



Complete Lyapunov Functions: Computation and Applications

Carlos Argáez¹(✉), Peter Giesl², and Sigurdur Hafstein¹

¹ Science Institute, University of Iceland, 107, Reykjavík, Iceland
{carlos,shafstein}@hi.is

² Department of Mathematics, University of Sussex, Falmer BN1 9QH, UK
P.A.Giesl@sussex.ac.uk
<http://www.hi.is/~carlos>
<http://users.sussex.ac.uk/~pag20/>
<http://www.hi.is/~shafstein>

Abstract. Many phenomena in disciplines such as engineering, physics and biology can be represented as dynamical systems given by ordinary differential equations (ODEs). For their analysis as well as for modelling purposes it is desirable to obtain a complete description of a dynamical system. Complete Lyapunov functions, or quasi-potentials, describe the dynamical behaviour without solving the ODE for many initial conditions. In this paper, we use mesh-free numerical approximation to compute a complete Lyapunov function and to determine the chain-recurrent set, containing the attractors and repellers of the system. We use a homogeneous evaluation grid for the iterative construction, and thus improve a previous method. Finally, we apply our methodology to several examples, including one to compute an epigenetic landscape, modelling a bistable network of two genes. This illustrates the capability of our method to solve interdisciplinary problems.

Keywords: Dynamical system · Complete Lyapunov Function · Quasi-potential · Mesh-free collocation · Radial basis functions

1 Introduction

Let us consider a general autonomous ordinary differential equation (ODE) $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$, where $\mathbf{x} \in \mathbb{R}^n$. A classical (strict) Lyapunov function [29] is a scalar-valued function that can be used to analyze the basin of attraction of one attractor such as an equilibrium or a periodic orbit. It attains its minimum at the attractor, and is otherwise strictly decreasing along solutions of the ODE.

A generalization of this idea is the notion of a complete Lyapunov function [10, 20, 28, 35], which characterizes the complete behaviour of the dynamical system. It is a scalar-valued function $V : \mathbb{R}^n \rightarrow \mathbb{R}$, defined on the whole phase space, not just in a neighbourhood of one particular attractor. It is non-increasing along solutions of the ODE. The phase space can be divided into the area where the

complete Lyapunov function strictly decreases along solution trajectories and the one where it is constant. For the first case, the complete Lyapunov function characterizes the gradient-like flow. There solutions pass through and the larger this area is, the more information is obtained from the complete Lyapunov function. Note that, by definition, the complete Lyapunov function needs to be constant along solution trajectories on each transitive component of the chain-recurrent set.

Furthermore, there are other methods to analyze the general behaviour of dynamical systems such as the direct simulation of solutions with many different initial conditions. This, however, is costly and can only give limited information about the general behaviour of the system, unless estimates are available, e.g. when shadowing solutions. More sophisticated methods include the computation of invariant manifolds, forming the boundaries of basins of attraction of attractors [37]; here, additional analysis of the parts with gradient-like flow is necessary. The cell mapping approach [18] or set oriented methods [12] divide the phase space into cells and compute the dynamics between these cells, see also for example [30]; these ideas have also been used to compute complete Lyapunov functions [9].

Since our first method was proposed in [31] to compute complete Lyapunov functions using mesh-free collocation and thus to divide the phase space into the chain-recurrent set and the gradient-like flow, several improvements have been proposed. We will shortly review such improvements and explain the novelties of this current work.

The general idea [31] is to compute a complete Lyapunov function v by approximating the solution of $V'(\mathbf{x}) = -1$, where $V'(\mathbf{x}) = \nabla V(\mathbf{x}) \cdot \mathbf{f}(\mathbf{x})$ denotes the orbital derivative, the derivative along solutions of the ODE. We use mesh-free collocation with Radial Basis Functions (RBF) to approximately solve this partial differential equation (PDE): we choose a finite set of collocation points X and compute an approximation v to V which solves the PDE at all collocation points. Note, however, that the PDE cannot be fulfilled at all points of the chain-recurrent set, such as an equilibrium or periodic orbit. For that reason, we used the failure of the method to approximate in certain areas to classify them [31]: we separate the collocation points X into a set X^0 , where the approximation fails, and X^- , where it works well.

In general, a complete Lyapunov function should be constant in X^0 . For that reason, in a subsequent step we solved the PDE $V'(\mathbf{x}) = 0$ for all $\mathbf{x} \in X^0$ and $V'(\mathbf{x}) = -1$ for all $\mathbf{x} \in X^-$, which is a more accurate approximation of a complete Lyapunov function. This procedure is then iterated, either for a specified number of iterations or until no more points are added to X^0 . The computed function v gives us the following information about the ODE under consideration: the set X^0 , where $v'(\mathbf{x}) \approx 0$, approximates the chain-recurrent set, including equilibria, periodic orbits and homoclinic orbits, while the set X^- , where $v'(\mathbf{x}) \approx -1$, approximates the area with gradient-like flow, where solutions pass through. The function v , through its level sets, gives additional information about the stability and attraction properties: minima of v correspond to attractors, while maxima represent repellers.

Let us give more details on how we determine whether the approximation near a collocation point is poor (or fails), resulting in placing this collocation point into the set X^0 , or whether the approximation is good, and the collocation point is placed into X^- . By construction, at the collocation points we have $v'(\mathbf{x}_j) = V'(\mathbf{x}_j)$, which thus will always be a perfect approximation. We thus evaluate $v'(\mathbf{x})$ for all \mathbf{x} of an evaluation grid $Y_{\mathbf{x}_j}$ of points near the collocation point \mathbf{x}_j . We fix a critical value $\gamma \leq 0$ and place the collocation point \mathbf{x}_j into X^0 if $v'(\mathbf{x}) > \gamma$ holds for at least one $\mathbf{x} \in Y_{\mathbf{x}_j}$; otherwise \mathbf{x}_j is placed into X^- .

While the basic method is already capable of classifying the chain-recurrent set in many examples, there were several shortcomings which were addressed in subsequent improvements: (a) in examples, where the velocity $\|\mathbf{f}(\mathbf{x})\|$ varies considerably, the chain-recurrent was either over- or under-estimated and (b) the method does not produce a complete Lyapunov function with a continuous derivative. Here, $\|\cdot\|$ denotes the Euclidean norm in \mathbb{R}^n .

To address (a), we have introduced a method [32] to analyze systems that have a large change in their velocity, namely where $\|\mathbf{f}(\mathbf{x})\|$ varies considerably over the phase space. We proposed to normalize the system by “almost” the norm. In particular, this is done by replacing the original system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$ by the system

$$\dot{\mathbf{x}} = \hat{\mathbf{f}}(\mathbf{x}), \quad \text{where } \hat{\mathbf{f}}(\mathbf{x}) = \frac{\mathbf{f}(\mathbf{x})}{\sqrt{\delta^2 + \|\mathbf{f}(\mathbf{x})\|^2}} \quad (1)$$

with small parameter $\delta > 0$. The dynamical system defined by (1) with $\hat{\mathbf{f}}(\mathbf{x})$ has the same solution trajectories as $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$, but these are traversed at a more uniform speed, namely $\|\hat{\mathbf{f}}(\mathbf{x})\| = \frac{\|\mathbf{f}(\mathbf{x})\|}{\sqrt{\delta^2 + \|\mathbf{f}(\mathbf{x})\|^2}} \approx 1$. The smaller δ is, the closer the speed is to 1. While the normalized method generally produces better results, it comes at a higher computational cost due to the evaluation of $\hat{\mathbf{f}}$.

To address problem (b), we have proposed different improvements. After solving $V'(\mathbf{x}) = -1$, we again split the collocation points into X^0 and X^- . However, instead of solving $V'(\mathbf{x}) = r(\mathbf{x})$ with $r(\mathbf{x}) = -1$ for $\mathbf{x} \in X^-$ and $r(\mathbf{x}) = 0$ for $\mathbf{x} \in X^0$, which results in a discontinuous function $r(\mathbf{x})$, we make the right-hand side $r(\mathbf{x})$ smooth by using the distance $\mathfrak{d}(\mathbf{x})$ between the point \mathbf{x} and the set X^0 . In detail, we solve the PDE

$$V'(\mathbf{x}) = r(\mathbf{x}) := \begin{cases} 0 & \text{if } \mathbf{x} \in X^0, \\ -\exp\left(-\frac{1}{\xi \cdot \mathfrak{d}^2(\mathbf{x})}\right) & \text{if } \mathbf{x} \in X^-, \end{cases} \quad (2)$$

where $\mathfrak{d}(\mathbf{x}) = \min_{\mathbf{y} \in X^0} \|\mathbf{x} - \mathbf{y}\|$ is the distance between the point \mathbf{x} and the set X^0 and $\xi > 0$ is a parameter [32].

Using (2) we guarantee that $r(\mathbf{x})$ and thus V is a smooth function such that for \mathbf{x} with a large distance to X^0 , $r(\mathbf{x})$ is close to -1 while for close distances it raises up to zero.

A different method to solve problem (b) defines $r(\mathbf{x}_j)$ to be the average value of $v'(\mathbf{y})$ over all $\mathbf{y} \in Y_{\mathbf{x}_j}$, this is done regardless of whether \mathbf{x}_j lies in X^0 or X^- .

This again results in a smooth function $r(\mathbf{x})$, and the distinction between X^0 and X^- is just used to determine the chain-recurrent set [33,34].

In previous work, we have used different sets for the evaluation points $Y_{\mathbf{x}_j}$. In [31], in two dimensions, we used points on two circumferences around each collocation point. Later we introduced a new evaluation grid consisting of points along the direction of the flow, $\mathbf{f}(\mathbf{x}_j)$ at each collocation point. That allowed to expand our method from two to higher dimensions without increasing the amount of evaluation points exponentially [34].

Combining this with fixing $r(\mathbf{x}_j)$ by averaging over all $v'(\mathbf{y})$ with $\mathbf{y} \in Y_{\mathbf{x}_j}$ may result in the right-hand side converging to zero as the number of iterations goes to infinity. This would result in a constant complete Lyapunov function, which is a trivial complete Lyapunov function and does provide any information about the dynamics. To fix that, we introduced a methodology in [33] that scales the orbital derivative condition, which shows to be efficient to avoid trivial solutions. However, for few iterations that consideration is not necessary.

In this paper, we use an evaluation grid to evaluate the complete Lyapunov function that consists of a homogeneous distribution of points on two circumferences around each collocation point. We use an iterative method to construct a complete Lyapunov function by replacing the right-hand side $r(\mathbf{x}_j)$ by the average of $v'(\mathbf{y})$ over all points of the evaluation grid at the respective collocation point \mathbf{x}_j . Depending on the example, we either consider the original equation $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$ or the normalized one, see (1). Furthermore, we apply our methodology to four examples. The first three are from [32]. The final example is a biological system to prove the capabilities of our method in applications.

Let us give an overview over the paper: in Sect. 2 we discuss complete Lyapunov functions as well as mesh-free collocation to approximate solutions of a general linear PDE. In Sect. 3 we present our algorithm to compute a complete Lyapunov function. In Sect. 4 we apply the method to four examples and discuss the results in detail.

2 Preliminaries

2.1 Complete Lyapunov Functions

We will consider a general autonomous ODE

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}), \text{ where } \mathbf{x} \in \mathbb{R}^n. \quad (3)$$

A complete Lyapunov function [10] is a continuous function $V: \mathbb{R}^n \rightarrow \mathbb{R}$ which is constant along solution trajectories on the chain-recurrent set, including local attractors and repellers, and decreasing along solution trajectories elsewhere. In contrast to classical Lyapunov functions [29], which are defined on the basin of attraction of just one attractor, a complete Lyapunov function characterizes the flow on the whole phase space and distinguishes between the chain-recurrent set and the gradient-like flow. Thus, it captures the long-term behaviour of the system.

Auslander [5] and Conley [10] proved the existence of complete Lyapunov functions for dynamical systems on a compact metric space. The idea is to consider corresponding attractor-repeller pairs and to construct a function which is 1 on the repeller, 0 on the attractor and decreasing in between. Then these functions are summed up over all attractor-repeller pairs. This was generalized to more general spaces by Hurley [19, 20, 28].

The smaller the part of the phase space where the complete Lyapunov function is constant, the more information is provided by a complete Lyapunov function. There exists a complete Lyapunov function which is only constant on the generalized chain-recurrent set [5], thus providing further information about the system as the generalized chain-recurrent set is a subset of the chain-recurrent set.

In [9, 16, 24] a computational approach to construct complete Lyapunov functions was proposed. The discrete-time system given by the time- T map was considered, the phase space was subdivided into cells and the dynamics between them were computed through an induced multivalued map using the computer package GAIO [11]. An approximate complete Lyapunov function is then computed using graph algorithms [9]. This approach requires a high number of cells even for low dimensions. We will use a different methodology, inspired by the construction of classical Lyapunov functions, which is faster and works well in higher dimensions. In [6], the approach of [9] is compared to the RBF method for equilibria (see below) for one particular example; here, the method of [9] works well only on the chain-recurrent set, while the RBF method is very efficient on the gradient-like part.

In [8], a complete Lyapunov is constructed as a continuous piecewise affine (CPA) function, affine on each simplex of a fixed simplicial complex. However, it is assumed that information about local attractors is available, while the proposed method in this paper does not require any information about the system under consideration.

In [1] a quasi-potential, which is very similar to a complete Lyapunov function, is constructed by numerical integration along solution trajectories with different initial conditions. The landscape (plot of the complete Lyapunov function) and the level sets are used to analyze a biological system modelling two genes that inhibit each other, forming a double-negative feedback loop structure. We will analyze this system with our method in Sect. 4.4.

2.2 Mesh-Free Collocation

For classical Lyapunov functions, several numerical construction methods have recently been proposed, e.g. [2–4, 7, 13, 17, 21–23, 25] see also the review [15]. Our algorithm will be based on the RBF (Radial Basis Function) method, a special case of mesh-free collocation, which approximates the solution of a linear PDE, specifying the orbital derivative.

Mesh-free methods, particularly based upon Radial Basis Functions, provide a powerful tool for solving generalized interpolation problems efficiently. We assume that the target function belongs to a Hilbert space H of continuous

functions (often a Sobolev space) with norm $\|\cdot\|_H$ and with reproducing kernel $\varphi : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$, given by a suitable Radial Basis Function Φ through $\varphi(\mathbf{x}, \mathbf{y}) := \Phi(\mathbf{x} - \mathbf{y})$, where $\Phi(\mathbf{x}) = \psi(\|\mathbf{x}\|)$ is a radial function. Examples for Radial Basis Functions include the Gaussians, multiquadratics and inverse multiquadratics; we, however, will use the compactly supported Wendland functions in this paper, which will be defined below.

We assume that the information $r_1, \dots, r_N \in \mathbb{R}$ of a target function $V \in H$ generated by N linearly independent functionals $\lambda_j \in H^*$ is known. The optimal reconstruction of the function V is the solution of the minimization problem $\min\{\|v\|_H : \lambda_j(v) = r_j, 1 \leq j \leq N\}$. It is well-known [36] that the solution can be written as $v(\mathbf{x}) = \sum_{j=1}^N \beta_j \lambda_j^\mathbf{y} \varphi(\mathbf{x}, \mathbf{y})$, where the coefficients β_j are determined by the interpolation conditions $\lambda_j(v) = r_j, 1 \leq j \leq N$.

In our case, we consider the PDE $V'(\mathbf{x}) = r(\mathbf{x})$, where $r(\mathbf{x})$ is a given function and $V'(\mathbf{x}) = \nabla V(\mathbf{x}) \cdot \mathbf{f}(\mathbf{x})$ denotes the orbital derivative. We choose N points $\mathbf{x}_1, \dots, \mathbf{x}_N \in \mathbb{R}^n$ of the phase space and define functionals $\lambda_j(v) := (\delta_{\mathbf{x}_j} \circ L)^\mathbf{y} v = v'(\mathbf{x}_j) = \nabla v(\mathbf{x}_j) \cdot \mathbf{f}(\mathbf{x}_j)$, where L denotes the linear operator of the orbital derivative $LV(\mathbf{x}) = V'(\mathbf{x})$ and δ is Dirac's delta distribution. The right-hand sides are $r_j = r(\mathbf{x}_j)$ for all $1 \leq j \leq N$. The approximation is then

$$v(\mathbf{x}) = \sum_{j=1}^N \beta_j (\delta_{\mathbf{x}_j} \circ L)^\mathbf{y} \Phi(\mathbf{x} - \mathbf{y}),$$

where Φ is a positive definite Radial Basis Function [36], and the coefficients $\beta_j \in \mathbb{R}$ can be calculated by solving a system of N linear equations. A crucial ingredient is the knowledge on the behaviour of the error function $|V'(\mathbf{x}) - v'(\mathbf{x})|$ in terms of the so-called fill distance $h = \sup_{\mathbf{y} \in K} \inf_{j=1, \dots, N} \|\mathbf{y} - \mathbf{x}_j\|$ which measures how dense the points $\{\mathbf{x}_1, \dots, \mathbf{x}_N\} \subset K$ are in the compact set $K \subset \mathbb{R}^n$, since it gives information when the approximate solution has a negative orbital derivative. Such error estimates were derived, for example in [13, 14], see also [26, 36].

The advantage of mesh-free collocation over other methods for solving PDEs is that scattered points can be added to improve the approximation, no triangulation of the phase space is necessary, the approximating function is smooth and the method works in any dimension.

In this paper, we use Wendland functions [27] as Radial Basis Functions through $\psi(\|\mathbf{x}\|) := \psi_{l,k}(c\|\mathbf{x}\|)$, where $c > 0$, $k \in \mathbb{N}$ is a smoothness parameter and $l = \lfloor \frac{n}{2} \rfloor + k + 1$. Wendland functions are positive definite Radial Basis Functions with compact support, which are polynomials on their support; the corresponding (Reproducing Kernel) Hilbert Space is norm-equivalent to the Sobolev space $W_2^{k+(n+1)/2}(\mathbb{R}^n)$. They are defined by recursion: for $l \in \mathbb{N}$, $k \in \mathbb{N}_0$ we define

$$\begin{aligned} \psi_{l,0}(r) &= (1 - r)_+^l \\ \psi_{l,k+1}(r) &= \int_r^1 t \psi_{l,k}(t) dt \end{aligned} \tag{4}$$

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

As collocation points $X \subset \mathbb{R}^n$ we use a hexagonal grid (5) with $\alpha \in \mathbb{R}^+$ constructed according to

$$\begin{aligned} & \left\{ \alpha \sum_{k=1}^n i_k w_k : i_k \in \mathbb{Z} \right\}, \text{ where} \\ & w_1 = (2e_1, 0, 0, \dots, 0) \\ & w_2 = (e_1, 3e_2, 0, \dots, 0) \\ & \vdots \quad \vdots \\ & w_n = (e_1, e_2, e_3, \dots, (n+1)e_n) \text{ and} \\ & e_k = \sqrt{\frac{1}{2k(k+1)}}, \quad k \in \mathbb{N}. \end{aligned} \tag{5}$$

We set $\psi_0(r) := \psi_{l,k}(cr)$ with positive constant c and define recursively $\psi_i(r) = \frac{1}{r} \frac{d\psi_{i-1}}{dr}(r)$ for $i = 1, 2$ and $r > 0$. The explicit formulas for v and its orbital derivative are

$$\begin{aligned} v(\mathbf{x}) &= \sum_{j=1}^N \beta_j \langle \mathbf{x}_j - \mathbf{x}, \mathbf{f}(\mathbf{x}_j) \rangle \psi_1(\|\mathbf{x} - \mathbf{x}_j\|), \\ v'(\mathbf{x}) &= \sum_{j=1}^N \beta_j \left[-\psi_1(\|\mathbf{x} - \mathbf{x}_j\|) \langle \mathbf{f}(\mathbf{x}), \mathbf{f}(\mathbf{x}_j) \rangle \right. \\ &\quad \left. + \psi_2(\|\mathbf{x} - \mathbf{x}_j\|) \langle \mathbf{x} - \mathbf{x}_j, \mathbf{f}(\mathbf{x}) \rangle \cdot \langle \mathbf{x}_j - \mathbf{x}, \mathbf{f}(\mathbf{x}_j) \rangle \right] \end{aligned}$$

where $\langle \cdot, \cdot \rangle$ denotes the standard scalar product in \mathbb{R}^n , β is the solution to $A\beta = \mathbf{r}$, $r_j = r(\mathbf{x}_j)$ and A is the $N \times N$ matrix with entries

$$\begin{aligned} a_{ij} &= \psi_2(\|\mathbf{x}_i - \mathbf{x}_j\|) \langle \mathbf{x}_i - \mathbf{x}_j, \mathbf{f}(\mathbf{x}_i) \rangle \langle \mathbf{x}_j - \mathbf{x}_i, \mathbf{f}(\mathbf{x}_j) \rangle \\ &\quad - \psi_1(\|\mathbf{x}_i - \mathbf{x}_j\|) \langle \mathbf{f}(\mathbf{x}_i), \mathbf{f}(\mathbf{x}_j) \rangle \end{aligned}$$

for $i \neq j$ and

$$a_{ii} = -\psi_1(0) \|\mathbf{f}(\mathbf{x}_i)\|^2.$$

More detailed explanations on this construction are given in [13, Chap. 3].

If no collocation point \mathbf{x}_j is an equilibrium for the system, i.e. $\mathbf{f}(\mathbf{x}_j) \neq \mathbf{0}$ for all j , and all collocation points are pairwise distinct, then the matrix A is positive definite and the system of equations $A\beta = \mathbf{r}$ has a unique solution. Note that this holds true independent of whether the underlying discretized PDE has a solution or not, while the error estimates are only available if the PDE has a solution.

3 Algorithm

Starting with scattered collocation points X , we solve the equation $V'(\mathbf{x}) = -1$, where $V'(\mathbf{x}) := \nabla V(\mathbf{x}) \cdot \mathbf{f}(\mathbf{x})$ denotes the orbital derivative, the derivative along

solutions of (3). Note that the equation $V'(\mathbf{x}) = -1$ does not have a solution on chain-recurrent sets in general; e.g. along a periodic orbit, the orbital derivative must integrate to 0. However, as mentioned above, we can still compute a (unique) approximation by the method described in Sect. 2.2.

In the next step we check for each collocation point \mathbf{x}_j in X whether the approximation was poor (then $\mathbf{x}_j \in X^0$) or good (then $\mathbf{x}_j \in X^-$). Then we approximate the solution of the new problem $V'(\mathbf{x}) = -1$ for $\mathbf{x} \in X^-$ and $V'(\mathbf{x}) = 0$ for $\mathbf{x} \in X^0$; the set X^0 indicates the (generalized) chain-recurrent set.

To determine whether the approximation was poor or good, we evaluate $v'(\mathbf{x})$ for test points \mathbf{x} around each collocation point \mathbf{x}_j – for a good approximation we expect $v'(\mathbf{x}) \approx -1$. In view of our goal to compute a complete Lyapunov function, we classify collocation points as poor if the orbital derivative near them is larger than 0 or a chosen critical value, i.e. $v'(\mathbf{x}) > \gamma$, for certain points \mathbf{x} near the collocation point \mathbf{x}_j . As points to check near a collocation point \mathbf{x}_j we choose points on two circumferences around each collocation point \mathbf{x}_j .

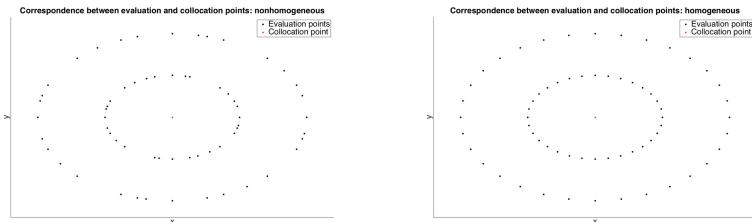


Fig. 1. Improvement over the evaluation points; both have 32 points on each circumference. Left: non-homogeneous distribution. Right: homogeneous distribution. The middle point represents the collocation point which is the centre of both circumferences.

Previously, in [31], we used points distributed on two concentric circumferences whose centre is the collocation point. However, the distribution of such points was not homogeneous along the circumferences, see Fig. 1. In this paper, however, we use points homogeneously distributed over two concentric circumferences whose centre is the collocation point. Originally in [31], the grid $Y_{\mathbf{x}_j}$ consisted of the points (6), while in this paper we use the points $Y_{\mathbf{x}_j}$ in (7).

$$\begin{aligned} \mathbf{x}_j &\pm r\alpha(\cos(\theta), -\sin(\theta)) \\ \mathbf{x}_j &\pm r\alpha(\sin(\theta), \cos(\theta)) \\ \mathbf{x}_j &\pm \frac{r}{2}\alpha(\cos(\theta), \sin(\theta)) \\ \mathbf{x}_j &\pm \frac{r}{2}\alpha(\cos(\theta), -\sin(\theta)) \end{aligned} \tag{6}$$

$$\begin{aligned} \mathbf{x}_j &+ r\alpha\left(\cos\frac{2\pi k}{M}, \sin\frac{2\pi k}{M}\right) \\ \mathbf{x}_j &+ \frac{r}{2}\alpha\left(\cos\frac{2\pi k}{M}, \sin\frac{2\pi k}{M}\right) \end{aligned} \tag{7}$$

For (6), we have used $\theta = \{0.0^\circ, 11.25^\circ, 22.5^\circ, 45^\circ, 56.25^\circ, 67.5^\circ, 75^\circ, 105^\circ\}$ and for (7) we use $M = 32$. In both cases, r is a scaling parameter and 32 is the total amount of points we distribute in each circumference, resulting in 64

points per collocation points in total. Figure 1 shows the distributions of both evaluation grids.

Figure 1 shows an improvement on the evaluation capabilities of our method. The fact that the left-hand side figure in Fig. 1 is not homogeneously distributed is seen with 4 blank spots in each of the two circumferences. Furthermore, it also shows regions in which we can find points close to each other, while in other regions the distance between points is large. Considering that the Lyapunov function is first computed on the collocation points and then evaluated over the evaluation grid, the fact that the evaluation is non-homogeneous as shown in Fig. 1, weakens the evaluation when compared with the homogeneous case.

In [31], the aim was to obtain a description of the chain-recurrent set X^0 and the gradient-like flow X^- . That description allowed to set the orbital derivative to 0 for points in X^0 and to keep the orbital derivative to be -1 for points in X^- . This results in a non-continuous right-hand side.

In this paper we use not only the information whether points lie in X^0 or X^- , but also the average value of $v'(\mathbf{y})$ around each collocation to determine the PDE in the subsequent step. In particular, in the subsequent iteration we set the orbital derivative at each collocation point to be the average value of the Lyapunov function derivative for all evaluation points around the respective collocation points. The evaluation grid, however, is different from [33, 34] in which a directional grid was used. The advantage of that particular grid is that it could be used for 3-dimensional systems without increasing its size exponentially. In our case, however, we use the averaging over the circumference values, which can also be generalized to higher dimensions.

The evaluation grid in this paper uses the contributions in all directions around each collocation point instead of just in one direction and will thus produce more accurate results. Setting $r = 0.5$ ensures that the evaluation grids for adjacent collocation points do not overlap. In our computations, as we did in [31], we use 64 evaluation points around each collocation point.

Next, we summarize our algorithm.

1. Create the collocation points X . Compute the approximate solution v_0 of $V'(\mathbf{x}) = -1$, set $i = 0$
2. For each collocation point \mathbf{x}_j , compute $v'_i(\mathbf{x})$: if $v'_i(\mathbf{x}) > \gamma$ for a point $\mathbf{x} \in Y_{\mathbf{x}_j}$, then $\mathbf{x}_j \in X^0$, otherwise $\mathbf{x}_j \in X^-$, where $\gamma \leq 0$ is a chosen critical value
3. Compute the approximate solution v_{i+1} of

$$V'(\mathbf{x}_j) = r(\mathbf{x}_j) = \frac{1}{2M} \left(\sum_{\mathbf{y} \in Y_{\mathbf{x}_j}} v'_i(\mathbf{y}) \right) -, \text{ where } z_- = z \text{ for } z \leq 0 \text{ and } z_- = 0$$

otherwise
4. Set $i \rightarrow i + 1$ and repeat steps 2. and 3. until no more points are added to X^0 or until a certain predefined number of iterations is reached

Note that the mesh-free collocation method only requires us to know the right-hand side $r(\mathbf{x})$ of the PDE at the collocation points, hence, it is sufficient to define r at all collocation points \mathbf{x}_j .

4 Examples

In the following we apply the method to four examples in two dimensions and then analyze their behaviour. Note that in all examples we use the notation $\mathbf{x} = (x, y)$. We have done 10 iterations of the method for each example. The first three systems (8), (9) and (10) have been previously studied in [31], and we compare our new method with the previous results. The final system in Sect. 4.4 models a bistable network of two genes.

For the first two examples we have used the original equation $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$ and have achieved very good results, while for the last two examples, we have employed the slightly slower “almost” normalized method, introduced in [32], using (1). For all examples, we have used the new evaluation grid (7) with $r = 0.5$ and $M = 32$.

4.1 Two Circular Periodic Orbits

We consider system (3) with right-hand side

$$\mathbf{f}(x, y) = \begin{pmatrix} -x(x^2 + y^2 - 1/4)(x^2 + y^2 - 1) - y \\ -y(x^2 + y^2 - 1/4)(x^2 + y^2 - 1) + x \end{pmatrix}. \quad (8)$$

This system has an asymptotically stable equilibrium at the origin, $\Omega_0 = \{(0, 0)\}$

since the Jacobian at the origin is $\begin{pmatrix} -\frac{1}{4} & -1 \\ 1 & -\frac{1}{4} \end{pmatrix}$ with 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 Wendland function $\psi_{5,3}$ with the parameter $c = 1$. The collocation points were set in a region $(-1.5, 1.5) \times (-1.5, 1.5) \subset \mathbb{R}^2$ and we used a hexagonal grid (5) with $\alpha = 0.018$. This setting gives a total amount of 36,668 collocation points and 2,346,752 evaluation points. We computed this example with the original system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$.

Figure 2 shows the approximation v_0 of $V'(\mathbf{x}) = -1$ as well as its orbital derivative v'_0 . The function v_0 (Fig. 2, left) clearly displays the stable periodic orbit and equilibrium as local minima of v_0 , while the unstable periodic orbit is a local maximum. The orbital derivative v'_0 (Fig. 2, right) is mostly -1 apart from the two periodic orbits and the equilibrium, where it is close to 0. The orbital derivative is clearly discontinuous.

Several improvements have been made to generate a continuous derivative [32–34] in subsequent iterations. In this paper, we use the method described above, solving $V'(\mathbf{x}) = r(\mathbf{x})$, where $r(\mathbf{x}_j)$ is given by the average value of the orbital derivative at the evaluation grid around \mathbf{x}_j , using the critical value $\gamma = -0.5$. Figure 3 shows v_{10} and its orbital derivative after 10 iterations.

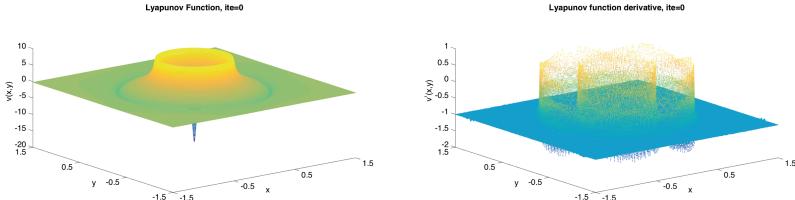


Fig. 2. Complete Lyapunov function for (8), iteration 0. The left-hand figure shows the approximated complete Lyapunov function $v_0(x, y)$. The right-hand figure shows its orbital derivative $v'_0(x, y)$. The Lyapunov function's minimum corresponds to the asymptotically stable equilibrium. The two periodic orbits with radius 1 and $1/2$ are asymptotically stable (Ω_1) and unstable (Ω_2) and are local minima and maxima of v_0 respectively. The orbital derivative v'_0 fails to be negative on Ω_1 , Ω_2 , and at the equilibrium, moreover, v'_0 is not continuous.

Figure 3 shows the tenth iteration of the new method. The orbital derivative v'_{10} is now continuous, which can be seen more clearly in Figs. 4 and 5, comparing the iterations 0 and 10 on a part of the phase space. Figure 4 shows a section of the Lyapunov function derivative v'_{10} over a part of the chain-recurrent set, namely Ω_2 . Figure 5 shows a section of the previous Fig. 4, which gives a clear picture of the improved continuity of the orbital derivative.

Finally, we are interested in the behaviour of the chain-recurrent set. Figure 6 shows the points \mathbf{y} of the evaluation grid, where $v'_i(\mathbf{y}) > \gamma$ with $\gamma = -0.5$, which approximate the chain-recurrent set. We find the three expected connected components of the chain-recurrent set already in iteration 0; and the sets do not change much when considering iteration 10.

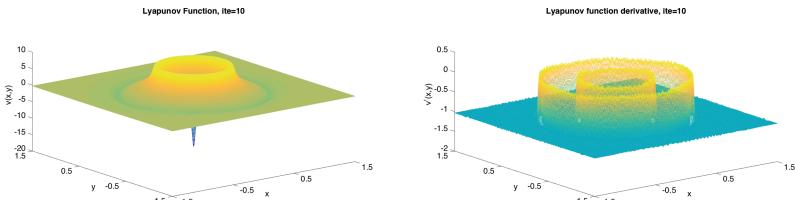


Fig. 3. Complete Lyapunov function for (8), iteration 10. The left-hand figure shows the approximated complete Lyapunov function $v_{10}(x, y)$. The right-hand figure shows its orbital derivative $v'_{10}(x, y)$; now continuous. The Lyapunov function's minimum corresponds to the asymptotically stable equilibrium. The two periodic orbits with radius 1 and $1/2$ are asymptotically stable (Ω_1) and unstable (Ω_2) and are local minima and maxima of v_{10} respectively. The orbital derivative v'_{10} fails to be negative on Ω_1 , Ω_2 , and at the equilibrium.

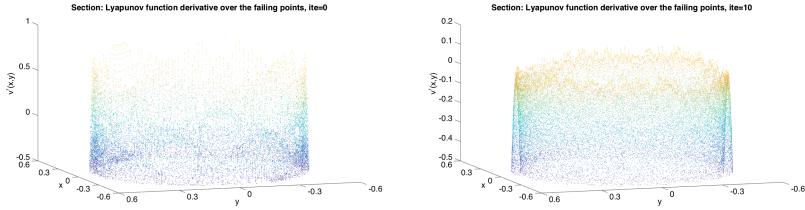


Fig. 4. Complete Lyapunov function derivative v'_0 (left) and v'_{10} for (8) over Ω_2 for iterations 0 and 10, $\gamma = -0.5$.

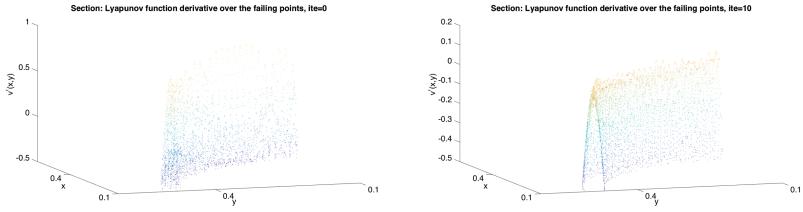


Fig. 5. Complete Lyapunov function derivative v'_0 (left) and v'_{10} for (8) over part of Ω_2 for iterations 0 and 10, $\gamma = -0.5$.

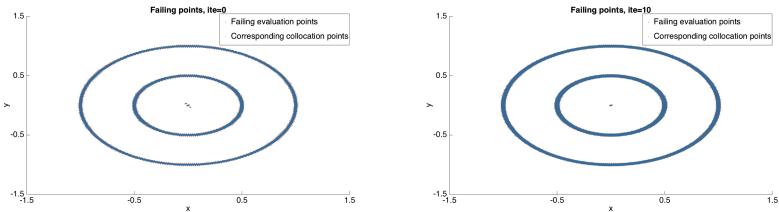


Fig. 6. The figures show the evaluation points y around each collocation point, where $v'(y) > \gamma = -0.5$; left: iteration 0 and right: iteration 10. The three chain-recurrent sets are clearly visible in both iterations.

4.2 Van-der-Pol Oscillator System

We consider system (3) with right-hand side

$$\mathbf{f}(x, y) = \begin{pmatrix} y \\ (1 - x^2)y - x \end{pmatrix}. \quad (9)$$

System (9) is the two-dimensional form of the Van-der-Pol oscillator. This describes the behaviour of a non-conservative oscillator reacting to a non-linear damping. The origin is an unstable focus, which can be seen from its Jacobian at the origin with eigenvalues $\lambda_{1,2} = 0.5 \pm 0.866025i$.

In this case, we have used the Wendland function $\psi_{4,2}$ with parameter $c = 1$. We set our collocation points in the region $(-4, 4) \times (-4, 4) \subset \mathbb{R}^2$, using a hexagonal grid (5) with $\alpha = 0.046$. We have a total amount of 36,668 collocation points and 2,346,751 evaluation points. We have used the original system, namely $\mathbf{f}(\mathbf{x})$.

Figure 7 shows the approximation v_0 of $V'(\mathbf{x}) = -1$ as well as its orbital derivative v'_0 . The function v_0 (Fig. 7, left) clearly displays the stable periodic orbit as minimum and the unstable equilibrium at the origin as maximum of v_0 . The orbital derivative v'_0 (Fig. 7, right) is mostly -1 apart from the periodic orbits and the equilibrium, where it is close to 0. The orbital derivative is clearly discontinuous.

Figure 8 shows the tenth iteration of the new method. The orbital derivative v'_{10} is now continuous, which can be seen more clearly in Figs. 9 and 10, comparing the iterations 0 and 10 on a part of the phase space. Figure 9 shows a section of the Lyapunov function derivative v'_{10} over a part of the chain-recurrent set, namely the periodic orbit. Figure 10 shows a section of the previous Fig. 9, which gives a clear picture of the improved continuity of the orbital derivative.

Figure 11 shows the points \mathbf{y} of the evaluation grid, where $v'_i(\mathbf{y}) > \gamma$ with $\gamma = -0.5$, which approximate the chain-recurrent set. We find the two expected connected components of the chain-recurrent set already in iteration 0; and the sets do not change much when considering iteration 10.

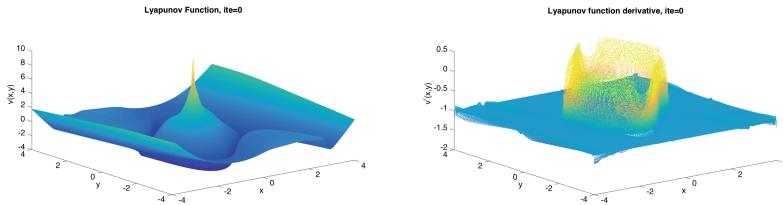


Fig. 7. Complete Lyapunov function for (9), iteration 0. The left-hand figure shows the approximated complete Lyapunov function $v_0(x, y)$. The right-hand figure shows its orbital derivative $v'_0(x, y)$. The Lyapunov function's maximum corresponds to the unstable equilibrium at the origin. The stable periodic orbit is a local minimum of v_0 . The orbital derivative v'_0 fails to be negative on the periodic orbit and at the equilibrium, moreover, v'_0 is not continuous.

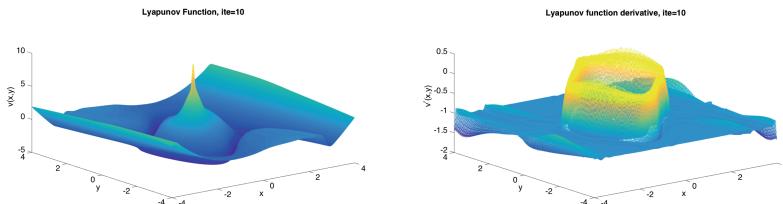


Fig. 8. Complete Lyapunov function for (9), iteration 10. The left-hand figure shows the approximated complete Lyapunov function $v_{10}(x, y)$. The right-hand figure shows its orbital derivative $v'_{10}(x, y)$; now continuous. The Lyapunov function's maximum corresponds to the unstable equilibrium at the origin. The stable periodic orbit is a local minimum of v_0 . The orbital derivative v'_0 fails to be negative on the periodic orbit and at the equilibrium.

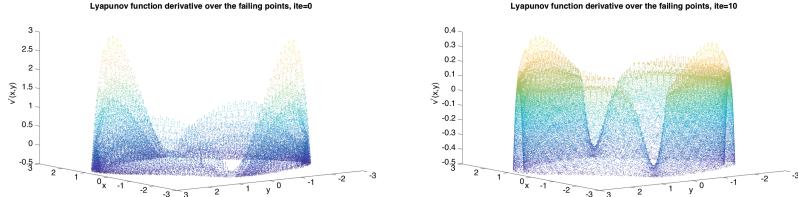


Fig. 9. Complete Lyapunov function derivative v'_0 (left) and v'_{10} for (9) over the periodic orbit for iterations 0 and 10, $\gamma = -0.5$.

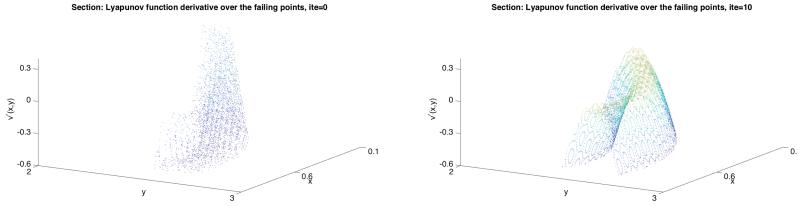


Fig. 10. Complete Lyapunov function derivative v'_0 (left) and v'_{10} for (9) over part of the periodic orbit for iterations 0 and 10, $\gamma = -0.5$.

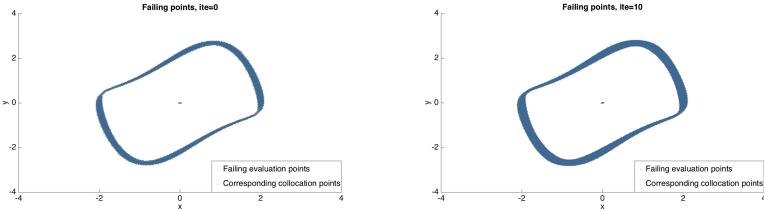


Fig. 11. The figures show the evaluation points y around each collocation point, where $v'(y) > \gamma = -0.5$; left: iteration 0 and right: iteration 10. The chain-recurrent sets are clearly visible in both iterations.

Note that there is a considerable improvement since [31]. Let us notice that we had improved the results of [31] in [32] by using the almost-normalized method (1), at a higher computational cost. However, in Fig. 11 we use the same (original) method as in [31]. The difference is in the amount of collocation points and the new distribution of the evaluation points. In particular, even after ten iterations, there are no points apart from the periodic orbit and the equilibrium marked as failing, whereas in previous methods, more and more “noise” was added at other parts of the phase space.

4.3 Homoclinic Orbit

As in [31], we also consider here the following example with right-hand side

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

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 have used the Wendland function $\psi_{4,2}$ with parameter $c = 1$. 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 (5) with $\alpha = 0.0125$. In this example, we have used the almost-normalized method, i.e. we have replaced \mathbf{f} by $\hat{\mathbf{f}}$ according to (1) with $\delta^2 = 10^{-8}$, and we have used $\gamma = -0.75$.

Figure 12 shows the approximation v_0 of $V'(\mathbf{x}) = -1$ as well as its orbital derivative v'_0 . The function v_0 (Fig. 12, left) clearly displays the stable homoclinic orbit as minimum and the unstable equilibrium at the origin as maximum of v_0 . The orbital derivative v'_0 (Fig. 12, right) is mostly -1 apart from the homoclinic orbits and the origin. The orbital derivative is clearly discontinuous. In contrast to previous examples, the orbital derivative has large values around the equilibrium at $(1, 0)$, corresponding to the homoclinic orbit.

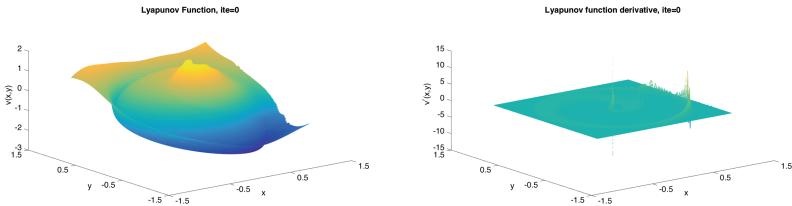


Fig. 12. Complete Lyapunov function for (10), iteration 0. The left-hand figure shows the approximated complete Lyapunov function $v_0(x, y)$. The right-hand figure shows its orbital derivative $v'_0(x, y)$. The Lyapunov function's maximum corresponds to the unstable equilibrium at the origin. The stable homoclinic orbit is a local minimum of v_0 . The orbital derivative v'_0 fails to be negative on the periodic orbit and at the origin, moreover, v'_0 is not continuous.

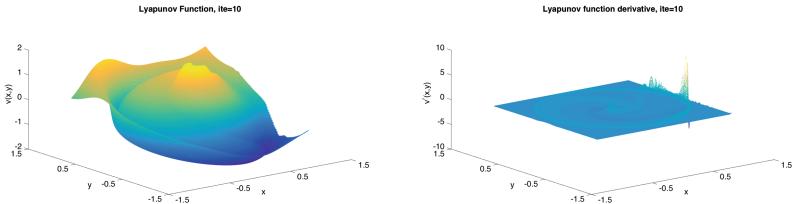


Fig. 13. Complete Lyapunov function for (10), iteration 10. The left-hand figure shows the approximated complete Lyapunov function $v_{10}(x, y)$. The right-hand figure shows its orbital derivative $v'_{10}(x, y)$; now continuous. The Lyapunov function's maximum corresponds to the unstable equilibrium at the origin. The stable homoclinic orbit is a local minimum of v_0 . The orbital derivative v'_0 fails to be negative on the periodic orbit and at the origin.

Figure 13 shows the tenth iteration of the new method. The orbital derivative v'_{10} is now continuous, which can be seen more clearly in Figs. 14 and 15, comparing the iterations 0 and 10 on a part of the phase space. Figures 14 shows a section of the Lyapunov function derivative v'_0 over the chain-recurrent set. Figure 15 shows a section of the previous Fig. 14.

Figure 16 shows the points \mathbf{y} of the evaluation grid, where $v'_i(\mathbf{y}) > \gamma$ with $\gamma = -0.75$, which approximate the chain-recurrent set. We find the two expected connected components of the chain-recurrent set already in iteration 0; and the sets do not change much when considering iteration 10.

We notice an important improvement when compared with [32] in which we notice that the amount of areas incorrectly marked as failing points is almost negligible.

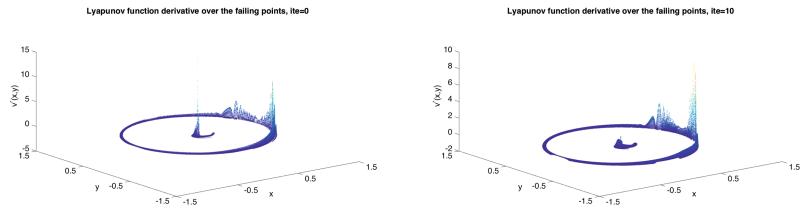


Fig. 14. Complete Lyapunov function derivative v'_0 (left) and v'_{10} for (10) over the homoclinic orbit and the origin for iterations 0 and 10, $\gamma = -0.75$.

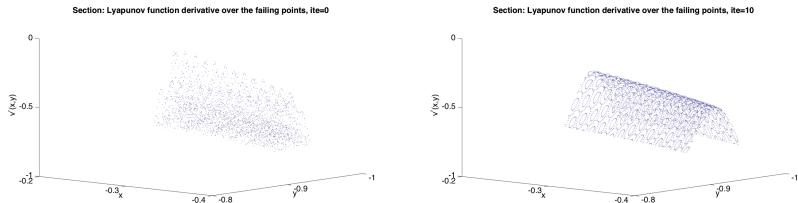


Fig. 15. Complete Lyapunov function derivative v'_0 (left) and v'_{10} for (10) over a part of the homoclinic orbit for iterations 0 and 10, $\gamma = -0.75$.

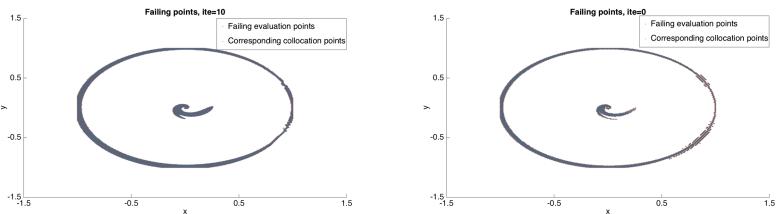


Fig. 16. The figures show the evaluation points \mathbf{y} around each collocation point, where $v'(\mathbf{y}) > \gamma = -0.75$; left: iteration 0 and right: iteration 10. The chain-recurrent sets are clearly visible in both iterations.

4.4 Application to Biology

As a final example, we consider a system from [1], Eqs. (6) and (7), which models a network of two genes that suppress each other to form a double-negative feedback loop. The general model is given by

$$\mathbf{f}(x, y) = \begin{pmatrix} B_X + \frac{\text{fold}_{YX} K_{DYX}^{n_H}}{K_{DYX} + y^{n_H}} - \deg_X x \\ B_Y + \frac{\text{fold}_{XY} K_{DXY}^{n_H}}{K_{DXY} + x^{n_H}} - \deg_Y y \end{pmatrix}. \quad (11)$$

where x and y represent the concentrations of the two gene products; for the meaning of the parameters, see [1]. We use the parameter values from [1] to obtain the following system

$$\mathbf{f}(x, y) = \begin{pmatrix} \frac{2}{10} + 2 \left(\frac{\left(\frac{7}{10}\right)^4}{\left(\frac{7}{10}\right)^4 + y^4} \right) - x \\ \frac{2}{10} + 2 \left(\frac{\left(\frac{5}{10}\right)^4}{\left(\frac{5}{10}\right)^4 + x^4} \right) - y \end{pmatrix}. \quad (12)$$

In [1], the authors compute a complete Lyapunov function, which they call quasi-potential, by numerical integration along solution trajectories with many different initial conditions. The resulting energy landscape, the plot of $v(x, y)$, is used to analyze the development and stability of cellular states.

Although this is a biological system in which only non-negative values of x and y are of biological interest, we set our collocation points in the region $(-1, 6) \times (-1, 6) \subset \mathbb{R}^2$ with a hexagonal grid (5) with $\alpha = 0.041$. We use the “almost” normalized method (1) with $\delta^2 = 10^{-8}$ and $\gamma = -0.5$. Our settings gave us a total amount of 35,728 collocation points and 2,286,592 evaluation points.

This system has three equilibria

$$\mathbf{z}_1 = (0.223344, 2.12342), \quad \mathbf{z}_2 = (0.542514, 1.03822), \quad \mathbf{z}_3 = (2.18526, 0.205466).$$

\mathbf{z}_1 and \mathbf{z}_3 are stable nodes; the eigenvalues of the corresponding Jacobians are $\lambda_1 = -1.23942$ and $\lambda_2 = -0.760578$, and $\lambda_1 = -1.05331$, $\lambda_2 = -0.946693$, respectively \mathbf{z}_2 is a saddle with corresponding eigenvalues of the Jacobian $\lambda_1 = -2.98147$, $\lambda_2 = 0.981468$.

Figure 17 shows the approximation v_0 of $V'(\mathbf{x}) = -1$ as well as its orbital derivative v'_0 . The function v_0 (Fig. 17, left) clearly displays the two stable equilibria as minima and the unstable equilibrium as saddle of v_0 . The orbital derivative v'_0 (Fig. 17, right) is mostly -1 apart from the three equilibria; this can be seen clearer in Fig. 20. Indeed, if we take a close up to the area where the critical values are, we see that the points after the tenth iteration are clearer distinguishable, see Fig. 20. Figure 18 shows the tenth iteration of the new method, which gives a similar picture.

The chain-recurrent set, given by the failing points, is shown in Fig. 19. We can identify the three critical points \mathbf{z}_1 , \mathbf{z}_2 and \mathbf{z}_3 easily with our method.

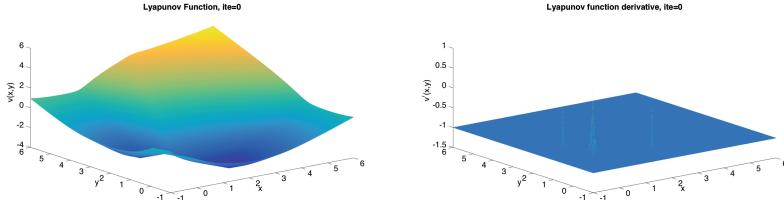


Fig. 17. Complete Lyapunov function for (12), iteration 0. The left-hand figure shows the approximated complete Lyapunov function $v_0(x, y)$. The right-hand figure shows its orbital derivative $v'_0(x, y)$. The complete Lyapunov function's saddle point corresponds to the unstable equilibrium while the two asymptotically stable equilibria are local minima of v_0 . The orbital derivative v'_0 fails to be negative at the three equilibria, which can be better seen in Fig. 20.

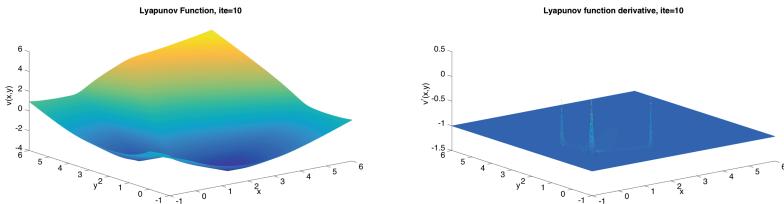


Fig. 18. Complete Lyapunov function for (12), iteration 10. The left-hand figure shows the approximated complete Lyapunov function $v_{10}(x, y)$. The right-hand figure shows its orbital derivative $v'_{10}(x, y)$; now continuous. The Lyapunov function's saddle point corresponds to the unstable equilibrium while the two asymptotically stable equilibria are local minima of v_0 . The orbital derivative v'_0 fails to be negative at the three equilibria.

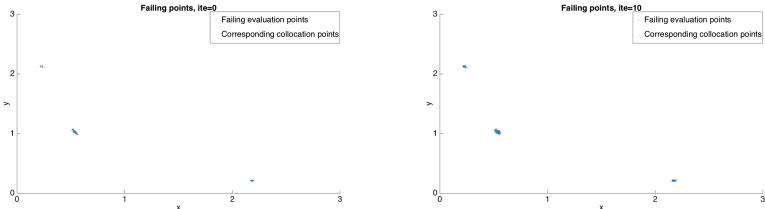


Fig. 19. The figures show the evaluation points \mathbf{y} around each collocation point, where $v'(\mathbf{y}) > \gamma = -0.5$; left: iteration 0 and right: iteration 10. The three equilibria are clearly visible in both iterations.

As in [1], we analyze the contour plots of v_{10} for iteration 10, Fig. 21. They are very similar to the results obtained in [1].

We see that our numerical computation of a complete Lyapunov function of system (12) is capable of reproducing the behaviour of the biological system. Furthermore, we can see that our method reproduces the results from [1] but

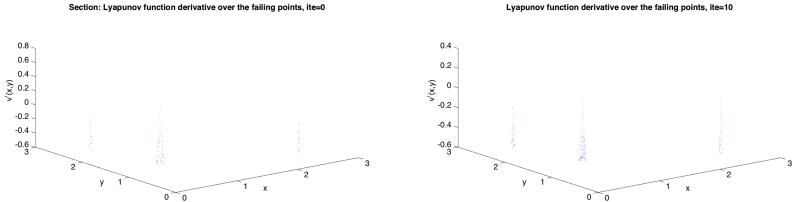


Fig. 20. Complete Lyapunov function derivative for iteration 0 (left) and iteration 10 (right) in the part of the phase space where it differs from -1 and in which the equilibria can be seen.

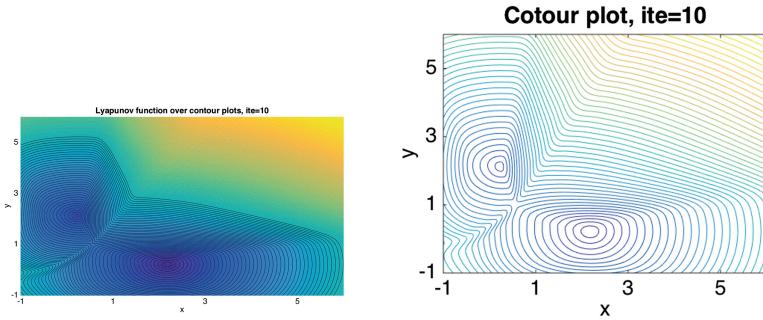


Fig. 21. Contour plots of the complete Lyapunov function v_{10} for iteration 10 and system (12).

without the need of computing numerical integration for particular solutions which shows that our method is a powerful to solve problems in other disciplines.

5 Conclusions and Outlook

In this paper we have presented several improvements to our previous methodology introduced in [31]. First of all, we used an evaluation grid which homogeneously distributes its points around the collocation points and thus avoids favouring specific regions. As seen in Fig. 1, in our previous method [31], an inhomogeneous distribution would favour the weighting at the regions in which the points are closer and hinder the regions in which the points are more distant. With a homogeneous grid, all points have the same weighting. Secondly, we also enhanced the results for (9) previously presented in [31] without using the “almost”-normalized method presented in [32]. Moreover, we are able to avoid overestimating the chain-recurrent set due to “noise”.

Using the “almost”-normalized method presented in [32] with the averaging idea presented in [34] we even managed to enhance the results previously seen in [31,32] for the system (10). As can be seen in Fig. 16, after 10 iterations the amount of “noise” that harms our definition of the chain-recurrent set, is reduced. We can then conclude that averaging all around the collocation point,

instead of in just one direction, has a better impact on determining the chain-recurrent sets. The reason is that close to the failing points, there are small areas that fail as well. Accordingly, if we take only one direction the failing points close a particular failing collocation point and away from that direction will not be noticed. However with a circumference they will be noticed and weighted. The directional evaluation grid has, however, another advantages. It favours the direction of the trajectories and it avoids exponential growth of the evaluation points.

Summarizing, in this paper we were capable of obtaining a good estimate for the chain-recurrent set for different systems using both the “almost”-normalized method presented in [32] as well as the common method. This shows that if the system does not present high variations of its speed the common method works as well.

In all our computations we have obtained very clear chain-recurrent sets with small “noise” or none of it. In short, we have presented an enhanced method to compute a complete Lyapunov function fulfilling the conditions for its derivative which is smooth and continuous. That allows to better determine the qualitative behaviour of a given ODE, using both the values of the complete Lyapunov function and its orbital derivative. Furthermore, we provide complete Lyapunov functions V that are constant along solutions in the chain-recurrent set, i.e. the orbital derivative is zero ($V'(\mathbf{x}) = 0$), and are strictly decreasing along solutions in the gradient-like set, i.e. $V'(\mathbf{x}) < 0$. We have shown that the method is a useful tool for applications and can determine the energy landscape of a quasi-potential.

Acknowledgements. First author in this paper is supported by the Icelandic Research Fund (Rannís) grant number 163074-052, Complete Lyapunov functions: Efficient numerical computation. Special thanks to Dr. Jean-Claude Berthet for all his good comments and advices on C++.

References

1. Bhattacharya, S., Zhang, Q., Andersen, M.E.: A deterministic map of Waddington’s epigenetic landscape for cell fate specification. *BMC Syst. Biol.* **5**, 85 (2011)
2. Doban A.: Stability domains computation and stabilization of nonlinear systems: implications for biological systems. Ph.D. thesis. Eindhoven University of Technology (2016)
3. Doban A., Lazar M.: Computation of Lyapunov functions for nonlinear differential equations via a Yoshizawa-type construction. In: 10th IFAC Symposium on Nonlinear Control Systems NOLCOS 2016, Monterey, California, USA, 23–25 August 2016, IFAC-PapersOnLine, 2016, vol. 49, no. 18, pp. 29–34 (2016). ISSN 2405-8963
4. Anderson, J., Papachristodoulou, A.: Advances in computational Lyapunov Analysis using sum-of-squares programming. *Discrete Contin. Dyn. Syst. Ser. B* **20**, 2361–2381 (2015)
5. Auslander, J.: Generalized recurrence in dynamical systems. *Contr. Diff. Eq.* **3**, 65–74 (1964)

6. Björnsson J., Giesl P., Hafstein S.: Algorithmic verification of approximations to complete Lyapunov functions. In: Proceedings of the 21st International Symposium on Mathematical Theory of Networks and Systems (no. 0180), pp. 1181–1188 (2014)
7. Björnsson J., Giesl P., Hafstein S., Kellett C., Li H.: Computation of continuous and piecewise affine Lyapunov functions by numerical approximations of the Massera construction. In: Proceedings of the CDC 53rd IEEE Conference on Decision and Control (2014)
8. Björnsson, J., Giesl, P., Hafstein, S., Kellett, C., Li, H.: Computation of Lyapunov functions for systems with multiple attractors. *Discrete Contin. Dyn. Syst.* **9**, 4019–4039 (2015)
9. Ban, H., Kalies, W.: A computational approach to Conley's decomposition theorem. *J. Comput. Nonlinear Dyn.* **1**, 312–319 (2006)
10. Isolated Invariant Sets and the Morse Index, C. Conley, American Mathematical Society, CBMS Regional Conference Series no. 3 (1978)
11. Dellnitz, M., Froyland, G., Junge, O.: The algorithms behind GAIO - set oriented numerical methods for dynamical systems. In: Ergodic Theory Analysis and Efficient Simulation of Dynamical Systems, pp. 145–174, 805–807. Springer, Berlin (2001)
12. Dellnitz, M., Junge, O.: Set oriented numerical methods for dynamical systems. In: Handbook of Dynamical Systems, North-Holland Amsterdam, vol. 2, pp. 221–264 (2002)
13. Giesl, P.: Construction of Global Lyapunov Functions Using Radial Basis Functions. Lecture Notes in Math., vol. 1904. Springer (2007)
14. Giesl, P., Wendland, H.: Meshless collocation: error estimates with application to Dynamical Systems. *SIAM J. Numer. Anal.* **45**, 1723–1741 (2007)
15. Giesl, P., Hafstein, S.: Review of computational methods for Lyapunov functions. *Discrete Contin. Dyn. Syst. Ser. B* **20**, 2291–2331 (2015)
16. Goullet, A., Harker, S., Mischaikow, K., Kalies, W., Kasti, D.: Efficient computation of Lyapunov functions for morse decompositions. *Discrete Contin. Dyn. Syst. Ser. B* **20**, 2419–2451 (2015)
17. Hafstein, S.: An algorithm for constructing Lyapunov functions. *Electron. J. Diff. Eqns.* (2007)
18. Hsu, C.S.: Cell-to-Cell Mapping. Applied Mathematical Sciences, vol. 64, Springer, New York (1987)
19. Hurley, M.: Noncompact chain recurrence and attraction. *Proc. Am. Math. Soc.* **115**, 1139–1148 (1992)
20. Hurley, M.: Lyapunov functions and attractors in arbitrary metric spaces. *Proc. Am. Math. Soc.* **126**, 245–256 (1998)
21. Johansson, M.: Piecewise linear control systems, Ph.D. thesis: Lund University Sweden (1999)
22. Johansen, T.: Computation of Lyapunov functions for smooth nonlinear systems using convex optimization. *Automatica* **36**, 1617–1626 (2000)
23. Kamyar, R., Peet, M.: Polynomial optimization with applications to stability analysis and control - an alternative to sum of squares. *Discrete Contin. Dyn. Syst. Ser. B* **20**, 2383–2417 (2015)
24. Kalies, W., Mischaikow, K., VanderVorst, R.: An algorithmic approach to chain recurrence. *Found. Comput. Math.* **5**, 409–449 (2005)
25. Marinósson, S.: Lyapunov function construction for ordinary differential equations with linear programming. *Dyn. Syst. Int. J.* **17**, 137–150 (2002)

26. Narcowich, F.J., Ward, J.D., Wendland, H.: Sobolev bounds on functions with scattered zeros with applications to radial basis function surface fitting. *Math. Comp.* **74**, 743–763 (2005)
27. Wendland, H.: Error estimates for interpolation by compactly supported Radial Basis Functions of minimal degree. *J. Approx. Theory* **93**, 258–272 (1998)
28. Hurley, M.: Chain recurrence semiflows and gradient. *J. Dyn. Diff. Eq.* **7**, 437–456 (1995)
29. Lyapunov A. M.: The general problem of the stability of motion. *Int. J. Control* **55**, 521–790 (1992). Translated by A. T. Fuller from Édouard Davaux's French translation (1907) of the 1892 Russian original
30. Osipenko, G.: Dynamical Systems Graphs and Algorithms. Lecture Notes in Mathematics, vol. 1889. Springer, Berlin (2007)
31. Argáez, C., Giesl, P., Hafstein, S.: Analysing dynamical systems towards computing complete Lyapunov functions. In: Proceedings of the 7th International Conference on Simulation and Modeling Methodologies Technologies and Applications, SIMULTECH 2017, Madrid, Spain (2017)
32. Argáez, C., Giesl, P., Hafstein, S.: Computational approach for complete Lyapunov functions. In: Proceedings in Mathematics and Statistics. Springer (2018, accepted for publication)
33. Argáez, C., Giesl, P., Hafstein, S.: Iterative construction of complete Lyapunov functions. Submitted
34. Argáez, C., Giesl, P., Hafstein, S.: Computation of complete Lyapunov functions for three-dimensional systems. Submitted
35. Conley, C.: The gradient structure of a flow I. *Ergodic Theory Dyn. Syst.* **8**, 11–26 (1988)
36. Wendland, H.: Scattered Data Approximation. Cambridge Monographs on Applied and Computational Mathematics, vol. 17. Cambridge University Press, Cambridge (2005)
37. Krauskopf, B., Osinga, H., Doedel, E. J., Henderson, M., Guckenheimer, J., Vladimirsky, A., Dellnitz, M., Junge, O.: A survey of methods for computing (un)stable manifolds of vector fields. *Int. J. Bifur. Chaos Appl. Sci. Eng.* **15**, 763–791 (2015)