2024 European Control Conference (ECC)
June 25-28, 2024. Stockholm, Sweden

A physics-informed neural network method to approximate homogeneous Lyapunov functions

Danilo R. Lima[1], Rosane Ushirobira[1], Denis Efimov[1]

Abstract— **This paper applies the physics-informed neural network approach to approximate globally-defined Lyapunov functions and stabilizing controls for homogeneous dynamical systems. The advantage of this class of systems is that all analysis and design can be made locally on a suitably defined unit sphere, which corresponds perfectly to the applicability conditions of neural networks.**

I. Introduction

Control theory and engineering heavily rely on the notion of stability of nonlinear dynamic systems. The literature contains various techniques to solve this issue, primarily based on the Lyapunov theory. The existence of a Lyapunov function is a necessary and sufficient condition for a large class of dynamical systems to be asymptotically stable. As no universal technique exists for designing such a function, determining a Lyapunov function candidate is not straightforward. The vast bulk of known procedures involves methods to either analytically design a Lyapunov function or attempt to build numeric approximations. Analytical methods are usually constrained to particular canonical forms of the system's model, while most currently used numeric techniques perform well in systems with a small number of state variables. However, these methods have a high computational cost when used in large-scale systems like recurrent neural networks, which might have numerous state variables.

Artificial neural networks (ANNs) became a trendy tool to deal with Lyapunov functions. A classical result, the universal approximation theorem, is of great help in this issue. According to this theorem, a neural network with at least one hidden layer may arbitrarily approximate any smooth function. The set of functions that neural networks represent with one hidden layer is then dense in the set of continuous functions [2], [9]. A stronger result is proved in these latter, namely that every continuous function with support in a compact set may be approximated with any desired degree of accuracy using standard multilayer feedforward networks with sufficient hidden units and any number of hidden layers. In [13], a direct approach for constructing a Lyapunov function modeled by a neural network is proposed, with the weights calculated by a genetic algorithm. For a training procedure to calculate approximations of Lyapunov functions for nonlinear ordinary differential equations, the author in [8] proposes a deep neural network architecture. If the systems admit a compositional Lyapunov function, it is shown that the number of neurons required to estimate a Lyapunov function with a fixed accuracy increases only polynomially in the state dimension. A control Lyapunov function is constructed in [12], using a neural network, which renders the closed-loop system stable after training. The authors in [10] provide the LyaNet technique for training ODEs that can give adequate prediction performance, quicker inference dynamics convergence, and more significant adversarial robustness. In [15], a deep neural network architecture is used to determine a numerical control Lyapunov function for nonlinear dynamics and an estimate of the region of the attraction, guaranteeing its stability.

The study in [3] describes a numerical method for solving partial differential equations with application neural network-based functions, employing an unconstrained minimization problem. However, it was in [14] that the Physics Informed Neural Network (PINN) method was first described. The authors utilized it for training a fully connected ANN endowed with a loss function that encodes physics laws represented by nonlinear PDEs. The paper [1] proposes PINN approaches combined with a coupled-automatic-numerical differentiation framework to enable robust and efficient training. Numerous other studies have attempted to apply, extend, or provide theoretical support for this approach. In particular, PINN was used in [8] to approximate a Lyapunov function, looking for small-gain theory applications in a large-scale system, where it was also observed that the approach could be applied only for local stability analysis, in the general case, and that there is no guarantee in a neighborhood of the origin.

An important problem, rarely discussed in the litera- ture, is the principal existence of a Lyapunov function with the chosen structure and regularity for the studied class of systems. According to the remarkable results in [7], under an appropriate nonlinear change of variables, global asymptotic stability (GAS) of the origin for a family of ordinary differential equations is equivalent to global exponential stability (GES) with a quadratic Lyapunov function. The change of variables can then be incorporated into an approximation of a Lyapunov function in the original coordinates. An important point is a sort of relaxation by taking the change of variables diffeomorphism only smooth out of the origin. For an arbitrary nonlinear system, this characteristic can be an insurmountable obstacle for a global approximation. Nevertheless, this obstruction does not hold for homogeneous systems, for which a Lyapunov function can also be chosen as homogeneous. So, here, we aim at approximating a Lyapunov function on a compact set, not including the origin, and then obtain a global expression by dilations.

In this note, following [7], [8], we first discuss the pos-

[1] D. R. Lima, R. Ushirobira, and D. Efimov are with Inria, Univ. Lille, CNRS, UMR 9189 - CRIStAL, F-59000 Lille, France. {danilo.rodrigues-de-lima,rosane.ushirobira, denis.efimov}@inria.fr.

sibilities of approximating a Lyapunov function by ANNs, showing the natural obstacles appearing close to the origin. Next, we aim to design a PINN that can efficiently store an approximation of a global Lyapunov function for homogeneous systems. Furthermore, this result is extended to approximate a stabilizing control algorithm simultaneously.

Notation

- Denote by $\mathbb{R}$ and $\mathbb{R}_+$ the sets of real and positive real numbers, respectively. Also, $\|\cdot\|$ is the Euclidean norm of a vector in $\mathbb{R}^n$.
- A continuous function $\alpha : \mathbb{R}_+ \to \mathbb{R}_+$ belongs to the class $\mathcal{K}$ if $\alpha(0) = 0$ and it is strictly increasing; α belongs to the class $\mathcal{K}_\infty$ if $\alpha \in \mathcal{K}$ and it is increasing to infinity. A continuous function $\beta : \mathbb{R}_+ \times \mathbb{R}_+ \to \mathbb{R}_+$ belongs to the class $\mathcal{KL}$ if $\beta(\cdot, t) \in \mathcal{K}$ for each fixed $t \in \mathbb{R}_+$ and $\beta(s, \cdot)$ is decreasing to zero for each fixed $s > 0$.
- The notation $DV(x)f(x)$ stands for the directional derivative of a continuously differentiable function $V : \mathbb{R}^n \to \mathbb{R}$ with respect to a vector field $f : \mathbb{R}^n \to \mathbb{R}^n$ evaluated at the point x (then $DV(x)$ is the gradient).
- For a set $\mathcal{A} \subset \mathbb{R}^n$, denote its boundary and interior by $\partial\mathcal{A}$ and $\mathrm{int}(\mathcal{A})$, respectively.
- The finite set of the first n positive integers is denoted by $\overline{1,n} := \{1, 2, ..., n\}$.

II. Preliminaries

Consider a continuous-time system:

$$\dot{x}(t) = f(x(t)),\ \forall t \in \mathbb{R}_+, \tag{1}$$

where $x : \mathbb{R}_+ \to \mathbb{R}^n$ represents the state function and $f : \mathbb{R}^n \to \mathbb{R}^n$ is a continuous vector field, locally Lipschitz continuous on $\mathbb{R}^n \setminus \{0\}$, with $f(0) = 0$. We suppose that the system (1) with initial condition $x_0 \in \mathbb{R}^n$ has a solution $x(t, x_0)$ defined for all $t \in \mathbb{R}_+$ (the system is forward complete, see [11] for basic definitions of properties of dynamical systems). In this note, we will sometimes omit the time dependency and write simply x. So, by $x \in \mathbb{R}^n$, we mean $x(t) \in \mathbb{R}^n$, for $t \in \mathbb{R}_+$.

A. Stability notions: GAS to GES, and the existence of Lyapunov functions

We refer to [7] for the concepts and results in this subsection. Let $\mathcal{O} \subset \mathbb{R}^n$ be an open set in the domain of attraction of the origin for (1).

Definition 1: A *Lyapunov function* on $\mathcal{O}$ is a continuously differentiable (C^1) function $V : \mathcal{O} \to \mathbb{R}$ such that $V(0) = 0$, $V(x) > 0$ for all $x \in \mathcal{O} \setminus \{0\}$, and the derivative of V in the direction of the vector field f satisfies $DV(x)f(x) = -g(x) \le 0$, for some function $g : \mathbb{R}^n \to \mathbb{R}_+$, with $g(x) > 0$ for all $x \in \mathcal{O} \setminus \{0\}$. In the case $\mathcal{O} = \mathbb{R}^n$ and $V(x) \to +\infty$ when $x \to +\infty$, we call V a *global Lyapunov function.*

The existence of a (global) Lyapunov function for (1) is a necessary and sufficient condition for $0 \in \mathbb{R}^n$ to be a (globally) asymptotically stable equilibrium, see for instance [11].

Using $\mathcal{KL}$ functions, we may also write that $x = 0$ is *globally asymptotically stable* if there exists a class $\mathcal{KL}$-function β such that $\|x(t, x_0)\| \le \beta(\|x_0\|, t)$, for all $t \in \mathbb{R}_+$ and all $x_0 \in \mathbb{R}^n$. Also, $x = 0$ is *globally exponentially stable*, if there exist $c \ge 1$ and $\alpha > 0$ such that

$$\|x(t, x_0)\| \le c\,\mathrm{e}^{-\alpha t}\,\|x_0\|, \tag{2}$$

for all $t \in \mathbb{R}_+$ and all $x_0 \in \mathbb{R}^n$.

A significant result in [7], Theorem 2, claims that the system (1), supposed to be GAS, can be transformed into a GES system:

Theorem 1 ([7]): Let $n \neq 4$,5 and consider any system (1) on $\mathbb{R}^n$, which is GAS. Then (1) can be transformed into a system that is GES. In particular, in (2), one can choose $c = 1$ and $\alpha = 1$.

The proposed transformation is realized through what the authors in [7] call a *change of variables*, that is, a homeomorphism $T : \mathbb{R}^n \to \mathbb{R}^n$ satisfying $T(0) = 0$, T C^1 on $\mathbb{R}^n$, and a diffeomorphism on $\mathbb{R}^n \setminus \{0\}$. These transformations are infinitely differentiable everywhere, except at the origin, where they are just continuously differentiable. Their respective inverses are continuous globally and infinitely differentiable away from the origin. In the new coordinates, for $y = T(x)$, the dynamics of (1) takes the form:

$$\dot{y} = DT(x)f(x) = DT\left(T^{-1}(y)\right) f\left(T^{-1}(y)\right) =: \widetilde{f}(y).$$

Remark that the change of variables can be seen as an approximation of a Lyapunov function, as suggested by the following proposition ([7], Proposition 1):

Proposition 1: For $n \neq 5$, let $V : \mathbb{R}^n \to \mathbb{R}$ be a proper (*i.e.*, for each $\beta \ge 0$, the set $\{x \in \mathbb{R}^n \mid V(x) \le \beta\}$ is bounded), positive definite, C^1 global Lyapunov function of (1). Assume that V is smooth on $\mathbb{R}^n \setminus \{0\}$. Then there exists a change of variables $T : \mathbb{R}^n \to \mathbb{R}^n$ with $DT(0) = 0$ such that $\widetilde{V}(y) = V\left(T^{-1}(y)\right) = \|y\|^2$ is the transformed Lyapunov function with $D\widetilde{V}(y)\widetilde{f}(y) = -\|y\|^2$ for all $y \in \mathbb{R}^n$.

Remark 1: As mentioned in [7], the main ingredient in the proof of the above proposition is related to the question of whether level sets of Lyapunov functions in $\mathbb{R}^n$ are diffeomorphic to the unit sphere in $\mathbb{R}^{n-1}$, based on results from F. W. Wilson [18]. In the case of homeomorphisms only, the case $n = 4$ results from the Poincaré conjecture, see [18], solved in 2003 by G. Perelman, and M. H. Freedman [6] proved the remaining case $n = 5$. However, the question of the existence of a diffeomorphism (as in the definition of change of variables) remains open.

Therefore, in the original coordinates, we have the expressions:

$$V(x) = \|T(x)\|^2 \quad \text{and} \quad DV(x)f(x) = -\|T(x)\|^2 \tag{3}$$

for all $x \in \mathbb{R}^n$.

B. Homogeneity

1) Definitions of homogeneity: Basic notions on homogeneous systems useful in the sequel are presented in this subsection. For any $r_i > 0$, $i \in \overline{1,n}$, and $\lambda > 0$, define the vector of weights $\mathbf{r} = \begin{pmatrix} r_1 & \dots & r_n \end{pmatrix}$ and the dilation matrix

$\Lambda_{\mathbf{r}}(\lambda) = \operatorname{diag}\{\lambda^{r_i}\}_{i=1}^{n}$. Also denote $r_{\min} = \min_{i\in\overline{1,n}}\ \{r_i\}$ and $r_{\max} = \max_{i\in\overline{1,n}}\ \{r_i\}$.

Definition 2 ([5], [20]):

1) A function $h : \mathbb{R}^n \to \mathbb{R}$ is called $\mathbf{r}$-*homogeneous*, if for some $\nu \in \mathbb{R}$ the relation $h(\Lambda_{\mathbf{r}}(\lambda)x) = \lambda^{\nu} h(x)$ holds for all $x \in \mathbb{R}^n$ and any $\lambda > 0$.
2) A vector field $f : \mathbb{R}^n \to \mathbb{R}^n$ is called $\mathbf{r}$-homogeneous, if for some $\nu \geq -r_{\min}$ the relation $f(\Lambda_{\mathbf{r}}(\lambda)x) = \lambda^{\nu}\Lambda_{\mathbf{r}}(\lambda)f(x)$ holds for all $x \in \mathbb{R}^n$ and any $\lambda > 0$.

In both cases, the constant ν is called the *degree of homogeneity*.

Definition 3: A dynamical system as in (1) is called $\mathbf{r}$-*homogeneous of degree* ν if this property is satisfied for the vector field f, in the sense of Definition 2.

Definition 4: For $x \in \mathbb{R}^n$ and $\varpi \geq r_{\max}$, the $\mathbf{r}$-*homogeneous norm* is defined as follows

$$\|x\|_{\mathbf{r}} = \left(\sum_{i=1}^{n} |x_i|^{\varpi/r_i}\right)^{1/\varpi}.$$

Notice that, for all $x \in \mathbb{R}^n$, its Euclidean norm $\|x\|$ is related to the homogeneous one: $\underline{\sigma}_r(\|x\|_{\mathbf{r}}) \leq \|x\| \leq \bar{\sigma}_r(\|x\|_{\mathbf{r}})$ for some $\underline{\sigma}_r, \bar{\sigma}_r \in \mathcal{K}_\infty$ [5].

In the following, due to this norm equivalence, the stability analysis with respect to the norm $\|\cdot\|$ can be replaced with the analysis for the norm $\|\cdot\|_{\mathbf{r}}$. The homogeneous norm has an important property: it is $\mathbf{r}$-homogeneous of degree 1, that is $\|\Lambda_{\mathbf{r}}(\lambda)x\|_{\mathbf{r}} = \lambda\|x\|_{\mathbf{r}}$ for all $x \in \mathbb{R}^n$ and $\lambda > 0$.

2) Lyapunov functions: One of the main features of homogeneous systems related to the Lyapunov function method is described in the following theorem:

Theorem 2 ([17], [19]): For an $\mathbf{r}$-homogeneous system (1) of degree ν, if the origin is GAS, then there exists an $\mathbf{r}$-homogeneous function $V : \mathbb{R}^n \to \mathbb{R}_+$ of degree $\mu > -\nu$, C^1 on $\mathbb{R}^n$, smooth on $\mathbb{R}^n\setminus\{0\}$, and such that for all $x \in \mathbb{R}^n$:

$$a\|x\|_{\mathbf{r}}^{\mu} \leq V(x) \leq b\|x\|_{\mathbf{r}}^{\mu}, \tag{4}$$

$$D^{+}V(x)f(x) \leq -cV^{1+\frac{\nu}{\mu}}(x) \leq -d\|x\|_{\mathbf{r}}^{\nu+\mu}, \tag{5}$$

for some $0 < a \leq b$, $c > 0$ and $d > 0$.

We describe next an interesting robustness result for homogeneous Lyapunov functions established in [5]:

Proposition 2: For a locally Lipschitz continuous and $\mathbf{r}$-homogeneous function $V : \mathbb{R}^n \to \mathbb{R}_+$ of degree $\mu > -\nu$, assume that the estimates (4), (5) are satisfied for constants $0 < a \leq b$ and $d > 0$. Let $\varepsilon : \mathbb{R}^n \to \mathbb{R}$ be a locally Lipschitz continuous and $\mathbf{r}$-homogeneous function of degree μ such that $-a < \underline{\varepsilon}$ and $d > \overline{\varepsilon}$, where $\underline{\varepsilon} = \inf_{y\in S_{\mathbf{r}}(1)} \varepsilon(y)$ and $\overline{\varepsilon} = \sup_{y\in S_{\mathbf{r}}(1)} D^{+}\varepsilon(y)f(y)$, with $S_{\mathbf{r}}(a) := \{x \in \mathbb{R}^n \mid \|x\|_{\mathbf{r}} = a\}$ $(a > 0)$. Then $V'(x) = V(x) + \varepsilon(x)$ is a locally Lipschitz continuous and $\mathbf{r}$-homogeneous Lyapunov function for (1).

C. ANNs & PINNs

1) Artificial neural networks: An *ANN* is a graph-like machine learning model. Each node is an artificial neuron, and each edge connecting nodes is a weight, which registers the degree of connection between neurons. Each neuron may be considered as a function that takes the output of nearby neurons as its input. Neurons are aggregated into L hidden layers, compressed between the input and output layers.

An ANN with more than one hidden layer ($L > 1$) between the input and output is called a *deep neural network*. For instance, in a *feedforward neural network*, the input is processed successively through the layers L, $L-1, \ldots, 1$. We will use networks with the input vector $x = \begin{pmatrix} x_1 & \ldots & x_n \end{pmatrix}^{\top} \in \mathbb{R}^n$ and output $W(x;\theta) \in \mathbb{R}^m$, where $\theta \in \mathbb{R}^p$ is the vectors of parameters to be learned. There are N_ℓ neurons in the layer numbered ℓ and y_k^{ℓ} is the output of the neuron at position k in the layer ℓ, giving a vector $y^{\ell} = \begin{pmatrix} y_1^{\ell} & \ldots & y_{N_\ell}^{\ell} \end{pmatrix}^{\top} \in \mathbb{R}^{N_\ell}$. At the lowest level, $y^{L+1} = x$, and the outputs of the neurons in the highest layer y^1 are used to compute the network's output $W(x;\theta)$. More precisely, in the hidden layers, the outputs of the neurons are

$$y_k^{\ell} = \sigma^{\ell}\left({w_k^{\ell}}^{\top} y^{\ell+1} + b_k^{\ell}\right),$$

for $k \in \overline{1, N_\ell}$, with $\sigma^{\ell} : \mathbb{R} \to \mathbb{R}$ an activation function, $w_k^{\ell} = \begin{pmatrix} w_{k,1}^{\ell} & \ldots & w_{k,N_{\ell+1}}^{\ell} \end{pmatrix}^{\top} \in \mathbb{R}^{N_{\ell+1}}$ the weights and $b_k^{\ell} \in \mathbb{R}$ the biases. The output is obtained by an affine linear combination of the values of the highest layer ($\ell = 1$) in the form:

$$W(x;\theta) = \sum_{k=1}^{N_1} a_k y_k^1 + c = \sum_{k=1}^{N_1} a_k \sigma^1\left({w_k^1}^{\top} y^2 + b_k^1\right) + c$$

where all parameters a_k, w_k^1, b_k^1, and c in the expression are gathered in the parameter θ (as well as other weights if $L > 1$).

2) Approximation of a continuous function by a neural network: In this subsection, we recall a popular method of approximating a continuous function.

The works of [9] and [2] have established that, given enough hidden units, standard multilayer feedforward networks with any number of hidden layers and arbitrary squashing function are capable of approximating, with any desired degree of accuracy, any continuous function of n variables with support in some compact set. In particular, the following result can be formulated:

Theorem 3 ([2]): Let $\phi : I_n \to \mathbb{R}$ be a continuous function defined on the n-dimensional unit cube I_n in $\mathbb{R}^n$, and $\varepsilon > 0$. Then, there is a $N_1 > 0$ such that

$$W(x;\theta) = \sum_{k=1}^{N_1} a_k \sigma^1\left({w_k^1}^{\top} x + b_k^1\right) + c \tag{6}$$

with σ^1 a sigmoidal function (*i.e.*, a non-decreasing function $\mathbb{R} \to \mathbb{R}$ with $\lim_{t\to+\infty} \sigma^1(t) = 1$ and $\lim_{t\to-\infty} \sigma^1(t) = 0$), $a_k, b_k^1, c \in \mathbb{R}$, and $w_k^1 \in \mathbb{R}^n$, for all $k \in \overline{1, N_1}$, satisfying

$$|W(x;\theta) - \phi(x)| < \varepsilon,\ \forall x \in I_n.$$

3) Application of ANNs for approximating Lyapunov functions: We wish to select θ such that $W(x;\theta)$ approximates a Lyapunov function $V(x)$, for all $x \in K$, with K a fixed compact set including the origin for a GAS (at zero) system (1). According to Theorem 3, we can use one single hidden layer, so $L = 1$ (a deeper ANNs can be used similarly). The formula of $W(x;\theta)$ is given by (6).

Following [8], to train the network and approximate a Lyapunov function by $W(x;\theta)$, a loss function $\mathcal{L} : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ must be defined. Then the *training phase* consists of finding parameters θ that search for the minimality of

$$J(\theta) = \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}\left(W(x^{(i)};\theta), DW(x^{(i)};\theta), x^{(i)}\right)$$

where $x^{(i)} \in K$ are test points (elements of a learning/test dataset). Unlike conventional ANN learning tasks, the loss function $\mathcal{L}$ additionally depends on the test points' values of the derivative DW of W with respect to x. Such a constraint is required to establish if W is a Lyapunov function [8]:

$$DW(x;\theta)f(x) \leq -\alpha(\|x\|), \ \forall x \in \mathbb{R}^n,$$

for a suitably chosen $\alpha \in \mathcal{K}$. The appearance of DW naturally opens a way of interpreting this training problem within a partial differential equation (PDE) framework. In such a case, the positivity of the Lyapunov function can be introduced through the boundary conditions:

$$W(0,\theta) = 0 \ \text{ and } \ W(x;\theta) > 0, \ \forall x \in \mathbb{R}^n.$$

However, it can be challenging to implement the boundary conditions in the above form numerically, either because the first expression involves a single point or because of the strict positive condition in the second one. We may then replace it with the following to palliate this:

$$\alpha_1(\|x\|) \leq W(x;\theta) \leq \alpha_2(\|x\|), \ \forall x \in \mathbb{R}^n,$$

for some functions α_1 and $\alpha_2 \in \mathcal{K}$.

Based on the above observations, in [8] the following loss function is considered:

$$\mathcal{L}(w,p,x) = [pf(x) + \alpha(\|x\|)]^2 + [w - \alpha_1(\|x\|)]_-^2 \\ + [w - \alpha_2(\|x\|)]_+^2,$$

where $[\cdot]_- = \min\{0,\cdot\}$ and $[\cdot]_+ = \max\{0,\cdot\}$. The drawback of this functional $\mathcal{L}$ is that its minimization and the obtained solution W depends on the choice of the tuning functions α, α_1, and α_2, and it could be the case that, for a chosen set of these functions, there is no Lyapunov function for (1).

4) PINN: Physics-Informed Neural Networks are artificial neural network models that use a priori information from the ODE or PDE system governing the dynamics under study and approximate a solution of this ODE/PDE for given initial conditions and time/space domain. This results in introducing terms into the loss function of the ANN that are somehow proportional to the considered physics/dynamics information.

A typical PINN uses a fully connected ANN architecture to represent the solution $V(x,t)$ of a dynamical system. For some domain $\Omega \subset \mathbb{R}^n$ and $T > 0$, the PINN model generates $W(x,t;\theta)$ predicting $V(x,t)$ for $x \in \Omega$ and $t \in [0,T]$, where θ are properly learned parameters of ANN as before. The accuracy of the PINN output is determined by θ, which is optimized with respect to the PINN loss function during the training. To derive the PINN loss function, we consider V to be described by differential equations of the general form:

$$\mathcal{D}_x(V(x,t)) + \mathcal{D}_t(V(x,t)) = 0, \ \forall x \in \Omega, \ t \in (0,T]; \\ V(x,0) = V_0(x), \forall x \in \Omega; \\ \mathcal{B}(V(x,t)) = g(x,t), \forall x \in \partial\Omega, \ t \in (0,T],$$

where $\mathcal{D}_x$ and $\mathcal{D}_t$ are differential operators in x and t, $\mathcal{B}$ is a suitable boundary functional, and the functions V_0 and g determine the initial and the boundary conditions. The loss function incorporates all information in the system [14]:

$$\mathcal{L}(W,x,t) = \nu_1\|\mathcal{D}_t(W(x,t;\theta)) + \mathcal{D}_x(W(x,t;\theta))\|^2 \\ + \nu_2\|W(x,0;\theta) - V_0(x)\|^2 \\ + \nu_3\|\mathcal{B}(W(x,t;\theta)) - g(x,t)\|^2,$$

where ν_1, ν_2, $\nu_3 > 0$ are tuning parameters. We can see that the functional $\mathcal{L}$ depends on the derivative of W with respect to x and t, which are the inputs of ANN producing $W(x,t;\theta)$. Then, the problem of approximating a Lyapunov function for the system (1) by an ANN solved in [8] is an example of exploitation of PINN. The functional $\mathcal{L}$ from the previous subsection contains the structure of the PDE describing the properties of a Lyapunov function.

III. Problem statement

Consider the vector field in (1). We wish to employ an ANN or PINN-based structure with an input vector $x = \begin{pmatrix} x_1 & \dots & x_n \end{pmatrix}^\top \in \mathbb{R}^n$ and a scalar output $\mathcal{W}(x;\theta) \in \mathbb{R}$ to approximate an unknown Lyapunov function of (1) at the origin for $x \in K$, with K being a compact set in $\mathbb{R}^n$ on which V must be computed, $\{0\} \in K$. Here, the vector $\theta \in \mathbb{R}^p$ represents the network's free parameters that must be learned to get the desired result.

To this end, we will refine the results of [8] by utilizing the result of Proposition 1 and the resulting PDE (3). Therefore, in our case, the ANN has to approximate the change of variables T defining $\mathcal{W}$ and $D\mathcal{W}$ due to (3). For brevity, let us use one single hidden layer ANN consisting of N_1 neurons without biases (in such a case, we naturally get that the transformation is zero at the origin). Consequently, consider $\theta = \operatorname{vec}\begin{pmatrix} w_1 & w_2 \end{pmatrix} \in \mathbb{R}^{2N_1 n}$, with $w_1 \in \mathbb{R}^{n \times N_1}$, $w_2 \in \mathbb{R}^{N_1 \times n}$, and define a smooth map $\mathcal{N}_\theta : \mathbb{R}^n \to \mathbb{R}^n$ as:

$$\mathcal{N}_\theta(x) = w_1 \bar{\sigma}^1(w_2 \, x), \forall x \in K, \tag{7}$$

where $\bar{\sigma}^1 : \mathbb{R}^{N_1} \to \mathbb{R}^{N_1}$ is a function defined as $\bar{\sigma}^1(z) = \begin{pmatrix} \sigma^1(z_1) & \dots & \sigma^1(z_{N_1}) \end{pmatrix}^\top \in \mathbb{R}^{N_1}$ for $z \in \mathbb{R}^{N_1}$, with σ^1 being a smooth activation function. Then such a function $\mathcal{N}_\theta(x)$ should approximate the transformation map T, and the Lyapunov function candidate will be approximated by

$$\mathcal{W}(x;\theta) = \mathcal{N}_\theta(x)^\top \mathcal{N}_\theta(x), \forall x \in K. \tag{8}$$

A similar structure of Lyapunov functions is proposed in [16] for polynomial systems using the sum-of-squares approach.

The function $\mathcal{W}(x;\theta)$ would be a Lyapunov function candidate, provided that the conditions for positive definiteness are verified, which in such a case can be rewritten as follows:

$$w_1^\top w_1 \geq I_{N_1},\ w_2^\top w_2 \geq I_n,$$

where I_n denotes the identity matrix in $\mathbb{R}^{n\times n}$. The remaining condition of (3) on the decay of $\mathcal{W}(x;\theta)$ along the trajectories of the system (1) can be presented as follows:

$$D\mathcal{W}(x;\theta)\ f(x) \leq -\mathcal{W}(x;\theta),\, \forall x \in K.$$

To define a loss function, we combine the conditions formulated above and write, with respect to the map $\mathcal{N}_\theta(x)$:

$$\mathcal{L}(w,p,x) = \left[2w^\top pf(x) + w^\top w\right]_+ - \left[w^\top w\right]_-, \qquad (9)$$

where, as before, w represents $\mathcal{N}_\theta$ and p corresponds to the derivative $D\mathcal{N}_\theta$ with respect to x. Note that $\mathcal{L}$ is not a square loss function, so its terms are not squares of the argument in comparison with the formulation of [8]. Therefore, it results in a simpler gradient with respect to θ. According to the results of [7], such a presentation of a Lyapunov function always exists for $n \neq 4, 5$ (see Proposition 1). In addition, we deal with inequalities and not with equality solutions. The functional (9) can be equivalently rewritten in the form explicitly dependent on θ:

$$\begin{aligned}\mathcal{L}(\theta,x) = &\left[2\mathcal{N}_\theta^\top(x)D\mathcal{N}_\theta(x)f(x) + \mathcal{N}_\theta^\top(x)\mathcal{N}_\theta(x)\right]_+ \\ &- \left[w_1^\top w_1 - I_{N_1}\right]_- - \left[w_2^\top w_2 - I_n\right]_-,\end{aligned}$$

which is ready for application of the optimization procedures in θ. In such a case, the Lyapunov function candidate (8) is a smooth function on $\mathbb{R}^n$, while in Proposition 1, the smoothness at the origin is not ensured for the studied class of systems in (1). Therefore, it is a mild limitation of this ANN approximation, which has some consequences.

Following the result of Theorem 3, for the vector of parameters θ properly tuned by a learning algorithm, for any $\varepsilon > 0$ and given compact set K, there is $N_1 > 0$ such that

$$\sup_{x\in K} \|V(x) - \mathcal{W}(x;\theta)\| \leq \varepsilon,$$

for some Lyapunov function V for the system (1). However, the function V is vanishing at the origin by Definition 1, while the error function $e(x;\theta) = V(x) - \mathcal{W}(x;\theta)$, being upper-bounded in the norm by ε in the set K, is not necessary vanishing for $x = 0$. Therefore, $\mathcal{W}(x;\theta)$ could be sign-indefinite close to the origin, and there is no theoretical guarantee of having a solution to this drawback. Additional research in this direction has a good perspective.

IV. PINN APPROXIMATION FOR HOMOGENEOUS SYSTEMS

As we have just seen, for the system (1) in its generality, the behavior at the origin of any Lyapunov function approximation can be a significant obstacle to ANN/PINN approximation. Fortunately, this obstruction does not apply to homogeneous systems for which a homogeneous Lyapunov function can be selected, whose design and analysis can be performed on the sphere only, leading to a global solution. Hence, the idea is to use a compact set that excludes the origin, estimate the Lyapunov function, and then use dilations to produce a global expression.

A. Stability

For homogeneous systems, we consider an approximation V_θ of an unknown Lyapunov function V defined globally, and where the ANN output $W(x;\theta)$ defined in (6) is needed on the set $S_{\mathbf{r}}(1)$ only:

$$V_\theta(x) = \|x\|_{\mathbf{r}}^{\mu}\ W\left(\Lambda_{\mathbf{r}}^{-1}(\|x\|_{\mathbf{r}})\ x;\theta\right),\ \forall x \in \mathbb{R}^n. \qquad (10)$$

Such a Lyapunov function V_θ is homogeneous by construction. Then, the conditions to be checked can be obtained from the estimates (4), (5) and restricted to the set $S_{\mathbf{r}}(1)$:

$$DW(y;\theta)f(y) \leq -1,\ W(y;\theta) \geq 1,\ \forall y \in S_{\mathbf{r}}(1),$$

which can be transformed into the following loss function:

$$\mathcal{L}(w,p,y) = [pf(y) + 1]_+ - [w-1]_-\,.$$

It is easy to see that, in this case, there is no problem with the vanishing of W somewhere on the set of approximation $S_{\mathbf{r}}(1)$. Moreover, for homogeneous systems and the chosen form of Lyapunov function approximation (10), there exist theoretical guarantees coming from Theorem 3, contrarily the general case of the system (1) discussed in the previous section (notice that (10) may be not smooth at the origin):

Theorem 4: There are a number of hidden units $N_1 > 0$ and a vector of parameters θ such that (10) is an $\mathbf{r}$-homogeneous global Lyapunov function for any degree $\mu > \max\{0, -\nu\}$ for an $\mathbf{r}$-homogeneous system (1) of degree ν with GAS origin.

Proof: For a properly learned parameters θ (it implies that a sufficiently dense grid has been selected guaranteeing that the behavior of f and W in between is sufficiently small), for $\varepsilon \in (0,1)$ sufficiently small, there is a sufficiently big number of neurons in the hidden layer $N_1 > 0$ such that

$$\sup_{y\in S_{\mathbf{r}}(1)} \|V(y) - W(y;\theta)\| \leq \varepsilon,$$

then it implies

$$DW(y;\theta)f(y) \leq -\rho,\ W(y;\theta) \geq 1-\varepsilon,\ \forall y \in S_{\mathbf{r}}(1),$$

for some $\rho > 0$, and (10) is the required homogeneous global Lyapunov function (see Proposition 2). ■

Comparing this result and its applicability conditions to the corresponding ones from [5], it is easy to conclude that PINN-based approach from Theorem 4 is much simpler. In addition, it allows many extensions to be straightforwardly obtained, as coupling with a stabilizing control design presented below.

B. Stabilization

Consider a continuous-time system:

$$\dot{x}(t) = F(x(t), u(t)),\ \forall t \in \mathbb{R}_+, \qquad (11)$$

where $x : \mathbb{R}_+ \to \mathbb{R}^n$ represents the state function, $F : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is a smooth vector field with $F(0,0) = 0$, and

$u : \mathbb{R}_+ \to \mathbb{R}^m$ is the input function. We need to assume the following symmetry relation extending the homogeneity property to the systems with input:

Assumption 1: There exist two vectors of weights $\mathbf{r} = \begin{pmatrix} r_1 & \dots & r_n \end{pmatrix}$ and $\tilde{\mathbf{r}} = \begin{pmatrix} \tilde{r}_1 & \dots & \tilde{r}_m \end{pmatrix}$ such that

$$F(\Lambda_{\mathbf{r}}(\lambda)x, \Lambda_{\tilde{\mathbf{r}}}(\lambda)u) = \lambda^{\nu}\Lambda_{\mathbf{r}}(\lambda)F(x,u)$$

for all $\lambda > 0$, all $x \in \mathbb{R}^n$, all $u \in \mathbb{R}^m$ and some $\nu \geq -r_{\min}$.

Hence, for $u = 0$, we recover the $\mathbf{r}$-homogeneity of F in the standard sense. Now, assume that there exists a stabilizing continuous feedback law admitting a symmetry relation:

Assumption 2: There is continuous function $u : \mathbb{R}^n \to \mathbb{R}^m$ such that

$$u(\Lambda_{\mathbf{r}}(\lambda)x) = \Lambda_{\tilde{\mathbf{r}}}(\lambda)u(x)$$

for all $\lambda > 0$ and all $x \in \mathbb{R}^n$, and the closed-loop system

$$\dot{x}(t) = F(x(t), u(x(t))) \tag{12}$$

admits the origin as a GAS equilibrium.

It is easy to verify that (12) is $\mathbf{r}$-homogeneous. Hence, by Theorem 2, there is an $\mathbf{r}$-homogeneous Lyapunov function $V(x)$. Let us try to calculate simultaneously the control u and the Lyapunov function V using PINNs.

As described in [5], homogeneous functions can be approximated on the unit sphere, and the expression (10) can be used to search $V(x)$. Similarly, define

$$U(x;\theta') = \Lambda_{\tilde{\mathbf{r}}}(\|x\|_{\mathbf{r}})\mathcal{N}_{\theta'}\left(\Lambda_{-\mathbf{r}}(\|x\|_{\mathbf{r}})x\right), \ \forall x \in \mathbb{R}^n$$

as an approximation of $u(x)$, where $\mathcal{N}_{\theta'}$ is given in (7) with m-dimensional output of ANN, and $N_1' > 0$ denotes the number of hidden units. The functional (9) can be rewritten in a respective form depending on θ and θ':

$$\begin{aligned}\mathcal{L}(\theta, \theta', y) = {} & [DW(y;\theta)F(y, \mathcal{N}_{\theta'}(y)) + 1]_+ \\ & - [W(y;\theta) - 1]_- .\end{aligned}$$

Its proper minimization allows the vectors of weights θ and θ' approximating the respective quantities, V and u, to be found.

Theorem 5: Let assumptions 1 and 2 be satisfied. There are numbers of hidden units $N_1 > 0$, $N_1' > 0$ and vectors of parameters θ, θ' such that (10) is an $\mathbf{r}$-homogeneous global Lyapunov function for any degree $\mu > \max\{0, -\nu\}$ for an $\mathbf{r}$-homogeneous system

$$\dot{x}(t) = F\left(x(t), U(x(t);\theta')\right) \tag{13}$$

possessing the GAS origin.

Proof: It repeats the main arguments of the proof of Theorem 4 by observing that the system (13) is $\mathbf{r}$-homogeneous of degree ν by construction. ■

V. Conclusion

In this note, we have proposed a method for approximating globally specified Lyapunov functions and stabilizing controllers for homogeneous dynamical systems using a particular type of ANN, physics-informed neural networks. The natural characteristics of Lyapunov functions and their derivatives are perfectly adaptable to employing PINN techniques. The great advantage of working with homogenous systems is that all analysis and design can be done locally on a well-specified unit sphere. This important feature also fully matches the criteria for applying neural networks. Using recent results on partial homogenization of nonholonomic systems [4], the proposed approach can be applied to this class of models.

References

[1] P.-H. Chiu, J. C. Wong, C. Ooi, M. H. Dao, and Y.-S. Ong. Can-pinn: A fast physics-informed neural network based on coupled-automatic–numerical differentiation method. *Computer Methods in Applied Mechanics and Engineering*, 395:114909, 2022.

[2] G. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4):303–314, Dec 1989.

[3] M. W. M. G. Dissanayake and N. Phan-Thien. Neural-network-based approximations for solving partial differential equations. *Communications in Numerical Methods in Engineering*, 10(3):195–201, 1994.

[4] D. Efimov and I. Kolmanovsky. Homogeneity with respect to a part of variables and accelerated stabilization. In *Proc. 62nd IEEE CDC*, Singapore, 2023.

[5] D. Efimov, R. Ushirobira, J. A. Moreno, and W. Perruquetti. Homogeneous Lyapunov functions: from converse design to numerical implementation. *SIAM Journal on Control and Optimization*, 56(5):3454–3477, 2018.

[6] M. H. Freedman. The topology of four-dimensional manifolds. *J. Differential Geom.*, 17(3):357–453, 1982. MR:0679066. Zbl:0528.57011.

[7] L. Grüne, E. D. Sontag, and F. R. Wirth. Asymptotic stability equals exponential stability, and ISS equals finite energy gain—if you twist your eyes. *Systems & Control Letters*, 38(2):127–134, 1999.

[8] L. Grüne. Computing Lyapunov functions using deep neural networks. *Journal of Computational Dynamics*, 8(2):131–152, 2021.

[9] K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989.

[10] I. D. Jimenez Rodriguez, A. Ames, and Y. Yue. LyaNet: A Lyapunov framework for training neural ODEs. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 18687–18703. PMLR, 17–23 Jul 2022.

[11] H.K. Khalil. *Nonlinear Systems*. Pearson Education. Prentice Hall, 2002.

[12] Y. Long and M.M. Bayoumi. Feedback stabilization: control Lyapunov functions modelled by neural networks. In *Proceedings of 32nd IEEE Conference on Decision and Control*, pages 2812–2814 vol.3, 1993.

[13] V. Petridis and S. Petridis. Construction of neural network based Lyapunov functions. In *The 2006 IEEE International Joint Conference on Neural Network Proceedings*, pages 5059–5065, 2006.

[14] M. Raissi, P. Perdikaris, and G.E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

[15] R. C. B. Rego and F. M. U. de Araújo. Learning-based robust neuro-control: A method to compute control Lyapunov functions. *International Journal of Robust and Nonlinear Control*, 32(5):2644–2661, 2022.

[16] S. M. Richards, F. Berkenkamp, and A. Krause. The Lyapunov neural network: Adaptive stability certification for safe learning of dynamical systems. In *Proceedings of The 2nd Conference on Robot Learning*, pages 466–476. PMLR, 2018.

[17] L. Rosier. Homogeneous Lyapunov function for homogeneous continuous vector field. *Systems & Control Letters*, 19(6):467–473, 1992.

[18] F.Wesley Wilson. The structure of the level surfaces of a lyapunov function. *Journal of Differential Equations*, 3(3):323–329, 1967.

[19] V. I. Zubov. *Methods of A. M. Lyapunov and Their Application*. P. Noordhoff Ltd, Groningen, 1964.

[20] V.I. Zubov. On systems of ordinary differential equations with generalized homogenous right-hand sides. *Izvestia vuzov. Mathematica.*, 1:80–88, 1958. in Russian.