Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 108C (2017) 655–664

Procedia Computer Science

International Conference on Computational Science, ICCS 2017, 12-14 June 2017, Zurich, Switzerland

Efficient Lyapunov Function computation for systems with multiple exponentially stable equilibria

Jóhann Björnsson[1] *and Sigurdur F. Hafstein[2]

[1] Reykjavik University, School of Science and Engineering, Menntavegur 1, 101 Reykjavik, Iceland
johannbj@ru.is

[2] University of Iceland, Faculty of Physical Sciences, Dunhagi 5, 107 Reykjavik, Iceland
shafstein@hi.is

Abstract

Recently a method was presented to compute Lyapunov functions for nonlinear systems with multiple local attractors [5]. This method was shown to succeed in delivering algorithmically a Lyapunov function giving qualitative information on the system's dynamics, including lower bounds on the attractors' basins of attraction. We suggest a simpler and faster algorithm to compute such a Lyapunov function if the attractors in question are exponentially stable equilibrium points. Just as in [5] one can apply the algorithm and expect to obtain partial information on the system dynamics if the assumptions on the system at hand are only partially fulfilled. We give four examples of our method applied to different dynamical systems from the literature.

Peer-review under responsibility of the scientific committee of the International Conference on Computational Science

Keywords: Lyapunov function, multiple equilibria, nonlinear system, algorithm

1 Introduction

We consider continuous time systems given by ordinary differential equations

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}), \tag{1.1}$$

where $\mathbf{f} \in C^2(\mathbb{R}^d, \mathbb{R}^d)$ is two-times continuously differentiable. We denote the solution to (1.1) started at $\boldsymbol{\xi}$ at time $t = 0$ by $t \mapsto \boldsymbol{\phi}(t, \boldsymbol{\xi})$. A so-called complete Lyapunov functions for the system (1.1) is a continuous function from the state-space to the real numbers that characterizes the decomposition of the flow into a gradient-like part and a chain-recurrent part [1, 7, 18]. For a more accessible overview of this fact, sometimes referred to as the *Fundamental Theorem of Dynamical Systems* cf. e.g. [25, 26]. A complete Lyapunov function is decreasing along solution trajectories on the gradient-like part of the flow and constant on the transitive components of the chain-recurrent part.

*Björnsson is supported by The Icelandic Research Fund, grant nr. 130677-052.

1877-0509 © 2017 The Authors. Published by Elsevier B.V.
Peer-review under responsibility of the scientific committee of the International Conference on Computational Science
10.1016/j.procs.2017.05.285

Whereas there have been numerous suggestions of how to compute Lyapunov function for systems on a domain containing one stable equilibrium, cf. e.g. [14] for a recent review, there have been much fewer publications on the numerical construction of Lyapunov functions with a more complicated chain-recurrent set.

In [5] a method was presented to compute a function V resembling a complete Lyapunov function for the system (1.1) on a compact subset of its state-space $\mathcal{D} \subset \mathbb{R}^d$, which is allowed to contain multiple attractors. In this method one first computes outer approximations of the attractors using a graph theoretic method [19, 15] followed by a subsequent numerical computation of a Massera-like Lyapunov function candidate [24], see [20] for an overview and classification of the different construction methods. The candidate is then used to parameterize a continuous and piecewise affine (CPA) Lyapunov function, of which the decrease condition along solution trajectories can be verified exactly by checking a certain set of linear inequalities. This set of linear inequalities comes from the so-called CPA method to compute Lyapunov functions, in which linear optimization is used to parameterize a CPA Lyapunov function satisfying these linear inequalities [23, 16, 13]. This method has been adapted to different kinds of systems like differential inclusions [2] and discrete-time systems [12] and to systems with different stability properties like ISS stability [21] and control systems [3]. The main advantage of the CPA method is that it delivers a function that is guaranteed to satisfy the conditions for a Lyapunov function exactly and its main drawback is that as it involves solving a large linear programming problem it is not particularly fast. It has therefore been used in combination with other faster methods to compute Lyapunov functions, the main idea being to compute a Lyapunov function candidate by the faster method and then use the CPA method to verify if the candidate indeed satisfies all conditions of a Lyapunov function. For this methodology cf. e.g. [4, 17, 22, 11] and the paper [5], on which we base this work.

1.1 Notation:

We write vectors $\mathbf{x} \in \mathbb{R}^d$ in boldface, $\|\mathbf{x}\|$ denotes the Euclidian norm of $\mathbf{x}$, and $\mathcal{B}_\varepsilon(\mathbf{x}) := \{\mathbf{y} \in \mathbb{R}^d : \|\mathbf{x} - \mathbf{y}\| < \varepsilon\}$ is an open ball centered at $\mathbf{x}$ with radius $\varepsilon > 0$. We write subsets $\mathcal{K} \subset \mathbb{R}^d$ in calligraphic and its interior is denoted by $\mathcal{K}^\circ$ and its closure by $\overline{\mathcal{K}}$. C^m stands for the set of all m-times continuously differentiable functions, the domain and codomain should always be obvious from the context. We denote by $\mathcal{A}(\mathbf{y}) := \{\mathbf{x} \in \mathbb{R}^d : \limsup_{t\to\infty} \|\boldsymbol{\phi}(t, \mathbf{x}) - \mathbf{y}\| = 0\}$ the *basin of attraction* of a stable equilibrium $\mathbf{y}$. A Lipschitz constant $L > 0$ for $\mathbf{f}$ on a set $\mathcal{K}$ is a constant such that $\|\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|$ for all $\mathbf{x}, \mathbf{y} \in \mathcal{K}$. If there exists a Lipschitz constant for $\mathbf{f}$ on every compact set $\mathcal{K} \subset \mathbb{R}^d$, $\mathbf{f}$ is said to be *locally Lipschitz*.

2 The Method

In [5] one first computes outer approximations $\mathcal{F}_i$ of the local attractors Ω_i, $i = 1, 2, \ldots, N$, of the system (1.1) contained in some predefined compact set $\mathcal{D} \subset \mathbb{R}^d$ of interest. Then one defines a sufficiently smooth functions $\gamma : \mathcal{D} \to \mathbb{R}^+$ ($\mathbb{R}^+ := [0, \infty)$) such that $\gamma(\mathbf{x}) = 0$ whenever $\mathbf{x} \in \bigcup_{i=1}^N \mathcal{F}_i$ and $\gamma(\mathbf{x}) > 0$ otherwise. As shown in [5, Theorem 3.2] the function

$$W(\mathbf{x}) := \int_0^T \gamma(\boldsymbol{\phi}(t, \mathbf{x}))dt$$

then has a negative orbital derivative

$$W'(\mathbf{x}) := \limsup_{h\to 0+} \frac{W(\boldsymbol{\phi}(h, \mathbf{x})) - W(\mathbf{x})}{h} \quad \Big(= \nabla W(\mathbf{x}) \cdot \mathbf{f}(\mathbf{x}) \;\text{ if } W \in C^1 \Big)$$