Algorithmic verification of approximations to complete Lyapunov functions

Jóhann Björnsson[1], Peter Giesl[2], and Sigurdur Hafstein[1]

***Abstract*— In [3] Conley showed that the state-space of a dynamical system can be decomposed into a gradient-like part and a chain-recurrent part, and that this decomposition is characterized by a so-called complete Lyapunov function for the system. In [14] Kalies, Mischaikow, and VanderVorst proposed a combinatorial method to compute discrete approximations to such complete Lyapunov functions. Their approach uses a finite subdivision of a compact subset of the state-space and a combinatorial multivalued map to approximate the dynamics. They proved that as the diameter of the elements of the subdivision approaches zero the resulting approximations to complete Lyapunov functions converge to a true complete Lyapunov function for the system. In [2] Ban and Kalies implemented this algorithm and used it to compute an approximation to a complete Lyapunov function for a time-T mapping of the van der Pol oscillator.**

The CPA method to compute Lyapunov functions uses linear programming to parameterize true Continuous and Piecewise Affine Lyapunov functions for continuous dynamical systems [11], [1], [8].

In this paper we propose using the CPA method to evaluate approximations to complete Lyapunov functions computed by the combinatorial method. Especially, we can explicitly compute the region of the state-space where the orbital derivative of the approximation is negative. Further, we use the RBF method [5] to solve a Zubov equation $\nabla V(\mathbf{x})\cdot\mathbf{f}(\mathbf{x}) = -p(\mathbf{x})$ and compare the solution V with the complete Lyapunov function computed by the combinatorial method for the van der Pol oscillator.

***Index Terms*— complete Lyapunov functions, nonlinear systems, combinatorial methods, continuous piecewise affine functions, radial basis functions**

I. INTRODUCTION

Let X be a metric space and $\phi : \mathbb{R}\times X \to X$ be a flow on the space, i.e. ϕ is continuous, $\phi(0,x) = x$ and $\phi(t,\phi(s,x)) = \phi(t+s,x)$ for all $x\in X$ and all $s,t\in\mathbb{R}$. The chain recurrent set of a flow is an adequate generalization of recurrent and almost recurrent phenomena and includes equilibrium points, periodic orbits, homoclinic orbits, etc. For a proper definition cf. e.g. [3]. Denote by $\mathcal{R}_C(\phi)$ the chain recurrent set of the flow ϕ. In [3] Conley showed that for a compact X there exists a so-called complete Lyapunov function for the flow ϕ, i.e. a continuous function $V : X \to [0,1]$ such that $t\mapsto V(\phi(t,x))$ is:

- constant for each $x\in\mathcal{R}_C(\phi)$
- strictly decreasing for each $x\in X\setminus\mathcal{R}_C(\phi)$

Thus, the flow is gradient-like on $X\setminus\mathcal{R}_C(\phi)$. Hurley in [13] proved similar results for time discrete semi-flows on separable metric spaces and in a recent short notice [15] Patrão adapted Hurley's results to prove the existence of a complete Lyapunov function for semi-flows on separable metric spaces.

In this paper we will discuss how to adapt the CPA method to evaluate the quality of approximations to complete Lyapunov functions. We will compute such approximations using the combinatorial methods presented in [2], [14] and the RBF method presented in [5]. We give a brief overview of the ideas behind these three methods and how we use them in our approach in Sections II to IV. Then, in Section V, we compute approximations to complete Lyapunov functions for the van der Pol oscillator and verify them with the adapted CPA method. Finally, we give some conclusions and outlook for further research in Section VI.

Notations: We write vectors in $\mathbb{R}^n$ and vector fields $\mathbb{R}^n\to\mathbb{R}^n$ in boldface, e.g. $\mathbf{x}$ and $\mathbf{f}(\mathbf{x})$. We consider the dynamical system

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) \tag{I.1}$$

where $\mathbf{f} : \mathbb{R}^n\to\mathbb{R}^n$ is (at least) locally Lipschitz. The solution to the system is denoted $\boldsymbol{\phi}(t,\boldsymbol{\xi})$, i.e. $t\mapsto\boldsymbol{\phi}(t,\boldsymbol{\xi})$ is the solution to the initial-value problem $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}),\ \mathbf{x}(0) = \boldsymbol{\xi}$. We denote the power-set of a set U by $\mathcal{P}(U)$. For a vector $\mathbf{x}\in\mathbb{R}^n$ and $p\geq 1$ we define the norm $\|\mathbf{x}\|_p = \left(\sum_{i=1}^n |x_i|^p\right)^{1/p}$. We also define $\|\mathbf{x}\|_\infty = \max_{i\in\{1,\dots,n\}}|x_i|$.

Let $\mathbf{x}_0,\mathbf{x}_1,\dots,\mathbf{x}_m$ be vectors in $\mathbb{R}^n$. The set of all *convex combinations* of these vectors is denoted by $\mathrm{co}\{\mathbf{x}_0,\mathbf{x}_1,\dots,\mathbf{x}_m\} := \{\sum_{i=0}^m \lambda_i\mathbf{x}_i : 0\leq\lambda_i\leq 1, \sum_{i=0}^m\lambda_i = 1\}$. The vectors $\mathbf{x}_0,\mathbf{x}_1,\dots,\mathbf{x}_m$ are called *affinely independent* if $\sum_{i=1}^m \lambda_i(\mathbf{x}_i-\mathbf{x}_0) = \mathbf{0}$ implies $\lambda_i = 0$ for all $i = 1,\dots,m$. If $\mathbf{x}_0,\mathbf{x}_1,\dots,\mathbf{x}_m$ are affinely independent, then the set $\mathfrak{S} := \mathrm{co}\{\mathbf{x}_0,\mathbf{x}_1,\dots,\mathbf{x}_m\}$ is called an m-simplex and the vectors $\mathbf{x}_0,\mathbf{x}_1,\dots,\mathbf{x}_m$ are said to be its vertices.

We denote the diameter of a bounded set $\mathcal{U}\subset\mathbb{R}^n$ with respect to the norm $\|\cdot\|_p$ by

$$\mathrm{diam}_p(\mathcal{U}) := \sup_{\mathbf{x},\mathbf{y}\in\mathcal{U}}\|\mathbf{x}-\mathbf{y}\|_p.$$

II. THE CPA METHOD

In the CPA method, linear programming is used to construct Lyapunov functions that are continuous and piecewise affine. A Lyapunov function V is parameterized by its values at the vertices of a triangulation and linear inequalities ensure

[1]Jóhann Björnsson (johannsb13@ru.is) and Sigurdur Hafstein (sigurdurh@ru.is) are with the School of Science and Engineering, Reykjavik University, Iceland. Björnsson is supported by The Icelandic Research Fund, grant nr. 130677-052.

[2]Peter Giesl (p.a.giesl@sussex.ac.uk) is with the Department of Mathematics, University of Sussex, United Kingdom

that V is strictly decreasing along all solution trajectories, also inside the simplices.

In this paper, we will use the CPA method not to construct, but to **verify** complete Lyapunov functions. We will use a different method to assign values to V at the vertices of the triangulation and then verify certain inequalities for these values. In other words, we construct a function, and then check if its CPA interpolation on a triangulation is a complete Lyapunov function. A similar approach was followed for classical Lyapunov functions constructed by the Yoshizawa method in [12]. In this paper, we will use two methods to construct complete Lyapunov functions, the combinatorial method described in Section III and the RBF method described in Section IV.

In this section, we first define triangulations and CPA functions. The main result, which we will use in the following, is Theorem 2.5.

A triangulation $\mathcal{T}$ of a subset of $\mathbb{R}^n$ consists of countably many n-simplices. To simplify notations, we often write $\mathcal{T} = (\mathfrak{S}_\nu)$, where it is to be understood that $\nu \in \{1, 2, \ldots, N\}$, if $\mathcal{T}$ has a finite number N of (different) simplices, or $\nu \in \mathbb{N}_{>0}$, if $\mathcal{T}$ is infinite. In this paper, we will only use triangulations with a finite number of simplices. We will briefly describe triangulations suited for our needs; for more details on triangulations for the CPA method, cf. [9], [10].

Definition 2.1 (Triangulation): Let $\mathcal{T}$ be a collection of n-simplices $\mathfrak{S}_\nu$ in $\mathbb{R}^n$. $\mathcal{T}$ is called a triangulation if for every $\mathfrak{S}_\nu, \mathfrak{S}_\mu \in \mathcal{T}$, $\nu \neq \mu$, either $\mathfrak{S}_\nu \cap \mathfrak{S}_\mu = \emptyset$ or $\mathfrak{S}_\nu$ and $\mathfrak{S}_\mu$ intersect in a common face. Recall that a face of an n-simplex $\mathrm{co}\{\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_n\}$ is a k-simplex $\mathrm{co}\{\mathbf{x}_{i_0}, \mathbf{x}_{i_1}, \ldots, \mathbf{x}_{i_k}\}$, where $0 \leq i_0, i_1, \ldots, i_k \leq n$ are pairwise different integers and $0 \leq k \leq n$.

For a triangulation $\mathcal{T}$ we define

$$\mathcal{V}_\mathcal{T} := \{\mathbf{x} \in \mathbb{R}^n : \mathbf{x} \text{ is a vertex of a simplex in } \mathcal{T}\}$$

and

$$\mathcal{D}_\mathcal{T} := \bigcup_{\mathfrak{S}_\nu \in \mathcal{T}} \mathfrak{S}_\nu.$$

We call $\mathcal{V}_\mathcal{T}$ the **vertex set** of the triangulation $\mathcal{T}$ and we say that $\mathcal{T}$ **is a triangulation** of the set $\mathcal{D}_\mathcal{T}$.

□

Let $\mathcal{T} = (\mathfrak{S}_\nu)$ be a triangulation of a set $\mathcal{D} \subset \mathbb{R}^n$. Then we can define a continuous, piecewise affine function $P : \mathcal{D} \to \mathbb{R}$ by fixing its values at the vertices of the simplices of the triangulation $\mathcal{T}$. More exactly:

Definition 2.2 (CPA function): Let $\mathcal{T} = (\mathfrak{S}_\nu)$ be a triangulation of a set $\mathcal{D} \subset \mathbb{R}^n$ and assume that for every $\mathbf{x} \in \mathcal{V}_\mathcal{T}$ we are given a number $P_\mathbf{x} \in \mathbb{R}$. Then we uniquely define a function $P : \mathcal{D} \to \mathbb{R}$ through:

i) $P(\mathbf{x}) := P_\mathbf{x}$ for every $\mathbf{x} \in \mathcal{V}_\mathcal{T}$.
ii) P is affine on every simplex $\mathfrak{S}_\nu \in \mathcal{T}$, i.e. there is a vector $\mathbf{a}_\nu \in \mathbb{R}^n$ and a number $b_\nu \in \mathbb{R}$, such that $P(\mathbf{x}) = \mathbf{a}_\nu^T \mathbf{x} + b_\nu$ for all $\mathbf{x} \in \mathfrak{S}_\nu$.

The set of all such continuous, piecewise affine functions $\mathcal{D} \to \mathbb{R}$ fulfilling i) and ii) is denoted by $\mathrm{CPA}[\mathcal{T}]$ or $\mathrm{CPA}[(\mathfrak{S}_\nu)]$. We identify $P \in \mathrm{CPA}[\mathcal{T}]$ with $(P_\mathbf{x})_{\mathbf{x} \in \mathcal{V}_\mathcal{T}}$ and write $P \sim (P_\mathbf{x})_{\mathbf{x} \in \mathcal{V}_\mathcal{T}}$.

For every $\mathfrak{S}_\nu \in \mathcal{T}$ we define $\nabla P_\nu = \nabla P|_{\mathfrak{S}_\nu} := \mathbf{a}_\nu$, where $\mathbf{a}_\nu \in \mathbb{R}^n$ is as in ii).

□

Note that $\mathbf{a}_\nu$ and thus ∇P_ν in ii) of Definition 2.2 is unique for every simplex $\mathfrak{S}_\nu$.

Remark 2.3: If $\mathbf{x} \in \mathfrak{S}_\nu = \mathrm{co}\{\mathbf{x}_0^\nu, \mathbf{x}_1^\nu, \ldots, \mathbf{x}_n^\nu\} \in \mathcal{T}$, then $\mathbf{x}$ can be written uniquely as a convex combination

$$\mathbf{x} = \sum_{i=0}^n \lambda_i \mathbf{x}_i^\nu, \ \sum_{i=0}^n \lambda_i = 1, \ 0 \leq \lambda_i \leq 1 \text{ for } i = 0, 1, \ldots, n,$$

of the vertices of $\mathfrak{S}_\nu$ and

$$P(\mathbf{x}) = P\left(\sum_{i=0}^n \lambda_i \mathbf{x}_i^\nu\right) = \sum_{i=0}^n \lambda_i P(\mathbf{x}_i^\nu) = \sum_{i=0}^n \lambda_i P_{\mathbf{x}_i^\nu}.$$

□

Definition 2.4: Let $P : \mathcal{U} \to \mathbb{R}$, $\mathcal{U} \subset \mathbb{R}^n$, be a function and let $\mathcal{T}$ be a triangulation with $\mathcal{D}_\mathcal{T} \subset \mathcal{U}$. We call the function $Q \sim (Q_\mathbf{x})_{\mathbf{x} \in \mathcal{V}_\mathcal{T}} \in \mathrm{CPA}[\mathcal{T}]$, where $Q_\mathbf{x} := P(\mathbf{x})$ for every $\mathbf{x} \in \mathcal{V}_\mathcal{T}$, the **CPA interpolation of the function** P **on** $\mathcal{T}$. Usually we write the CPA interpolation of P as P_C.

□

The following theorem follows directly from the proof of Theorem 4.5 in [1].

Theorem 2.5: Consider the system (I.1), let $\mathcal{T} = (\mathfrak{S}_\nu)$ be a triangulation, and assume that $\mathbf{f}$ in (I.1) is C^2 on $\mathcal{D}_\mathcal{T}$. Let $V \sim (V_\mathbf{x})_{\mathbf{x} \in \mathcal{V}_\mathcal{T}} \in \mathrm{CPA}[\mathcal{T}]$.

Define for every $\mathfrak{S}_\nu \in \mathcal{T}$ the constants $h_\nu := \mathrm{diam}_2(\mathfrak{S}_\nu)$ and

$$E_\nu := \frac{nB_\nu}{2} h_\nu^2, \text{ where} \tag{II.1}$$

$$B_\nu \geq \max_{m,r,s=1,2,\ldots,n} \max_{\mathbf{z} \in \mathfrak{S}_\nu} \left| \frac{\partial^2 f_m}{\partial x_r \partial x_s}(\mathbf{z}) \right|.$$

Assume that for a simplex $\mathfrak{S}_\nu = \mathrm{co}\{\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_n\} \in \mathcal{T}$ the inequality

$$0 > \nabla V_\nu \cdot \mathbf{f}(\mathbf{x}_i) + E_\nu \|\nabla V_\nu\|_1 \tag{II.2}$$

holds true for every vertex $\mathbf{x}_i \in \mathfrak{S}_\nu$. Then

$$0 > \nabla V_\nu \cdot \mathbf{f}(\mathbf{x}) \ \text{ for all } \mathbf{x} \in \mathfrak{S}_\nu.$$

□

III. The Combinatorial Method

Consider the dynamical system (I.1) on a compact set $\mathcal{C} \subseteq \mathbb{R}^n$, where $\mathbf{f}$ is a C^1 vector-field on $\mathcal{C}$. We additionally assume that $\mathcal{C}$ is an invariant set for the dynamics, i.e. if $\mathbf{x} \in \mathcal{C}$ then $\boldsymbol{\phi}(t, \mathbf{x}) \in \mathcal{C}$ for all $t \in \mathbb{R}$. Our aim is to construct an approximate Lyapunov function for the system by discretizing both the state-space $\mathcal{C}$ and the time-T map of the flow generated by the vector field $\mathbf{f}$ in order to construct a finite directed graph which encapsulates the behaviour of the system.

We begin by discretizing $\mathcal{C}$. We call a family $U = \{\mathcal{U}_i\}_{i \in I}$ of compact sets a grid for the set $\mathcal{C}$ if the following conditions are met:

1) $\mathcal{C} \subseteq \bigcup \mathcal{U}_i$.
2) $\overline{\mathcal{U}_i^\circ} = \mathcal{U}_i$ for all i.
3) $\mathcal{U}_i^\circ \bigcap \mathcal{U}_j^\circ = \varnothing$ if $i \neq j$.

We refer to the sets $\mathcal{U}_i$'s as the cells of the grid. For $\epsilon > 0$ we define the diameter of U to be

$$\text{diam}(U) := \sup_{i \in I} \text{diam}_2(\mathcal{U}_i)$$

and note that since $\mathcal{C}$ is compact, there exists for each $\epsilon > 0$ a finite grid U for $\mathcal{C}$ with $\text{diam}(U) \leq \epsilon$.

We now wish to encode the dynamics as a map

$$\Phi : U \to \mathcal{P}(U)$$

and thus obtain a directed graph $(U, \widetilde{\Phi})$ which contains information about the original dynamical system. We shall construct the map Φ in two steps. First we fix a time T and compute the time-T map $\boldsymbol{\phi}_T(\mathbf{x}) := \boldsymbol{\phi}(T, \mathbf{x})$ of the system. We do this by first integrating the system numerically for selected initial-values. As we shall see later, different values of T give us quite different results. We then use the time-T map $\boldsymbol{\phi}_T$ to construct the map Φ. This can be done in numerous ways, the only condition is that the map Φ covers the time-T map, that is for each $\mathcal{U}_i$ we have

$$\boldsymbol{\phi}_T(\mathcal{U}_i) \subset \left(\bigcup_{\mathcal{S} \in \Phi(\mathcal{U}_i)} \mathcal{S} \right)^\circ . \qquad \text{(III.1)}$$

If Φ fulfills this condition we say that it is an outer approximation of $\boldsymbol{\phi}_T$. Obviously $\Phi(\mathcal{U}_i)$ should not be much larger than necessary to fulfill (III.1) if Φ is supposed to replicate the dynamics of the discrete-time system

$$\mathbf{x}_{k+1} = \boldsymbol{\phi}_T(\mathbf{x}_k) \qquad \text{(III.2)}$$

in any useful way.

Once an outer approximation Φ has been constructed we can use it to construct the graph $(U, \widetilde{\Phi})$, where $\widetilde{\Phi}$ is the set of arcs:

$$(\mathcal{U}_i, \mathcal{U}_j) \in \widetilde{\Phi} \iff \mathcal{U}_j \in \Phi(\mathcal{U}_i).$$

We want to construct a complete Lyapunov function V on the graph, i.e. a function $V : U \to [0, 1]$ such that

$$(\mathcal{U}_i, \mathcal{U}_j) \in \widetilde{\Phi} \implies V(\mathcal{U}_j) \leq V(\mathcal{U}_i). \qquad \text{(III.3)}$$

Moreover, if there is **not** a directed path in $(U, \widetilde{\Phi})$ from $\mathcal{U}_j$ to $\mathcal{U}_i$, then the inequality in (III.3) must be strict, i.e. $V(\mathcal{U}_j) < V(\mathcal{U}_i)$.

We shall now go into some details on how we construct the Lyapunov function for the graph $(U, \widetilde{\Phi})$. As a simple example consider the graph in Figure 1. We start by using the well known algorithm of Tarjan [16], which computes the strongly connected components of the graph and sorts the resulting graph of equivalence classes topologically. This delivers the graph in Figure 2 for the the graph $(U, \widetilde{\Phi})$ in Figure 1. Each vertex in this new graph represents a strongly connected component of the original graph. We refer to this new graph as the component graph of $(U, \widetilde{\Phi})$. For two vertices X and Y, $X \neq Y$, in the component graph, there is an arc from

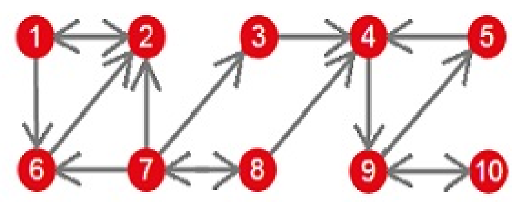

Fig. 1. An exemplary figure of a graph $(U, \widetilde{\Phi})$. Some values of Φ are $\Phi(\mathcal{U}_1) = \{\mathcal{U}_2, \mathcal{U}_6\}$, $\Phi(\mathcal{U}_4) = \{\mathcal{U}_9\}$, and $\Phi(\mathcal{U}_9) = \{\mathcal{U}_5, \mathcal{U}_{10}\}$.

Fig. 2. The graph delivered by Tarjan's algorithm when applied to the graph $(U, \widetilde{\Phi})$ from Figure 1. Its vertices represent strongly connected components in the original graph.

X to Y if and only if there exist $x \in X$ and $y \in Y$ such that there is an arc from x to y in the original graph, or equivalently $y \in \Phi(x)$.

In order to construct a complete Lyapunov function for the original graph $(U, \widetilde{\Phi})$ we first construct a Lyapunov function for its component graph. To do this consider the following partial ordering over the component graph: $X < Y$ if and only if there exists a directed path from X to Y. We assign a natural number i_A to every vertex A of the component graph, such that

$$X < Y \implies i_X < i_Y.$$

We can then define a complete Lyapunov function $\widetilde{V}$ for the component graph by assigning appropriate numerical values to $\widetilde{V}(X)$ for all vertices X. $\widetilde{V}$ can then be extended to a complete Lyapunov function V on $(U, \widetilde{\Phi})$ in the canonical way $V(x) := \widetilde{V}(X)$ for all $x \in X$.

In [14] and [2] it is shown how to compute appropriate values for $\widetilde{V}(X)$, such that it approximates a Lyapunov function for the system (III.2) arbitrary close. Due to space constraints we do not discuss the general case, but in Section V we write down the formulas for our particular example.

IV. The RBF collocation method

Meshless collocation, in particular using Radial Basis Functions (RBF), has been successfully used to construct (classical) Lyapunov functions on the basin of attraction of an exponentially stable equilibrium. We will briefly recall the method, and then describe how we have generalised the method in this paper to approximate a complete Lyapunov function.

We consider the system given by (I.1) with $\mathbf{f} \in C^\sigma$, where σ will be defined later, and assume that $\mathbf{x}_0$ is an exponentially stable equilibrium with basin of attraction $\mathcal{A}(\mathbf{x}_0)$. A classical Lyapunov function is a function $V \in C^1(\mathcal{A}(\mathbf{x}_0), \mathbb{R})$ attaining its minimum at $\mathbf{x}_0$ and with strictly negative orbital derivative

$V'(\mathbf{x}) := \nabla V(\mathbf{x}) \cdot \mathbf{f}(\mathbf{x}) < 0$ for all $\mathbf{x} \in \mathcal{A}(\mathbf{x}_0) \setminus \{\mathbf{x}_0\}$. In particular, the solution of the first-order linear PDE, with differential operator D,

$$DV(\mathbf{x}) := V'(\mathbf{x}) = \nabla V(\mathbf{x}) \cdot \mathbf{f}(\mathbf{x}) = -\|\mathbf{x} - \mathbf{x}_0\|_2^2 \quad \text{(IV.1)}$$

is such a classical Lyapunov function. The RBF method approximates the solution of (IV.1) using meshless collocation. Thereto, we fix a compact set $\mathcal{C}$ and choose a grid, i.e. a finite set of pairwise distinct points $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ in $\mathcal{C}$, not containing an equilibrium. Furthermore, we fix a Radial Basis Function $\Phi(\mathbf{x}) = \psi(\|\mathbf{x}\|_2)$. For this paper we use the family of Wendland's compactly supported RBF $\psi_{l,k}(r)$ [17] each of which is a polynomial on its support, and define $\psi(r) = \psi_{l,k}(cr)$, where $c > 0$ is a constant and $l := \lfloor \frac{n}{2} \rfloor + k + 1$. Then $k \in \mathbb{N}$, $k \geq 2$ reflects the smoothness of the function $\Phi(\mathbf{x}) = \psi_{l,k}(c\|\mathbf{x}\|_2) \in C^{2k}(\mathbb{R}^n)$. We require $\mathbf{f}$ to be smooth, namely, $\mathbf{f} \in C^\sigma$ with $\sigma = \lceil k + \frac{n+1}{2} \rceil$. In this paper we use $c = 1$ and the Wendland function

$$\psi_{4,2}(r) = (1 - r)_+^6 [35r^2 + 18r + 3],$$

where $x_+ = x$ for $x > 0$ and $x_+ = 0$ for $x \leq 0$.

The approximate solution of (IV.1) is now given by

$$V_R(\mathbf{x}) = \sum_{j=1}^{N} \alpha_j (\delta_{\mathbf{x}_j} \circ D)^{\mathbf{y}} \Phi(\mathbf{x} - \mathbf{y})$$

where $\delta_{\mathbf{x}_j} g(\mathbf{x}) = g(\mathbf{x}_j)$, D denotes the differential operator of the orbital derivative, and the superscript $\mathbf{y}$ denotes the application of the operator with respect to $\mathbf{y}$. The coefficients $\alpha_j \in \mathbb{R}$ are determined by the collocation conditions $V_R'(\mathbf{x}_i) = V'(\mathbf{x}_i) = -\|\mathbf{x}_i - \mathbf{x}_0\|^2 =: r_i$ for all $i = 1, \dots, N$ and are calculated by solving the linear system of equations $A\boldsymbol{\alpha} = \mathbf{r}$, where $\mathbf{r} = (r_1, r_2, \dots, r_N)^T$ and

$$A = \big((\delta_{\mathbf{x}_i} \circ D)^{\mathbf{x}} (\delta_{\mathbf{x}_j} \circ D)^{\mathbf{y}} \Phi(\mathbf{x} - \mathbf{y})\big)_{i,j=1,\dots,N}.$$

Note that the symmetric matrix A is positive definite when choosing a positive definite Radial Basis Function and the grid $\mathcal{X}$ as above and thus nonsingular so the equation $A\boldsymbol{\alpha} = \mathbf{r}$ always has a solution, cf. [7].

Error estimates of the form

$$\|V'(\mathbf{x}) - V_R'(\mathbf{x})\|_2 \leq C h_{\mathcal{X},\mathcal{C}}^{k-1/2} \|V\|_{W_2^{k+(n+1)/2}(\mathcal{C})}$$

for all $\mathbf{x} \in \mathcal{C}$ hold [7, Corollary 4.11], where V_R and k were defined above and

$$h_{\mathcal{X},\mathcal{C}} = \max_{\mathbf{y} \in \mathcal{C}} \min_{\mathbf{x} \in \mathcal{X}} \|\mathbf{x} - \mathbf{y}\|_2$$

is the fill distance, measuring how dense the grid points of $\mathcal{X}$ lie in $\mathcal{C}$. Note that $V \in W_2^{k+(n+1)/2}(\mathcal{C})$, denoting the Sobolev space, holds due to the smoothness assumptions on $\mathbf{f}$. Due to the error estimate, even an approximation, if the error is small enough, has negative orbital derivative, and thus is a classical Lyapunov function. However, near the equilibrium, where $V'(\mathbf{x}) = 0$, this argument fails and the approximation can have points with positive orbital derivative. Solutions to this local problem are discussed in [5], [6].

A **complete** Lyapunov function for a general system (I.1) is a continuous function $V : \mathbb{R}^n \to \mathbb{R}$ that is strictly decreasing along all solution trajectories not in the chain recurrent set of the flow. If $V \in C^1$ this implies that $V'(\mathbf{x}) = \nabla V(\mathbf{x}) \cdot \mathbf{f}(\mathbf{x}) < 0$ for all $\mathbf{x}$ **not** in the chain recurrent set. In order to approximate a function with these properties we consider the linear first-order PDE

$$V'(\mathbf{x}) := \nabla V(\mathbf{x}) \cdot \mathbf{f}(\mathbf{x}) = -\|\mathbf{f}(\mathbf{x})\|_2 \quad \text{(IV.2)}$$

on a compact set $\mathcal{C} \subset \mathbb{R}^n$. Note that this equation does not necessarily have a solution. Indeed, if $\mathcal{C}$ contains e.g. a periodic orbit, then (IV.2) cannot have a solution. To see this, assume the system has a periodic orbit given by the periodic solution $t \mapsto \boldsymbol{\phi}(t, \mathbf{p})$, where $\boldsymbol{\phi}(T, \mathbf{p}) = \mathbf{p}$ with period $T > 0$, $\boldsymbol{\phi}(\mathbb{R}, \mathbf{p}) \subset \mathcal{C}$. Then we have

$$\begin{aligned} 0 &= V(\boldsymbol{\phi}(T, \mathbf{p})) - V(\mathbf{p}) \\ &= \int_0^T V'(\boldsymbol{\phi}(t, \mathbf{p}))\, dt \\ &= -\int_0^T \|\mathbf{f}(\boldsymbol{\phi}(t, \mathbf{p}))\|_2\, dt \\ &< 0 \end{aligned}$$

as $\mathbf{f}(\mathbf{x}) \neq 0$ for all points $\mathbf{x}$ on the periodic orbit, which is a contradiction.

Nevertheless, we follow the approach above, define a finite set of pairwise distinct points $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\} \subset \mathcal{C}$, none of whom is an equilibrium, and solve the linear equation $A\boldsymbol{\alpha} = \mathbf{r}$ with A as above and $\mathbf{r} = (r_1, r_2, \dots, r_N)^T$ defined by $r_i = -\|\mathbf{f}(\mathbf{x}_i)\|_2$ for $i = 1, 2, \dots, N$. Even if the problem (IV.2) has no solution, this collocation problem always has a solution. Indeed, the proof is the same as above, see [7], as the operator D is the same. The matrix A is positive definite since the functionals $\delta_{\mathbf{x}_j} \circ D$, $j = 1, \dots, N$ are linearly independent; this follows from the fact that all $\mathbf{x}_j$ are pairwise distinct and none of them is an equilibrium point, which are the only singular points of D in the sense of [7, Definition 3.2]. We expect that the approximation will not have negative orbital derivative on and close to the chain recurrent set, such as periodic orbits, but we anticipate to obtain an approximation with negative orbital derivative in the gradient-like part. The only certainty, however, is that the solution V_R cannot have a negative orbital derivative at all points of a transitive component of the chain recurrent set, which is fully contained in $\mathcal{C}$.

V. Case study: the Van Der Pol oscillator

In order to numerically investigate our approach we apply it to the system of equations

$$\begin{aligned} \dot{x} &= y, \\ \dot{y} &= (1 - x^2)y - x, \end{aligned} \quad \text{(V.1)}$$

known as the van der Pol equations and we consider it on the set $\mathcal{C} = [-4, 4] \times [-4, 4]$. Then $\mathcal{C}$ contains an unstable equilibrium at the origin and an asymptotically stable periodic orbit that encircles the origin. We compute an approximation to a complete Lyapunov function for the system, both with the combinatorial method and the RBF method, and then we verify these approximations with the CPA method.

First we apply the combinatorial method as presented in [2], [14]. We subdivide the interval $[-4, 4]$ on both axes into 128 even intervals to obtain a grid of $16,384$ elements. For every grid element $\mathcal{U}_i$ we denote its midpoint by $\mathbf{u}_i$.

To determine the map Φ we need to compute $\boldsymbol{\phi}_T(\mathbf{x})$ for a fixed $T > 0$ and for some $\mathbf{x} \in \mathcal{C}$. We did this by numerical integration using the Runge-Kutta method RK4 with a fixed time-step of 0.01. Second, we note that for every T the time-T map $\boldsymbol{\phi}_T$ is locally Lipschitz, i.e. there exists a constant $L > 0$ such that

$$\|\boldsymbol{\phi}_T(\mathbf{x}) - \boldsymbol{\phi}_T(\mathbf{y})\|_\infty \leq L\|\mathbf{x} - \mathbf{y}\|_\infty. \tag{V.2}$$

Therefore we can construct an outer approximation Φ of $\boldsymbol{\phi}_T$ by insisting that $\mathcal{U}_j \in \Phi(\mathcal{U}_i)$ if and only if

$$\mathcal{U}_j \cap \left\{ \mathbf{x} \in \mathcal{C} \,:\, \frac{\|\mathbf{x} - \boldsymbol{\phi}_T(\mathbf{u}_i)\|_\infty}{\operatorname{diam}_\infty(\mathcal{U}_i)} \leq \frac{1}{2}L \right\} \neq \varnothing$$

where $\operatorname{diam}_\infty(\mathcal{U}_i) = 8/128 = 1/16$.

To compute a Lipschitz constant L for $\boldsymbol{\phi}_T$ one might be tempted to use the usual estimate from Gronwall's lemma

$$\|\boldsymbol{\phi}_T(\boldsymbol{\xi}) - \boldsymbol{\phi}_T(\boldsymbol{\eta})\|_\infty \leq \|\boldsymbol{\xi} - \boldsymbol{\eta}\|_\infty e^{L^* T}$$

where

$$L^* := \sup_{\substack{t\in[0,T] \\ \boldsymbol{\xi},\boldsymbol{\eta}\in\mathcal{C} \\ \boldsymbol{\xi}\neq\boldsymbol{\eta}}} \frac{\|\boldsymbol{\phi}(t,\boldsymbol{\xi}) - \boldsymbol{\phi}(t,\boldsymbol{\eta})\|_\infty}{\|\boldsymbol{\xi} - \boldsymbol{\eta}\|_\infty},$$

and then set $L := e^{L^* T}$. This estimate, however, is so conservative that the outer approximation Φ is not able to encode the dynamics of the system (III.2) in a useful way for our numerical computations. Indeed, our computations for several different T with such an L always resulted in the component graph consisting of one vertex and the computed combinatorial Lyapunov function being constant on $\mathcal{C}$.

We thus resorted to estimating the constant L numerically. By computing

$$\frac{\|\boldsymbol{\phi}_T(\mathbf{x}) - \boldsymbol{\phi}_T(\mathbf{y})\|_\infty}{\|\mathbf{x} - \mathbf{y}\|_\infty}$$

directly for $5 \cdot 10^5$ pairs of midpoints $\mathbf{x}$ and $\mathbf{y}$ of adjacent cells we obtained the upper bound $L = 1.78$ for $T = 0.2$ and $L = 2.55$ for $T = 1$. With these constants $\Phi(\mathcal{U}_i) \subset U$ contains 4 to 9 cells when $T = 0.2$ and 9 to 16 cells when $T = 1$.

Note that by using these values for L instead of a theoretical upper bound we cannot be sure that Φ actually covers the time-T map $\boldsymbol{\phi}_T$ in the sense of (III.1). We can only assume that (V.2) should hold for most $\mathcal{U}_i$.

Since $\mathcal{C}$ is not an invariant set as Figure 3 obtained from the software GAIO [4] indicates, we have to modify our map slightly to mend this. Those midpoints $\mathbf{u}_i$ that are not mapped by $\boldsymbol{\phi}_T$ into $\mathcal{C}$ are iterated repeatedly under $\boldsymbol{\phi}_T$ until we have obtained a point in $\mathcal{C}$. That is, if $\boldsymbol{\phi}_T(\mathbf{u}_i) \notin \mathcal{C}$, we set $\mathcal{U}_j \in \Phi(\mathcal{U}_i)$, where $\boldsymbol{\phi}(rT, \mathbf{u}_i) \notin \mathcal{C}$ for $r = 1, 2, \ldots, s-1$ and $\mathbf{y} := \boldsymbol{\phi}(sT, \mathbf{u}_i) \in \mathcal{C}$, whenever

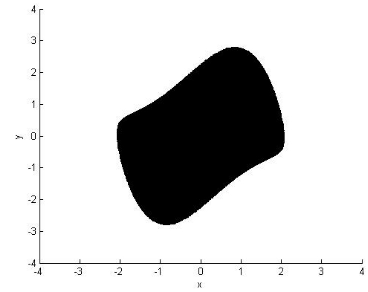

Fig. 3. The relative attractor for the system (V.1) in $\mathcal{C}$ computed with GAIO. This set is a global attractor of the system.

$$\mathcal{U}_j \cap \left\{ \mathbf{x} \in \mathcal{C} \,:\, \frac{\|\mathbf{x} - \mathbf{y}\|_\infty}{\operatorname{diam}_\infty(\mathcal{U}_i)} \leq \frac{1}{2}L \right\} \neq \varnothing.$$

This is guaranteed to work for our system because the periodic orbit in $\mathcal{C}$ is also a global attractor for the system so all solutions end up in $\mathcal{C}$.

To compute a Lyapunov function V_G for the graph $(U, \widetilde{\Phi})$ we followed the algorithm in [2]. There is only one attractor-repeller pair which we have to consider. The attractor A is depicted in Figure 6 for time-step $T = 0.2$ and in Figure 8 for time-step $T = 1$. In both cases the dual repeller A^* consist of the four cells $\mathcal{U}_j \in U$ containing the origin. We define the functions $v, v^* : U \to [0, 1]$ by

$$v(\mathcal{U}_i) := \frac{\min_{\mathcal{U}_a \in A} \|\mathbf{u}_i - \mathbf{u}_a\|_2}{\min_{\mathcal{U}_a \in A} \|\mathbf{u}_i - \mathbf{u}_a\|_2 + \min_{\mathcal{U}_{a^*} \in A^*} \|\mathbf{u}_i - \mathbf{u}_{a^*}\|_2}$$

and

$$v^*(\mathcal{U}_i) := \max_{\mathcal{U}_j \in \Gamma_+(\mathcal{U}_i)} v(\mathcal{U}_i),$$

where $\Gamma^+(\mathcal{U}_i) \subset U$ is the forward image of $\mathcal{U}_i$ under Φ, containing $\mathcal{U}_i$ and all $\mathcal{U}_k \in U$ such that there exists a directed path from $\mathcal{U}_i$ to $\mathcal{U}_k$ in the graph $(U, \widetilde{\Phi})$. Then, by Theorem 3.3 in [2], the function

$$V_G(\mathcal{U}_i) = \begin{cases} 0, & \text{if } \mathcal{U}_i \in A, \\ 1, & \text{if } \mathcal{U}_i \in A^*, \\ \frac{1}{2}v^*(\mathcal{U}_i) + \frac{1}{2}\max\limits_{\mathcal{U}_j \in \Phi(\mathcal{U}_i)} V_G(\mathcal{U}_j), & \text{otherwise.} \end{cases}$$

is a Lyapunov function for the graph $(U, \widetilde{\Phi})$ that approximates a continuous Lyapunov function for (III.2) arbitrarily close for a grid U with a small enough diameter. Note that the recursively defined function V_G is in fact well defined on the grounds of Conley's fundamental theorem, cf. Theorem 1.5 in [14].

Now we come to the CPA approximation. A part of the triangulation which we use for the CPA method is depicted in Figure 5. The vertices of the triangles are (x, y) where

$$x = 4 \cdot \frac{i - \frac{1}{2}}{64} \quad \text{and} \quad y = 4 \cdot \frac{j - \frac{1}{2}}{64}, \; i, j = -63, -62, \ldots, 64.$$

These triangles cover the square $[-4 \cdot \frac{63.5}{64}, 4 \cdot \frac{63.5}{64}]^2 \subset \mathcal{C}$. We assign values to the CPA Lyapunov function V_C from

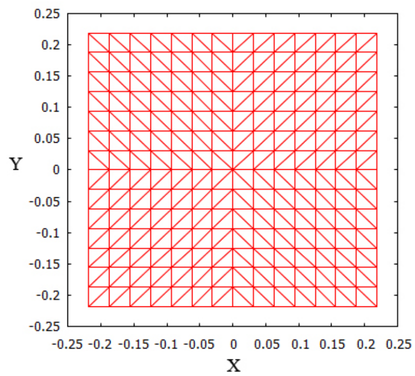

Fig. 4. The triangulation we use in this paper. The figure shows a part of the triangulation near the origin. The triangulation is extended in the obvious way to cover the domain $[-4 \cdot b, 4 \cdot b]^2$, where $b = \frac{127}{128}$, for all examples in this paper. It consists of $2 \cdot (2 \cdot 127)^2 = 129,032$ triangles. The factor b is explained in the text.

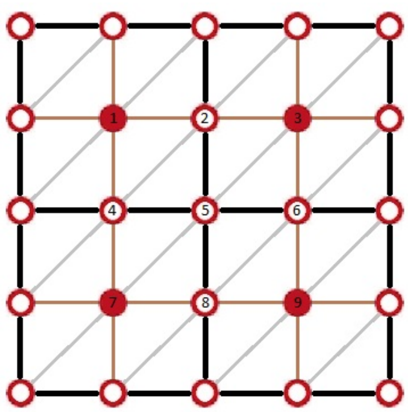

Fig. 5. An exemplary picture of four cells $\mathcal{U}_i$ of the grid for the combinatorial method and 25 vertices of the triangulation for the CPA method. Some vertices are at the midpoints of the cells and some are at the boundaries. The combinatorial method delivers values for V_G at the midpoints (the filled dots). In the text we explain how we assign values to V_C at the vertices.

V_G as follows. If $(x, y) = \mathbf{u}_i$ is the centre of an $\mathcal{U}_i$, we set $V_C(x, y) := V_G(\mathcal{U}_i)$. For other vertices (x, y) we assign the average of the closest $V_G(\mathcal{U}_i)$ to $V_C(x, y)$. This is best explained graphically. In Figure 5 four cells $\mathcal{U}_i$ and 25 vertices are depicted. The vertices numbered 1, 3, 7, and 9 are the centres of cells $\mathcal{U}_i$ and we assign to V_C at these vertices the values of V_G at the cell. At vertex 2, V_C is assigned the average of the values of G at vertices 1 and 3. At vertex 4, V_C is assigned the average of the values of G at vertices 1 and 7, and for the vertices 6 and 8 we proceed identically. At vertex 5, V_C is assigned the average of the values of V_G at vertices 1, 3, 7, and 9. Note that a finer triangulation for the CPA would deliver the same results due to the Lyapunov function constructed by the combinatorial method and the way how we assign values to the CPA Lyapunov function.

We now set the time step $T = 0.2$ and calculate an approximation V_G to a complete Lyapunov function for the

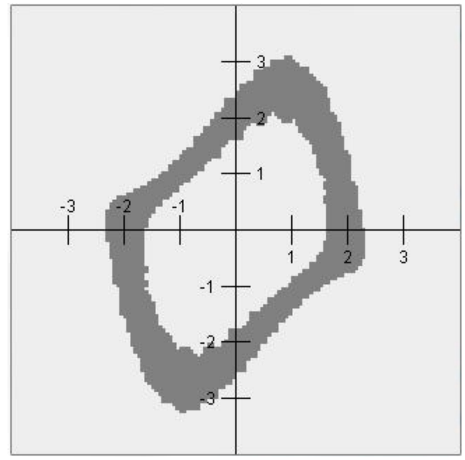

Fig. 6. The attractor of ϕ_T with $T = 0.2$ for the system (V.1) computed with the combinatorial algorithm.

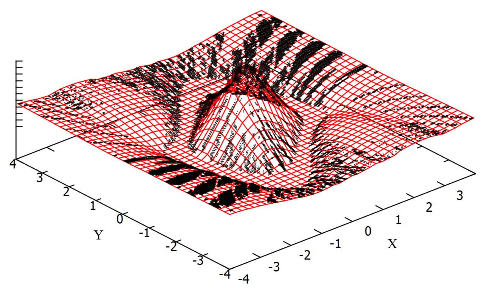

Fig. 7. The CPA approximation V_C to a complete Lyapunov function for ϕ_T with $T = 0.2$ for the system (V.1). The area where the orbital derivative is positive for the continuous system (V.1) is painted black. The orbital derivative is positive in 27068 of 129032 triangles or 20.98%.

time-T map ϕ_T. In Figure 6 we see the attractor computed by the combinatorial method. We used the CPA method from Section II to investigate where the orbital derivative of the CPA approximation V_C of V_G is positive for the continuous system (V.1). The results are depicted in Figure 7. Note that a complete Lyapunov function for the time-T mapping ϕ_T is not necessarily a complete Lyapunov functions for the continuous system [2], and as seen in Figure 7 the orbital derivative is positive in a rather large subset of $\mathcal{C}$. We did the same calculations for the time step $T = 1$. The results are depicted in Figure 8. As expected, the approximation to the attractor is now better, see Figure 8, and the approximation to a complete Lyapunov function for the continuous system (I.1) is poorer, cf. Figure 9. The computation time of the complete Lyapunov function for $T = 0.2$ was 27 sec. and for $T = 1$ it was 55 sec. on a modern PC. The verification by the CPA methods takes 0.025 sec.

We now turn to the RBF method. We computed V_R for three different grids $\mathcal{X}$, one with $11 \times 11 = 121$ grid points, one with $21 \times 21 = 441$, and one with $41 \times 41 = 1681$ grid points. The grid points have coordinates

$$(x, y) = \frac{4}{q}(i, j), \quad i, j = -q, -q + 1, \dots, q,$$

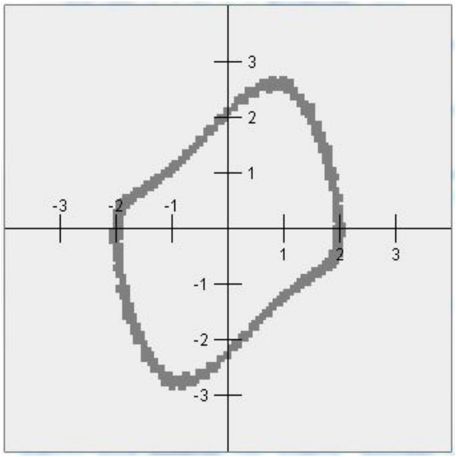

Fig. 8. The attractor of ϕ_T with $T = 1$ for the system (V.1) computed with the combinatorial algorithm.

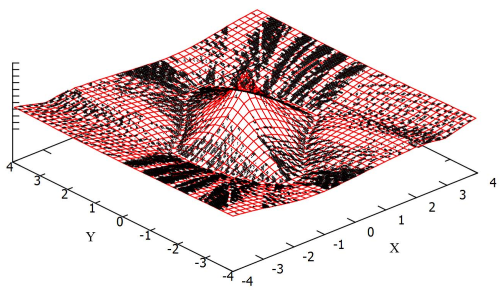

Fig. 9. The CPA approximation V_C to a complete Lyapunov function for ϕ_T with $T = 1$ for the system (V.1). The area where the orbital derivative is positive for the continuous system (V.1) is painted black. The orbital derivative is positive in 37222 of 129032 triangles or 28.8%.

where we used $q = 5$, $q = 10$, and $q = 20$ respectively. The most laborious computation with $q = 20$, i.e. 1681 grid points, takes 6 sec. on a modern PC and the verification by the CPA method takes 0.025 sec. In Figures 10 to 15 the results are shown. The orbital derivative is negative in a much larger area than with the combinatorial method. This is to be expected because the combinatorial method is computing an approximation to a complete Lyapunov function for the time-T mapping, $T = 0.2$ and $T = 1$, which is not necessarily a complete Lyapunov function for the continuous system. The RBF method, however, is not designed to identify attractors and thus does not produce a Lyapunov function with a vanishing orbital derivative on the attractor.

VI. Summary and Outlook

The complete Lyapunov functions computed by the combinatorial method uses a discrete-time approximation (III.2) to the continuous-time system (I.1). Our numerical results indicate that it is a rather poor approximation to a complete Lyapunov function of the continuous time van der Pol oscillator. There are at least two different paths that one might follow to proceed. First, one could use a variation of the CPA method for discrete-time systems as in [9] to

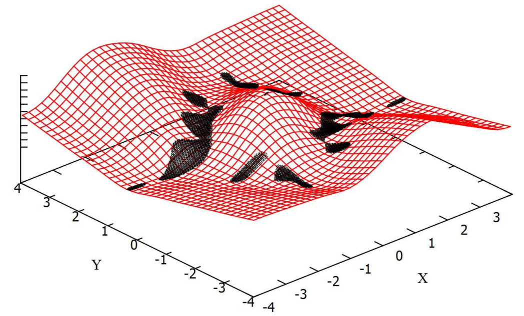

Fig. 10. The CPA approximation V_C to a complete Lyapunov function computed by the RBF collocation method using an 11×11 grid. The area where the orbital derivative is positive for the continuous system (V.1) is painted black. The orbital derivative is positive in 9434 of 129032 triangles or 7.3%.

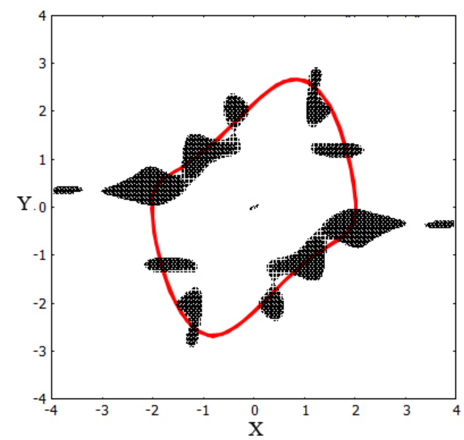

Fig. 11. The periodic orbit of the system (V.1) (red) and the area where the orbital derivative of the CPA approximation to the complete Lyapunov function computed by the RBF method using an 11×11 grid is positive (black). The orbital derivative is positive in 9434 of 129032 triangles or 7.3%.

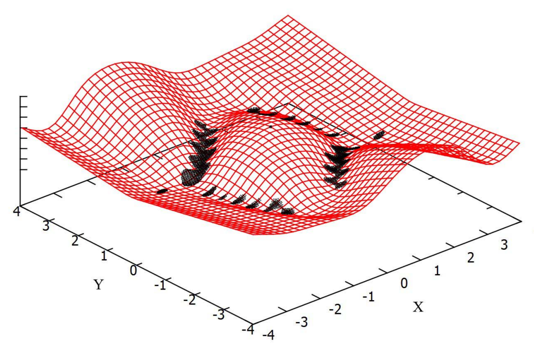

Fig. 12. The CPA approximation V_C to a complete Lyapunov function computed by the RBF collocation method using an 21×21 grid. The area where the orbital derivative is positive for the continuous system (V.1) is painted black. The orbital derivative is positive in 4592 of 129032 triangles or 3.6%.

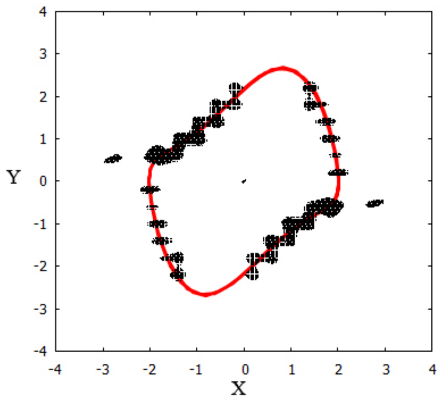

Fig. 13. The periodic orbit of the system (V.1) (red) and the area where the orbital derivative of the CPA approximation to the complete Lyapunov function computed by the RBF method using an 21×21 grid is positive (black). The orbital derivative is positive in 4592 of 129032 triangles or 3.6%.

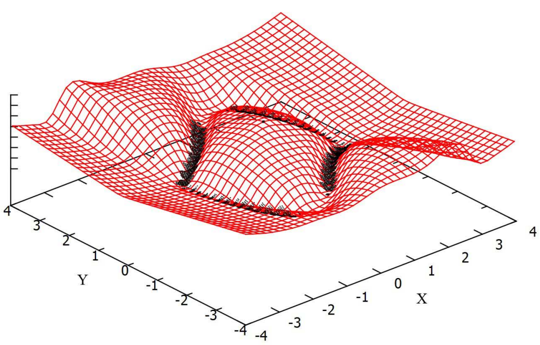

Fig. 14. The CPA approximation V_C to a complete Lyapunov function computed by the RBF collocation method using an 41×41 grid. The area where the orbital derivative is positive for the continuous system (V.1) is painted black. The orbital derivative is positive in 2888 of 129032 triangles or 2.2%.

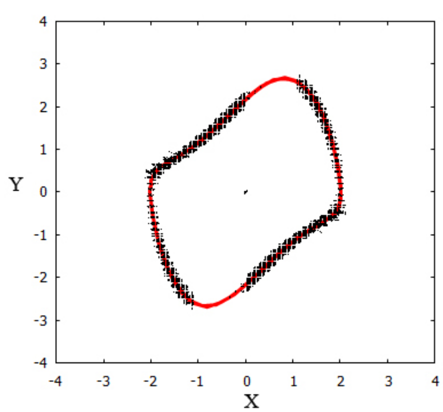

Fig. 15. The periodic orbit of the system (V.1) (red) and the area where the orbital derivative of the CPA approximation to the complete Lyapunov function computed by the RBF method using an 41×41 grid is positive (black). The orbital derivative is positive in 2888 of 129032 triangles or 2.2%.

verify the quality of the Lyapunov function for the discrete-time system (III.2) directly. Second, one could use Lemma 7.7 in [14] or Theorem 1.1 in [15] to compute a complete Lyapunov function for the continuous-time system (I.1) from the one computed for its discrete-time approximation (III.2) and then verify this function by the CPA method. We intend to follow both paths in the future.

For the van der Pol oscillator, the combinatorial method works very well to identify the attractor, i.e. in this case the periodic orbit; the constructed complete Lyapunov function is indeed constant on the periodic orbit. The RBF method, on the other hand, fails to produce a function with vanishing orbital derivative on the periodic orbit or the equilibrium, but works very well away from it in producing a function with strictly negative orbital derivative. This suggests that a suitable combination of both methods is a promising approach to construct a complete Lyapunov function. The CPA verification method thus was the tool to rigorously analyze the constructed functions and identify areas where they need improvement.

References

[1] R. Baier, L. Grüne and S. Hafstein, Linear programming based Lyapunov function computation for differential inclusions, Discrete Contin. Dynam. Systems - B, **17** (2012), 33–56.

[2] H. Ban and W. Kalies, A computational approach to Conley's decomposition theorem, J. Comput. Nonlinear Dynam., **1-4** (2006), 312–319.

[3] C. Conley "Isolated invariant sets and the Morse index", Vol. 38 of CBMS, American Math. Soc., 1978.

[4] M. Dellnitz, G. Froyland and O. Junge, The Algorithms Behind GAIO Set Oriented Numerical Methods for Dynamical Systems, in Ergodic Theory, Analysis, and Efficient Simulation of Dynamical Systems, Ed. B. Fiedler, Springer, (2001) 145–174

[5] P. Giesl, "Construction of Global Lyapunov Functions Using Radial Basis Functions", Lecture Notes in Mathematics **1904**, Springer, 2007.

[6] P. Giesl, Construction of a local and global Lyapunov function using radial basis functions. *IMA J. Appl. Math.* **73-5** (2008), 782–802.

[7] P. Giesl & H. Wendland, Meshless Collocation: Error Estimates with Application to Dynamical Systems. *SIAM J. Numer. Anal.* **45-4** (2007), 1723–1741.

[8] P. Giesl and S. Hafstein, Revised CPA method to compute Lyapunov functions for nonlinear systems, J. Math. Anal. Appl., **410** (2014), 292–306.

[9] P. Giesl and S. Hafstein, Computation of Lyapunov functions for nonlinear discrete time systems by linear programming, J. Differ. Equ. Appl., **20-4**, (2014), pp. 610-640.

[10] P. Giesl and S. Hafstein, Implementation of a Simplicial Fan Triangulation for the CPA Method to compute Lyapunov Functions, To appear at the 2014 American Control Conference.

[11] S. Hafstein, "An algorithm for constructing Lyapunov functions", Electron. J. Differential Equ. Monogr., **8** (2007).

[12] S. Hafstein, H. Li, and C. Kellett, Continuous and Piecewise Affine Lyapunov Functions using the Yoshizawa Construction, To appear at the 2014 American Control Conference.

[13] M. Hurley, Lyapunov functions and attractors in arbitrary metric spaces, Proc. Amer. Math. Soc. **126** (1998), 245–256.

[14] W. Kalies, K. Mischaikow and R. VanderVorst, An algorithmic approach to chain recurrence, Found. Comput. Math., **5-4** (2005), 409–449.

[15] M. Patrão, Existence of complete Lyapunov functions for semiflows on separable metric spaces, Far East Journal of Dynamical Systems, **17-1**, (2011), 49–54.

[16] R. Tarjan, Depth-first search and linear graph algorithms, SIAM J. Comput. **1-2** (1972), 146-160.

[17] H. Wendland: Error estimates for interpolation by compactly supported radial basis functions of minimal degree. *J. Approx. Theory* **93** (1998), 258–272.