2019 18th European Control Conference (ECC)
Napoli, Italy, June 25-28, 2019

# Improved estimation of the chain-recurrent set

Carlos Argáez[1] and Peter Giesl[2] and Sigurdur Freyr Hafstein[3]

*Abstract*— **When studying the behaviour of dynamical systems, one particular goal is to find and isolate the periodic solutions and the equilibria. They are a subset of the chain-recurrent set of the dynamical system. In recent work, many improvements have been achieved in computing an approximation of a complete Lyapunov function of a given dynamical system and thus to identify the chain-recurrent set. A weak point in this approach, however, has been an over-estimation of the chain-recurrent set. In this work, we introduce a heuristic algorithm that reduces the overestimation in a simple and efficient way. Furthermore, a new and improved grid to evaluate the complete Lyapunov function is introduced to avoid unevaluated regions in the domain of the function.**

## I. INTRODUCTION

Let us assume that we have a dynamical system expressed by a general autonomous ordinary differential equation (ODE),

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

where $\mathbf{x} \in \mathbb{R}^n$, $n \in \mathbb{N}$.

In general terms we are interested in describing the qualitative behaviour of system (1) and specially to find attractors and repellers.

There are many methods to analyze the general behaviour of dynamical systems: from direct simulations of solutions with many different initial conditions, to computation of invariant manifolds which form the boundaries of attractors' basins of attraction [17]. Another group of techniques include set oriented methods [9] or the cell mapping approach [13]. All these methods require large computational effort.

Aleksandr Lyapunov proposed [18] in 1893 a new way to studying the stability of an equilibrium point, or more generally an attractor, in which the solution to the differential equation is not required. For this, he introduced an auxiliary scalar-valued function whose domain is a subset of the state-space, and which is strictly decreasing along all solution trajectories in a neighbourhood of an attractor, such as an asymptotically stable equilibrium point or periodic orbit. Nowadays, such a function is known as a strict *Lyapunov function* in his honor. Furthermore, a Lyapunov function attains its minimum on the attractor. As a consequence, solutions that start close to the latter converge to it. This is the classical definition of a strict Lyapunov function. However, this function is only defined in the neighbourhood of one attractor. A natural extension is a function defined on the whole state space, a *complete Lyapunov function*, which was introduced in [7], [8], [14], [15].

A complete Lyapunov function characterizes the complete qualitative behaviour of the dynamical system on the whole phase space and not just in a neighbourhood of one particular attractor. It allows dividing the state-space into two disjoint areas: The gradient-like flow, where the system flows through, and the chain-recurrent set, where infinitesimal perturbations can make the system recurrent. On these two areas, the system behaves in fundamentally different ways.

The first mathematical existence proof was given by Conley, who proved the existence of complete Lyapunov functions [7] for a dynamical system defined on a compact metric space. Hurley [15] extended these results to separable metric spaces. For some very recent results see also [10], [6].

In this paper we use an algorithm to compute a complete Lyapunov function which was introduced and developed in [1], [4], [5], [12], [3] and which has proven to be computationally efficient. It is a modification of a method to compute classical Lyapunov functions for one stable equilibrium using Radial Basis Functions [11].

The general idea is to approximate a "solution" to the ill-posed problem $V'(\mathbf{x}) = -1$, where $V'(\mathbf{x}) = \nabla V(\mathbf{x}) \cdot \mathbf{f}(\mathbf{x})$ is the derivative along solutions of the ODE, i.e. the orbital derivative. A function $v$ is computed using Radial Basis Functions, a mesh-free collocation technique, such that $v'(\mathbf{x}) = -1$ is fulfilled at all points $\mathbf{x}$ in a finite set of collocation points $X$.

The discretized problem of computing $v$ is well-posed and possesses a unique solution. However, the computed function $v$ cannot fulfill the PDE at all points of the chain-recurrent set, such as an equilibrium or a periodic orbit. This is the key component of our general algorithms to locate the chain-recurrent sets; we use the area where $v$ fails to fulfill the PDE to determine the chain-recurrent set. Furthermore, our method has the advantage of not being overly computationally demanding.

## II. ALGORITHM

We base our method to compute complete Lyapunov functions on the algorithms described in [1], [2], [4], [5], [12], [3]. In this paper, however, we extend the methodology to reduce the overestimation of the chain-recurrent set.

Like in the methodology introduced in [4], we also normalize the right-hand side of the ODE. This means that the original dynamical system (1) gets substituted by

$$\dot{\mathbf{x}} = \hat{\mathbf{f}}(\mathbf{x}), \quad \text{where} \quad \hat{\mathbf{f}}(\mathbf{x}) = \frac{\mathbf{f}(\mathbf{x})}{\sqrt{\delta^2 + \|\mathbf{f}(\mathbf{x})\|^2}}, \tag{2}$$

*The first author in this paper is supported by the Icelandic Research Fund (Rannís) grant number 163074-052, Complete Lyapunov functions: Efficient numerical computation.

[1] Carlos Argáez is with the Science Institute, University of Iceland, Dunhagi 5, 107 Reykjavík, Iceland carlos@hi.is

[2] Peter Giesl is with the Department of Mathematics, University of Sussex, United Kingdom P.A.Giesl@sussex.ac.uk

[3] Sigurdur Freyr Hafstein is with the Science Institute, University of Iceland, Dunhagi 5, 107 Reykjavík, Iceland shafstein@hi.is

978-3-907144-01-5 ©2018 EUCA

with a small parameter $\delta > 0$ and where $\|\cdot\|$ denotes the Euclidean norm. The reason for this is explained in detail in [4]. In short, the main motivation is that the systems (1) and (2) have the same trajectories, but the speed of the solutions to (2) is nearly uniform. This was shown to deliver superior results.

### A. *MESH-FREE COLLOCATION*

Mesh-free collocation methods are a powerful methodology for solving generalized interpolation problems. Radial Basis Functions (RBF) [11] can serve as basis functions for such methods. They are particular real-valued functions, whose evaluation depends only on the distance from the origin; Gaussians, multiquadrics and Wendland functions are examples of such functions.

Mesh-free collocation enables us to use locally a high resolution of collocation points to solve PDE's. Scattered points can be added or removed to improve the approximation and no triangulation of the phase space is needed. In our work we use Wendland functions as RBF. These are compactly supported functions [19], that have been intensively used for computing Lyapunov functions. Furthermore, they are positive definite functions that are polynomials on their compact support and the corresponding Reproducing Kernel Hilbert Space is norm-equivalent to a Sobolev space[11]. Note that in the context of RBF positive definite refers to the matrix $\left(\psi(\|\mathbf{x}_i - \mathbf{x}_j\|)\right)_{i,j}$ being positive definite for $X = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$, where $\mathbf{x}_i \neq \mathbf{x}_j$ if $i \neq j$.

The Wendland functions have the general form $\psi(\mathbf{x}) := \psi_{\ell,k}(c\|\mathbf{x}\|)$, where $c > 0$ and $k \in \mathbb{N}$ is a smoothness parameter. The parameter $l$ is fixed as $\ell = \lfloor \frac{n}{2} \rfloor + k + 1$. The Reproducing Kernel Hilbert Space corresponding to $\psi_{\ell,k}$ contains the same functions as the Sobolev space $W_2^{k+(n+1)/2}(\mathbb{R}^n)$ and the spaces are norm equivalent. The Wendland functions are defined by the following recursion:
For $\ell \in \mathbb{N}$ and $k \in \mathbb{N}_0$, we define

$$\begin{aligned} \psi_{\ell,0}(r) &= (1-r)_+^{\ell}, \\ \psi_{\ell,k+1}(r) &= \int_r^1 t\,\psi_{\ell,k}(t)\,\mathrm{d}t \end{aligned} \tag{3}$$

for $r \in \mathbb{R}_0^+$, where $x_+ = x$ for $x \geq 0$ and $x_+ = 0$ for $x < 0$.

The construction of $X$ can be done in different ways. We build our collocation points $X = \{\mathbf{x}_1, \dots, \mathbf{x}_N\} \subset \mathbb{R}^n$, as a subset of a hexagonal grid with fineness-parameter $\alpha_{\text{Hexa-basis}} \in \mathbb{R}^+$ which is constructed according to the next equation:

$$\left\{ \alpha_{\text{Hexa-basis}} \sum_{k=1}^{n} i_k \omega_k : i_k \in \mathbb{Z} \right\}, \text{ where} \tag{4}$$

$$\begin{aligned} \omega_1 &= (2\varepsilon_1, 0, 0, \dots, 0) \\ \omega_2 &= (\varepsilon_1, 3\varepsilon_2, 0, \dots, 0) \\ \vdots \quad & \quad \vdots \\ \omega_n &= (\varepsilon_1, \varepsilon_2, \varepsilon_3, \dots, (n+1)\varepsilon_n) \text{ and} \\ \varepsilon_k &= \sqrt{\frac{1}{2k(k+1)}}, \quad k \in \mathbb{N}. \end{aligned}$$

This grid has been shown to optimally balance the opposing aims of a small fill distance to give good error estimates and a large separation distance of collocation points to keep the condition numbers of the collocation matrices as small as possible. As a consequence of this choice, the condition numbers of the collocation matrices do not become overly large [16].

Since at all equilibrium points $\mathbf{x}$ we have $\mathbf{f}(\mathbf{x}) = \mathbf{0}$, we need to remove all such points from the set of the collocation points $X$, since otherwise the collocation matrix would be singular.

The approximated $v$ is then given by the function that satisfies the PDE $v'(\mathbf{x}) = -1$ at all collocation points and it is norm minimal in the corresponding Reproducing Kernel Hilbert space. Practically, we find $v$ by solving a system of $N$ linear equations, where $N$ is the number of collocation points.

To determine the chain-recurrent set, we need to establish at which collocation points a function $V$ fulfilling $V'(\mathbf{x}) = -1$ is properly approximated by $v$. However, all collocation points satisfy the PDE by construction. Therefore, we need a special evaluation grid $Y_{\mathbf{x}_j}$ around each collocation point $\mathbf{x}_j$. Such an evaluation grid can be constructed in many different ways, but keep in mind that it is important to be able to correlate each evaluation point to a particular collocation point.

Let us review the different approaches for the evaluation grid so far. In [1], we used points distributed inhomogeneously along two circumferences around each collocation point. In particular, in $\mathbb{R}^2$, for a collocation point $\mathbf{x}_j \in \mathbb{R}^2$, we used the following points $Y_{\mathbf{x}_j}$:

$$\begin{aligned} &\mathbf{x}_j \pm r\xi\alpha(\cos(\theta), -\sin(\theta)) \\ &\mathbf{x}_j \pm r\xi\alpha(\sin(\theta), \cos(\theta)) \end{aligned} \tag{5}$$

where $\xi = 1$ for the outer circumference and $\xi = 1/2$ for the inner one, $r > 0$, is a parameter to scale the circumferences and $\theta = 0, 11.25, 22.5, 45, 56.25, 67.5, 75$ and $105$ in units of sexagesimal degrees. In paper [4], we used an improved version of this grid, in which the points were distributed homogeneously.

$$\begin{aligned} Y_{\mathbf{x}_j} = &\{\mathbf{x}_j + r\alpha_{\text{Hexa-basis}}(\cos(\theta), \sin(\theta))\} \\ &\cup \{\mathbf{x}_j + \tfrac{r}{2}\alpha_{\text{Hexa-basis}}(\cos(\theta), \sin(\theta))\} \\ &\text{where } \theta \in \{2\pi/32, 4\pi/32, 6\pi/32, \dots, 2\pi\} \end{aligned} \tag{6}$$

as before $r > 0$ is a parameter and $\alpha_{\text{Hexa-basis}}$ is the parameter used to build the hexagonal grid defined above.

The grid (6) can be generalized to higher dimension. However, the growth of the number of evaluation points is exponential. To avoid this we resorted to a different version of the evaluation grid in [5], [3], where we used evaluation points that were aligned along the flow of the ODE system.

$$\begin{aligned} &Y_{\mathbf{x}_j} \\ &= \left\{ \mathbf{x}_j \pm \frac{r \cdot k \cdot \alpha_{\text{Hexa-basis}} \cdot \hat{\mathbf{f}}(\mathbf{x}_j)}{m} : k \in \{1, \dots, m\} \right\} \end{aligned} \tag{7}$$

Like before, $\alpha_{\text{Hexa-basis}}$ is the parameter used to build the hexagonal grid defined above, $r \in (0,1)$ is the ratio up to which the evaluation points will be placed and $m \in \mathbb{N}$ denotes the number of points in the evaluation grid that will be placed on both sides of the collocation points aligned to the flow.

These grids are shown in Fig. 1.

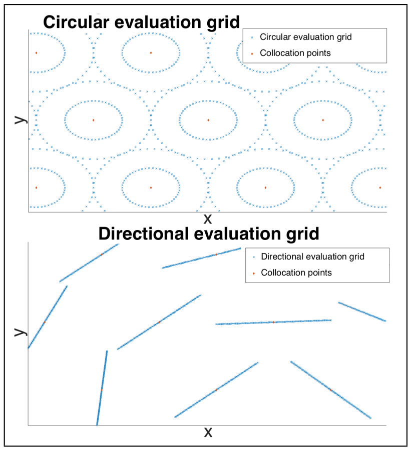

Fig. 1. Upper figure shows the circular grid points (blue) around each collocation points (red) according to 6. Lower figure shows the directional grid points (blue) around each collocation point according to 7.

It can be clearly seen in Fig. 1 that there are several empty regions that will not be evaluated, i.e. in between the circumferences or between the direction of the flow. Even if we increase the amount of evaluation points in the circumferences or in the direction of the flow, that does not solve the problem, for we would be placing more points in the same regions of the phase space. In order to reduce the error in the computation of the approximation to the complete Lyapunov function, it is necessary to increase the amount of points in the collocation grid. That will increase, consequently, the amount of evaluation points.

We propose a new evaluation grid that is built using a reduced (in area) sized version of (4) around each collocation point of (4). The way of building such grid is explained as follows:

- First, we need to build a set of coordinates in the canonical basis. The amount of pairs in such a set is defined by the maximum value to be taken $\mathbb{Z}$. For example, if we chose $i_{max} = 5$ for a two-dimensional system, then the elements of such a grid would be: $\mathbb{G}_c = \{(-5,-5),(-5,-4),\ldots,(0,0),\ldots,(5,4),(5,5)\}$ and we have 120 points in the new canonical grid to be transformed. The total amount of points in this grid for an $n$-dimensional system is given by: $(2i_{max}+1)^n - 1$, where $n$ is the dimension of the problem given in (1) and $i_{max} \in \mathbb{Z}_+$ is the maximum value used to construct the grid.
- The Cartesian or canonical grid is now transformed into (4) in which now the element $(0,0)$ is substituted by the coordinate of the collocation point around which we will adjust this new evaluation grid. This will transport $\mathbb{G}_c$ to the coordinate of the colocation point. The transformation of the new grid to be adjusted around the collocation point is:

$$\left\{ \frac{\alpha_{\text{Hexa-basis}}}{2i_{max}+1} \sum_{k=1}^{n} i_k \omega_k : i_k \in \mathbb{G}_c \right\} \tag{8}$$

where $2i_{k_{max}}+1$ is the factor that allows us to guarantee that all points are distributed homogeneously in the whole phase-space regardless of to which collocation point they belong. $\alpha_{\text{Hexa-basis}}$, as before, is the parameter used to build the collocation grid defined above.

Note that by equation (8) the evaluation points corresponding to the collocations points are aligned with the direction of the hexagonal grid.

The new evaluation grid will, however, grow exponentially. It can be seen around one collocation point in Fig. 2.

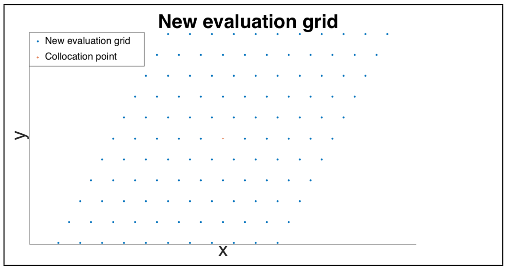

Fig. 2. Homogenization of the evaluation grids (blue) around all collocation points (red).

As before, we start by approximating the solution of $V'(\mathbf{x}) = -1$ by $v$, using the collocation points $X$. We define a tolerance parameter $-1 < \gamma \leq 0$, and mark a collocation point $\mathbf{x}_j$ to be poorly approximated, i.e., an element of our approximation of the chain-recurrent set ($\mathbf{x}_j \in X^0$), if there is at least one point $\mathbf{y} \in Y_{\mathbf{x}_j}$ such that $v(\mathbf{y}) > \gamma$. The well-approximated points, i.e., for which the condition $v'(\mathbf{y}) \leq \gamma$ holds for all $\mathbf{y} \in Y_{\mathbf{x}_j}$, belong to our approximation of the area of the gradient-like flow ($\mathbf{x}_j \in X^-$).

We compute the next approximation $v_{i+1}$ by solving the PDE

$$V'(\mathbf{x}_j) = \tilde{r}_j := \left( \frac{1}{|Y_{\mathbf{x}_j}|} \sum_{\mathbf{y} \in Y_{\mathbf{x}_j}} v_i'(\mathbf{y}) \right)_- , \tag{9}$$

where $\chi_- = \chi$ if $\chi \leq 0$ and $\chi_- = 0$ otherwise. Note that the method only requires us to know the right-hand side of the PDE at the collocation points. Summarizing, in subsequent iterations, the new right-hand side value is obtained by averaging the values around each collocation point and bounding by 0.

### B. Reducing the over-estimation of the chain-recurrent sets

To reduce the overestimation of the chain-recurrent set, we firstly analyze the behaviour of our benchmark systems. For example, we do notice [1], [4], [5], [12], [3] that the size of the chain-recurrent set increases over iterations. So, in practice, the only stopping criterion is defined by the predefined maximum amount of iterations because the chain-recurrent set does not converge since more points are added to it at each cycle.

However, it is also noticed in [1], [4], [5], [12], [3] that a reasonable approximation to the chain-recurrent set is obtained within few iterations. A large amount of iterations smooth out the complete Lyapunov function though it also increases and over-estimates the chain-recurrent set.

We base our algorithm on the assumption that the components of the chain-recurrent set are subsets of $(n-1)-$dimensional manifolds in an $n$-dimensional system. An assumption that is appropriate for many interesting systems. In our particular benchmark problems, we are exploring the symmetry of the systems to assume that the chain-recurrent set is formed by concentric circumferences.

The new algorithm works as follows using the collocation points $X = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$:

1) Compute the approximate Lyapunov function $v_i(\mathbf{y})$ and the orbital derivative $v_i'(\mathbf{y})$ for $i = 0$ by solving $V'(\mathbf{x}_j) = -1$ at the collocation points.
2) Approximate the chain-recurrent set by $X^0$ by computing $v_i'(\mathbf{y})$ for all $\mathbf{y} \in Y_{\mathbf{x}_j}$ for each collocation point $\mathbf{x}_j$, see (5). If $v_i'(\mathbf{y}) > \gamma$ for any $\mathbf{y} \in Y_{\mathbf{x}_j}$, then $\mathbf{x}_j \in X^0$, else $\mathbf{x}_j \in X^-$, where $\gamma \leq 0$ is a predefined critical value.
3) Measure all radii (distances) from the origin to the different failing points.
4) Classify all different radii and define the different intervals of the failing points, separated by a large interval of radii with no failing points.
5) For each interval $[r_{min}, r_{max}]$ of failing points we only keep the failing points in $[r_1, r_2]$ in $X_0$, defined below and reject the other ones (move them to $X^-$)..

$$r_1 = r_{min} + 0.52 * (r_{max} - r_{min})$$
$$r_2 = r_{max} - 0.52 * (r_{max} - r_{min})$$

   In Fig. 4, we show an overview of these radii in an example.
6) For the points close to the origin: If the distance of the failing points to the origin is smaller than $3\alpha_{\text{Hexa-basis}}$, then we consider those points to be an overestimation of the equilibrium at the origin. All those points are then removed; remember we have also removed the origin from the collocation points. Then, the origin is considered to be in the chain-recurrent set.
7) Define $r_j = \left( \frac{1}{|Y_{\mathbf{x}_j}|} \sum_{\mathbf{y} \in Y_{\mathbf{x}_j}} v_i'(\mathbf{y}) \right)_-$
8) Compute the approximate solution $v_{i+1}$ of $V'(\mathbf{x}_j) = r_j$ for $j = 1, \dots, N$.
9) Set $i \rightarrow i+1$ and repeat steps 2) to 8) until no more points are added to the reduced $X^0$.

## III. Results

Figure 2 shows the homogenization of the grid. When compared with Figure 1, we can see the strong disadvantage of to many blanks spaces left out of analysis in contrast with Figure 2, in which the evaluation points are distributed nicely in all the phase-space.

Next, we apply our algorithm to two benchmark problems in which the periodic orbits are circles.

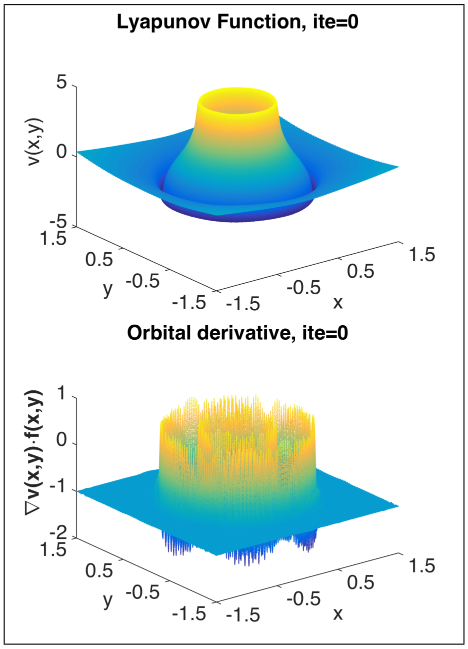

Fig. 3. First step of the algorithm. Upper figure: Complete Lyapunov function for system (10). Lower figure: The orbital derivative.

### A. Two circular periodic orbits

We consider system (1) with right-hand side

$$\mathbf{f}(x,y) = \begin{pmatrix} -x(x^2+y^2-1/4)(x^2+y^2-1)-y \\ -y(x^2+y^2-1/4)(x^2+y^2-1)+x \end{pmatrix}. \tag{10}$$

This system has an asymptotically stable equilibrium at the origin, $\Omega_0 = \{(0,0)\}$ since the Jacobian at the origin has eigenvalues: $\lambda_{1,2} = -0.25 \pm i$.

Moreover, the system has two periodic circular orbits: an asymptotically stable periodic orbit at $\Omega_1 = \{(x,y) \in \mathbb{R}^2 \mid x^2+y^2 = 1\}$ and a repelling periodic orbit at $\Omega_2 = \{(x,y) \in \mathbb{R}^2 \mid x^2+y^2 = 1/4\}$.

To compute the Lyapunov function with our method we used the Wendland function $\psi_{5,3}$.

The collocation points were set in a region $(-1.5, 1.5) \times (-1.5, 1.5) \subset \mathbb{R}^2$, with $i_{max} = 11$ and we used a hexagonal grid (4) with $\alpha_{\text{Hexa}-basis} = 0.03$. This setting gives a total amount of $11,600$ collocation points and $6,124,800$ evaluation points. We computed this example with the almost-normalized method $\dot{\mathbf{x}} = \hat{\mathbf{f}}(\mathbf{x})$ with $\delta^2 = 10^{-8}$ and $\gamma = -0.25$. The Complete Lyapunov function the orbital derivative are shown in 3.

Figure 3 is given by the first iteration. It could also be given by the last one or any other. What it is important here, it is to have a closed orbit identified.

We obtain the chain-recurrent set and we show the overestimation of the chain-recurrent set after 100 iterations in Fig. 4.

After applying our algorithm for only two iterations, we obtain the results shown in Figure 5.

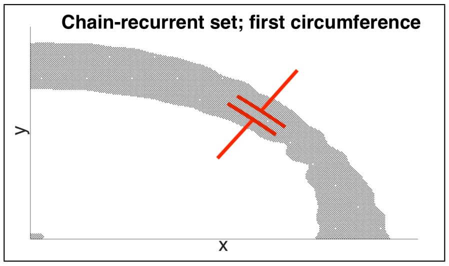

Fig. 4. Lower: Overview of the cut off radii defined in Eq. (10). This is a pedagogical example to show what the algorithm would in general consider to be in the chain-recurrent set and what should be removed from it. System (10).

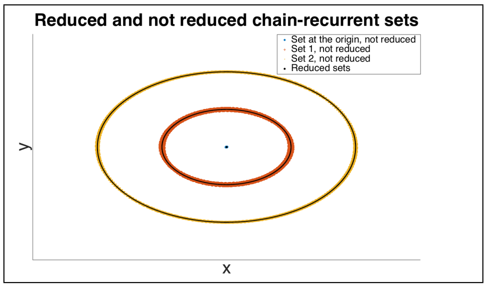

Fig. 5. Superposition of the reduced chain-recurrent set (black) over the not reduced one for system (10). Both cases correspond to the second iteration.

### B. Homoclinic orbit

As in [1], we also consider here the following example with right-hand side

$$\mathbf{f}(x,y)=\begin{pmatrix} x(1-x^2-y^2)-y((x-1)^2+(x^2+y^2-1)^2) \\ y(1-x^2-y^2)+x((x-1)^2+(x^2+y^2-1)^2) \end{pmatrix}. \tag{11}$$

The origin is an unstable focus, which can be seen from the eigenvalues of its Jacobian at the origin, which are $\lambda_{1,2}=1\pm 2i$.

Furthermore, the system has an asymptotically stable homoclinic orbit at a circle centred at the origin and with radius 1, connecting the equilibrium $(1,0)$ with itself.

We used the Wendland function $\psi_{4,2}$ for our computations.

We set our collocation points in the region $(-1.5,1.5)\times(-1.5,1.5)\subset\mathbb{R}^2$ with a hexagonal grid (4) with $\alpha_{\mathrm{Hexa}-basis}=0.02$. In this example, we have used the normalized method, i.e. we replaced $\mathbf{f}$ by $\hat{\mathbf{f}}$ as in (2) with $\delta^2=10^{-8}$, and we used $\gamma=-0.75$. This setting gives a total amount of $26,100$ collocation points and evaluation points $13,780,800$ using $i_{max}=11$. The Lyapunov function is shown in Fig. 6 and the superposition of the reduced chain-recurrent set over the over-estimated one is seen in Fig. 7.

## IV. Discussion and further improvements

Taking measures as discussed in this paper to reduce the overestimation delivered better approximations of the chain-recurrent set for our benchmark-problems than in earlier attempts. Our new grid has the benefit of allowing us to evaluate the whole space phase. Unfortunately, the amount of points grows considerably. For example, for a collocation grid built with $\alpha_{\mathrm{Hexa}-basis}=0.03$, the total amount of collocation-points is only $11,600$, but the number of evaluation points is $6,124,800$ with $i_{max}=11$.

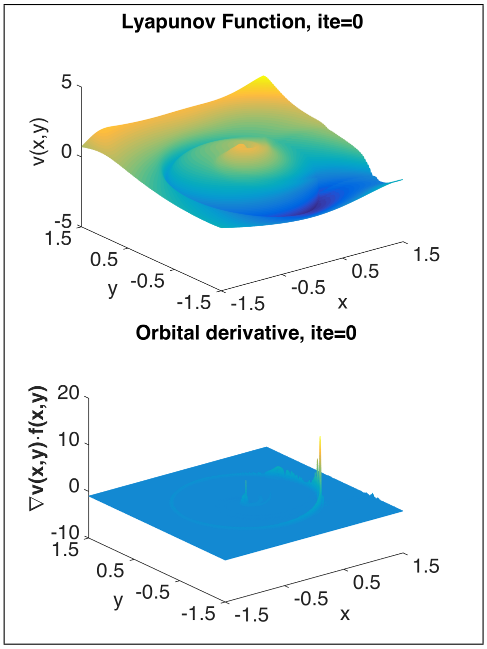

Fig. 6. First step of the algorithm. Upper figure: Complete Lyapunov function. Lower figure: Orbital derivative. System (11).

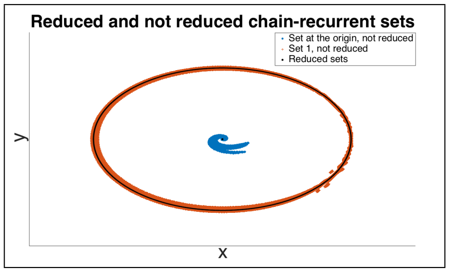

Fig. 7. Superposition of the reduced approximation of the chain-recurrent set (black) drawn over the unreduced one (red) for system (11).

However, since the evaluations are perfectly parallelizable they are not overly time consuming on a state of the art computer using multithreading. For example, the computation of system (10) with $6,124,800$ points to evaluate and $11,600$ points in the collocation lasted only 8 minutes on a PC an i7-4790K CPU (4 cores@4.00GHz, hyper-threading) using 8 threads. For the system (11), with $13,780,800$ points to evaluate and $26,100$ in the collocation, the computation lasted 45 minutes.

### A. The stopping criterion

So far we have explained an algorithm that works only for cases in which the components of the chain-recurrent set are circular and, unlike the previous cases [1], [4], [5], [12], [3], we have managed to obtain a working stopping criterion. For system (10) the algorithm stopped at the iteration 91, while for system (11) the algorithm stopped at the iteration 4. In both cases the stopping criterion was set to be an unchanged or a decreasing amount of points in the chain-recurrent set for three consecutive iterations. This means that it is worth the effort to further develop this algorithm to use clustering to reduce the over-estimation of the chain-recurrent set.

Figure 8 shows the variation of the amount of elements in the reduced chain-recurrent set for the system (10) for 1000 iterations. It also shows the amount of points in a close region to the iteration where it stops.

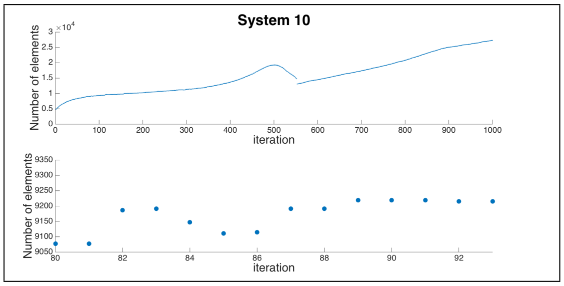

Fig. 8. System (10). Upper figure: Amount of elements in the chain-recurrent set as a function of iterations. Lower figure: Amount of points of the chain-recurrent set at the moment the algorithm stopped.

In Fig. 8 can be seen that the amount of elements varies considerably. At iteration 500, before the amount of numbers in the chain-recurrent set decays, the graph in Fig. 8 attains a local maximum. That could also be used as a stopping criterion.

The over-estimation comes from the function's smoothing throughout the iterations, because near to the chain-recurrent set the value of $v'(\mathbf{x})$ is close to zero. Now, however, we have been capable of isolating the periodic behaviour in our examples and to reduce the over-estimation.

## V. Conclusions

In this work a simple but important improvement to the algorithms presented in [1], [4], [5], [12], [3] was suggested. First, we constructed an evaluation grid homogeneously distributed in the whole phase-space.

Furthermore, we have improved the effort required to obtain the chain-recurrent sets by reducing its over-estimation. In contrast with our previous work, this is now used as a stopping criterion. A usable stopping criterion is to stop when no more points are added to the approximation of the chain-recurrent set for two or three iterations.

We have shown that there is a considerable variability on the amount of points in the approximation to chain-recurrent set as a function of iterations.

Future work includes the development of an optimised stopping criterion as well as to generalize algorithm for more general chain-recurrent sets.

## References

[1] C. Argáez, P. Giesl and S. Hafstein, *Analysing dynamical systems towards computing complete Lyapunov functions*, Proceedings of the 7th International Conference on Simulation and Modeling Methodologies, Technologies and Applications (SIMULTECH) 2017, Madrid, Spain, Pag. 134–144

[2] C. Argáez, S. Hafstein, P. Giesl, *Wendland Functions - A C++ Code to Compute Them*, Proceedings of the 7th International Conference on Simulation and Modeling Methodologies, Technologies and Applications (SIMULTECH) 2017, Madrid, Spain, Pag. 323–330

[3] C. Argáez, P. Giesl, and S. Hafstein, *Computation of complete Lyapunov functions for three-dimensional systems*, Proceedings of the 57rd IEEE Conference on Decision and Control (CDC) 2018, Miami Beach, FL, USA, Pag. 4059–4064.

[4] C. Argáez, P. Giesl and S. Hafstein, *Computational approach for complete Lyapunov functions*, In: Dynamical Systems in Theoretical Perspective Series: Springer Proceedings in Mathematics and Statistics 248 ed. J. Awrejcewicz, Springer 2018. Pag. 1-11.

[5] C. Argáez, P. Giesl, and S. Hafstein, *Iterative construction of complete Lyapunov functions*, Proceedings of the 8th International Conference on Simulation and Modeling Methodologies, Technologies and Applications (SIMULTECH) 2018, Porto, Portugal, Pag. 211-222

[6] P. Bernhard and S. Suhr, *Lyapounov Functions of Closed Cone Fields: From Conley Theory to Time Functions*, Commun. Math. Phys., Vol. 359, Pag 467–498, 2018

[7] C. Conley, *Isolated Invariant Sets and the Morse Index*, American Mathematical Society, CBMS Regional Conference Series no. 38, 1978

[8] C. Conley, *The gradient structure of a flow I*, Ergodic Theory Dynam. Systems, Vol. 8, Pag. 11–26, 1988

[9] M. Dellnitz and O. Junge, *Set oriented numerical methods for dynamical systems*, Handbook of dynamical systems, Pag. 221–264, 2002

[10] A. Fathi and P. Pageault, *Smoothing Lyapunov functions*, Trans. Amer. Math. Soc., Vol. 371, Pag. 1677–1700, 2019

[11] P. Giesl, *Construction of Global Lyapunov Functions Using Radial Basis Functions*, Springer-Verlag Berlin Heidelberg, Lecture Notes in Mathematics 1904, 2007

[12] P. Giesl C. Argáez, S. Hafstein and H. Wendland,*Construction of a Complete Lyapunov Function using Quadratic Programming*, Proceedings of the 15th International Conference on Informatics in Control, Automation and Robotics (ICINCO) 2018, Pag. 587–594

[13] C. S. Hsu, *Cell-to-cell mapping; a method of global analysis for nonlinear systems*. Applied mathematical sciences 64. New York: Springer, 1987.

[14] M. Hurley, *Chain recurrence, semiflows, and gradients*, J Dyn Diff Equat, Vol. 7, No. 3, Pag 437–456, 1995

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

[16] A. Iske, *Perfect Centre Placement for Radial Basis Function Methods*, TU Munich, Germany, 1998

[17] B. Krauskopf, H. Osinga, E. J. Doedel, M. Henderson, J. Guckenheimer, A. Vladimirsky, M. Dellnitz and O. Junge, *A survey of methods for computing (un)stable manifolds of vector fields*, Internat. J. Bifur. Chaos Appl. Sci. Engrg., Pag 763–791, Vol. 15, 2015

[18] A. M. Lyapunov, *The general problem of the stability of motion*. Internat. J. Control, Vol. 55, No. 3, Pag. 521–790. Translated by A. T. Fuller from Édouard Davaux's French translation (1907) of the 1892 Russian original, With an editorial (historical introduction) by Fuller, a biography of Lyapunov by V. I. Smirnov, and the bibliography of Lyapunov's works collected by J. F. Barrett, Lyapunov centenary issue.

[19] H. Wendland, *Error estimates for interpolation by compactly supported Radial Basis Functions of minimal degree*, J. Approx. Theory, Vol. 93, Pag 258–272, 1998