^k$, for $z \in \mathbb{R}$ and $k \geq 1$, whose approximation properties have been recently studied in [4].

In (2.2) we can see that, as the width p increases, the number of discontinuities L decreases with the same rate, meaning that width and depth play a similar role in the steering. Nevertheless, a result on the optimal design of our interpolating models can be derived:

Corollary 2. *For the family of controls given by theorem 1 that ensure interpolation of $\mathcal{D}$, the minimal complexity is*

$$\kappa_{min} = (4d + 2)(N + 1),$$

obtained when $p = 1$, i.e., when the neural ODE belongs to the narrow model (1.6).

The complete transition from the narrow model (1.6) to the shallow model (1.5), characterized by $L = 0$ is not attained in (2.2). Due to the division into two steps in the proposed algorithm, whenever $p > N$ the selected control will exhibit a single switch ($L = 1$), reaching a two-layer architecture, rather than the autonomous ansatz (1.5). The restriction naturally raises the question of whether simultaneous control is possible in shallow neural ODEs (1.5). For this task, a reconsideration of the algorithm presented in [28] becomes necessary. In the high dimensional setting, and more precisely, when the dimension exceeds the number of data points ($d > N$), we can refine the statement of theorem 1 to include the case $L = 0$:

Corollary 3. *Let $N \geq 1$, $d \geq 2$ with $d > N$, and $T > 0$ be fixed. Consider the dataset $\mathcal{D}$ as defined in (2.1). For any $p \geq 1$, there exists a piecewise constant control*

$$(W, A, \mathbf{b}) \in L^\infty\left((0, T); \mathbb{R}^{d \times p} \times \mathbb{R}^{p \times d} \times \mathbb{R}^p\right)$$

such that the flow $\Phi_T(\cdot; W, A, \mathbf{b})$ generated by (1.3) interpolates the dataset $\mathcal{D}$. Furthermore, the number of discontinuities of $(W, a, \mathbf{b})$ is

$$L = 2\left(\lceil N/p \rceil - 1\right).$$

The key idea is that, when $d > N$, the first step in the proof of theorem 1 can be suppressed. This is done by transforming the $x^{(1)}-$axis so that each $\mathbf{x}_n$ shares the same first coordinate with $\mathbf{y}_n$, or, equivalently, by optimally re-orienting the hyperplanes represented in section 4.2. For more insights on the proof, see fig. 6 in section 4.

Remark 3. Corollary 3 suggests ideas similar to those in [14, Theorem 5.1]. In that result, interpolation is established for $d \geq N$ in a simplified neural ODE, when $\sup_{n=1,\dots,N} |\mathbf{x}_n - \mathbf{y}_n| < \varepsilon$ for a sufficiently small $\varepsilon > 0$, under a geometric assumption on the images of the targets through $\boldsymbol{\sigma}$. Moreover, an estimation of the control cost is obtained, which is linear with respect to ε. The generation of new synthetic coordinates until $d \geq N$ is not typically a problem, as discussed in [9], where the technique of embedding the dataset in $\mathbb{R}^d \times \{0, \dots, 0\}$ is proposed and its computational advantages are studied.

In practice, N tends to be larger than d. In that case, interpolation with constant controls can be obtained for $p = N$ under a certain separability hypothesis on $\mathcal{D}$:

Assumption 1. *Let $\mathcal{D} = \{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^N \subset \mathbb{R}^d \times \mathbb{R}^d$ as defined in (2.1). There exist a vector $\mathbf{a} \in \mathbb{S}^{d-1}$, a permutation τ of N elements and a sequence $-\infty < b_{N+1} < b_N < \cdots < b_1 < \infty$ such that, for all $n = 1, \dots, N-1$,*

$$-b_n < \mathbf{a} \cdot \mathbf{x}_{\tau(n)} < -b_{n+1} \quad and \quad -b_n < \mathbf{a} \cdot \mathbf{y}_{\tau(n)} < -b_{n+1}.$$

Assumption 1 claims that we can diagonally separate each pair $(\mathbf{x}_n, \mathbf{y}_n)$ from the rest, in the sense that we can define $N + 1$ parallel hyperplanes $H_n = \{\mathbf{a} \cdot \mathbf{x} + b_n = 0\}$ such that the strip S_n bounded by H_n and H_{n+1} contains only the point $\mathbf{x}_n$ and its target $\mathbf{y}_n$, for $n = 1, \dots, N$ (see section 2.1). While the hypothesis might seem overly restrictive, it is noteworthy that if the points are randomly sampled from a compact set, the probability that the condition is fulfilled converges to 1 when the dimension grows:

Proposition 4. *Let $\mu \in \mathcal{P}_{ac}^c(\mathbb{R}^d)$ such that the random variables $\pi_i X$ are independent and identically distributed (i.i.d.) for $i = 1, \dots, d$, where $X \sim \mu$ and π_i is the canonical projection on the i-th coordinate. If every $\mathbf{x}_n$ and $\mathbf{y}_n$ in $\mathcal{D}$ is sampled from μ, and N is sufficiently large, then the probability P that assumption 1 is satisfied is bounded as*

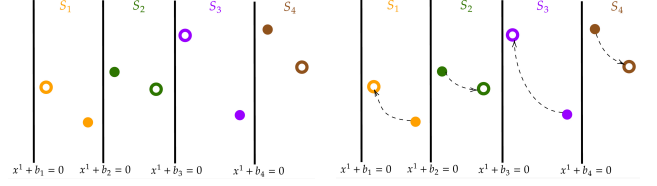$$1 - \left[1 - \frac{1}{\sqrt{2}}\left(\frac{e}{2N}\right)^N\right]^d \leq P \leq 1.$$

Figure 2: Left: separability condition in assumption 1, for $\mathbf{a} = \mathbf{e}_1$. Right: trajectories for exact control in the same example.

The hypothesis of proposition 4 are fulfilled by the uniform probability measure in any hypercube, or by any isotropic Gaussian distribution.

Now, under assumption 1, we can build a constant control such that the flow of (1.5), taking a width $p = N$, interpolates $\mathcal{D}$. Our result is somehow a dynamic version of [36, Theorem 1], under certain geometric conditions. A representation of the trajectories can be seen in section 2.1.

Corollary 5. *Consider a dataset $\mathcal{D} \subset \mathbb{R}^d \times \mathbb{R}^d$ for $d \geq 2$, under assumption 1. For any fixed $T > 0$, there exists a control $(W, A, \mathbf{b}) \in \mathbb{R}^{d \times N} \times \mathbb{R}^{N \times d} \times \mathbb{R}^N$ such that the flow Φ_T generated by (1.5) interpolates the dataset $\mathcal{D}$.*

All in all, new strategies are required to study simultaneous control in the autonomous model (1.5) under general conditions. A natural starting point to assess the problem's feasibility is to relax it by admitting an error $\varepsilon > 0$, and use density tools provided by universal approximation theorems (UATs); see [7, 27]. In deep learning, UATs establish the density of neural networks in function spaces over compact domains. The decay rate of the error in relation to the number of parameters of the network has been quantified for certain spaces [3, 8]. Specifically, these studies bound the uniform error decay rate when the target function is Lipschitz continuous in a compact domain.

We will approach the UAT, often interpreted in a static manner, from our dynamic control perspective. In this regard, shallow neural ODEs provide a vector field that transitions initial data to final data, with its flow at time T approximating the target function. First, we establish the existence of a time-independent field whose integral curves guide each input point $\mathbf{x}_n$ in $\mathcal{D}$ to its corresponding target $\mathbf{y}_n$ within a fixed time T. This field is constructed based on purely geometric considerations (see fig. 3) and can be chosen to be Lipschitz continuous.

Proposition 6. *Let $N \geq 1$, $d \geq 2$ and $T > 0$ be fixed. Consider the dataset $\mathcal{D} \subset \mathbb{R}^d \times \mathbb{R}^d$ as defined in (2.1), and any compact subset $\Omega \subset \mathbb{R}^d$ such that $\text{Int}(\Omega)$ is connected and $\mathcal{D} \subset \text{Int}(\Omega) \times \text{Int}(\Omega)$. Then, there exists a vector field $\mathbf{V} \in \text{Lip}\left(\mathbb{R}^d, \mathbb{R}^d\right)$ such that the flow $\Psi_{T,\mathbf{V}}$ of the equation*

$$\dot{\mathbf{x}} = \mathbf{V}(\mathbf{x})$$

interpolates the dataset $\mathcal{D}$, and the N curves given by

$$\mathcal{C}_n := \{\Psi_{t,\mathbf{V}}(\mathbf{x}_n) : t \in [0, T]\} \qquad (n = 1, \dots, N),$$
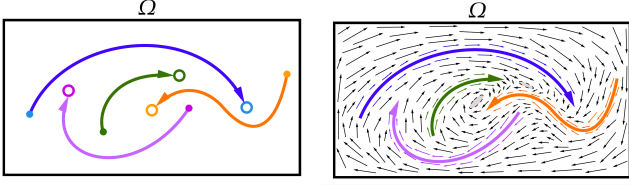
are contained in $\text{Int}(\Omega)$.

Figure 3: Construction of the Lipschitz field $\mathbf{V}$ in proposition 6 which interpolates $\mathcal{D}$ in a compact domain Ω that contains all the points and curves.

The subset $\Omega \subset \mathbb{R}^d$, which will serve as our domain of approximation, can always be established as $\Omega = [-R, R]^d$ for a sufficiently large $R > 0$. Consequently, for any dataset $\mathcal{D} \subset \mathbb{R}^d \times \mathbb{R}^d$, there exists a field $\mathbf{V} \in \mathrm{Lip}(\mathbb{R}^d, \mathbb{R}^d)$ whose integral curves $\mathcal{C}_n$ interpolate $\mathcal{D}$, i.e., the space

$$\mathcal{V}_{\mathcal{D}} := \left\{ \mathbf{V} \in \mathrm{Lip}\left(\mathbb{R}^d, \mathbb{R}^d\right) : \Psi_{T,\mathbf{V}} \text{ interpolates } \mathcal{D} \right\}$$

is non-empty. Moreover, we can define

$$L_0 := \inf_{\mathbf{V} \in \mathcal{V}_{\mathcal{D}}} L_V,$$

so L_0 only depends on $\mathcal{D}$ and the chosen domain of approximation Ω. It suffices then to combine the UAT from [8] (see lemma 12 in section 4) with classical results on the stability of ODEs to obtain the following theorem:

Theorem 7. *Let $N \geq 1$, $d \geq 2$ and $T > 0$ be fixed. Consider the dataset $\mathcal{D}$ as defined in (2.1). For each $p \geq 1$, there exists a control $(W, A, \mathbf{b}) \in \mathbb{R}^{d \times p} \times \mathbb{R}^{p \times d} \times \mathbb{R}^p$ such that the flow Φ_T generated by (1.5) satisfies*

$$\sup_{i=1,\ldots,N} |\mathbf{y}_n - \Phi_T(\mathbf{x}_n; W, A, \mathbf{b})| \leq C_{d,\mathcal{L},T} \frac{\log_2(\kappa)}{\kappa^{1/d}}, \quad (2.3)$$

where $\kappa = (d+2)dp$ is the complexity of the NODE, and

$$C_{d,\mathcal{L},T} = C_{d,L_0} T \exp\left(\mathcal{L} T\right),$$

being $\mathcal{L} = \min\{L_0, \|W\| \cdot \|A\|\}$ where $\|\cdot\|$ is the spectral norm, and $C_{d,L_0} > 0$ a constant depending on d and L_0 but independent of κ.

Remark 4. The argument employed in this theorem extends beyond neural networks. Since we solely rely on a density result that provides a convergence rate, other dense families of functions like polynomials, trigonometric, finite element methods or wavelets could also be considered, with their corresponding convergence rates.

Remark 5. Given a domain of approximation Ω, for the bound (2.3) to be optimal, it is natural to pose the problem of finding the interpolating field $\mathbf{V} \in \mathcal{V}_{\mathcal{D}}$ which has the smallest possible Lipschitz constant L_V within Ω.

Remark 6. When $d \geq 3$, the construction of a field $\mathbf{V} \in \mathcal{V}_{\mathcal{D}}$ is generally very simple. Since, in that case, two arbitrary curves are unlikely to intersect, we can generally consider the N segments that connect each pair $(\mathbf{x}_n, \mathbf{y}_n) \in \mathcal{D}$ and build $\mathbf{V}$ as one of the piecewise constant fields having these segments as integral curves. Selecting the optimal field then becomes a combinatorial problem.

2.2. Transport control

So far, we have considered the system (1.3) with a finite set of points $\mathcal{D}$ as initial data. A natural extension of this setting, particularly pertinent when the data points are sampled from an underlying distribution, is to consider as input a probability measure μ_0 on $\mathbb{R}^d$. The scenario where this distribution is a finite combination of Dirac deltas aligns with the study previously conducted in section 2.1.

Specifically, we consider the space $\mathcal{P}_{ac}^c(\mathbb{R}^d)$ of compactly supported and absolutely continuous probability measures on $\mathbb{R}^d$. Our goal is to transform any given $\mu_0 \in \mathcal{P}_{ac}^c(\mathbb{R}^d)$ into a fixed target probability measure μ_* through the push-forward map generated by a neural ODE, that is,

$$\Phi_T(\cdot; W, A, \mathbf{b})_\# \mu_0 = \mu_*.$$

This question can be reformulated as the control problem of a transport equation. For each $t \in [0, T]$, we consider the family of measures $\mu(t) = \Phi_{t\#}\mu_0$, where Φ_t represents the flow at time t generated by (1.3). Given that the field

$$\sum_{i=1}^{p} \mathbf{w}_i(t), \sigma(\mathbf{a}_i(t) \cdot \mathbf{x} + b_i(t))$$

is Lipschitz continuous with respect to $\mathbf{x}$, if $\mu_0 \in \mathcal{P}_{ac}^c(\mathbb{R}^d)$ then the curve of measures $\{\mu(t)\}_{t \in [0,T]}$ is contained in $\mathcal{P}_{ac}^c(\mathbb{R}^d)$. For each t, $\mu(t)$ is defined by a density function $\rho(t)$ that satisfies the neural transport equation

$$\begin{cases} \partial_t \rho + \mathrm{div}_{\mathbf{x}} \left(\rho \sum_{i=1}^p \mathbf{w}_i \, \sigma(\mathbf{a}_i \cdot \mathbf{x} + b_i) \right) = 0 \\ \rho(0) = \rho_0. \end{cases} \quad (2.4)$$

Here, we have assumed that μ_0 has density ρ_0, and

$$(\mathbf{w}_i, \mathbf{a}_i, b_i)_{i=1}^p \subset L^\infty\left((0,T); \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}\right)$$

serve again as control functions. The projected characteristics of (2.4) solve the neural ODE (1.3) in $(0, T) \times \mathbb{R}^d$. If the controls are step functions, and since the ReLU function is Lipschitz, the continuity equation (2.4) is well-posed and the total mass is conserved. Therefore, we aim to find some controls such that the corresponding solution of (2.4) with initial condition ρ_0 satisfies

$$\rho(T) = \rho_*.$$

This task, however, can be very hard to achieve, so we consider a relaxation of the problem to approximate control of (2.4). For this purpose, first we must choose a function to quantify the difference between any two measures.

Definition 7. For any $q \geq 1$, the *Wasserstein-q distance* between $\mu, \nu \in \mathcal{P}_{ac}^c(\mathbb{R}^d)$ is defined as

$$W_q(\mu, \nu) := \left(\min_{\gamma \in \Pi(\mu,\nu)} \int_{\mathbb{R}^d \times \mathbb{R}^d} |\mathbf{x} - \mathbf{y}|^q d\gamma(x,y) \right)^{1/q}, \quad (2.5)$$

where $\Pi(\mu, \nu)$ denotes the set of measures γ on $\mathbb{R}^d \times \mathbb{R}^d$ that couple μ and ν in the sense that $\gamma(\cdot \times \mathbb{R}^d) = \mu(\cdot)$

5

and $\gamma(\mathbb{R}^d \times \cdot) = \nu(\cdot)$. Note that $\mu \in \mathcal{P}^c_{ac}(\mathbb{R}^d)$ has finite q-th momentum for every $q \geq 1$, hence the Wasserstein$-q$ distance is well-defined in this space. Moreover, recalling the Monge formulation of optimal transport, if μ and ν belong to $\mathcal{P}^c_{ac}(\mathbb{R}^d)$ then

$$W_q(\mu, \nu) = \left(\min_T \left\{ \int_{\mathbb{R}^d} |\mathbf{x} - T(\mathbf{x})|^q d\mu : T_{\#}\mu = \nu \right\} \right)^{1/q}, \tag{2.6}$$

where $T : \mathbb{R}^d \to \mathbb{R}^d$ measurable, see [34].

Problem-Definition. Let μ_0 and μ_* be two compactly supported, absolutely continuous probability measures with respective densities ρ_0 and ρ_*. For any fixed time horizon $T > 0$ and $\varepsilon > 0$, find controls

$$\{(\mathbf{w}_i, \mathbf{a}_i, b_i)\}_{i=1}^p \subset L^\infty \left((0, T); \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R} \right),$$

for some $p \geq 1$, such that the solution of (2.4) in time T *approximately interpolates* the initial condition ρ_0 to the target density ρ_*. This is achieved when the W_q-error of the corresponding measures (for some $q \geq 1$) satisfies:

$$W_q(\mu(T), \mu_*) < \varepsilon.$$

The following theorem offers a partial solution to this problem. It assumes that $1 \leq q < \frac{d}{d-1}$ and targets the uniform measure in $[0, 1]^d$. While this bears resemblance to achieving null controllability, the nonlinear nature of the problem prevents from directly extending this result to arbitrary targets.

Theorem 8. *Let $d \geq 1$, $\mu_0 \in \mathcal{P}^c_{ac}(\mathbb{R}^d)$ with density ρ_0, μ_* the uniform measure in $[0, 1]^d$, and $T > 0$ be fixed. For any $\varepsilon > 0$, $q \in [1, \frac{d}{d-1})$ and $p \geq 1$, there exists a piecewise constant control*

$$(W, A, \mathbf{b}) \in L^\infty \left((0, T); \mathbb{R}^{d \times p} \times \mathbb{R}^{p \times d} \times \mathbb{R}^p \right)$$

such that the measure $\mu(t) \in \mathcal{P}^c_{ac}(\mathbb{R}^d)$ whose density $\rho(t)$ solves (1.4) taking ρ_0 as initial condition, satisfies

$$W_q(\mu(T), \mu_*) < \varepsilon.$$

Furthermore, the number of discontinuities of $(W, A, \mathbf{b})$ is

$$L = \lceil 2d/p \rceil + \max\{\lceil n/p_1 \rceil, \ldots, \lceil n^d/p_d \rceil\} - 1,$$

for any $p_1, \ldots, p_d \geq 1$ such that $p_1 + \cdots + p_d = p$, and

$$n := \left(\frac{3d^{1/2+1/q}}{\varepsilon} \right)^{\frac{1}{1+d/q-d}}.$$

For a given $\varepsilon > 0$, the behavior of L resembles that described in theorem 1, as it decreases with an increase in p, reaching $L = 1$ when p is large enough. Our proof is similar to a strategy from [10], and based on the specific movements that the neural ODE (1.3) allows. We compress the support of μ_0 to $[0, 1]^d$ and divide it into hyperrectangles, each with a mass of $O(\varepsilon^d)$. These subsets are then transformed to match a similar partition of $[0, 1]^d$ corresponding to the uniform measure μ_*.

Remark 8. If $\varepsilon > 0$ is sufficiently small, and we choose $p_1 = \cdots = p_{d-1} = 1$, $p_d = p - d + 1$, it follows that

$$L = \lceil 2d/p \rceil + \left\lceil \frac{1}{p - d + 1} \left(\frac{3^{1+d/q}\sqrt{d}}{\varepsilon} \right)^{\frac{d}{1+d/q-d}} \right\rceil - 1.$$

For $q = 1$, this expression simplifies to:

$$L = \lceil 2d/p \rceil + \left\lceil \frac{1}{p - d + 1} \left(\frac{3^{1+d}\sqrt{d}}{\varepsilon} \right)^d \right\rceil - 1.$$

3. Discussion

3.1. Conclusions

We have established several results on the capacity of neural ODEs for interpolation and its relationship with the chosen architecture, determined by the depth p and width L. More precisely, we have provided explicit dependencies between these two parameters that are sufficient to (exactly or approximately) interpolate either two sets of N different points in $\mathbb{R}^d$ or any compactly supported, absolutely continuous probability measure in $\mathbb{R}^d$ with the uniform measure in $[0, 1]^d$. Our work reveals that p and L can play similar roles in the algorithms, thereby exhibiting a degree of exchangeability in the network's structure.

Specifically, theorem 1 proves that a neural ODE with p neurons can interpolate any dataset of N pairs of points using a piecewise constant control with $L = 2\lceil N/p \rceil - 1$ discontinuities. Although increasing p reduces the number of discontinuities, we find a limiting case of a 2-hidden layer neural ODE when $p \geq N$. Explicit controls for interpolation with a shallow neural ODE ($L = 0$) are obtained in corollary 3 when $d > N$; or in corollary 5, with $p = N$ under assumption 1. More generally, theorem 7 provides an error decay rate with respect to the number of parameters for shallow neural ODEs. Finally, theorem 8 explores the Wasserstein-q approximate control of the neural transport equation to a uniform distribution on $[0, 1]^d$ using piecewise constant controls. As in theorem 1, the number of discontinuities diminishes as p increases.

3.2. Open questions

Some new objectives can be derived from our work:

1. Approaching the autonomous regime. As we have discussed, both $d > N$ and assumption 1 are only special cases where we can find controls to interpolate in the autonomous regime of shallow neural ODE. The question of finding such a construction for any $d \geq 1$, or at least under a less restrictive hypothesis on the dataset than assumption 1, is still open. A first step could involve assuming the relaxed condition that the projections of points onto a line with direction $\mathbf{a} \in \mathbb{S}^{d-1}$ are ordered as $\mathbf{a} \cdot \mathbf{x}_{\tau(1)} < \cdots < \mathbf{a} \cdot \mathbf{x}_{\tau(N)}$ and $\mathbf{a} \cdot \mathbf{y}_{\tau(1)} < \cdots < \mathbf{a} \cdot \mathbf{y}_{\tau(N)}$, for a certain permutation τ of N elements. The strategy to be adopted is clear. It entails a combination of the
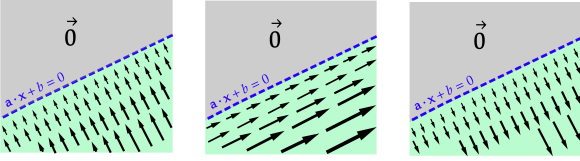
Figure 4: Left to right: Compression, parallel motion, expansion.

one-dimensional control delineated in lemma 13 with the transversal control required to prove corollary 5, followed by a case-by-case analysis.

2. Universal approximation. In [28], the authors demonstrate that a narrow neural ODE can approximate any simple function with compact support. The proof hinges on three key aspects: interpolation capacity, the compressive nature of neural ODEs, and a control strategy that bounds the support of the target function irrespective of required approximation accuracy. However, minimizing the number of time discontinuities required for this control, particularly by increasing p, is a non-trivial task that may require an entirely different approach.

3. Neural transport equation. The problem of controlling the neural transport equation (2.4) using constant controls is as yet unresolved. One possible approach involves approximating both the initial and target measures with atomic measures of the form $\rho_N = \frac{1}{N} \sum_{n=1}^{N} \beta_n \delta_{\alpha_n}$, with $\beta_n > 0$ and $\alpha_n \in \mathbb{R}^d$, and then interpolating those Dirac deltas by controlling the characteristic curves. However, a potential issue arises as $N \to \infty$: the distance between $\rho(T)$ and $\rho^N(T)$, the solutions to the transport equation with initial conditions ρ_0 and ρ_0^N, respectively, may diverge significantly. This error can be quantified using the Grönwall inequality, which suggests that the Lipschitz constant could increase unboundedly with N, especially as the number of controlled points grows.

4. Minimizing the number of time jumps. Another interesting question is how to frame the reduction of discontinuities as an optimal control problem. For instance, one could penalize the frequency of time jumps by targeting the total variation seminorm. However, this seminorm lacks regularity, and moreover the class of piecewise constant functions is not a closed set of admissible controls.

5. Switching dimensions. In our simplified ResNet (1.1), the dimension remains constant across layers. However, strategically varying the hidden dimension by defining $p = p(t)$ could offer advantages, either by reducing complexity through dimension shrinkage or by creating space through dimension increase. Exploring effective methods to implement these transitions, whether by employing projections or by applying nonlinear transformations to the data, constitutes a compelling area for research.

4. Proofs

4.1. Basic dynamics

We describe the simplest dynamics that we can generate via (1.6) by conveniently choosing $(\mathbf{w}, \mathbf{a}, b)$:

1. For each $t > 0$, the term $\mathbf{a}(t) \cdot \mathbf{x} + b(t)$ identifies a hyperplane in $\mathbb{R}^d$. For instance, taking $\mathbf{a} = \mathbf{e}_k$ and $b = -c$, we fix the hyperplane h with the equation $x^{(k)} - c = 0$.

2. The application of σ and the product with the vector $\mathbf{w}(t)$ yields the field $\mathbf{w}(t) \max\{x^{(k)} - c, 0\}$, which exhibits distinct dynamics in two complementary half-spaces: $H^+ \equiv \{x^{(k)} - c > 0\}$, where the field equals $\mathbf{w}(t)(x^{(k)} - c)$, and $H^- \equiv \{x^{(k)} - c \leq 0\}$, where the field is zero, meaning this set remains stationary under the flow.

3. The choice of $\mathbf{w}(t)$ specifies the orientation and magnitude of the field. For example, $\mathbf{w}(t) = \pm \mathbf{e}_k$ yields $\dot{x}^{(k)}(t) = \pm \max\{(x^{(k)} - c), 0\}$, so the points in H^+ can either be attracted to or repelled from h, enabling compression or expansion along the k-th coordinate. Conversely, $\mathbf{w}(t) = \mathbf{e}_i$, with $i \neq k$, results in $\dot{x}^{(i)}(t) = \max\{(x^{(k)} - c), 0\}$. In this case, we generate in H^+ a movement that is parallel to h, i.e., along the coordinate i.

The three basic operations of compression, expansion and movement in parallel with the hyperplane (represented in fig. 4) constitute our toolbox for many subsequent proofs.

4.2. Proof of theorem 1.

We will employ the following lemma, whose proof we postpone to the end of this subsection:

Lemma 9. *Let $N \geq 1$, $d \geq 2$ and consider the dataset $\mathcal{D} = \{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^{N}$ as defined in (2.1). There exists a change of coordinates in $\mathbb{R}^d$ such that*

$$x_n^{(1)} \neq x_m^{(1)} \qquad and \qquad y_n^{(2)} \neq y_m^{(2)}, \qquad if\ n \neq m. \quad (4.1)$$

Under the separability condition (4.1), we achieve the exact control by building on the methods developed in [28]. Let $p \geq 1$ be fixed. We divide the proof in two steps, illustrated in section 4.2 and section 4.2.

Step 1: Control of $d - 1$ coordinates. By (4.1), we can relabel the data $\{\mathbf{x}_n\}_{n=1}^{N}$ to impose the ordering

$$x_1^{(1)} < \cdots < x_N^{(1)}.$$

We define a partition of $\{\mathbf{x}_n\}_{n=1}^{N}$ in $\lceil N/p \rceil$ subsets by increasing order of the $x^{(1)}$-coordinate. The j-th subset is

$$X_j := \{\mathbf{x}_{(j-1) \cdot p + 1}, \ldots, \mathbf{x}_{j \cdot p}\}, \quad \text{for } j = 1, \ldots, \lceil N/p \rceil - 1,$$

and $X_{\lceil N/p \rceil}$ contains the remaining $N - p\lfloor N/p \rfloor$ points. We describe the control of the first subset X_1. We take controls $\mathbf{a}_i = \mathbf{e}_1$ and $b_i \in \mathbb{R}$, for $i = 1, \ldots, p$, satisfying

$$-b_1 < x_1^{(1)} < -b_2 < x_2^{(1)} < \cdots < -b_p < x_p^{(1)}.$$

These controls define a family of parallel hyperplanes, given by $\mathbf{a}_i \cdot \mathbf{x} + b_i = x^{(1)} + b_i = 0$, which separate the points
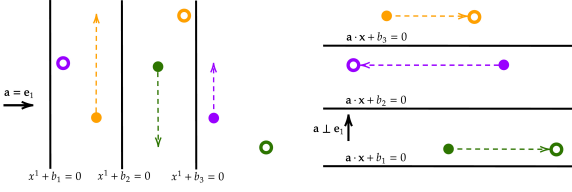
Figure 5: Left: Step 1. Fix $x^{(1)}$ and control $x^{(2)}, \ldots, x^{(d)}$. Right: Step 2. Control $x^{(1)}$ while $x^{(2)}, \ldots, x^{(d)}$ are fixed.

of X. In this way, $i-1$ terms of the sum in (1.3) cancel inside the strip $-b_i < x^{(1)} < -b_{i+1}$ for each $i = 1, \ldots, p$, so (1.3) simplifies to

$$\dot{\mathbf{x}} = \sum_{l=1}^{i} \left\{ \mathbf{w}_l x^{(1)} + \mathbf{w}_l b_l \right\}. \tag{4.2}$$

We consider velocities of the form $\mathbf{w}_i = (0, w_i^{(2)}, \ldots, w_i^{(d)})$, where the components $w_i^{(k)} \in \mathbb{R}$ have to be defined in order to achieve the exact control in time $T = 1$. The first point, $\mathbf{x}_1$, is subject only to one velocity, $\mathbf{w}_1$, so

$$x_1^{(k)}(t) = w_1^{(k)}(x_1^{(1)} + b_1)t + x_1^{(k)},$$

while $x_1^{(1)}$ remains fixed. Therefore, it is enough to take

$$w_1^{(k)} = \frac{y_1^{(k)} - x_1^{(k)}}{x_1^{(1)} + b_1}.$$

Similarly, for $i = 2, \ldots, p$, having fixed $\mathbf{w}_1, \ldots, \mathbf{w}_{i-1}$ it is enough to take

$$w_i^{(k)} = \frac{y_i^{(k)} - x_i^{(k)} - \sum_{l=1}^{i-1} w_l^{(k)}(x_i^{(1)} + b_l)}{x_i^{(1)} + b_i},$$

for $k = 1, \ldots, d$. The described procedure can be simultaneously done for each X_j, with $j = 2, \ldots \lceil N/p \rceil$, taking into account that the fields used to control $X_1, \ldots, X_{j-1}$ (all of them orthogonal to $\mathbf{e}_1$) will be added as new terms in (4.2). In the end, we will have, for every $n = 1, \ldots, N$:

$$\Phi_1(\mathbf{x}_n; W, A, \mathbf{b})^{(k)} = y_n^{(k)}, \qquad \text{for } k = 2, \ldots, d.$$

The total number of iterations employed in this step is $\lceil N/p \rceil$, which corresponds to $\lceil N/p \rceil - 1$ switches.

Step 2: Control of the remaining coordinate. In a slight abuse of notation, we redefine $\mathbf{x}_n := \Phi_1(\mathbf{x}_n)$, where Φ_1 is the flow resulting from step 1. Once again, we can relabel the data, now assuming

$$x_1^{(2)} < \cdots < x_N^{(2)}.$$

Following the increasing order of the $x^{(2)}$-coordinate, we define $X_1, \ldots, X_{\lceil N/p \rceil - 1}$, each being a subset of $\{\mathbf{x}_n\}_{n=1}^N$ with p points, and $X_{\lceil N/p \rceil}$, which contains the remaining $N - p \lfloor N/p \rfloor$ points.

We follow an analogous methodology to step 1. For each j, we define controls $\mathbf{a}_i = \mathbf{e}_2$ and b_i (for $i = 1, \ldots, p$)

that separate the points of X_j using p parallel hyperplanes, each described by the equation $x^{(2)} = b_i$. Now, we consider velocities of the form $\mathbf{w}_i = w_i \mathbf{e}_1$, where the values w_i are determined, as in step 1, to ensure

$$\Phi_1(\mathbf{x}_n; W, A, \mathbf{b})^{(1)} = y_n^{(1)}, \qquad \text{for } n = 1, \ldots, N.$$

The number of switches employed in step 2 is $\lceil N/p \rceil - 1$. Then, by adding one more to transition between steps, the whole control requires $L = 2 \lceil N/p \rceil - 1$ switches, hence proving theorem 1.

Proof of lemma 9. The set of vectors in $\mathbb{R}^d$ that are orthogonal to any point $\mathbf{x}_n$ from $\mathcal{D}$, is a finite union of hyperplanes in $\mathbb{R}^d$. Therefore, one can always choose $\mathbf{u}_1 \in \mathbb{S}^{d-1}$ inside the complement of this set in $\mathbb{R}^d$.

Consider the orthogonal subspace $S = \langle \mathbf{u}_1 \rangle^{\perp} \subset \mathbb{R}^d$, of dimension $d-1$. With a similar argument, one can select a vector $\mathbf{u}_2 \in \mathbb{S}^{d-1} \cap S$ that has a non-zero scalar product with all of the vectors $\mathbf{y}_n$ $(n = 1, \ldots, N)$. Completing the pair $(\mathbf{u}_1, \mathbf{u}_2)$ to form a orthonormal basis $\mathcal{B} = \{\mathbf{u}_1, \ldots, \mathbf{u}_d\}$ of $\mathbb{R}^d$, the dataset $\mathcal{D}$ will satisfy the separability condition eq. (4.1) when expressed in $\mathcal{B}$. $\qquad \square$

4.3. Proof of corollary 3

We aim to eliminate the initial step in the algorithm defined in the proof of theorem 1. To achieve this, we seek a new vector basis in $\mathbb{R}^d$ where the input-target pairs inherently share the first coordinate. When $d \leq N$, this condition usually cannot be met. However, when $d > N$, there exists an orthonormal vector basis $\mathcal{B} \subset \mathbb{R}^d$ such that for all $n = 1, \ldots, N$, the first coordinates of the N pairs with respect to $\mathcal{B}$ satisfy $x_n^{(1)} = y_n^{(1)}$.

To construct such a vector system, without loss of generality we can assume that $d = N+1$. Let $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^d \times \mathbb{R}^d$ with $\mathbf{x} \neq \mathbf{y}$. We seek a vector $\mathbf{u} \in \mathbb{S}^{d-1}$ such that $\mathbf{u} \cdot \mathbf{x} = \mathbf{u} \cdot \mathbf{y}$. This condition is equivalent to $\mathbf{u} \cdot (\mathbf{x} - \mathbf{y}) = 0$, which is satisfied by any unit vector $\mathbf{u}$ contained in the linear hyperplane orthogonal to $\mathbf{x} - \mathbf{y}$.

For $d-1$ input-target pairs of points, we consider the corresponding hyperplanes $\{H_n\}_{n=1}^{d-1}$. Note that some of these hyperplanes can be repeated. So, the intersection $\bigcap_{i=1}^{d-1} H_i$ yields a linear subspace of dimension at most $d-1$. We choose any unit vector $\mathbf{e}_1'$ contained in that subspace. Any completion to an orthonormal basis $\mathcal{B} = \{\mathbf{e}_1', \ldots, \mathbf{e}_d\} \subset \mathbb{R}^d$ will satisfy the desired condition for $d-1$ points. The procedure is illustrated in fig. 6.

4.4. Proof of corollary 5

Proof of proposition 4. We carry out a similar analysis to the one in [2], where the probability of requiring k hyperplanes to separate two sets of N points for binary classification was estimated. We introduce the random variable $Z_{d,2N}^p(\mathcal{D})$, which assigns to each possible dataset

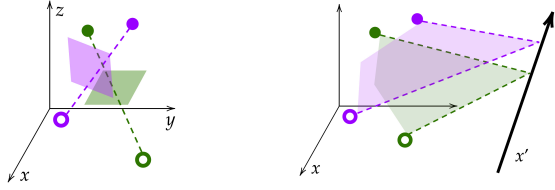$$\mathcal{D} = \{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^N \subset \text{supp}(\mu) \times \text{supp}(\mu)$$

Figure 6: For $N = 2$, $d = 3$, construction of a new basis of $\mathbb{R}^d$ in which the first coordinates of the pairs $(\mathbf{x}_n, \mathbf{y}_n)$ are matched. theorem 1 is applied afterwards.

the minimum number of parallel hyperplanes needed to separate in $\mathbb{R}^d$ every pair $(\mathbf{x}_n, \mathbf{y}_n)$ from the others. Note that $\min_{\mathcal{D}} Z^p_{d,2N}(\mathcal{D}) = N - 1$. We estimate the probability $P_{d,N} := P(Z^p_{d,2N}(\mathcal{D}) = N - 1)$ for any d and N.

First, we consider the one-dimensional case. Since all the points are sampled from the same distribution, every possible configuration of the $2N$ points in the real line will have the same probability, i.e., their distribution is uniform on the finite space of all possible orderings. Therefore, we can compute:

$$P_{1,N} = \frac{\text{favorable configurations}}{\text{total configurations}} = \frac{N! \, 2^N}{(2N)!}.$$

Now, we apply Stirling's formula $\frac{n^n \sqrt{2\pi n}}{n! e^n} \xrightarrow{n \to \infty} 1$ to approximate, for sufficiently large N:

$$P_{1,N} \approx \frac{N^N \sqrt{2\pi N}(2/e)^N}{(2N)^{2N}\sqrt{4\pi N}/e^{2N}} = \frac{1}{\sqrt{2}}\left(\frac{e}{2N}\right)^N.$$

Let $Z^{p,c}_{d,2N}$ be similarly defined to $Z^p_{d,2N}$ but restricting the hyperplanes to be orthogonal to one of the d canonical axes. Then, for any $d \geq 1$, we can bound:

$$P(Z^{p,c}_{d,2N} = N - 1) \leq P(Z^p_{d,2N} = N - 1). \tag{4.3}$$

By hypothesis, the d random variables defined as $Z^p_{1,2N}$ over the projection of $\mathcal{D}$ on each canonical axis are i.i.d. to $Z^p_{1,2N}$, so we can write:

$$\begin{aligned} P(Z^{p,c}_{d,2N} > N - 1) &= [1 - P(Z_{1,2N} = N - 1)]^d \\ &= \frac{N! \, 2^N}{(2N)!} \approx \left[1 - \frac{1}{\sqrt{2}}\left(\frac{e}{2N}\right)^N\right]^d, \end{aligned}$$

if $N \gg 1$. By (4.3), the complementary provides the desired lower bound for $P_{d,N}$. $\qquad\square$

Proof of corollary 5. Let $\mathbf{a} \in \mathbb{S}^{d-1}$, $\{b_n\}_{n=1}^{N+1} \subset \mathbb{R}$ and τ be given by assumption 1. With no loss of generality, we can assume that $\mathbf{a} = \mathbf{e}_1$ and τ is the identity permutation. The argument that we will use is similar to the one employed in the proof of theorem 1, but now the motion must be longitudinal as well as transverse. It also hinges on the fact that, , inside the n-th strip

$$S_n := \{\mathbf{x} \in \mathbb{R}^d : -b_n < \mathbf{a} \cdot \mathbf{x} < -b_{n+1}\},$$

the equation (1.5) simplifies to (4.2). The simultaneous control of the data points is achieved inductively, in increasing order of the first coordinates, by appropriately defining the field $\mathbf{w}_n$ associated with each hyperplane H_n. Both the base case and the inductive step are established by the following two lemmas, which will be proven later.

Lemma 10. *Consider two points $\mathbf{x}_1, \mathbf{y}_1 \in \mathbb{R}^d$ with $\mathbf{x}_1 \neq \mathbf{y}_1$. For any $T > 0$ and $b \in \mathbb{R}$ satisfying $x_1^{(1)} + b > 0$ and $y_1^{(1)} + b > 0$, there exists a unique $\mathbf{w} \in \mathbb{R}^d$ such that the solution of*

$$\begin{cases} \dot{\mathbf{x}} &= \mathbf{w}\sigma(x^{(1)} + b), \\ \mathbf{x}(0) &= \mathbf{x}_1 \in \mathbb{R}^d \end{cases} \tag{4.4}$$

reaches $\mathbf{x}(T) = \mathbf{y}_1$.

Having controlled $\mathbf{x}_1, \ldots, \mathbf{x}_{n-1}$ to $\mathbf{y}_1, \ldots, \mathbf{y}_{n-1}$ in a time horizon $T > 0$, and using parameters $\{\mathbf{w}_i\}_{i=1}^{n-1} \subset \mathbb{R}^d$ and $\{b_i\}_{i=1}^{n-1} \subset \mathbb{R}$ such that $b_{n-1} < \cdots < b_1$ and

$$x_i^{(1)} + b_i > 0 \quad \text{and} \quad y_i^{(1)} + b_i > 0 \quad \text{for } i = 1, \ldots, N,$$

steering $\mathbf{x}_n$ involves overcoming an autonomous drift field

$$\mathbf{d}(\mathbf{x}) := \sum_{i=1}^{n-1} \mathbf{w}_i \sigma(\mathbf{a} \cdot \mathbf{x} + b_i) = \sum_{i=1}^{n-1} \mathbf{w}_i \sigma(x^{(1)} + b_i). \tag{4.5}$$

The drift field $\mathbf{d}$ becomes more intense as the first coordinate increases, owing to the characteristics of the ReLU function. However, the following lemma shows that the control is possible:

Lemma 11. *Consider $\mathbf{x}_n, \mathbf{y}_n \in \mathbb{R}^d$ with $\mathbf{x}_n \neq \mathbf{y}_n$. With the above notation, for any $T > 0$ there exists a unique $\mathbf{w}_n \in \mathbb{R}^d$ and some $b_n \in \mathbb{R}$ satisfying*

$$b_n < b_{n-1}, \quad x_n^{(1)} + b_n > 0 \quad \text{and} \quad y_n^{(1)} + b_n > 0$$

such that the solution of the Cauchy problem

$$\begin{cases} \dot{\mathbf{x}} &= \mathbf{d}(\mathbf{x}) + \mathbf{w}_n \sigma(x^{(1)} + b_n), \\ \mathbf{x}(0) &= \mathbf{x}_n, \end{cases} \tag{4.6}$$

where $\mathbf{d}$ is given by (4.5), reaches $\mathbf{x}(T) = \mathbf{y}_n$.

With lemmas 10 and 11, the inductive argument is almost complete. It is left to show that the trajectory of each initial datum $\mathbf{x}_n$ will remain in its corresponding strip S_n for all $t \in (0, T)$.

On one hand, lemma 11 guarantees that the trajectory $\mathbf{x}(t)$ originating from $\mathbf{x}_n$ will reach the endpoint $\mathbf{y}_n$. On the other hand, taking into account that the field is autonomous and invariant along each hyperplane $x^{(1)} = \text{const}$, then $\dot{x}^{(1)}(t)$ cannot change sign at any time, that is to say, the $x^{(1)}(t)$ does not change its direction. Consequently, the entire trajectory will be contained within the strip bounded by $x = x_n^{(1)}$ and $x = y_n^{(1)}$, which in turn is contained in S_n. $\qquad\square$

Proof of lemma 10. In the half-space $\{x^{(1)} + b > 0\}$, the equation (4.4) is written as

$$\dot{\mathbf{x}} = \mathbf{w}\sigma(x^{(1)} + b) = \mathbf{w}x^{(1)} + \mathbf{w}\,b. \qquad (4.7)$$

We can assume that $b = 0$, so the solution of (4.7) is

$$\mathbf{x}(t) = \frac{x_1^{(1)}}{w^{(1)}}\mathbf{w}\left(e^{w^{(1)}t} - 1\right) + \mathbf{x}_1,$$

which can be driven to $\mathbf{x}(T) = \mathbf{y}_1$ by taking

$$w^{(1)} = \frac{1}{T}\ln\left(\frac{y_1^{(1)}}{x_1^{(1)}}\right) \quad \text{and} \quad w^{(k)} = \frac{y_1^{(k)} - x_1^{(k)}}{y_1^{(1)} - x_1^{(1)}}\,w^{(1)},$$

or $w^{(k)} = y_1^{(k)} - x_1^{(k)}$ if $x_1^{(1)} = y_1^{(1)}$, for $k = 2,\ldots,d$. Moreover, $\mathbf{x}(t)$ stays in the half-space $x^{(1)} > 0$ for $t \in [0,T]$ because $x^{(1)}(t)$ is monotone in that interval. $\qquad\square$

Proof of lemma 11. First, take any $b_n \in \mathbb{R}$ satisfying

$$b_n < b_{n-1}, \quad x_n^{(1)} + b_n > 0 \quad \text{and} \quad y_n^{(1)} + b_n > 0.$$

For simplicity, we rewrite (4.4) as

$$\dot{\mathbf{x}} = (\mathbf{s}_{n-1} + \mathbf{w}_n)\,x^{(1)} + \mathbf{w}_n b_n + \mathbf{c}_{n-1},$$

where $\mathbf{s}_{n-1} = \sum_{i=1}^{n-1}\mathbf{w}_i$, $\mathbf{c}_{n-1} = \sum_{i=1}^{n-1}\mathbf{w}_i b_i$. If we restrict to the first coordinate, we have:

$$\begin{cases} \dot{x}^{(1)} &= \left(s_{n-1}^{(1)} + w_n^{(1)}\right)x^{(1)} + w_n^{(1)}b_n + c_{n-1}^{(1)}, \\ x^{(1)}(0) &= x_n^{(1)}, \end{cases}$$

which has solution

$$x^{(1)}(t) = \frac{w_n^{(1)}b_n + c_{n-1}^{(1)}}{s_{n-1}^{(1)} + w_n^{(1)}}\left[e^{\left(s_{n-1}^{(1)} + w_n^{(1)}\right)t} - 1\right] \\ + x_n^{(1)}e^{\left(s_{n-1}^{(1)} + w_n^{(1)}\right)t}.$$

First, we want to see if there exists $\hat{w}_n^{(1)} \in \mathbb{R}$ such that $x^{(1)}(T) = y_n^{(1)}$, or, equivalently, if the function

$$f(z) = \frac{z\,b_n + c_{n-1}^{(1)}}{s_{n-1}^{(1)} + z}\left[e^{\left(s_{n-1}^{(1)} + z\right)T} - 1\right] \\ + x_n^{(1)}e^{\left(s_{n-1}^{(1)} + z\right)T} - y_n^{(1)} \qquad (4.8)$$

has a real root. For that task, we compute:

$$\lim_{z \to \infty} f(z) = \text{sign}\left(b_n + x_n^{(1)}\right) \cdot \infty = +\infty,$$
$$\lim_{z \to -\infty} f(z) = -b_n - y_n^{(1)} < 0.$$

On the other hand,

$$\lim_{z \to -s_{n-1}^{(1)}\,^+} f(z) = \left(c_{n-1}^{(1)} - b_n s_{n-1}^{(1)}\right)T + x_n^{(1)} - y_n^{(1)}$$
$$= \lim_{z \to -s_{n-1}^{(1)}\,^-} f(z),$$

so f is continuous. Hence, we can assure that there exists $\hat{w}_n^{(1)} \in \mathbb{R}$ such that $f\left(\hat{w}_n^{(1)}\right) = 0$. Now, we denote

$$s_n^{(1)} = \hat{w}_n^{(1)} + s_{n-1}^{(1)} \quad \text{and} \quad c_n^{(1)} = \hat{w}_n^{(1)}b_n + c_{n-1}^{(1)}.$$

For each component $j \in \{2,\ldots,d\}$,

$$\begin{cases} \dot{x}^{(j)} &= c_n^{(1)}\frac{s_{n-1}^{(j)} + w_n^{(j)}}{s_n^{(1)}}\left[e^{s_n^{(1)}t} - 1\right] \\ &\quad + x_n^{(1)}\left(s_{n-1}^{(j)} + w_n^{(j)}\right)e^{s_n^{(1)}t} + w_n^{(j)}b_n + c_{n-1}^{(j)}, \\ x^{(j)}(0) &= x_n^{(j)}, \end{cases}$$

which has solution

$$x^{(j)}(t) = x_n^{(j)} + \left[w_n^{(j)}b_n + c_{n-1}^{(j)} - c_n^{(1)}\frac{s_{n-1}^{(j)} + w_n^{(j)}}{s_n^{(1)}}\right]t \\ + \left[c_n^{(1)}\frac{s_{n-1}^{(j)} + w_n^{(j)}}{s_n^{(1)2}} + x_n^{(1)}\frac{s_{n-1}^{(j)} + w_n^{(j)}}{s_n^{(1)}}\right]e^{s_n^{(1)}t}.$$

Now, we want to find a solution of $g(z) = 0$ for

$$g(z) = x_n^{(j)} + \left[zb_n + c_{n-1}^{(j)} - c_n^{(1)}\frac{s_{n-1}^{(j)} + z}{s_n^{(1)}}\right]T \\ + \left[c_n^{(1)}\frac{s_{n-1}^{(j)} + z}{s_n^{(1)2}} + x_n^{(1)}\frac{s_{n-1}^{(j)} + z}{s_n^{(1)}}\right]e^{s_n^{(1)}T} - y_n^{(j)}.$$

This is an affine function in z, so $g(z) = 0$ has a unique solution $\hat{w}_n^{(j)} \in \mathbb{R}$ if and only if the slope is non-zero. Suppose that the b_n we have chosen yields a zero slope, i.e.,

$$\left[\frac{c_n^{(1)}}{s_n^{(1)}} + x_n^{(1)}\right]e^{s_n^{(1)}T} + (b_n s_{n-1}^{(1)} - c_{n-1}^{(1)})T = 0.$$

Recalling that $f(\hat{w}_n^{(1)}) = 0$, we can write this equation as

$$y_n^{(1)} + \frac{c_n^{(1)}}{s_n^{(1)}} + (b_n s_{n-1}^{(1)} - c_{n-1}^{(1)})T = 0,$$

so

$$\frac{c_n^{(1)}}{s_n^{(1)}} = (c_{n-1}^{(1)} - b_n s_{n-1}^{(1)})T - y_n^{(1)} \qquad (4.9)$$

and we can solve for $\hat{w}_n$ in this equation as

$$\hat{w}_n^{(1)} = \frac{\left[(c_{n-1}^{(1)} - b_n s_{n-1}^{(1)})T - y_n^{(1)}\right]s_{n-1}^{(1)} - c_{n-1}^{(1)}}{b_n - \left(c_{n-1}^{(1)} - b_n s_{n-1}^{(1)}\right)T - y_n^{(1)}}.$$

Meanwhile, substituting (4.9) in $f(\hat{w}_n^{(1)}) = 0$, we get:

$$\left[x_n^{(1)} - y_n^{(1)}\right]e^{s_n^{(1)}T} - T(c_{n-1}^{(1)} + b_n s_{n-1}^{(1)})[e^{s_n^{(1)}T} - 1] = 0,$$

which can also be solved for $\hat{w}_n^{(1)}$ as

$$\hat{w}_n^{(1)} = \frac{1}{T}\ln\left(\frac{(c_{n-1}^{(1)} - b_n s_{n-1}^{(1)})T}{(c_{n-1}^{(1)} - b_n s_{n-1}^{(1)})T - y_n^{(1)} + x_n^{(1)}}\right) - s_{n-1}^{(1)}.$$

Equalizing both expressions of $\hat{w}_n^{(1)}$, we obtain an equation with different analytic functions of b_n and not involving $\hat{w}_n^{(1)}$. Therefore, changing slightly the chosen value of b_n, these expressions become different, independently of the corresponding value of $\hat{w}_n^{(1)}$, so there is a perturbation of b_n that satisfies the statement of the lemma while $g(z) = 0$ has a unique solution.

$\square$

4.5. Proof of theorem 7

First, we prove proposition 6 with an inductive argument of topological nature.

Proof of proposition 6. First, we aim to build a family of N disjoint C^∞ curves contained in $\text{Int}(\Omega)$, each connecting the two points of a corresponding pair $(\mathbf{x}_n, \mathbf{y}_n) \in \mathcal{D}$. When $d \geq 2$, any connected open set in $\mathbb{R}^d$ is path-connected. Therefore, we can take a continuous path C that connects any two given points $(\mathbf{x}, \mathbf{y})$ inside $U := \text{Int}(\Omega) \setminus K$, where K represents any finite union of disjoint curves contained in $\text{Int}(\Omega)$. Moreover, by a well-known approximation argument this path can be chosen to be C^∞.

Now, we have N disjoint C^∞ curves $\{C_n\}_{n=1}^N$ contained in $\text{Int}(\Omega)$, each connecting a corresponding pair of points $(\mathbf{x}_n, \mathbf{y}_n)$. The tangent velocity field of each curve is also C^∞, so the assembled field $\mathbf{V}'$, defined in $\bigcup_{n=1}^N C_n$, is smooth too. Since its domain is compact, it is also Lipschitz-continuous. The required vector field $\mathbf{V}$ is provided by Kirzsbraun's Theorem (see [33]), which ensures the existence of a Lipschitz-continuous map $\mathbf{V} : \mathbb{R}^d \to \mathbb{R}^d$ that extends $\mathbf{V}'$ sharing the same Lipschitz constant. $\square$

The proof of theorem 7 employs the following lemma from [8, Section 7.2.2]:

Lemma 12 (Approximation rate for Lipschitz functions). *Let K be the unit ball in $\text{Lip}(\Omega, \mathbb{R})$, where $\Omega = [-R, R]^d$. We have*

$$\frac{1}{[\kappa \log_2 \kappa]^{1/d}} C_{d,R} \leq E(K, \Sigma_\kappa)_{C(\Omega)} \leq C_{d,R} \frac{\log_2 \kappa}{\kappa^{1/d}},$$

where:

- $\Sigma_\kappa := \{S_\kappa : \mathbb{R}^d \to \mathbb{R}\}$ *is the space of shallow neural networks with $\kappa = (d+2)p$ parameters, p being the number of neurons in the hidden layer;*

- $E(K, \Sigma_\kappa)_{C(\Omega)} = \sup_{f \in K} \inf_{S \in \Sigma_\kappa} \|f - S\|_{C(\Omega)}$ *measures the capacity of Σ_κ to approximate any function in K.*

This result was in turn derived from [3, Proposition 6], where the upper bound is obtained for K_L, the ball of radius L in $\text{Lip}(\Omega, \mathbb{R})$:

$$E(K_L, \Sigma_\kappa)_{C(\Omega)} \leq C_{d,R} L \frac{\log_2 \kappa}{\kappa^{1/d}}. \qquad (4.10)$$

Proof of theorem 7. Proposition 6 ensures that we can find a Lipschitz-continuous field $\mathbf{V} : \mathbb{R}^d \to \mathbb{R}^d$ such that the flow Ψ_T of the ODE

$$\dot{\mathbf{x}} = \mathbf{V}(\mathbf{x}) \qquad (4.11)$$

interpolates the dataset. The classical UAT result in [7] guarantees that we can uniformly approximate in Ω with precision $\varepsilon/\sqrt{d}$ each of its components $V^{(i)}$ using a corresponding shallow neural network $S^i_{\kappa_i} : \Omega \to \mathbb{R}$, with $\kappa_i \geq 1$ for $i = 1, \ldots, d$. Moreover, lemma 12 quantifies the dependence of the error with respect to the number of parameters of each $S^i_{\kappa_i}$, ensuring that

$$\kappa_1 = \kappa_2 = \cdots = \kappa_d = (d+2)p$$

parameters suffice to ensure (4.10) on each component. The assembled field

$$\mathbf{V}_{NN} = \left(S^1_{\kappa_1}, \ldots, S^d_{\kappa_d}\right) : \mathbb{R}^d \to \mathbb{R}^d$$

is of the form $\mathbf{V}_{NN}(\mathbf{x}) = W\boldsymbol{\sigma}(A\mathbf{x} + \mathbf{b})$ with complexity $\kappa = (d+2)pd$, and satisfies $\sup_{\mathbf{x} \in \Omega} |\mathbf{V}(\mathbf{x}) - \mathbf{V}_{NN}(\mathbf{x})| < \varepsilon$. Consider the neural ODE given by

$$\dot{\mathbf{x}} = \mathbf{V}_{NN}(\mathbf{x}). \qquad (4.12)$$

Let $\mathbf{X}_V(t; \mathbf{x}_0)$ and $\mathbf{X}_{NN}(t; \mathbf{x}_0)$ be the respective trajectories in time $t > 0$ that a point $\mathbf{x}_0 \in \Omega$ will follow under the dynamics provided by (4.11) and (4.12). The deviation

$$\mathbf{z}(t) = |\mathbf{X}_V(t; \mathbf{x}_0) - \mathbf{X}_{NN}(t; \mathbf{x}_0)|$$

is bounded as

$$\mathbf{z}(t) \leq \int_0^t |\mathbf{V}(\mathbf{X}_V(s; \mathbf{x}_0)) - \mathbf{V}_{NN}(\mathbf{X}_{NN}(s; \mathbf{x}_0))| \, \mathrm{d}s$$

$$\leq \int_0^t \big\{ |\mathbf{V}(\mathbf{X}_V(s; \mathbf{x}_0)) - \mathbf{V}(\mathbf{X}_{NN}(s; \mathbf{x}_0))|$$

$$+ |\mathbf{V}(\mathbf{X}_{NN}(s; \mathbf{x}_0)) - \mathbf{V}_{NN}(\mathbf{X}_{NN}(s; \mathbf{x}_0))| \big\} \mathrm{d}s$$

$$\leq L_V \int_0^t |\mathbf{X}_V(s; \mathbf{x}_0) - \mathbf{X}_{NN}(s; \mathbf{x}_0)| \, \mathrm{d}s + \varepsilon \, t$$

$$= L_V \int_0^t \mathbf{z}(s) \mathrm{d}s + \varepsilon \, t.$$

By Grönwall's inequality, it follows that

$$\mathbf{z}(t) \leq \varepsilon \, t \, \exp\{L_V \, t\}.$$

On the other hand, in the second line, we could alternatively add and subtract $\mathbf{V}_{NN}(\mathbf{X}_V(s; \mathbf{x}_0))$. Then, if we denote by L_{NN} the Lipschitz constant of $\mathbf{V}_{NN}$, we have:

$$\mathbf{z}(t) \leq \int_0^t \big\{ |\mathbf{V}(\mathbf{X}_V(s; \mathbf{x}_0)) - \mathbf{V}_{NN}(\mathbf{X}_V(s; \mathbf{x}_0))|$$

$$+ |\mathbf{V}_{NN}(\mathbf{X}_V(s; \mathbf{x}_0)) - \mathbf{V}_{NN}(\mathbf{X}_{NN}(s; \mathbf{x}_0))| \big\} \mathrm{d}s$$

$$\leq \varepsilon \, t + L_{NN} \int_0^t |\mathbf{X}_{NN}(s; \mathbf{x}_0) - \mathbf{X}_V(s; \mathbf{x}_0)| \, \mathrm{d}s$$

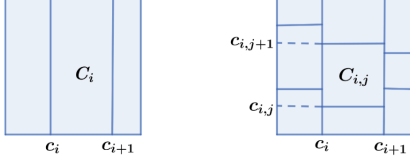$$= \varepsilon \, t + L_{NN} \int_0^t \mathbf{z}(s) \mathrm{d}s.$$

Figure 7: Division of $[0,1]^2$ into rectangles, each containing a mass of $1/n^2$ following the distribution given by ρ_0.

By Grönwall's inequality, it follows that

$$\mathbf{z}(t) \leq \varepsilon\, t\, \exp\left\{L_{NN}\, t\right\}.$$

Taking $\mathbf{x}_0 = \mathbf{x}_n$ and $t = T$, the two bounds for $\mathbf{z}(t)$ give:

$$|\mathbf{y}_n - \Phi_T(\mathbf{x}_n)| \leq \varepsilon\, T\, \exp\left\{\min\{L_V, L_{NN}\}\, T\right\}.$$

Note that $L_{NN} \leq \|W\| \cdot \|A\|$ because $\boldsymbol{\sigma}$ is 1-Lipschitz, so the approximation rate (2.3) is obtained by direct application of (4.10). □

4.6. Proof of theorem 8

We seek to find $(W, A, \mathbf{b})$ such that the generated vector field moves, compresses and stretches the mass distributed following ρ_0, to drive it approximately to the target density, given by ρ_*. This is achieved in four steps, illustrated in figs. 7 to 10 for the case $d = 2$.

1. Preparation. We compress $\operatorname{supp}(\mu_0)$ into $[0,1]^d$. To do this, we aim to find a control

$$(W, A, \mathbf{b}) \in L^\infty\left((0,T); \mathbb{R}^{d\times p} \times \mathbb{R}^{p\times d} \times \mathbb{R}^p\right)$$

such that, in a time $T_1 > 0$, the flow Φ_{T_1} of (1.3) satisfies

$$\Phi_{T_1}(\operatorname{supp}(\mu_0)) \subset [0,1]^d$$

For $k = 1, \ldots, d$, we fix the hyperplane $x^{(k)} = 0$ and a compressive velocity field by taking $(\mathbf{w}, \mathbf{a}, b) = (-\mathbf{e}_k, \mathbf{e}_k, 0)$. We choose $T_{1,k} > 0$ sufficiently large to ensure

$$\Phi_{T_{1,k}}\left(\operatorname{supp}(\mu_0) \cap \{x^{(k)} \geq 0\}\right) \subset \{0 \leq x^{(k)} \leq 1\}.$$

We repeat the operation with the hyperplanes $x^{(k)} = 1$ for $k = 1, \ldots, d$, taking $(\mathbf{w}, \mathbf{a}, b) = (\mathbf{e}_k, -\mathbf{e}_k, 1)$ and $T'_{1,k} > 0$ such that

$$\Phi_{T'_{1,k}} \circ \Phi_{T_{1,k}}\left(\operatorname{supp}(\mu_0) \cap \{x^{(k)} \leq 0\}\right) \subset \{0 \leq x^{(k)} \leq 1\}.$$

Both operations are possible in a finite time because μ_0 has compact support. In the end, we will have built piecewise constant controls $(\mathbf{w}, \mathbf{a}, b)$ that take $2d$ values, such that

$$\Phi_{T_1}(\operatorname{supp}(\mu_0)) \subset [0,1]^d, \quad \text{for } T_1 := \sum_{k=1}^{d}\left(T_{1,k} + T'_{1,k}\right).$$

Using p neurons, we can simultaneously apply p controls, because the characteristic curves of (2.4) when $(\mathbf{w}, \mathbf{a}, b) = (\pm\mathbf{e}_k, \pm\mathbf{e}_k, 1)$ are parallel to the hyperplanes $\{x^{(l)} = 0\}$

for every $l \neq k$. So, the total number of values taken by $(W, A, \mathbf{b})$ is $\lceil 2d/p \rceil$.

2. Partition. We aim to divide $[0,1]^d$ into a collection of n^d hyperrectangles, each containing a mass of $1/n^d$, as distributed by μ_0. The process can be visualized in fig. 7 for $d = 2$. For simplicity, we redefine $\mu_0 := \Phi_{T_1} \# \mu_0$ with density ρ_0, now satisfying $\operatorname{supp}(\mu_0) \subset [0,1]^d$. Let $n \geq 1$ and consider the function

$$t \longmapsto \int_{[0,t)\times[0,1]^{d-1}} d\mu_0 = \int_{[0,t)\times[0,1]^{d-1}} \rho_0.$$

This function is continuous, strictly increasing (by absolute continuity), equal to 0 at $t = 0$ and equal to 1 at $t = 1$. Therefore, we can choose $n + 1$ numbers

$$c_0 = 0 < c_1 < \cdots < 1 = c_n$$

such that, for $i_1 = 0, \ldots, n-1$,

$$\int_{[c_{i_1}, c_{i_1+1}]\times[0,1]^{d-1}} \rho_0 = \frac{1}{n}.$$

Similarly, for each $i_1 = 0, \ldots, n-1$ we choose $n+1$ numbers

$$c_{i_1,0} = 0 < c_{i_1,1} < \cdots < 1 = c_{i_1,n}$$

such that, for $i_2 = 0, \ldots, n-1$,

$$\int_{[c_{i_1},c_{i_1+1}]\times[c_{i_1,i_2}, c_{i_1,i_2+1}]\times[0,1]^{d-2}} \rho_0 = \frac{1}{n^2}.$$

Repeating this operation recursively for each coordinate, we end up with n^d hyperrectangles

$$C^0_{i_1,\ldots,i_d} := [c_{i_1}, c_{i_1+1}] \times \cdots \times [c_{i_1,\ldots,i_d}, c_{i_1,\ldots,i_d+1}] \subset \mathbb{R}^d,$$

with $i_k \in \{0, \ldots, n-1\}$ for every $k = 1, \ldots, d$, such that

$$\int_{C^0_{i_1,\ldots,i_d}} \rho_0 = \frac{1}{n^d}.$$

The analogous partition for the uniform measure μ_* is

$$G_{i_1,\ldots,i_d} := \left[\frac{i_1}{n}, \frac{i_1+1}{n}\right] \times \cdots \times \left[\frac{i_d}{n}, \frac{i_d+1}{n}\right].$$

For the sake of readability, we will denote each multi-index by $I = (i_1, \ldots, i_d) \in \{0, \ldots, n-1\}^d$ and write C^0_I and G_I.

3. Control. We aim to define the controls that expand and compress the mass until the hyperrectangles C^0_I approximate a corresponding collection of n^d hypercubes, each of them containing the same mass $1/n^d$, as distributed by μ_*. Ideally, we would build $(W, A, \mathbf{b})$ such that the flow of the ODE (1.3) satisfied

$$\Phi_T(C^0_I; W, A, \mathbf{b}) = G_I \qquad (4.13)$$

for each $I \in \{0, \ldots, n-1\}^d$. This would be done by transforming each hyperplane $\{x^{(k)} = c_{i_1,\ldots,i_k}\}$ into a target hyperplane $\{x^{(k)} = i_k/n\}$.
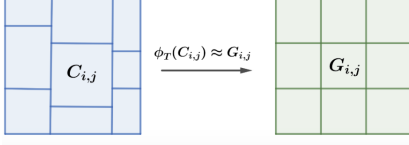
12

Figure 8: Representation of the ideal transformation of the rectangles C_I into the corresponding ones G_I.

Figure 9: Construction of the partition in rectangles G_I^δ.

However, this task is not possible in general, since any $\{x^{(k)} = i_k/n\}$ can be a target for multiple distinct hyperplanes $\{x^{(k)} = c_{i_1,\dots,i_k}\}$. We therefore relax the problem to δ-approximate control by considering δ-displacements of the target hyperplanes $\{x^{(k)} = i_k/n\}$. Now we aim to control each C_I^0 to a corresponding target G_I^δ that is δ-close to G_I, for a sufficiently small $\delta > 0$.

Let us build the new hyperrectangles G_I^δ, a process shown in fig. 9. For each $k \in \{2,\dots,d\}$, $(i_1,\dots,i_{k-1}) \in \{0,\dots,n-1\}^{k-1}$ and $i_k \in \{0,\dots,n\}$, we define

$$g_{i_1,\dots,i_k}^\delta := i_k/n + \delta(c_{i_1,\dots,i_k} - \tilde{c}_{i_k}),$$

where

$$\tilde{c}_{i_k} := \min\{c_{i_1',\dots,i_{k-1}',i_k} : (i_1',\dots,i_{k-1}') \in \{0,\dots,n-1\}^{k-1}\}.$$

Note that $g_{i_1,\dots,i_k}^\delta = 1$ whenever $i_k = n$. By construction,

$$g_{i_1,\dots,i_{k-1},i_k}^\delta < g_{i_1',\dots,i_{k-1}',i_k}^\delta \iff c_{i_1,\dots,i_{k-1},i_k} < c_{i_1',\dots,i_{k-1}',i_k}$$

and

$$g_{i_1,\dots,i_{k-1},i_k}^\delta = g_{i_1',\dots,i_{k-1}',i_k}^\delta \iff c_{i_1,\dots,i_{k-1},i_k} = c_{i_1',\dots,i_{k-1}',i_k}.$$

By recursion, we define a new partition of $[0,1]^d$ into a collection of rectangles G_I^δ with $I \in \{0,\dots,n-1\}^d$, where

$$G_I^\delta \subset G_I + \{0\} \times [-\delta,\delta]^{d-1}.$$

Moreover, this partition mimics the structure of the partition defined for μ_0, in the sense that there is the same number of distinct target hyperplanes as initial ones to be controlled. To sum up, if we take $\delta < 1/n$, we end up with:

$$\left\{ C_I^0 : I \in \{0,\dots,n-1\}^d \right\} \text{ s.t. } \int_{C_I^0} d\mu_0 = \frac{1}{n^d}, \quad (4.14)$$

$$\left\{ G_I^\delta : I \in \{0,\dots,n-1\}^d \right\} \text{ s.t. } \int_{G_I^\delta} d\mu_* \leq \frac{3^d}{n^d},$$

$$\text{and } \operatorname{diam}(G_I^\delta) \leq 3\sqrt{d}\,\frac{1}{n}.$$

It is left to map C_I^0 to G_I^δ for each I. This is based on the following lemma, whose proof we postpone for readability.

Lemma 13. *Let $d \geq 2$, $\mu_0 \in \mathcal{P}_{ac}^c(\mathbb{R}^d)$ with density ρ_0, ρ_* the uniform density in $[0,1]^d$, and $T > 0$ be fixed. Let $n \geq 1$ and consider a family of hyperrectangles such as*
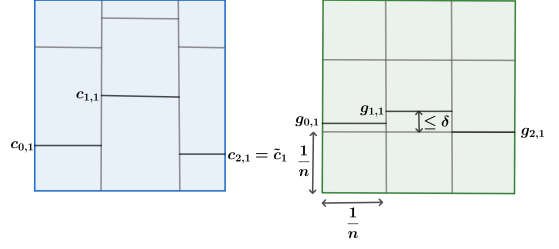
(4.14). *For any $p_1,\dots,p_d \geq 1$, there exists a piecewise constant control*

$$(W, A, \mathbf{b}) \in L^\infty((0,T); \mathbb{R}^{p\times d} \times \mathbb{R}^{d\times p} \times \mathbb{R}^p)$$

with $p = p_1 + \dots + p_d$ such that, for each $I = (i_1,\dots,i_d) \in \{0,\dots,n-1\}^d$, the flow Φ_T generated by (1.3) satisfies

$$\Phi_T(C_I^0; W, A, \mathbf{b}) = G_I^\delta$$

Furthermore, the number of discontinuities of $(W, A, \mathbf{b})$ is

$$L = \max\{\lceil n/p_1 \rceil, \dots, \lceil n^d/p_d \rceil\} - 1.$$

4. Estimates. We compute the W_q-distance between both measures to verify the approximate control. Let Φ_T be the flow given by lemma 13, satisfying

$$\Phi_T(C_I) = G_I^\delta, \text{ for } I = (i_1,\dots,i_d) \in \{0,\dots,n-1\}^d.$$

Let us quantify, in the Wasserstein-q distance, the proximity of $\Phi_{T\#}\mu_0$ to μ_*. We have

$$W_q(\mu(T),\mu_*) = W_q(\Phi_{T\#}\mu_0,\mu_*)$$
$$\leq \sum_{I \in \{0,\dots,n-1\}^d} W_q(\Phi_{T\#}\mu_0|_{G_I^\delta},\mu_*|_{G_I^\delta}), \quad (4.15)$$

see [34] for the inequality. For each $I \in \{0,\dots,n-1\}^d$, let $\gamma_I : \mathbb{R}^d \to \mathbb{R}^d$ be the measurable function that satisfies

$$\gamma_{I\#}(\Phi_{T\#}\mu_0|_{G_I^\delta}) = \mu_*|_{G_I^\delta},$$

attaining the minimum in Monge formulation (2.6) for W_q. In particular, γ_I only redistributes the mass inside G_I^δ, so

$$\int_{\mathbb{R}^d} |x - \gamma_I(x)|^q d\mu_*|_{G_I^\delta} = \int_{G_I^\delta} |x - \gamma_I(x)|^q d\mu_*$$
$$\leq \operatorname{diam}(G_I^\delta)^q \int_{G_I^\delta} d\mu_* \leq \frac{3^{q+d}d^{q/2}}{n^{q+d}}.$$

Plugging this bound into (4.15), we conclude that

$$W_q(\mu(T),\mu_*) \leq 3^{1+d/q}\sqrt{d}\,n^d \left(\frac{1}{n^{q+d}}\right)^{1/q}$$
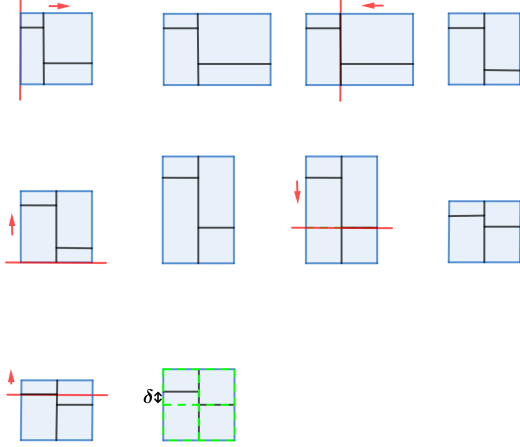$$= 3^{1+d/q}\sqrt{d}\,n^{-(1+d/q-d)}.$$

13

Figure 10: From left to right, by rows: We transform the rectangles C_I to approximate the corresponding ones G_I^δ (in green)

It follows that

$$W_q(\mu(T), \mu_*) \xrightarrow{n \to \infty} 0 \iff q < \frac{d}{d-1},$$

and, in that case, $W_q(\mu(T), \mu_*) < \varepsilon$ is obtained for

$$n > \left(\frac{3^{1+d/q}\sqrt{d}}{\varepsilon} \right)^{\frac{1}{1+d/q-d}},$$

hence proving theorem 8.

Proof of lemma 13. We control the rectangles by mapping each hyperplane to its corresponding target hyperplane. This strategy is illustrated in fig. 10.

The proof is divided into three steps. First, we achieve simultaneous control for any p hyperplanes orthogonal to a fixed direction. Second, for $N \geq p$ hyperplanes, we iteratively control p-subsets. Third, we show that this approach is applicable to all d canonical directions simultaneously.

Step 1. Let $k \in \{1, \ldots, d\}$, $p \geq 1$ and

$$-\infty < c_1 < \cdots < c_p < \infty,$$
$$-\infty < g_1 < \cdots < g_p < \infty.$$

We aim to find constant controls $\{(\mathbf{w}_i, \mathbf{a}_i, b_i)\}_{i=1}^p \subset \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}$ such that the flow generated by (1.3) satisfies

$$\Phi_T(\{x^{(k)} = c_i\}) = \{x^{(k)} = g_i\}, \quad i = 1, \ldots, p. \quad (4.16)$$

In particular, we will take $\mathbf{a}_i = \mathbf{e}_k$ for all i, and b_i such that $-b_i < \min\{c_i, g_i\}$. Since the field

$$\sum_{i=1}^p \mathbf{w}_i \sigma(\mathbf{a}_i \cdot \mathbf{x} + b_i) = \sum_{i=1}^p \mathbf{w}_i \sigma(x^{(k)} + b_i)$$

only depends on the $x^{(k)}$-coordinate, it is projectable onto the $x^{(k)}$-axis, i.e., the forward evolution of a hyperplane orthogonal to any coordinate-axis is a hyperplane orthogonal to the same coordinate-axis, for any time. Therefore,

we can identify each $\{x^{(k)} = c_i\}$ and $\{x^{(k)} = g_i\}$ with the point $c_i \in \mathbb{R}$ or $g_i \in \mathbb{R}$ and study their evolution in the real line, so the problem becomes one-dimensional. Thus, we identify $x^{(k)} \equiv x$ and fix $\mathbf{w}_i = w_i \mathbf{e}_k$ with $w_i \in \mathbb{R}$, so we aim to find $(w_i)_{i=1}^p \subset \mathbb{R}$ and $(b_i)_{i=1}^p \subset \mathbb{R}$ such that $\Phi_T(c_i) = g_i$ for $i = 1, \ldots, p$. We proceed by induction on p.

First, we consider $p = 1$ and let $c_1, g_1 \in \mathbb{R}$. Take any $-b < \min\{c_1, g_1\}$, so $\{c_1, g_1\} \subset \{x + b > 0\}$, and $w = \text{sign}(g_1 - c_1)$, pointing from c_1 to g_1. Then,

$$\Phi_{\tilde{T}}(c_1) = g_1, \quad \text{for } \tilde{T} = \left| \log\left(\frac{g_1 - b}{c_1 - b} \right) \right|.$$

Using a time rescaling argument identical to the one in the proof of 1, this control can be done in any time $T > 0$. Also note that $\{x + b \leq 0\}$ is fixed by the flow Φ_t.

In the inductive step, we assume that the statement is true for some $p \geq 1$, and consider

$$-\infty < c_1 < \cdots < c_{p+1} < \infty,$$
$$-\infty < g_1 < \cdots < g_{p+1} < \infty.$$

Let $\{(w_i, b_i)\}_{i=1}^p \subset \mathbb{R} \times \mathbb{R}$ with $-b_i < \min\{c_i, g_i\}$ for all i, such that the flow of the one-dimensional problem satisfies

$$\Phi_T(c_i) = g_i, \quad i = 1, \ldots, p.$$

We want to add a new pair $(w_{p+1}, b_{p+1}) \in \mathbb{R} \times \mathbb{R}$ such that

$$\Phi_T(c_i) = g_i, \quad i = 1, \ldots, p + 1.$$

Case 1: $c_p < g_p$. Take $b_{p+1} = -g_p$, so (1.3) becomes

$$\dot{x} = \sum_{i=1}^p w_i(x + b_i)\mathbb{1}_{c_i < x}(x) + w_{p+1}(x - g_p)\mathbb{1}_{g_p < x}(x).$$

The added velocity $w_{p+1}(x - g_p)$ only acts on the half-space $\{x \geq g_p\}$, so the points $\{c_i\}_{i=1}^p$ are only subject to the drift field

$$d(x) := \sum_{i=1}^p w_i(x + b_i)\mathbb{1}_{c_i < x}(x).$$

Therefore, when adding (w_{p+1}, b_{p+1}), we still have

$$\Phi_T(c_i) = g_i, \quad i = 1, \ldots, p.$$

If $c_{p+1} \leq g_p$, there is $0 \leq s < T$ such that $\Phi_s(c_{p+1}) = g_p$. Otherwise, if $c_{p+1} > g_p$, consider $s = 0$. Note that s only depends on the p first neurons, on c_{p+1} and on g_p, and it is thus independent of (w_{p+1}, b_{p+1}). Therefore, c_{p+1} is only subject to $d(x)$ for $t \in (0, s)$, and to $d(x) + w_{p+1}(x - g_p)$ for $t \in (s, T)$. More precisely:

$$\frac{d}{dt}\Phi_t(c_{p+1}) = d(\Phi_t(c_{p+1})) + w_{p+1}(\Phi_t(c_{p+1}) - g_p)\mathbb{1}_{s \leq t < T}(t).$$

14

Then, with a similar computation to that of lemma 11:

$$\Phi_T(c_{p+1}) = \Phi_{T-s} \circ \Phi_s(c_{p+1})$$
$$= \left(g_p + \frac{\sum_{i=1}^p w_i b_i - w_{p+1} g_p}{\sum_{i=1}^p w_i + w_{p+1}}\right) e^{(T-s)\sum_{i=1}^{p+1} w_i}$$
$$- \frac{\sum_{i=1}^p w_i b_i - w_{p+1} g_p}{\sum_{i=1}^p w_i + w_{p+1}}.$$

The expression converges to g_p when $w_{p+1} \to -\infty$, and diverges to ∞ when $w_{p+1} \to \infty$. Therefore, by continuity, and since $g_p < g_{p+1}$, there exists $w_{p+1} \in \mathbb{R}$ such that $\Phi_T(c_{p+1}) = g_{p+1}$.

Case 2: $g_p \le c_p < g_{p+1}$. Take $b_{p+1} = -c_p$, so (1.3) becomes

$$\dot{x} = \sum_{i=1}^p w_i(x + b_i)\mathbb{1}_{c_i < x}(x) + w_{p+1}(x - c_p)\mathbb{1}_{c_p < x}(x),$$

Again, the points $\{c_i\}_{i=1}^p$ are only subject to $d(x)$ because the added velocity only acts on $\{x \ge c_p\}$, so

$$\Phi_T(c_i) = g_i, \quad i = 1, \ldots, p.$$

The point c_{p+1} is subject to the total velocity at $t = 0$. A similar computation to Case 1 leads to

$$\Phi_T(c_{p+1}) = \left(c_p + \frac{\sum_{i=1}^p w_i b_i - w_{p+1} c_p}{\sum_{i=1}^p w_i + w_{p+1}}\right) e^{T \sum_{i=1}^{p+1} w_i}$$
$$- \frac{\sum_{i=1}^p w_i b_i - w_{p+1} c_p}{\sum_{i=1}^p w_i + w_{p+1}}.$$

$\Phi_T(c_{p+1})$ tends to c_p when $w_{p+1} \to -\infty$, and diverges to ∞ when $w_{p+1} \to \infty$. By continuity, and since $c_p < g_{p+1}$, there exists $w_{p+1} \in \mathbb{R}$ such that $\Phi_T(c_{p+1}) = g_{p+1}$.

Case 3: $g_{p+1} \le c_p$. Take $b_{p+1} = -c_p$, so (1.3) becomes

$$\dot{x} = \sum_{i=1}^p w_i(x + b_i)\mathbb{1}_{c_i < x}(x) + w_{p+1}(x - c_p)\mathbb{1}_{c_p < x},$$

and

$$\Phi_T(c_i) = g_i, \quad i = 1, \ldots, p.$$

Let $0 < s \le T$ be the first time such that $\Phi_s(c_{p+1}) = c_p$. An analogous computation to the previous cases gives

$$\Phi_s(c_{p+1}) = \left(c_{p+1} + \frac{\sum_{i=1}^p w_i b_i - w_{p+1} c_p}{\sum_{i=1}^{p+1} w_i}\right) e^{s \sum_{i=1}^{p+1} w_i}$$
$$- \frac{\sum_{i=1}^p w_i b_i - w_{p+1} c_p}{\sum_{i=1}^{p+1} w_i},$$

so

$$s = \frac{1}{\sum_{i=1}^{p+1} w_i} \log\left(\frac{\sum_{i=1}^p w_i(c_p - b_i) + 2w_{p+1} c_p}{\sum_{i=1}^p w_i(c_{p+1} + b_i) + w_{p+1}(c_{p+1} - c_p)}\right).$$

By varying w_{p+1} in $(-\sum_{i=1}^p w_i, +\infty)$, we can ensure that s can take any value in $(0, T)$. Then,

$$\Phi_T(c_{p+1}) = \Phi_{T-s} \circ \Phi_s(c_{p+1})$$
$$= \left(c_p + \frac{\sum_{i=1}^p w_i b_i}{\sum_{i=1}^p w_i}\right) e^{(T-s)\sum_{i=1}^p w_i}$$
$$- \frac{\sum_{i=1}^p w_i b_i}{\sum_{i=1}^p w_i}.$$

For $s = T$, the expression is equal to c_p, and for $s = 0$ it equals $\Phi_T(c_p) = g_p$. By continuity, and also because $g_p < g_{p+1} \le c_p$, an argument like in case 1 ensures the existence of $w_{p+1} \in \mathbb{R}$ such that $\Phi_T(c_{p+1}) = g_{p+1}$.

Step 2. Let $k \in \{1, \ldots, d\}$, $p \ge 1$, and $N \ge p$ and

$$-\infty < c_1 < \cdots < c_N < \infty, \quad -\infty < g_1 < \cdots < g_N < \infty.$$

We will show that there exist piecewise constant controls $(w_i, a_i, b_i)_{i=1}^p$ such that the flow of (1.3) satisfies

$$\Phi_T(c_i) = g_i, \quad i = 1, \ldots, N,$$

and the number of discontinuities is $\lceil N/p \rceil - 1$. We use a similar argument to the one in the proof of theorem 1. We divide $\{c_i\}_{i=1}^N$ and $\{g_i\}_{i=1}^N$ into subsets of p points

$$C_j := \{c_{(j-1)\cdot p + 1}, \ldots, c_{j \cdot p}\}, \quad G_j := \{g_{(j-1)\cdot p + 1}, \ldots, g_{j \cdot p}\},$$

for $j = 1, \ldots, \lceil N/p \rceil - 1$, and $C_{\lceil N/p \rceil}$, $G_{\lceil N/p \rceil}$ with the remaining $N - p\lfloor N/p \rfloor$ points.

The piecewise constant controls are obtained by induction on j. In each iteration, we apply step 1 to define p constant controls $(w_i^j, a_i^j, b_i^j)_{i=1}^p$ that map the p points of C_j to the corresponding ones in G_j in time $\frac{T}{\lceil N/p \rceil}$. Note that the initialization of induction in step 1 ensures that the previously controlled subsets $C_1, \ldots, C_{j-1}$ can remain fixed during the subsequent iterations, which trivializes the induction. Finally, we have the piecewise constant controls

$$(w_i, a_i, b_i)_{i=1}^p = \sum_{j=1}^{\lceil N/p \rceil} (w_i^j, a_i^j, b_i^j)_{i=1}^p \mathbb{1}_{\left(\frac{(j-1)T}{\lceil N/p \rceil}, \frac{jT}{\lceil N/p \rceil}\right)}(t),$$

which achieve the desired objective.

Step 3. For every $k = 1, \ldots, d$, let $p_k, N_k \ge 1$, and

$$-\infty < c_1^k < \cdots < c_{N_k}^k < \infty, \quad -\infty < g_1^k < \cdots < g_{N_k}^k < \infty.$$

For each fixed direction $k \in \{1, \ldots, d\}$, step 2 is used to build piecewise constant controls

$$(\mathbf{w}_{j,k}, \mathbf{a}_{j,k}, b_{j,k})_{j=1}^{p_k} = (w_{j,k}\,\mathbf{e}_k, \mathbf{e}_k, b_{j,k})_{j=1}^{p_k}$$

with $\lceil N_k/p_k \rceil - 1$ discontinuities, such that

$$\Phi_T(\{x^{(k)} = c_i^k\}) = \{x^{(k)} = g_i^k\}, \quad i = 1, \ldots, N.$$

Moreover, only the k-th coordinate is varying on each flow, as argued in steps 1 and 2 when we simplified the problem

to one dimension. We define each tern of the assembled control $(\mathbf{w}_j, \mathbf{a}_j, b_j)_{j=1}^p$, with $p = \sum_{k=1}^d p_k$, by

$$(\mathbf{w}_j, \mathbf{a}_j, b_j) = \left(\mathbf{w}_{j-\sum_{i=1}^{k-1} p_i, p_k}, \mathbf{a}_{j-\sum_{i=1}^{k-1} p_i, p_k}, b_{j-\sum_{i=1}^{k-1} p_i, p_k}\right),$$

for $\sum_{i=1}^{k-1} p_i + 1 \leq j \leq \sum_{i=1}^k p_i$. Therefore, the resulting neural ODE (1.3) on each coordinate writes

$$\dot{x}^{(k)} = w_{1,k}^{(k)}(x^{(k)} + b_{1,k}) + \cdots + w_{p_k,k}^{(k)}(x^{(k)} + b_{p_k,k})$$

All the equations of the system are independent, so each movement does not interfere with the other $d - 1$ movements. Therefore, the corresponding flow of (1.3) satisfies

$$\Phi_T(\{x^{(k)} = c_i^k\}) = \{x^{(k)} = g_i^k\}$$

for $k = 1, \ldots, d$ and $i = 1, \ldots, N_k$, and moreover, the number of discontinuities in the controls is

$$L = \max\{\lceil N_1/p_1 \rceil - 1, \ldots, \lceil N_d/p_d \rceil - 1\}$$
$$= \max_{k=1,\ldots,d} \lceil N_k/p_k \rceil - 1.$$

Recalling that $N_k = n^k$ for $k = 1, \ldots, d$ (by construction of the rectangles C_I^0 and G_I^δ), it follows the desired result. $\square$

5. Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

6. Acknowledgments

References

[1] Agrachev, A.A., Sarychev, A.V., 2021. Control on the manifolds of mappings with a view to the deep learning. Journal of Dynamical and Control Systems 28, 989–1008.

[2] Alvarez-Lopez, A., Orive-Illera, R., Zuazua, E., 2023. Optimized classification with neural odes via separability. arXiv:2312.13807.

[3] Bach, F., 2017. Breaking the curse of dimensionality with convex neural networks. Journal of Machine Learning Research 18, 1–53.

[4] Cabanilla, K.I.M., Mohammad, R.Z., Lope, J.E.C., 2024. Neural networks with relu powers need less depth. Neural Networks 172, 106073.

[5] Chang, B., Meng, L., Haber, E., Tung, F., Begert, D., 2018. Multi-level residual networks from dynamical systems view. arXiv:1710.10348.

[6] Chen, R.T.Q., Rubanova, Y., Bettencourt, J., Duvenaud, D., 2018. Neural ordinary differential equations, in: Proceedings of the 32nd International Conference on Neural Information Processing Systems, Curran Associates Inc.. p. 6572–6583.

[7] Cybenko, G., 1989. Approximation by superpositions of a sigmoidal function. Mathematics of Control, Signals and Systems 2, 303–314.

[8] DeVore, R., Hanin, B., Petrova, G., 2021. Neural network approximation. Acta Numerica 30, 327–444.

[9] Dupont, E., Doucet, A., Teh, Y.W., 2019. Augmented neural odes, in: Proceedings of the 33rd International Conference on Neural Information Processing Systems, Curran Associates Inc.. p. 3140–3150.

[10] Duprez, M., Morancey, M., Rossi, F., 2017. Controllability and optimal control of the transport equation with a localized vector field, in: 2017 25th Mediterranean Conference on Control and Automation (MED), pp. 74–79.

[11] E, W., 2017. A proposal on machine learning via dynamical systems. Communications in Mathematics and Statistics 5, 1–11.

[12] Elamvazhuthi, K., Gharesifard, B., Bertozzi, A.L., Osher, S., 2022. Neural ode control for trajectory approximation of continuity equation. IEEE Control Systems Letters 6, 3152–3157.

[13] Eldan, R., Shamir, O., 2015. The Power of Depth for Feedforward Neural Networks. JMLR: Workshop and Conference Proceedings 49, 1–34.

[14] Esteve, C., Geshkovski, B., Pighin, D., Zuazua, E., 2021. Large-time asymptotics in deep learning. arXiv:2008.02491.

[15] Esteve-Yagüe, C., Geshkovski, B., 2023. Sparsity in long-time control of neural ODEs. Systems Control Lett. 172, Paper No. 105452, 14.

[16] Fan, F., Lai, R., Wang, G., 2020. Quasi-equivalence of width and depth of neural networks.

[17] Grathwohl, W., Chen, R.T.Q., Bettencourt, J., Sutskever, I., Duvenaud, D., 2019. FFJORD: free-form continuous dynamics for scalable reversible generative models, in: 7th International Conference on Learning Representations, ICLR 2019, New Orleans.

[18] Haber, E., Ruthotto, L., 2017. Stable architectures for deep neural networks. Inverse Problems 34, 014004.

[19] Hardt, M., Ma, T., 2017. Identity matters in deep learning, in: International Conference on Learning Representations.

[20] Huang, G.B., 2003. Learning capability and storage capacity of two-hidden-layer feedforward networks. IEEE Transactions on Neural Networks 14, 274–281.

[21] Kobyzev, I., Prince, S.J., Brubaker, M.A., 2021. Normalizing flows: An introduction and review of current methods. IEEE Transactions on Pattern Analysis and Machine Intelligence 43, 3964–3979.

[22] Li, Q., Lin, T., Shen, Z., 2022. Deep learning via dynamical systems: An approximation perspective. Journal of the European Mathematical Society 25, 1671–1709.

[23] Lin, H., Jegelka, S., 2018. Resnet with one-neuron hidden layers is a universal approximator, in: Proceedings of the 32nd International Conference on Neural Information Processing Systems, p. 6172–6181.

[24] Lu, Z., Pu, H., Wang, F., Hu, Z., Wang, L., 2017. The expressive power of neural networks: A view from the width, in: Proceedings of the 31st International Conference on Neural Information

Processing Systems, Curran Associates Inc., Red Hook, NY, USA. p. 6232–6240.

[25] Mhaskar, H., Liao, Q., Poggio, T., 2017. When and why are deep networks better than shallow ones?, in: Proceedings of the AAAI conference on artificial intelligence.

[26] Papamakarios, G., Nalisnick, E., Rezende, D.J., Mohamed, S., Lakshminarayanan, B., 2021. Normalizing flows for probabilistic modeling and inference. J. Mach. Learn. Res. 22.

[27] Pinkus, A., 1999. Approximation theory of the mlp model in neural networks. Acta Numerica 8, 143–195.

[28] Ruiz-Balet, D., Zuazua, E., 2023. Neural ODE control for classification, approximation, and transport. SIAM Rev. 65, 735–773.

[29] Ruiz-Balet, D., Zuazua, E., 2024. Control of neural transport for normalising flows. Journal de Mathématiques Pures et Appliquées 181, 58–90.

[30] Ruthotto, L., 2024. Differential equations for continuous-time deep learning. arXiv:2401.03965.

[31] Scagliotti, A., 2023. Deep learning approximation of diffeomorphisms via linear-control systems. Mathematical Control and Related Fields 13, 1226–1257.

[32] Tabuada, P., Gharesifard, B., 2023. Universal approximation power of deep residual neural networks through the lens of control. IEEE Transactions on Automatic Control 68, 2715–2728.

[33] Valentine, F.A., 1945. A lipschitz condition preserving extension for a vector function. American Journal of Mathematics 67, 83–93.

[34] Villani, C., 2008. Optimal transport – Old and new. Springer Berlin, Heidelberg. volume 338. pp. xxii+973.

[35] Yun, C., Sra, S., Jadbabaie, A., 2019. Small relu networks are powerful memorizers: A tight analysis of memorization capacity, in: Proceedings of the 33rd International Conference on Neural Information Processing Systems, Curran Associates Inc.. p. 15558–15569.

[36] Zhang, C., Bengio, S., Hardt, M., Recht, B., Vinyals, O., 2016. Understanding deep learning requires rethinking generalization. Communications of the ACM 64.