On the Existence and Neural Network Representation of Separable Control Lyapunov Functions ⋆

Mario Sperl [a], Jonas Mysliwitz [a], Lars Grüne [a]

[a] *Mathematical Institute, University of Bayreuth, Bayreuth, Germany*

Abstract

In this paper, we investigate the ability of neural networks to mitigate the curse of dimensionality in representing control Lyapunov functions. To achieve this, we first prove an error bound for the approximation of separable functions with neural networks. Subsequently, we discuss conditions on the existence of separable control Lyapunov functions, drawing upon tools from nonlinear control theory. This enables us to bridge the gap between neural networks and the approximation of control Lyapunov functions. Moreover, we present a network architecture and a training algorithm to illustrate the theoretical findings on a 10-dimensional control system.

Key words: control Lyapunov functions; neural networks; curse of dimensionality.

1 Introduction

Control Lyapunov functions (clfs) serve as a certificate of asymptotic null-controllability and can also be used to examine robustness against uncertainties and disturbances or to study performance criteria. However, their most common application lies in designing stabilizing feedback laws using the clf as guidance towards the equilibrium. Since, in general, it is quite hard to compute clfs analytically, we rely on numerical methods. However, traditional numerical methods, which rely on a grid-based approach for the computation of the derivative of the clf, suffer from the curse of dimensionality. This means that, to achieve a certain accuracy, the number of required grid points and, thus, the numerical effort grows exponentially in the dimension of the state space. Consequently, such approaches become impractical in high dimensions.

This paper concerns the use of neural networks (NNs) to circumvent the curse of dimensionality for approximating clfs. Our approach is related to the work Sontag (1991), which investigates structural properties on control systems that allow for an exact representation of a (possibly discontinuous) stabilizing feedback by NNs. Further, there exist several papers that present algorithms for the computation of clfs by NNs, see, e.g. Khansari-Zadeh and Billard (2014); Long and Bayoumi (1993). However, while the algorithms therein have similarities with our numerical approach, none of them provides a complexity analysis regarding the curse of dimensionality. Establishing conditions for mitigating the curse of dimensionality in computing clfs is the main contribution of this work. Addressing this challenge requires identifying a suitable class of functions that enables NNs to avoid the curse of dimensionality.

There exist various recent papers that discuss results regarding a curse-of-dimensionality-free approximation of solutions of particular kinds of partial differential equations, see, e.g., Beck et al. (2023); Darbon et al. (2020); Gonon and Schwab (2023). In particular, some of these references exploit the smoothness of solutions of 2nd order Hamilton-Jacobi-Bellman equations. However, when it comes to computing a clf for a deterministic system, which can be characterized as a solution of a particular first-order Hamilton-Jacobi-Bellman equation, we cannot expect such a level of smoothness. Thus, we rely on a different structural assumption that allows NNs to mitigate the curse of dimensionality. To this end, we consider so-called separable functions. Informally speaking, a mapping is called separable if it can be written as a sum of functions that are each defined on some lower-dimensional domain. Separable functions fall into the

⋆ A preliminary version of this paper was presented at the 12th IFAC Symposium on Nonlinear Control Systems NOLCOS 2022, see Grüne and Sperl (2023). Corresponding author M. Sperl.

Email addresses: mario.sperl@uni-bayreuth.de (Mario Sperl), jonas.mysliwitz@uni-bayreuth.de (Jonas Mysliwitz), lars.gruene@uni-bayreuth.de (Lars Grüne).

class of compositional functions. The ability of NNs to avoid the curse of dimensionality for compositional functions has been discussed for instance in Dahmen (2023); Kang and Gong (2022); Poggio et al. (2017). Compared to general compositional functions, separable functions have a simpler structure that allows for more precise estimates, while the classes of control systems admitting separable clfs are still non-trivial.

Contribution

In this paper, we bridge the gap between NN approximation theory and the computation of clfs via NNs. First, we provide error estimates for the approximation of separable functions with NNs. While Grüne (2021); Grüne and Sperl (2023) state asymptotic results for $\mathcal{L}_\infty$ approximation, this paper derives detailed error bounds, including all relevant constants. Furthermore, under additional assumptions, we prove an extension of the approximation result towards partial derivatives. Afterwards, we extend the results for Lyapunov functions in Grüne (2021) to clfs. Specifically, based on Grüne and Sperl (2023) we use methods from nonlinear control theory to identify conditions on the control system such that a separable clf exists. Additionally, we explore achieving separability through a state space transformation. Overall, we identify scenarios where NNs can provably avoid the curse of dimensionality in the computation of clfs. Finally, we propose a network architecture and training algorithm, and provide an empirical evaluation of the benefit of the separable structure. Compared to Grüne and Sperl (2023), this paper contributes detailed theoretical results on approximation errors, results on separability after a suitable transformation, and extended numerical results. In this context, we would also like to mention those topics that are not part of this paper. While this paper provides an expressivity result and proposes a training algorithm, it does not delve into the analysis of the convergence of the training algorithm or the generalization properties of the NN. Regarding the last point, which is of high importance for practical usage, we would like to refer to the works Dai et al. (2021); Liu et al. (2023, 2024), where methods to verify that the NN output satisfies the Lyapunov conditions have been developed, thus providing a tool to verify generalization properties. In particular, we would like to point out that Liu et al. (2024) leverages a compositional structure of the control system for verification, aligning well with the use of separability for efficient representation discussed in this paper. Specifically, the separable structure might be beneficial for formal verification by enabling the decomposition of the verification process for high-dimensional systems into the verification of smaller, lower-dimensional subsystems. Moreover, we only consider the case in which smooth clfs exist, which allows us to better focus on the main results of this paper. Nonsmooth clfs will be addressed in future research.

Outline

The remainder of this paper is organized as follows: The problem formulation is introduced in the next section. Afterwards, we provide a complexity analysis regarding the approximation of separable functions with NNs. In Section 4 we focus on the existence of separable clfs, while numerical test cases are performed in Section 5. Finally, Section 6 concludes the paper.

Notation

For $n \in \mathbb{N}$ we set $[n] := \{1, \dots, n\}$. We denote the infinity norm for continuous functions f on some compact set K via $\|f\|_{\infty,K} := \sup_{x\in K}\|f(x)\|$. The symbol D is used to denote the classic differential operator. Moreover, for some multi-index $\alpha \in \mathbb{N}^n$ we use D_α to denote the higher-order partial derivative with respect to α. We make use of the comparison functions $\mathcal{K}$ and $\mathcal{K}_\infty$, where $\mathcal{K}$ denotes all continuous and strictly increasing functions $\gamma\colon \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ with $\gamma(0) = 0$ and $\mathcal{K}_\infty$ comprises all $\mathcal{K}$-functions that satisfy $\lim_{r\to\infty} \gamma(r) = \infty$.

2 Problem formulation

We consider a control system of the form

$$\dot{x} = f(x, u), \tag{1}$$

where the right-hand side $f\colon \mathbb{R}^n \times U \to \mathbb{R}^n$ is continuous, locally Lipschitz in x, and has an equilibrium at 0, i.e., $f(0, 0) = 0$. The input set is denoted as $U \subset \mathbb{R}^m$ and the admissible control functions are given as the set of measurable and locally essentially bounded functions $u\colon \mathbb{R}_{\geq 0} \to U$. In order to avoid technicalities, we assume our system (1) to be defined on the whole domain $\mathbb{R}^n$. We are interested in stabilizing the system towards the origin. To this end, we assume the control system (1) to be asymptotically controllable. In (Sontag, 1983, Theorem 2.5) it has been shown that asymptotic controllability is equivalent to the existence of a clf in the sense of Dini. However, in the scope of this paper, we will only consider the case where our control system (1) admits a continuously differentiable clf, where the Dini derivative equals the gradient. This allows us to ensure compatibility with some theorems from the literature cited in the subsequent sections and avoids distracting technical difficulties. The important case that no smooth clf exists will be investigated in future research, cf. Section 6.

Definition 1 *A continuously differentiable function $V\colon \mathbb{R}^n \to \mathbb{R}$ is called (smooth) control Lyapunov function (clf) for (1) if there exist $\alpha_1, \alpha_2 \in \mathcal{K}_\infty$ and $\alpha_3 \in \mathcal{K}$ such that for $x \in \mathbb{R}^n$*

$$\alpha_1(\|x\|) \leq V(x) \leq \alpha_2(\|x\|), \tag{2a}$$

$$\inf_{u\in U} DV(x)f(x, u) \leq -\alpha_3(\|x\|). \tag{2b}$$

3 Neural networks approximating separable functions

3.1 Preliminaries on neural networks

From a mathematical point of view, a neural network (NN) is a mapping $x \mapsto W(x;\theta)$ that takes some input vector $x \in \mathbb{R}^n$ and processes it according to its parameters θ in order to return some output. In case of a feedforward network, the value of a neuron $y_k^{(l)}$ in layer l with number k is determined via

$$y_k^{(l)} = \sigma_l\Big(\sum_{i=1}^{N_{l-1}} w_{k,i}^{(l)} y_i^{l-1} + b_k^{(l)}\Big), \tag{3}$$

where $w_{k,i}^{(l)} \in \mathbb{R}$ are weights, $b_k^{(l)} \in \mathbb{R}$ are bias terms and $\sigma_l\colon \mathbb{R} \to \mathbb{R}$ is the activation function of layer l. We solely consider feedforward networks with a one-dimensional output $W(x;\theta) \in \mathbb{R}$ and the identity as activation function in the last layer. It has been shown in Cybenko (1989) that the set of NNs with one hidden layer and a continuous sigmoidal activation function is dense in $C([0,1]^n)$. Since we are interested in the numerical effort, we need a quantitative version of an approximation theorem. To this end, we characterize the complexity of a NN by the number of neurons in its hidden layers. Further, for $p \in \mathbb{N}$, $r \in \mathbb{R}_{>0}$, and $K \subset \mathbb{R}^n$ compact we define

$$W_{p,r}(K) := \left\{F \in C^p(K,\mathbb{R}) \mid \|F\|_{W_p(K)} \le r\right\},$$

where $\|F\|_{W_p(K)} := \sum_{0 \le |\alpha| \le p} \|D_\alpha F\|_{\infty,K}$.

Theorem 2 *Let $p \in \mathbb{N}$, $r, R \in \mathbb{R}_{>0}$ and $\sigma \in \mathcal{C}^\infty(\mathbb{R},\mathbb{R})$ be not a polynomial. Then for every $n \in \mathbb{N}$ there exists $\mu_n > 0$ such that for all $M \in \mathbb{N}$, any NN of the form*

$$W(x;\theta) = \sum_{k=1}^{M} w_k^{(2)} \sigma\left(\sum_{i=1}^{n} w_{k,i}^{(1)} x_i + b_k\right), \tag{4}$$

and any $F \in C^p(K,\mathbb{R})$ there exists a parameter vector $\theta_F = (w_F^{(1)}, w_F^{(2)}, b_F)$ such that

$$\|W(\cdot;\theta_F) - F(\cdot)\|_{\infty,K} \le \mu_n M^{-\frac{p}{n}} \widetilde{R} \|F\|_{W_p(K)}, \tag{5}$$

where $K := [-R,R]^n$, $\widetilde{R} := \max\{R^p, 1\}$. Further, assume $p \ge 2$ and $\|\sigma''\|_{\infty,\mathbb{R}} \le C_\sigma$ for some $C_\sigma > 0$. Then for a family of functions $\mathcal{F} \subset W_{p,r}(K)$ for which $\|w_F^{(1)}\|_\infty$ and $\|w_F^{(2)}\|_1$ are uniformly bounded by some $C_\theta > 0$ for all $F \in \mathcal{F}$ and $M \in \mathbb{N}$, in addition we have for each $|\alpha| = 1$ and $F \in \mathcal{F}$

$$\|D_\alpha W(\cdot;\theta_F) - D_\alpha F(\cdot)\|_{\infty,K} \le \sqrt{\mu_n} M^{-\frac{p}{2n}} \sqrt{C\widetilde{R}},$$

where $C = 2r(r + C_\theta^3 C_\sigma)$ is independent of n and M.

PROOF. The first statement follows immediately from (Mhaskar, 1996, Theorem 2.1). For the second statement, observe that under the assumptions for all $F \in \mathcal{F}$ and the corresponding $W_F = W(\cdot;\theta_F)$ the derivatives $D_\alpha F$ and $D_\alpha W_F(\cdot)$ are Lipschitz with Lipschitz constants $L_{F'} = r$ and $L_{W'} = C_\theta^3 C_\sigma$, respectively. Then for $e = \alpha^T$ the mean value theorem implies that

$$\left| D_\alpha F(x) - \frac{F(x+he) - F(x)}{h} \right| \le L_{F'} h,$$

$$\left| D_\alpha W_F(x) - \frac{W_F(x+he) - W_F(x)}{h} \right| \le L_{W'} h.$$

By the triangle inequality we obtain

$$\begin{aligned}
&|D_\alpha W_F(x) - D_\alpha F(x)| \\
&\le \left| D_\alpha W_F(x) - \frac{W_F(x+he) - W_F(x)}{h} \right| \\
&+ \left| \frac{W_F(x+he) - W_F(x)}{h} - \frac{F(x+he) - F(x)}{h} \right| \\
&+ \left| D_\alpha F(x) - \frac{F(x+he) - F(x)}{h} \right| \\
&\le L_{W'} h + 2\mu_n M^{-\frac{p}{n}} \widetilde{R} r/h + L_{F'} h \\
&\le \sqrt{\mu_n M^{-\frac{p}{n}} 2r(L_{F'} + L_{W'})\widetilde{R}},
\end{aligned}$$

where we used $h = \sqrt{2\mu_n M^{-\frac{p}{n}} \widetilde{R} r/(L_{F'} + L_{W'})}$ in the last inequality, which minimizes the expression in the second last step. This yields the claim. □

Note that the requirement for σ to have a uniformly bounded second derivative is satisfied by many commonly used smooth activation functions, such as the softplus function $\sigma(x) = \log(1 + e^x)$, the sigmoid function $\sigma(x) = \frac{1}{1+e^{-x}}$, and the hyperbolic tangent $\sigma(x) = \tanh(x)$. In the following, for a function $F \in C^p(K,\mathbb{R})$ we denote by $\theta_F = (w_F^{(1)}, w_F^{(2)}, b_F)$ the vector of parameters satisfying (5) for a NN of the form (4). We can conclude from Theorem 2 that the number of neurons needed to provide an approximation up to some accuracy $\varepsilon > 0$ is given by $M = \mathcal{O}(\varepsilon^{-\frac{n}{p}})$, which has been shown in Mhaskar (1996) to be best possible. Thus, in general, NNs suffer from the curse of dimensionality.

3.2 Mitigating the curse of dimensionality with neural networks

In this section, we derive an error bound for the approximation of so-called separable functions with NNs. This demonstrates that NNs can mitigate the curse of dimensionality for a suitable class of functions and provides a detailed expression of the dependency on the involved constants.

Definition 3 *Let $F \in \mathcal{C}^p(\mathbb{R}^n, \mathbb{R})$ and $d \in [n]$. Then F is called (strictly) d-separable if for some $s \in [n]$ there exist $d_1, \dots, d_s \in [d]$ and functions $F_1, \dots, F_s$ with $F_j \in \mathcal{C}^p(\mathbb{R}^{d_j}, \mathbb{R})$, such that for all $x \in \mathbb{R}^n$ it holds*

$$F(x) = \sum_{j=1}^{s} F_j(z_j), \tag{6}$$

where $z_j = (x_{k_{j-1}}, \dots, x_{k_j - 1})$ with $k_0 := 1$ and $k_j := k_{j-1} + d_j$, $j \in [s]$.

If F is a *strictly* d-separable function, its domain can be split into s subspaces intersecting only at the origin, allowing F to be written as a sum of s functions, which are defined on the these subspaces. For simplicity, we will omit the term "strictly" in what follows. The separable structure can be exploited by a NN (see Figure 1), with two hidden layers: the first using identity activation and the second employing a smooth, non-polynomial activation function $\sigma_2 \in \mathcal{C}^\infty(\mathbb{R}, \mathbb{R})$. The second layer's sublayers can learn the functions F_j in (6).

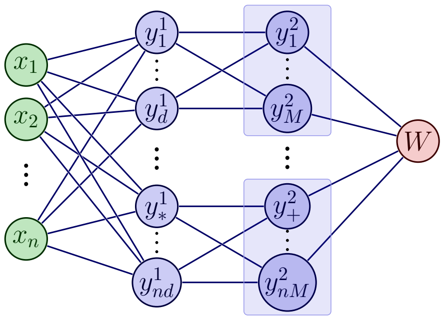


Fig. 1. Architecture of the NN with $* = (n-1)d + 1$, $+ = (n-1)M + 1$, and $W = W(x; \theta)$.

In the following, let $d, p \in \mathbb{N}$ and $r, R \in \mathbb{R}_{>0}$. Define $K_n := [-R, R]^n$ for $n \in \mathbb{N}$ and

$$\mathcal{F}_{r,d,p}^{(n)} := \Big\{ F \in W_{p,r}(K_n) \;\Big|\; F \text{ is } d\text{-separable}, F(0) = 0 \Big\}.$$

Lemma 4 *For all $F \in \mathcal{F}_{r,d,p}^{(n)}$ we can write $F = \sum_{j=1}^{s} F_j$ for some $F_j \in W_{p,r}(K_{d_j})$ with $d_j \in [d]$ for $j \in [s]$.*

PROOF. Since F is d-separable, we can write $F(x) = \sum_{j=1}^{s} \tilde{F}_j(z_j)$. As $\sum_{j=1}^{s} \tilde{F}_j(0) = 0$ by defining $F_j(z_j) := \tilde{F}_j(z_j) - F_j(0)$, we have $F_j(0) = 0$. This yields

$$F_j(z_j) = F_j(z_j) + \sum_{i \neq j} F_i(0) = F(0, \dots, 0, z_j, 0, \dots, 0). \tag{7}$$

Further, observe that for $x = (z_1, \dots, z_s) \in \mathbb{R}^n$

$$DF(x) = \Big[DF_1(z_1) \; DF_2(z_2) \; \cdots \; DF_s(z_s) \Big]. \tag{8}$$

Consequently, together with (7) it follows

$$\begin{aligned} \|F_j\|_{W_p(K_{d_j})} &= \sum_{0 \le |\alpha| \le p} \|D_\alpha F_j\|_{\infty, K_{d_j}} \\ &\le \sum_{0 \le |\alpha| \le p} \|D_\alpha F\|_{\infty, K_n} \le r. \qquad \square \end{aligned}$$

Leveraging Lemma 4, we will henceforth assume that for any $\mathcal{F}_{r,d,p}^{(n)} \ni F = \sum_{j=1}^{s} F_j$, the components F_j belong to $W_{p,r}(K_{d_j})$, without explicitly restating this assumption.

Theorem 5 *There exists a constant $\mu_d > 0$ such that for all $n \in \mathbb{N}$ and $M \in \mathbb{N}$ the NN $W(x; \theta)$ depicted in Figure 1 with $n(d + M)$ neurons satisfies that for all $F \in \mathcal{F}_{r,d,p}^{(n)}$ there exists θ_F such that*

$$\|F(\cdot) - W(\cdot; \theta_F)\|_{\infty, K_n} \le n r \mu_d M^{-\frac{p}{d}} \tilde{R},$$

where $\tilde{R} := \max\{R^p, 1\}$. Further, assume $\|\sigma''\|_{\infty, \mathbb{R}} \le C_\sigma$ for some $C_\sigma > 0$, $p \ge 2$, and let $\mathcal{F} \subset \mathcal{F}_{r,d,p}^{(n)}$ such that for each $F = \sum_{j=1}^{s} F_j \in \mathcal{F}$ the parameters θ_{F_j} satisfy that $\|w_{F_j}^{(1)}\|_\infty$ and $\|w_{F_j}^{(2)}\|_1$ are uniformly bounded by some $C_\theta > 0$ over $\mathcal{F}$ and $M \in \mathbb{N}$. Then, for each $|\alpha| = 1$

$$\|W_\alpha(\cdot; \theta_F) - F_\alpha(\cdot)\|_{\infty, K} \le \sqrt{\mu_n} M^{-\frac{p}{2d}} \sqrt{C \tilde{R}}$$

holds for all $F \in \mathcal{F}$, where $C = 2r(r + C_\theta^3 C_\sigma)$.

PROOF. We set the parameters corresponding to the first hidden layer of the network depicted in Figure 1 such that its first s sublayers contain the vectors z_j, $j \in [s]$, respectively. For the output layer we choose $w_{1,i}^3 = 1$ for $i \in [dM]$, $w_{1,i}^3 = 0$ for $i > dM$, and $b_1^3 = 0$. Observe that

$$W(x; \theta) = \sum_{j=1}^{n} \sum_{i=1}^{M} y_{(j-1)M+i}^2 = \sum_{j=1}^{s} \sum_{i=1}^{M} y_{(j-1)M+i}^2, \tag{9}$$

where for each $j \in [s]$, the output $\sum_{i=1}^{M} y_{(j-1)M+i}^2$ of the j-th sublayer can be viewed as the output of a NN with input z_j and a hidden layer of M neurons (cf. Figure 1), denoted by $W_j(z_j; \theta_j)$. Applying Theorem 2, we get

$$\|F_j(\cdot) - W_j(\cdot; \theta_{F_j})\|_{\infty, K_{d_j}} \le r \mu_d M^{-\frac{p}{d}} \tilde{R}.$$

For the resulting θ_F, by (9) we have for $x \in K_n$

$$\begin{aligned} \|F(x) - W(x; \theta_F)\| &= \Big\| \sum_{j=1}^{s} F_j(z_j) - W_j(z_j; \theta_{F_j}) \Big\| \\ &\le \sum_{j=1}^{s} \|F_j(z_j) - W_j(z_j; \theta_{F_j})\| \le s r \mu_d M^{-\frac{p}{d}} \tilde{R}. \end{aligned}$$

As $s \le n$, this shows the first claim. Since

$$DW(x;\theta) = \left[DW_1(x;\theta_1)\ \dots\ DW_s(x;\theta_s)\right],$$

together with (8), the second claim immediately follows from Theorem 2. □

In the proof of Theorem 5, the first (linear) layer of the network in Figure 1 computes the decomposition of the state x into vectors z_j, $1 \le j \le s$, as in Definition 3. This layer maps $x \mapsto W^1 x$, where $W^1 \in \mathbb{R}^{nd\times n}$ represents the decomposition. Since the weights $w^1_{k,i}$, $k \in [nd]$, $i \in [n]$, can take any real value, the first layer can express any matrix $W^1 \in \mathbb{R}^{nd\times n}$, motivating the following definition.

Definition 6 *Let $d \in [n]$, $F \in \mathcal{C}^1(\mathbb{R}^n, \mathbb{R})$, and $T \in \mathbb{R}^{n\times n}$ be invertible. Then F is called linearly d-separable with respect to T if the mapping $x \mapsto F(Tx)$ is (strictly) d-separable. Further, a function $G \in \mathcal{C}^1(\mathbb{R}^n, \mathbb{R}^l)$ is called linearly d-separable if each of its l component functions is linearly d-separable.*

The following corollary generalizes Theorem 5 to linearly d-separable functions. For $c \in \mathbb{R}_{>0}$, let GL^c_n denote the set of invertible matrices $T \in \mathbb{R}^{n\times n}$ with $\|T\|_\infty \le c$ and $\|T^{-1}\|_\infty \le c$. Note that any $T \in \mathbb{R}^{n\times n}$ with condition number $\le c^2$ can be rescaled to lie in GL^c_n. Further, define

$$\mathcal{F}^{(n)}_{r,d,p,c} := \Big\{ F \in W_{p,r}(K_n) \;\Big|\; F \text{ is linearly } d\text{-separable w.r.t. some } T \in GL^c_n, F(0) = 0 \Big\}.$$

Corollary 7 *There exists a constant $\mu_d > 0$ such that for all $n \in \mathbb{N}$ and $M \in \mathbb{N}$ the NN $W(x;\theta)$ depicted in Figure 1 with $n(d+M)$ neurons satisfies that for all $F \in \mathcal{F}^{(n)}_{r,d,p,c}$ there exists θ_F such that*

$$\|F(\cdot) - W(\cdot;\theta_F)\|_{\infty,K} \le cnr^p\mu_d \max\{\tilde{R}, 1\} M^{-\frac{p}{d}},$$

where $\tilde{R} := \max\{(cR)^p, 1\}$. Further, assume $\|\sigma''\|_{\infty,\mathbb{R}} \le C_\sigma$ for some $C_\sigma > 0$, $p \ge 2$, and let $\mathcal{F} \subset \mathcal{F}^{(n)}_{r,d,p,c}$ such that for each $F(T\cdot) = \sum_{j=1}^s F_j(\cdot) \in \mathcal{F}$ the parameters θ_{F_j} satisfy that $\|w^{(1)}_{F_j}\|_\infty$ and $\|w^{(2)}_{F_j}\|_1$ are uniformly bounded by some $C_\theta > 0$ over $\mathcal{F}$ and $M \in \mathbb{N}$. Then, for each $|\alpha| = 1$ the inequality

$$\|W_\alpha(\cdot;\theta_F) - F_\alpha(\cdot)\|_{\infty,K} \le \sqrt{\mu_d} M^{-\frac{p}{2d}} \sqrt{\widetilde{R}C}$$

holds for all $F \in \mathcal{F}$, where $C = 2c^p r(c^p r + C^3_\theta C_\sigma)$.

PROOF. Let $F \in \mathcal{F}^{(n)}_{r,d,c,p}$. Consider the mapping $G\colon T^{-1}K_n \to \mathbb{R}, x \mapsto F(Tx)$. By assumption, G is a d-separable function. Further, note that $G(0) = 0$ and $T^{-1}K_n \subset cK_n = [-cR, cR]^n$. Moreover, it holds that

$$\begin{aligned}\|G\|_{W_1(T^{-1}K_n)} &= \sum_{0\le|\alpha|\le p} \|D_\alpha F(T\cdot)\|_{\infty,T^{-1}K_n} \\ &\le c^p \sum_{0\le|\alpha|\le p}^{n} \|F_\alpha(T\cdot)\|_{\infty,T^{-1}K_n} \le c^p\|F\|_{W_p(K_n)}.\end{aligned}$$

Hence, applying Theorem 5 yields $\widetilde{\theta}_F$ such that

$$\|G(\cdot) - W(\cdot;\widetilde{\theta}_F)\|_{\infty,T^{-1}K_n} \le nc^p r\mu_d \max\{\widetilde{R}, 1\} M^{-\frac{p}{d}}, \tag{10}$$

where $W(x;\theta)$ is the NN constructed in the proof of Theorem 5. Recall that the output of the first hidden layer is $\hat{W}^1 x$ for some $\hat{W}^1 \in \mathbb{R}^{nd\times n}$. Replacing $T^{-1}K^n$ as the input space with K^n and adjusting the first hidden layer weights to $W^1 := \hat{W}^1 T^{-1}$ proves the first claim. For the second claim, assume $F \in \mathcal{F}$ and observe that for any $|\alpha| = 1$ we have

$$\begin{aligned}&\|D_\alpha W(\cdot,\theta_F) - D_\alpha F(\cdot)\|_{\infty,K_n} \\ \le&\|T^{-1}\|\|D_\alpha W(\cdot;\widetilde{\theta}_F) - D_\alpha G_\alpha(\cdot)\|_{\infty,T^{-1}K_n} \\ \le& c\sqrt{\mu_d} M^{-\frac{p}{2d}} \sqrt{\widetilde{R}C},\end{aligned}$$

where the last inequality follows from Theorem 5 applied for $G = F(T\cdot) \in \mathcal{F}^{(n)}_{c^p r,d,p}$. □

The estimate (9) in Corollary 7 can also be derived using Theorem 4.10 from Kang and Gong (2022). Representing the separable function as a compositional function and estimating the constants in (Kang and Gong, 2022, Remark 4.12) yields arguments similar to those in the proofs of Theorem 5 and Corollary 7. As a consequence of Theorem 5 and Corollary 7, by counting the total number of neurons in the hidden layers in Figure 1, we obtain that the number of hidden neurons needed to approximate (linearly) d-separable functions grows only polynomially in the state dimension n. For $\mathcal{L}_\infty$ approximation this asymptotic result has already been stated in Grüne (2021). The following Corollary extends it to the W_1-norm.

Corollary 8 *Let $\varepsilon > 0$ and consider the setting from Corollary 7. Then for $n \in \mathbb{N}$ the number of hidden neurons $N \in \mathbb{N}$ needed to ensure*

$$\sup_{F\in\mathcal{F}} \inf_\theta \|F(\cdot) - W(\cdot;\theta)\|_{W_1([-R,R]^n)} \le \varepsilon$$

is given by $N = \mathcal{O}\Big(nd + \left(\frac{n}{\varepsilon}\right)^{\frac{2d}{p}+1}\Big)$.

4 Existence of separable control Lyapunov functions

In this section, we use methods from nonlinear systems theory for providing conditions for the existence of (linearly) separable clfs. Thus, by invoking the results from Section 3 we can identify classes of systems that allow NNs to mitigate the curse of dimensionality.

4.1 Separability via small-gain theory and active nodes

This subsection proves the existence of separable clfs based on small gain theory, leveraging the notion of active nodes from Chen and Astolfi (2020, 2024). We consider a control system (1) and assume that it can be decomposed into $s \in \mathbb{N}$ subsystems denoted by

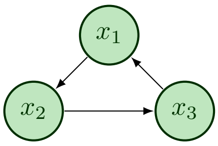
$$\Sigma_j : \quad \dot{z}_j = f_j(x, \tilde{u}_j) = f_j(z_j, z_{-j}, \tilde{u}_j), \quad j \in [s], \tag{11}$$

$$x = \begin{pmatrix} z_1 \\ \vdots \\ z_s \end{pmatrix}, \quad u = \begin{pmatrix} \tilde{u}_1 \\ \vdots \\ \tilde{u}_s \end{pmatrix}, \quad f(x,u) = \begin{pmatrix} f_1(x, \tilde{u}_1) \\ \vdots \\ f_s(x, \tilde{u}_s) \end{pmatrix},$$

with $z_j \in \mathbb{R}^{d_j}$, $U = U_1 \times \cdots \times U_s$, $\tilde{u}_j \in U_j$, $f_j : \mathbb{R}^n \times U_j \to \mathbb{R}^{d_j}$, and

$$z_{-j} := \Big(z_1, \ldots, z_{j-1}, z_{j+1}, \ldots, z_s\Big)^T \in \mathbb{R}^{n-d_j}.$$

We explicitly allow for the case that some subsystems Σ_j are independent of the control u, which corresponds to the case $U_j = \{0\}$. Such a decomposition in now represented as a directed graph that consists of s nodes. Each node belongs to one subsystem and there exists an edge from node i to node j, $j \neq i$, if the subsystem i influences the subsystem j, i.e., if the function f_j depends on the vector z_i. Figure 2 illustrates the graph corresponding to a decomposition into 1-dimensional subsystems of the control system (12) from Chen and Astolfi (2020).



$$\begin{aligned} \dot{x}_1 &= x_3 + u, \\ \dot{x}_2 &= x_1 - x_2 + x_1^2, \\ \dot{x}_3 &= x_2 - x_3. \end{aligned} \tag{12}$$

Fig. 2. A control system and its corresponding graph.

Assumption 9 *For each $j \in [s]$ there exists a feedback function $F_j : \mathbb{R}^{d_j} \to U_j$, comparison functions $\alpha_j \in \mathcal{K}_\infty$, $\gamma_{i,j} \in \mathcal{K}_\infty$, $i \neq j$, as well as a positive-definite and radially unbounded function $V_j \in \mathcal{C}^1(\mathbb{R}^{d_j}, \mathbb{R})$ such that*

$$\begin{aligned} &DV_j(z_j) f_j(z_j, z_{-j}, F_j(z_j)) \\ &\leq -\alpha_j(V_j(z_j)) + \sum_{i \neq j} \gamma_{i,j}(V_i(z_i)). \end{aligned} \tag{13}$$

Note that for a subsystem Σ_j that is not influenced by the control, the left-hand side in (13) does not depend on any feedback function F_j. In particular, Assumption 9 states that for all $j \in [s]$, the function V_j is an ISS-Lyapunov function (see Sontag and Wang (1995)) for the system $\dot{z}_j = f_j(z_j, z_{-j}, F_j(z_j))$, where z_{-j} is seen as the external input. Given such a stability assumption on each of the subsystems, small-gain theory can be used to obtain a stability property of the overall system, see, for instance, Dashkovskiy et al. (2010); Rüffer (2007). In the following, we focus on the theory developed in Chen and Astolfi (2024) that allows to formulate a graph-based criterion for the existence of a separable clf. Note that we do not impose regularity conditions on F_j in Assumption 9 since this is not necessary in order to apply the results from Chen and Astolfi (2024), whereas regularity of F_j is of course required for the existence of solutions of the control system.

Definition 10 (cf. Chen and Astolfi (2024)) *Let $j \in [s]$ and consider a subsystem Σ_j as in (11). The subsystem is called active if there exist $\bar{\alpha}_j$, $\gamma_{i,j} \in \mathcal{K}_\infty$, $i \neq j$, and a function $V_j \in \mathcal{C}^1(\mathbb{R}^{d_j}, \mathbb{R})$ such that for all $\alpha_j > \bar{\alpha}_j$ there exists $F_j : \mathbb{R}^{d_j} \to U_j$ such that (13) holds.*

Intuitively, Definition 10 implies that, for given gain functions $\gamma_{i,j}$, the rate of decrease of V_j along the direction of the vector field can be made as steep as desired by applying an appropriate feedback F_j. Using this notion of active subsystems (or active nodes), the results of Chen and Astolfi (2024) yield the following proposition.

Proposition 11 *Consider a control system of the form (1) given through subsystems of the form (11) and let Assumption 9 hold. Moreover, assume that in each cycle of the directed graph corresponding to the decomposition (11) there is at least one active subsystem. Then there exists a d-separable clf for the system (1).*

PROOF. Let V_j, $j \in [s]$, denote the ISS-Lyapunov functions obtained from Assumption 9. Applying Theorem 4 and Theorem 5 in Chen and Astolfi (2024), respectively, yields the existence of continuous, positive definite functions $\lambda_j : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$, $j \in [s]$, such that

$$V(\cdot) := \sum_{j=0}^{s} \int_0^{V_j(\cdot)} \lambda_j(s)\, ds$$

is a Lyapunov function for

$$\dot{z}_j = f_j(z_j, z_{-j}, F_j(z_j)), \quad j \in [s].$$

This implies that V satisfies condition (2b), whence V is a clf for (1). This gives us the decomposition of V as d-separable function as in Definition 3. $\square$

Revisiting the control system in (12), we can check that $V_j(x_j) = x_j^2$ is an ISS-Lyapunov function for each subsystem and that the first subsystem is active. Thus, Proposition 11 yields the existence of a 1-separable clf for Example 12. Overall, by invoking Corollary 8 we can conclude that Proposition 11 identifies a class of control systems, where a clf can be approximated by a NN without the curse of dimensionality.

4.2 Linear separability via linearization

This subsection extends the discussion from Subsection 4.1 to the existence of linearly d-separable clfs. This is motivated by the system

$$\dot{x}_1 = x_3 + u, \quad \dot{x}_2 = x_1 - x_2 + x_2^2, \quad \dot{x}_3 = x_2 + x_3, \quad (14)$$

which is a variation of (12) for which it has been shown in Grüne and Sperl (2023) that it has a clf, but no 1-separable clf. However, it can be shown to possess a linearly 1-separable clf in a neighborhood around the origin. We first prove that stabilizable linear systems always admit a linearly 1-separable clf.

Proposition 12 *Consider a linear control system of the form*

$$\dot{x} = Ax + Bu, \quad (15)$$

where $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times m}$. Assume that (A, B) is stabilizable. Then there exists a linearly 1-separable clf V for the system (15). The function V is quadratic, i.e., $V(x) = x^T Px$ for some $P \in \mathbb{R}^{n\times n}$ and satisfies

$$\inf_{u\in\mathbb{R}^m} DV(x)(Ax + Bu) \le DV(x)(A + BF)x \le -c\|x\|_2^2 \quad (16)$$

for a suitable feedback matrix $F \in \mathbb{R}^{n\times m}$ and some $c > 0$.

PROOF. It is known that for a linear and stabilizable system, there always exists a clf of the form $V(x) = x^T Px$ for a suitable symmetric and positive definite matrix $P \in \mathbb{R}^{n\times n}$. The inequality (16) then follows from the usual matrix Lyapunov inequality. As P is symmetric and positive-definite, there exists an orthogonal matrix $T \in \mathbb{R}^{n\times n}$ such that

$$\widetilde{P} := T^{-1}PT = T^T PT = \mathrm{diag}(p_1, \dots, p_n)$$

is a diagonal matrix. Thus,

$$V(Tx) = (Tx)^T P(Tx) = x^T \widetilde{P} x = \sum_{i=1}^{n} p_i x_i^2$$

is a 1-separable function. □

Proposition 12 implies that linearizable control systems locally possess linearly 1-separable clfs.

Corollary 13 *Consider a control system (1) with a $\mathcal{C}^1$-function f and assume that its linearization at the origin is stabilizable. Then the control system (1) possesses a linearly 1-separable clf on some neighborhood of the origin.*

PROOF. Write $f(x, u) = Ax + Bu + g(x, u)$ with

$$\lim_{\|(x,u)\|\to 0} \frac{\|g(x, u)\|}{\|(x, u)\|} = 0.$$

Since (A, B) is stabilizable, Proposition 12 yields the existence of $c \in \mathbb{R}_{\ge 0}$, $F \in \mathbb{R}^{n\times m}$, and a linearly 1-separable function $V(x) = x^T Px$ such that (16) holds. Following the proof of (Sontag, 1998, Theorem 19), we obtain

$$\inf_u DV(x)f(x, u) \le -c\|x\|_2^2 + 2xPg(x, F(x)) < 0$$

for x sufficiently small, since $\frac{\|2xPg(x,F(x))\|}{\|x\|^2} \to 0$ for $x \to 0$. Hence, V is a clf for the nonlinear system (1) in a suitable neighborhood of the origin. □

4.3 Linear separability via feedback linearization

Next we explore a class of systems for which Proposition 12 can be employed to achieve linear separability through a potential nonlinear transformation. To this end, we extend the definition of feedback linearizability from (Sontag, 1998, Section 5.3) to multi-input systems.

Definition 14 *An affine control system*

$$\dot{x} = f(x) + \sum_{j=1}^{m} g_j(x)u_j$$

with control input $u = (u_1, \dots, u_m)^T \in \mathbb{R}^m$ is called feedback linearizable, if there exists a diffeomorphism $S \in \mathcal{C}^1(\mathbb{R}^n, \mathbb{R}^n)$ as well as maps $a_j, b_j \colon \mathbb{R}^n \to \mathbb{R}$, $j \in [m]$, such that the transformed control system

$$\dot{\tilde{x}} = \tilde{f}(\tilde{x}) + \sum_{j=1}^{m} \tilde{g}_j(\tilde{x})v_j$$

with transformed state $\tilde{x} = S(x)$, new control input $v = (v_1, \dots, v_m)^T \in \mathbb{R}^m$ and

$$\tilde{f}(\tilde{x}) = DS(x)\Big(f(x) + \sum_{j=1}^{m} a_j(x)g_j(x)\Big),$$
$$\tilde{g}_j(\tilde{x}) = b_j(x)DS(x)g_j(x),$$

is a linear control system, i.e., if there exist matrices $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times m}$ such that $\tilde{f}(\tilde{x}) = A\tilde{x}$ and $(\tilde{g}_1(\tilde{x}), \dots, \tilde{g}_m(\tilde{x})) = B$ holds for all $\tilde{x} \in \mathbb{R}^n$.

Theorem 15 *Consider a feedback linearizable affine control system with transformation S satisfying $S(0) = 0$, for which the pair (A, B) is stabilizable. Then the control system has a clf V of the form $V(x) = \widetilde{V}(S(x))$ with a linearly 1-separable function $\widetilde{V} : \mathbb{R}^n \to \mathbb{R}$.*

PROOF. According to Proposition 12, we have

$$\inf_{v\in\mathbb{R}^m} D\widetilde{V}(\tilde{x})(A\tilde{x}+Bv) \le D\widetilde{V}(\tilde{x})(A\tilde{x}+BF\tilde{x}) \le -c\|\tilde{x}\|_2^2$$

for suitable $c \in \mathbb{R}_{\ge 0}$, $F \in \mathbb{R}^{n\times m}$, and some linearly 1-separable mapping $\widetilde{V}$. For $V(x) = \widetilde{V}(S(x))$ and $u_j = a_j(x) + b_j(x)v_j$ we then obtain

$$\begin{aligned}
&DV(x)\big(f(x) + \sum_{j=1}^m g_j(x)u_j\big)\\
&= D\widetilde{V}(S(x))DS(x)\big(f(x) + \sum_{j=1}^m g_j(x)(a_j(x) + b_j(x)v_j)\big)\\
&= D\widetilde{V}(\tilde{x})\big(\tilde{f}(\tilde{x}) + \sum_{j=1}^m \tilde{g}_j(\tilde{x})v_j\big) \;=\; D\widetilde{V}(\tilde{x})(A\tilde{x} + Bv).
\end{aligned}$$

This implies

$$\inf_{u\in\mathbb{R}^m} DV(x)\Big(f(x) + \sum_{j=1}^m g_j(x)u_j\Big) \le -c\|S(x)\|_2^2.$$

Since S is a diffeomorphism with $S(0) = 0$, there exist $\alpha_1, \alpha_2 \in \mathcal{K}_\infty$ with $\alpha_1(\|x\|_2) \le (\|S(x)\|_2) \le \alpha_2(\|x\|_2)$, see Lemma 1 in Kellett and Dower (2015). Thus, V satisfies all inequalities in (2), whence it is a clf. □

Corollary 16 *Consider the setting of Theorem 15 and assume that the transformation map S is linearly k-separable for some $k \in [n]$. Then the control system has a clf V that is a composition of a linearly 1-separable function with a linearly k-separable function.*

Note that Corollary 16 in particular applies to linear mappings S, as linear mappings are always 1-separable. We can conclude that in the setting of Corollary 16 there exists a curse-of-dimensionality-free approximation with a NN that is built as in Figure 1, but has one additional hidden layer at the beginning, which is used to represent the k-separable transformation S.

5 Numerical illustration

5.1 Network structure and training algorithm

The structure of the NN that we use for the computation of a linearly separable clf is exactly the one depicted in Figure 1 with the modification of introducing a hyperparameter s for the number of sublayers, i.e., replacing the n sublayers in Figure 1 by s sublayers. An important feature of this network architecture is the fact that the decomposition of the state vector x into the vectors z_j, $1 \le j \le s$, is determined by the first hidden layer. Thus, the detection of a suitable splitting of the state space (see Definition 3) is part of the training process. This means that the numerical algorithm presented in this section does not need to know the splitting or coordinate transformation discussed in Section 4. Rather, this structure will be "learned" by the network in the training process.

It is possible to incorporate the linear transformation computed by the first hidden layer in Figure 1 into the second hidden layer, that is, to merge the two hidden layers into one hidden fully-connected layer. Since the NN in Figure 1 can be viewed as a fully-connected NN with some particular weights set to 0, a fully connected NN still preserves the property of mitigating the curse of dimensionality for separable clfs. However, in our numerical test cases, the NN with two hidden layers as depicted in Figure 1 frequently demonstrated an improved numerical performance. On the other hand, if no a priori estimates of the hyperparameters d and s are possible, the usage of a fully connected NN is more practical. A detailed comparison of these NN architectures, including different numbers of hidden layers, is of high importance but is deferred to future research due to space limitations.

We define a loss-function L that penalizes the violation of the three inequalities defining a smooth clf in Definition 1. For any point $x \in K$ we set

$$\begin{aligned}
&L(x, W(x;\theta), DW(x;\theta)) :=\\
&\quad ([W(x;\theta) - \alpha_1(\|x\|)]_-)^2 + ([W(x;\theta) - \alpha_2(\|x\|)]_+)^2\\
&\quad + \eta\Big(\Big[\alpha_3(\|x\|) + \inf_{u\in U} DW(x;\theta)f(x,u)\Big]_+\Big)^2, \qquad (17)
\end{aligned}$$

where $\alpha_1, \alpha_2 \in \mathcal{K}_\infty$, $\alpha_3 \in \mathcal{K}$, $[\cdot]_+ := \max(\cdot, 0)$, $[\cdot]_- := \min(\cdot, 0)$, and $\eta > 0$ is a weighting factor. Note that the functions α_1, α_2, and α_3, as well as the parameter η are hyperparameters of the algorithm. Their choice can significantly affect the training process, whence a system approach for selecting these hyperparameters is important and will be investigated in future research.

Note that L depends on the point x, the evaluation $W(x;\theta)$ and the orbital derivative $DW(x;\theta)f(x,u)$. We calculate this orbital derivative alongside the evaluation of $W(x;\theta)$ via automatic differentiation. This means that the orbital derivative is computed on the fly from the separable network, using the built-in differentiation via backpropagation in Tensorflow (see Abadi et al. (2015)). Thus, the derivative does not need to be stored separately, whence separability of the orbital derivative, which cannot be expected since f is not separable, is not needed. Moreover, we need to evaluate the expression $\inf_{u\in U} DW(x;\theta)f(x,u)$. This expression can be simplified for systems with $U=[-C,C]^m$ for some $C>0$ and an affine linear control input of the form $\dot{x}=f(x,u)=h(x)+g(x)u$, since then we have

$$\begin{aligned}&\inf_{u\in U} DW(x;\theta)f(x,u)\\ =&DW(x;\theta)h(x)-C\|DW(x;\theta)g(x)\|_1,\end{aligned} \tag{18}$$

cf. (Grüne and Sperl, 2023, Lemma 6). The training process of the NN is then performed by minimizing the value of the loss function (17) over a finite set of training data $\mathcal{D}_T\subset K$.

Remark 17 *Clfs can be characterized as solutions of Zubov's equation Camilli et al. (2008); Grüne and Wirth (2000), i.e., as optimal value functions for suitable optimal control problems. For such problems, NN approaches have been proposed in the literature, see e.g. Albi et al. (2022); Liu et al. (2023); Nakamura-Zimmerer et al. (2022); Zhou et al. (2024). However, they are difficult to apply in our setting, because while we assume that a separable clf exists, we do not know its precise form and thus also not the corresponding optimal control problem.*

Furthermore, in our numerical tests it has turned out that the most significant error usually lies around the origin. We tackle this by adding the term $W(0;\theta)^2+\|DW(0;\theta)\|^2$ to the loss-function used for the training of the network, cf. Chang et al. (2019). Adding these terms to the loss function encourages the used optimization routine to stay at $W(0;\theta)=0$ and $DW(0;\theta)=0$ during the training. While this approach produced the best results for us, different ways to address issues at the origin have successfully been implemented in the literature, for instance by transforming the NN output, cf. Gaby et al. (2022); Mukherjee et al. (2022).

5.2 Numerical test case

Finally, we illustrate the presented algorithm on the following 10-dimensional control system

$$\dot{x}=f(x,u)=\begin{pmatrix}-x_1+x_1x_2-0.1x_9^2\\ -x_2u_1\\ -x_3+x_3x_4-0.1x_1^2\\ -x_4u_2\\ -x_5+x_5x_6+0.1x_7^2\\ -x_6u_3\\ -x_7+x_7x_8\\ -x_8u_4\\ -x_9+x_9x_{10}\\ -x_{10}u_5+0.1x_2^2\end{pmatrix} \tag{19}$$

with $U=[-1,1]^5$. It consists of 5 two-dimensional bilinear subsystems of the form $\dot{y}=-y+zy, \dot{z}=-uz$ coupled with small non-linearities. For $u=1$ this recovers the ODE presented in Ahmadi et al. (2011), where it is shown that there does not exist a polynomial Lyapunov function for this system on $\mathbb{R}^2$. While there still exists a quadratic clf with appropriate coefficients on compact sets, enlarging the training domain makes it more difficult to recover it. This can cause the NN to defer from a quadratic influence of the variables, cf. Figure 3. To illustrate the ability of our approach to determine subspaces that lead to separability, we consider the transformed system $\dot{x}=T^{-1}f(Tx,u)$, where $T=I_{10}+P\in\mathbb{R}^{10\times 10}$ with P being normally distributed around 0 with scale 0.1. Note that the subsystem that are computed during the training process are typically not the original subsystems from (19). We employed the hyperparameters $\alpha_1(r)=0.5r^2, \alpha_2(r)=10r^2, \alpha_3(r)=0.01r^2$, as well as $d=2$, $s=5$, and $M=64$ in a training process with 2×10^5 training data, a batch size of 64, and the softplus-function as activation function in the second hidden layer. The training process was conducted to compute a clf on the domain $[-4,4]^{10}$, where we beforehand transformed $x\mapsto \frac{1}{4}x$ and performed the training on $[-1,1]^{10}$ for numerical reasons. Our computations are carried out with Python 3.10.6 and Tensorflow 2.11.0 on an NVIDIA GeForce RTX 3070 GPU. The optimization has been performed with the ADAM stochastic gradient descent method. After 30 epochs and a training time of 470 seconds, the algorithm reached an $\mathcal{L}_1$ error of 9.8×10^{-5} in the training data. An evaluation at independently chosen 2×10^5 validation data showed an $\mathcal{L}_1$ error of 9.3×10^{-7} and an $\mathcal{L}_\infty$ error of 9.3×10^{-2}.

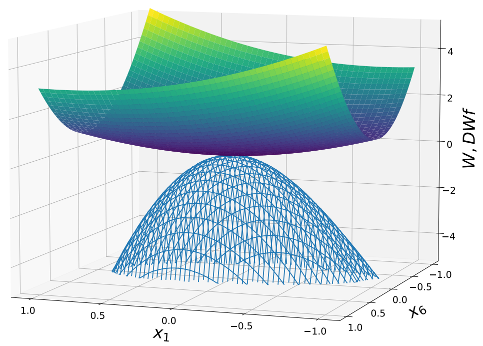


Fig. 3. Approximate clf (solid) and its corresponding orbital derivative (mesh) on the (x_1,x_3)-plane.

Figure 3 shows the computed NN output $W(x;\theta)$ projected onto the $(x_1,x_6)-$axis as surface plot. Further, the directional derivative $DW(x;\theta)f(x,u^*)$ with

$$u^*(x)=\arg\min DW(x;\theta)f(x,u) \tag{20}$$

is calculated according to (18) and depicted as wireframe plot. Figure 4 depicts the evaluation of $W(x;\theta)$ alongside 20 trajectories with initial values randomly sampled in $[-0.5,0.5]^{10}$ and control $u^*(x)$ as in (20). Note that

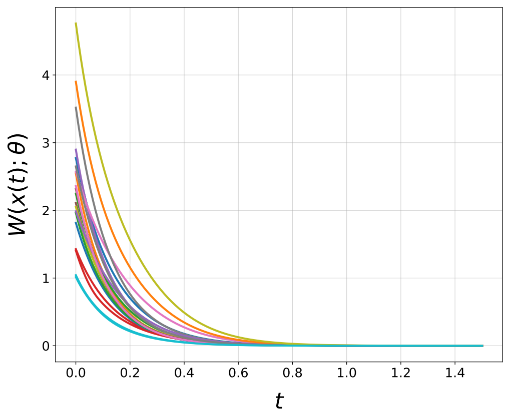


Fig. 4. Evaluation of $W(x;\theta)$ along trajectories.

the convergence of the trajectories in Figure 4 towards 0 as well as the plots in Figure 3 provide empirical evidence that the computed NN output might indeed by a clf. However, there is no formal guarantee that the Lyapunov conditions are met at every point. For verification techniques, we refer to the corresponding discussion in the introduction.

Further, we trained neural networks towards clfs for the control system (19), adjusted to dimensions $n = 2, 4, \ldots, 16$. For each dimension, we utilized a network architecture as shown in Figure 4 with $n/2$ sublayers. The sublayer size M was decreased as far as possible while still achieving an L_1 loss below a tolerance of 10^{-4} on both the training and independently chosen validation data. The resulting total number of neurons in the network and its number of trainable parameters in dependence of the dimension are shown in Figure 5. Notably, the growth in both the number of neurons and trainable parameters is non-exponential and appears almost linear. This trend arises because the minimal sublayer size M remains in $\{2, 3, 4, 5\}$ across all dimensions. This indicates that the bounds derived in Section 3 are conservative for this example, as expected, since the theoretical error bounds consider the worst-case scenario where each subsystem simultaneously attains its maximal error for the same input x. Our Tensorflow code is available on `https://github.com/MarioSperl/SeparableCLF-NN`.

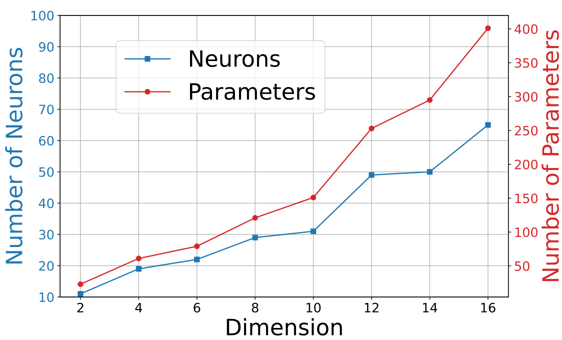


Fig. 5. Scaling of neurons and parameters with dimension.

6 Conclusion

In this paper, we have discussed the capability of NNs to approximate clfs in high space dimensions. To this end, we have shown that NNs can mitigate the curse of dimensionality for approximating (linearly) separable functions and provided conditions for the existence of (linearly) separable clfs. Thus, we have identified control systems that allow for a representation of a clf with a NN mitigating the curse of dimensionality. Moreover, a numerical algorithm was presented. For future research, we intend to systematically study the influence of the hyperparameters determining the NN architecture and the loss function. Afterwards, a comparison to other numerical methods is of interest, as it was for example done in Zhou et al. (2024). Moreover, we aim to investigate the approximation of non-smooth clfs with NNs.

Acknowledgements

The authors acknowledge funding by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) in the frame of the priority programme SPP 2298 “Theoretical Foundations of Deep Learning” - project number 463912816.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., et al., 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Ahmadi, A., Krstic, M., Parrilo, P., 2011. A globally asymptotically stable polynomial vector field with no polynomial Lyapunov function, in: 2011 IEEE 50th Conference on Decision and Control (CDC), IEEE. pp. 7579–7580.

Albi, G., Bicego, S., Kalise, D., 2022. Gradient-augmented supervised learning of optimal feedback laws using state-dependent Riccati equations. IEEE Control Syst. Lett. 6, 836–841.

Beck, C., Jentzen, A., Kleinberg, K., Kruse, T., 2023. Nonlinear monte carlo methods with polynomial runtime for Bellman equations of discrete time high-dimensional stochastic optimal control problems. preprint arXiv:2303.03390 .

Camilli, F., Grüne, L., Wirth, F., 2008. Control Lyapunov functions and Zubov's method. SIAM J. Control Optim. 47, 301–326.

Chang, Y., Roohi, N., Gao, S., 2019. Neural Lyapunov control, in: Advances in Neural Information Processing Systems, Curran Associates, Inc.

Chen, K., Astolfi, A., 2020. On the active nodes of network systems, in: 2020 59th IEEE Conference on Decision and Control (CDC), IEEE. pp. 5561–5566.

Chen, K., Astolfi, A., 2024. Active nodes of network systems with sum-type dissipation inequalities. IEEE Trans. Automat. Control 69, 3896–3911.

Cybenko, G., 1989. Approximation by superpositions of a sigmoidal function. Math. Control Signals Systems 2, 303–314.

Dahmen, W., 2023. Compositional sparsity, approximation classes, and parametric transport equations. preprint arXiv: 2207.06128 .

Dai, H., Landry, B., Gao, S., Yang, L., Pavone, M., Tedrake, R., 2021. Lyapunov-stable neural-network control, in: Proceedings of Robotics: Science and Systems.

Darbon, J., Langlois, G., Meng, T., 2020. Overcoming the curse of dimensionality for some Hamilton-Jacobi partial differential equations via neural network architectures. Res. Math. Sci. 7.

Dashkovskiy, S., Rüffer, B., Wirth, F., 2010. Small gain theorems for large scale systems and construction of ISS Lyapunov functions. SIAM J. Control Optim. 48, 4089–4118.

Gaby, N., Zhang, F., Ye, X., 2022. Lyapunov-net: A deep neural network architecture for Lyapunov function approximation, in: 2022 IEEE 61st Conference on Decision and Control (CDC), IEEE. pp. 2091–2096.

Gonon, L., Schwab, C., 2023. Deep ReLU neural networks overcome the curse of dimensionality for partial integrodifferential equations. Anal. Appl. 21, 1–47.

Grüne, L., 2021. Computing Lyapunov functions using deep neural networks. J. Comput. Dyn. 8, 131–152.

Grüne, L., Sperl, M., 2023. Examples for separable control Lyapunov functions and their neural network approximation. IFAC-PapersOnLine 56, 19–24.

Grüne, L., Wirth, F., 2000. Computing control Lyapunov functions via a Zubov type algorithm, in: Proceedings of the 39th IEEE Conference on Decision and Control, IEEE. pp. 2129–2134.

Kang, W., Gong, Q., 2022. Feedforward neural networks and compositional functions with applications to dynamical systems. SIAM J. Control Optim. 60, 786–813.

Kellett, C.M., Dower, P.M., 2015. Input-to-state stability, integral input-to-state stability, and $\mathcal{L}_2$-gain properties: Qualitative equivalences and interconnected systems. IEEE Trans. Automat. Control 61, 3–17.

Khansari-Zadeh, S., Billard, A., 2014. Learning control Lyapunov function to ensure stability of dynamical system-based robot reaching motions. Robotics and Autonomous Systems 62, 752–765.

Liu, J., Meng, Y., Fitzsimmons, M., Zhou, R., 2023. Physics-informed neural network Lyapunov functions: PDE characterization, learning, and verification.

Liu, J., Meng, Y., Fitzsimmons, M., Zhou, R., 2024. Compositionally verifiable vector neural Lyapunov functions for stability analysis of interconnected nonlinear systems. preprint arXiv:2403.10007 .

Long, Y., Bayoumi, M., 1993. Feedback stabilization: Control Lyapunov functions modelled by neural networks, in: Proceedings of 32nd IEEE Conference on Decision and Control, IEEE. pp. 2812–2814.

Mhaskar, H., 1996. Neural networks for optimal approximation of smooth and analytic functions. Neural Comput. 8, 164–177.

Mukherjee, S., Drgoňa, J., Tuor, A., Halappanavar, M., Vrabie, D., 2022. Neural Lyapunov differentiable predictive control, in: 2022 IEEE 61st Conference on Decision and Control (CDC), IEEE. pp. 2097–2104.

Nakamura-Zimmerer, T., Gong, Q., Kang, W., 2022. Neural network optimal feedback control with guaranteed local stability. IEEE Open Journal of Control Systems 1, 210–222.

Poggio, T., Mhaskar, H., Rosasco, L., Miranda, B., Liao, Q., 2017. Why and when can deep-but not shallow-networks avoid the curse of dimensionality: A review. International Journal of Automation and Computing 14, 503–519.

Rüffer, B.S., 2007. Monotone dynamical systems, graphs, and stability of large-scale interconnected systems. Ph.D. thesis. Universität Bremen.

Sontag, E.D., 1983. A Lyapunov-like characterization of asymptotic controllability. SIAM J. Control Optim. 21, 462–471.

Sontag, E.D., 1991. Feedback stabilization using two-hidden-layer nets, in: 1991 American control conference, IEEE. pp. 815–820.

Sontag, E.D., 1998. Mathematical Control Theory. Springer New York.

Sontag, E.D., Wang, Y., 1995. On characterizations of the input-to-state stability property. Systems Control Lett. 24, 351–359.

Zhou, R., Fitzsimmons, M., Meng, Y., Liu, J., 2024. Physics-informed extreme learning machine lyapunov functions. IEEE Control Systems Letters 8, 1763–1768.