Numerical Methods and Software for Solving Boundary Value ODEs for Bifurcation Analysis

by

Lixin Liu

B.Sc., Beijing Normal University, 1985

M.Sc., Simon Fraser University, 1988

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
in the Department
of
Mathematics & Statistics

SIMON FRASER UNIVERSITY
December 1993

APPROVAL

Name: Lixin Liu

Degree: Doctor of Philosophy

Title of thesis: Numerical Methods and Software for Solving Boundary Value ODEs for Bifurcation Analysis

Examining Committee: Dr. B. R. Alspach
Chair

Dr. R. D. Russell
Senior Supervisor

Dr. G. Bojadziev

Dr. T. Tang

Dr. R. W. Lardner

Dr. E. J. Doedel
External Examiner
Department of Computer Science
Concordia University

Date Approved: December 7, 1993

PARTIAL COPYRIGHT LICENSE

Title of Thesis/Project/Extended Essay

Numerical Methods and Software for Solving

Boundary Value ODEs for Bifurcation Analysis

Author: ______________________

(signature)

Lixin Liu

(name)

Nov. 25, 1993

(date)

Abstract

The development of numerical methods and software for dynamical systems has become a very active research topic during the last two decades. One of the most successful examples is the software AUTO. This software is capable of doing bifurcation analysis, tracing the different solution branches and determining the stability of the solutions for steady state problems as well as boundary value problems. However, robustness has been sacrificed in some places in order to achieve efficiency (e.g., in the linear system solver for BVPs), and better numerical methods can sometimes be implemented to improve the stability, accuracy and speed of the code (e.g., in the numerical methods for computing periodic solutions and connecting orbit). In this thesis, some new numerical methods are analyzed and implemented in AUTO. These numerical methods include a new linear system solver for the boundary value problems and a new algorithm for computing periodic solutions and connecting orbits (homoclinic orbits and heteroclinic orbits). A comparison between the original AUTO and our implementation is done. Finally, a pseudo-spectral method for computing connecting orbits is investigated.

Acknowledgements

I am deeply indebted to my senior supervisor, Dr. Robert D. Russell for his guidance, encouragement and patience during the preparation of this thesis.

I would also like to thank Dr. Weiwei Sun, Dr. Weizhang Huang and members of the examining committee for their important comments on this thesis, Dr. Tim Swartz, Ms. Lisa Kristiansen, Ms. Helen Verrall and Mr. Martin Gilchrist for the correction of my English. Thanks also go to Mrs. Sylvia Holmes and Mr. Yingjie Liu for their help.

Finally, financial support from Dr. R. D. Russell and Simon Fraser University is also appreciated.

Dedication

To

Hua

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Nonlinear phenomena appear virtually everywhere in our daily life. These problems arise not only in the traditional areas in mechanics, physics and chemistry, but also increasingly in mathematical modeling of biological, ecological, social and economic systems. As a result, the research in nonlinear dynamical systems has become increasingly attractive to scientists and engineers, analysts and experimentalists in many disciplines. Many of these problems are remarkably difficult and their solutions have extremely rich structures. Despite the effort by many great scientists, analytical solutions for nonlinear problems can only be obtained for some special cases. Most of the problems can only be solved numerically.

With the increasing computer power during the past 30 years, numerical software has become extremely useful for studying the problems in dynamical systems. Therefore, numerical analysis plays an important role in producing efficient and robust numerical methods, algorithms and software for the study of dynamical systems. Various numerical software has been developed during the 1970s and 1980s, such as, BIFPACK [65] written by Seydel, AUTO [25, 28] by Doedel, COLCON [7] by Bader and Kuker, and several other packages. Among them, AUTO is one of the most successful ones due to its capacity, efficiency, flexibility, well-designed user interface and its graphics facilities. It has been widely used in the engineering and applied mathematics research and development. This software is capable of doing bifurcation analysis for steady state problems, boundary value problems and discrete systems. Using a pseudo-arclength continuation method, it is able to compute the solution branches and to determine their stability and bifurcation. AUTO was first released in 1980 and was revised in 1986. Since then, there have been a number of improvements. Fairgrieve and Jepson [31] improve the method for computing Floquet multipliers of the periodic solutions without a significant increase of the cost. Taylor [70] has developed a graphical user interface software for SGI machines so that users can use AUTO interactively.

Despite all the success of AUTO, there are still some places in the code where robustness has been sacrificed in order to achieve efficiency. One example is the numerical algorithm for solving the linear systems for boundary value problems, where AUTO uses a potentially unstable solver. In many other places, better numerical algorithms and implementations should be used to improve the efficiency and stability of the code. Some examples include the numerical continuation for the periodic solutions, connecting orbits, limit points, period doubling bifurcation points and tori bifurcation points. One purpose of this thesis is to address the problems of instability and inefficiency of the numerical methods in AUTO and to improve its performance by incorporating stable and efficient numerical algorithms into the software.

Dynamical systems described by ordinary differential equations and finite difference equations are called *continuous* and *discrete* dynamical systems, respectively. In the real-life models, people are usually concerned about the long-term behaviour of the solutions. Some of these solutions may be very complicated. However, they stay bounded on certain regions of the domain for all time. One interesting problem in studying dynamical systems is to compute the different types of *invariant manifolds* and their *stability*. An invariant manifold defines a certain region of the domain on which, if the initial conditions are given on it, the solution remains for all time. The stability characterizes the long-term behaviour of the solution near the manifold. If this solution always stays close enough to the invariant manifold, the invariant manifold is *stable*; otherwise, it is *unstable*. Invariant manifolds for continuous dynamical systems include steady state solutions, periodic solutions, connecting orbits (either homoclinic or heteroclinic orbits), invariant tori, and other more complicated structures like strange attractors.

The properties of an invariant manifold may change suddenly due to the change of the condition (usually a change of parameters) of the system; this phenomenon is called *bifurcation*. The bifurcation brings the qualitative change for the solutions of the model, and it is always important to be able to identify such a case. Bifurcation points are usually determined by continuously computing the invariant manifolds and monitoring the properties (such as the stability) of the manifolds along the change of the conditions or parameters. Therefore, in bifurcation analysis, it is very important to develop numerical methods for the computation and continuation of the invariant manifolds as well as the determination of the stability for the invariant manifolds. For the continuous dynamical systems, numerical methods for computing steady state solutions involve solving some nonlinear algebraic equations where Newton's methods are usually used. The stability and bifurcation for the solution branch can be obtained by studying the eigen-structures for linearized problems. To compute periodic solutions and connecting orbits, certain nonlinear boundary value ODEs need to be solved. These boundary value problems may contain some types of singularities. For example, computing a connecting orbit involves solving a BVP defined on an infinite domain, where the standard numerical BVP solvers may not be suitable. In this thesis, we will try to discuss some of the issues related to the stability of solving boundary value ODEs and the numerical computations of homoclinic and heteroclinic orbits.

In chapter 2, we will briefly review the stability issues for steady state problems, initial value problems, and boundary value problems. Our discussion about bifurcations is focused on the steady state solutions and periodic solutions. Many types of bifurcation will be demonstrated with examples. Numerical methods for the continuation, bifurcation analysis and their implementation in AUTO will also be studied.

The ability to solve boundary value ODEs is one of the essential parts for the study of dynamical systems. Problems like computing periodic orbits and connecting orbits all involve solving BVPs. Many numerical methods for the discretization of a BVP such as the multiple shooting method, the finite difference method and the collocation method usually produce a linear system with a special structure, and it should be solved efficiently and accurately. The linear system solver for the boundary value problems in AUTO has its own advantages. It is of capable of dealing with the non-separated boundary conditions, the integral conditions and the scalar parameters directly. It is also able to obtain the Floquet multipliers for periodic solutions with little extra cost and thus to determine the stability and bifurcation for periodic solutions. However, under certain circumstance, this algorithm is known to be equivalent to a compactification algorithm, and may suffer from poor numerical stability. In chapter 3, a new linear system solver based on the QR factorization is studied and implemented in AUTO. We will discuss the stability and efficiency of this solver as opposed to AUTO's original linear solver with several numerical examples.

Structural changes in dynamical systems are often related to the appearance or disappearance of solutions connecting two stationary points. These solutions can be either homoclinic orbits or heteroclinic orbits. They are one of the important types of invariant manifolds in the continuous dynamical systems and are extremely useful in the study of the structural stability of dynamical systems. Both homoclinic orbits and heteroclinic orbits can arise as the limiting cases of the periodic orbits where the period tends to infinity. The applications of the heteroclinic orbits can also be found in travelling wave problems for nonlinear parabolic partial differential equations. However, numerical computation of such connecting orbits can be very difficult due to solving the boundary value problems on a infinite domain. In chapter 4, we will study the applications of the homoclinic and heteroclinic orbits, and the conditions for the existence and uniqueness of such solutions. We will develop a new collocation algorithm based upon the arclength parametrization of the connecting orbits and discuss its convergence properties and implementation in AUTO. Our numerical results will be shown with several examples. In the final part of the chapter, we will apply rational spectral methods to compute the connecting orbits. In this method, we use a rational spectral approach with the implementation of a special type of boundary condition and a phase condition.

In chapter 5, we will summarize our conclusions and provide the information for the future research.

Chapter 2

Continuation and Bifurcation

A continuous dynamical system consists of one or several nonlinear ordinary differential equations and parameters. A series of solutions for a dynamical system with respect to the change of parameters is known as a *solution branch.* To study the properties of the dynamical system, one would like to investigate the qualitative and the quantitative changes of the solution branches. This study requires a numerical algorithm to do the continuation of the solution branches in accordance with change in the parameters. These changes can sometime result in the birth or death of solution branches or the change of the stability for the solutions. Such qualitative change of the solution is called a *bifurcation.* The numerical methods for determining bifurcation points are also very important in studying the dynamical systems. In this chapter, we will review the stability and bifurcation properties for several types of solutions for continuous dynamical systems, and numerical continuation and bifurcation analysis of these systems.

2.1 Stability and Bifurcation

We start our discussion with the following ODE system

$$\mathbf{u}'(t) = \mathbf{f}(\mathbf{u}(t), \boldsymbol{\lambda}) \tag{2.1}$$

where $t \in \mathbf{R}$, $\mathbf{u}, \mathbf{f} \in \mathbf{R}^n$ and $\boldsymbol{\lambda} \in \mathbf{R}^p$. Our interest is to study the properties of the solution $\mathbf{u}(t)$ with respect to the parameter $\boldsymbol{\lambda}$. We call the pair $(\mathbf{u}(t), \boldsymbol{\lambda})$ a *solution branch.* We are particularly interested in the long-term behavior of the solution branches, their local stability and bifurcation. We first define the stability for the *initial value problems.*

Definition 2.1 *A solution* $\mathbf{u}(t)$ *of ODE system* (2.1) *where* $t > a$ *is said to be locally stable if* $\forall\epsilon > 0$, *there is a* $\delta > 0$ *such that any solution* $\hat{\mathbf{u}}(t)$ *satisfying*

$$||\mathbf{u}(a) - \hat{\mathbf{u}}(a)|| < \delta$$

also satisfies

$$||\mathbf{u}(t) - \hat{\mathbf{u}}(t)|| < \epsilon$$

$\forall t > a$. *Otherwise, the solution* $\mathbf{u}(t)$ *is said to be unstable. Moreover,* $\mathbf{u}(t)$ *is called asymptotically stable if it is stable and*

$$\lim_{t\to\infty} ||\mathbf{u}(t) - \hat{\mathbf{u}}(t)|| = 0.$$

2.1.1 Steady State Solutions

A solution of (2.1) satisfying $\mathbf{u}'(t) = \mathbf{0}$ is called a *steady state solution* or *fixed point* or *stationary solution.* Here we assume that there is only one free parameter $\lambda \in \mathbf{R}$. Note that the steady state solution $\mathbf{u}$ is independent of t and the ODE system (2.1) is reduced to a nonlinear algebraic system

$$\mathbf{f}(\mathbf{u}, \lambda) = \mathbf{0}. \tag{2.2}$$

The definition of local stability for steady state solutions follows from Definition 2.1, where $\mathbf{u}(t) \equiv \mathbf{u}$.

If $\mathbf{u}_0$ is a steady state solution of (2.2) at $\lambda = \lambda_0$, then the local stability of $\mathbf{u}_0$ may be determined by the eigenvalues of $\mathbf{f}_\mathbf{u}(\mathbf{u}_0, \lambda_0)$. We assume that $\mathbf{u}_0$ is a hyperbolic fixed point, i.e., that no eigenvalue of $\mathbf{f}_\mathbf{u}(\mathbf{u}_0, \lambda_0)$ has zero real part. If all of the real parts of the eigenvalues are negative, then $\mathbf{u}_0$ is locally asymptotically stable; otherwise, it is unstable. For certain parameter values, if at least one eigenvalue of $\mathbf{f}_\mathbf{u}(\mathbf{u}_0, \lambda_0)$ appears on the imaginary axis, $\mathbf{u}_0$ is a singular point of the solution branch. We will only discuss the following types of singular point: simple bifurcation, quadratic turning point and Hopf bifurcation.

Definition 2.2 $(\mathbf{u}_0, \lambda_0)$ *is called a simple bifurcation point if*

(1) $\mathbf{f}(\mathbf{u}_0, \lambda_0) = \mathbf{0}$;

(2) $\operatorname{rank}(\mathbf{f_u}(\mathbf{u}_0, \lambda_0)) = n - 1$;

(3) $\operatorname{rank}(\mathbf{f_u}(\mathbf{u}_0, \lambda_0) | \mathbf{f}_\lambda(\mathbf{u}_0, \lambda_0)) = n - 1$;

(4) *exactly two branches intersect with two distinct tangents.*

Example 2.1 *Transcritical Bifurcation*

$$u' = \lambda u - u^2.$$

In this example, there are two branches of the steady state solutions $u = 0$ and $u = \lambda$. These two branches intersect at $\lambda = 0$ and change their stability.

Example 2.2 *Supercritical Pitchfork Bifurcation*

$$u' = \lambda u - u^3.$$

In this example, there is only one solution $u = 0$ when $\lambda < 0$ and it is stable; when $\lambda > 0$, the solution $u = 0$ becomes unstable, and two other stable steady state solution branches $u = \pm\sqrt{\lambda}$ appear.

Example 2.3 *Subcritical Pitchfork Bifurcation*

$$u' = \lambda u + u^3.$$

When $\lambda < 0$, the solution $u = 0$ is stable and the solution $u = \pm\sqrt{-\lambda}$ are unstable; when $\lambda > 0$, there is only one solution $u = 0$ and it is unstable.

Definition 2.3 $(\mathbf{u}_0, \lambda_0)$ *is called a simple or quadratic turning point if*

(1) $\mathbf{f}(\mathbf{u}_0, \lambda_0) = \mathbf{0}$;

(2) $\operatorname{rank}(\mathbf{f_u}(\mathbf{u}_0, \lambda_0)) = n - 1$;

(3) $\operatorname{rank}(\mathbf{f_u}(\mathbf{u}_0, \lambda_0)|\mathbf{f}_\lambda(\mathbf{u}_0, \lambda_0)) = n$;

(4) *there exits a parametrization* $(\mathbf{u}(s), \lambda(s))$ *such that* $(\mathbf{u}(s_0), \lambda(s_0)) = (\mathbf{u}_0, \lambda_0)$ *and* $d^2\lambda(s_0)/ds^2 \neq 0$.

Example 2.4 *Turning Point*

$$u' = \lambda - u^2.$$

A steady state solution does not exist for $\lambda < 0$. For $\lambda > 0$, the stable steady state solution $u = \sqrt{\lambda}$ turns into an unstable steady state solution $u = -\sqrt{\lambda}$ at $\lambda = 0$. The turning point is also referred as a *saddle-node bifurcation* or a *limit point* in some literature.

Definition 2.4 $(\mathbf{u}_0, \lambda_0)$ *is called a Hopf bifurcation point if*

(1) $\mathbf{f}(\mathbf{u}_0, \lambda_0) = \mathbf{0}$;

(2) $\mathbf{f_u}(\mathbf{u}_0, \lambda_0)$ *has a simple pair of purely imaginary eigenvalues* $\mu(\lambda_0) = \pm i\beta$ *and no other eigenvalue has zero real part;*

(3) $d(\operatorname{Re}\mu(\lambda_0))/d\lambda \neq 0$.

Example 2.5 *Hopf bifurcation*

$$\begin{aligned} u_1' &= -u_2 + u_1(\lambda - u_1^2 - u_2^2) \\ u_2' &= u_1 + u_2(\lambda - u_1^2 - u_2^2). \end{aligned}$$

The steady state solution $u_1 = u_2 = 0$ is stable when $\lambda < 0$ and unstable when $\lambda > 0$. At this point, a stable branch of periodic solutions $u_1^2 + u_2^2 = \lambda$ appears with period $T = 2\pi$.

Theorem 2.1 *If* $(\mathbf{u}_0, \lambda_0)$ *is a Hopf bifurcation point, then there is a birth of periodic solutions and the initial period is* $T_0 = 2\pi/\beta$.

2.1.2 Initial Value Problems

We now consider the ODE system

$$\mathbf{u}'(t) = \mathbf{f}(\mathbf{u}(t), \boldsymbol{\lambda}) \tag{2.3}$$

with the initial condition

$$\mathbf{u}(a) = \mathbf{u}_0 \tag{2.4}$$

where $t > a$, $\mathbf{u} \in \mathbf{R}^n$, $\mathbf{f} \in \mathbf{R}^n$, $\lambda \in \mathbf{R}^p$. The stability of an IVP is defined in Definition 2.1. We first study the stability condition for the linear initial value problem

$$\mathbf{u}'(t) = A(t)\mathbf{u}(t) + \mathbf{q}(t), \tag{2.5}$$

where $t > a$, $A(t) \in \mathbf{R}^{n\times n}$, $\mathbf{u} \in \mathbf{R}^n$ and $\mathbf{q}(t) \in \mathbf{R}^n$. The initial condition is given by (2.4).

Definition 2.5 *A matrix $Y(t;\tau) \in \mathbf{R}^{n\times n}$ is called a fundamental solution (matrix) of differential equation system* (2.5) *if* $\forall \tau > a$

$$\begin{aligned} Y'(t;\tau) &= A(t)Y(t;\tau), \qquad t > a &(2.6)\\ Y(\tau;\tau) &= I. &(2.7)\end{aligned}$$

We will also refer to $Y(t) := Y(t;\tau) \in \mathbf{R}^{n\times n}$ as a fundamental solution if $Y(t)$ satisfies (2.6) but not necessarily (2.7).

Theorem 2.2 *Let $Y(t) := Y(t;a)$ be a fundamental solution of the ODE* (2.5). *Then the solution $\mathbf{u}(t)$ of* (2.5) *and* (2.4) *is stable if and only if $\sup_{t\geq a} \|Y(t)\|$ is bounded. Moreover, $\mathbf{u}(t)$ is asymptotically stable if and only if $\|Y(t)\| \to 0$ as $t \to \infty$.*

To investigate the stability for the nonlinear IVP (2.3) and (2.4), we consider the linearization of the ODE (2.3). Let $\hat{\mathbf{u}}(t)$ be a solution of (2.3) with $\hat{\mathbf{u}}(a) \approx \mathbf{u}_0$, and $\mathbf{v}(t) = \mathbf{u}(t) - \hat{\mathbf{u}}(t)$. We define the *variational equation* for (2.3) as

$$\mathbf{v}'(t) = A(t)\mathbf{v}(t) \tag{2.8}$$

where $A(t) := \mathbf{f_u}(\mathbf{u}(t), \lambda)$.

Theorem 2.3 *Suppose that*

$$\frac{\|\mathbf{r}(t,\mathbf{u},\hat{\mathbf{u}})\|}{\|\mathbf{v}(t)\|} \to 0 \quad \text{as} \quad \|\mathbf{v}(t)\| \to 0$$

where

$$\mathbf{r}(t,\mathbf{u},\hat{\mathbf{u}}) := \mathbf{f}(\mathbf{u}(t),\lambda) - \mathbf{f}(\hat{\mathbf{u}}(t),\lambda) + A(t)[\mathbf{u}(t) - \hat{\mathbf{u}}(t)]$$

is the remainder term. If the fundamental solution $Y(t;a)$ satisfies

$$\|Y(t;a)\| < K, \qquad t \geq a$$

for some constant K, then the solution $\mathbf{u}(t)$ is stable. Moreover, if

$$\lim_{t\to\infty} \|Y(t;a)\| = 0,$$

$\mathbf{u}(t)$ *is asymptotically stable.*

2.1.3 Periodic Solutions

A solution $\mathbf{u}(t)$ of the equation (2.1) is called a *periodic solution* or *periodic orbit* if there exists $T > 0$ such that

$$\mathbf{u}(t+T) = \mathbf{u}(t) \tag{2.9}$$

for all t. T is called the *period*. Throughout this thesis, we shall assume that T is the "minimum" period, i.e., T is the smallest positive constant satisfying (2.9). Obviously, a steady state solution is also a periodic solution with period any positive number. However, we are interested in the time-dependent periodic solutions, and we exclude the steady state solutions in our definition. Periodic solutions are an important type of invariant manifold in continuous dynamical systems and have many real-life applications.

The definition of stability for periodic solutions also follows from Definition 2.1 and it can be determined by the eigenvalues of the *monodromy matrix.*

Definition 2.6 *Let $\mathbf{p}(t;\lambda_0)$ be a periodic solution of* (2.1) *with period T, and let $Y(t;0) \in \mathbf{R}^{n\times n}$ with $Y(0;0) = I$ be a fundamental solution matrix of the variational equation*

$$\mathbf{v}'(t) = \mathbf{f_u}(\mathbf{p}(t;\lambda_0);\lambda_0)\mathbf{v}(t). \tag{2.10}$$

Then $Y(T;0)$ is called a monodromy matrix, and its eigenvalues are called the Floquet multipliers.

Note that $\mathbf{u}(0) = \mathbf{u}(T) = Y(T;0)\mathbf{u}(0)$, so 1 is an eigenvalue of $Y(T;0)$. We denote the Floquet multiplers as $\mu_1, \mu_2, \cdots, \mu_n$ with $\mu_n = 1$ and summarize the stability results in the following theorem.

Theorem 2.4 *Let* $\mathbf{p}(t;\lambda_0)$ *be a periodic solution of* (2.1) *with period* T. *Then* 1 *is a Floquet multiplier of* $\mathbf{u}(t)$. *Furthermore,*

(1) $\mathbf{p}(t;\lambda_0)$ *is stable if* $|\mu_i| < 1$ *for all* $1 \leq i \leq n-1$;

(2) $\mathbf{p}(t;\lambda_0)$ *is unstable if there exists* $1 \leq i \leq n-1$ *such that* $|\mu_i| > 1$.

The stability of the periodic solutions changes when one or several Floquet multipliers cross the unit circle. Similar to the steady state solutions, several types of bifurcation such as *transcritical bifurcation*, *pitchfork bifurcation* and *turning points* also exist for the periodic solutions. However, there are number of interesting types of bifurcation for the periodic solutions which do not exist for the steady state problems. We demonstrate these cases in the following examples. Unless stated otherwise, the computations throughout this thesis were done by the author using AUTO.

Example 2.6 *Period Doubling Bifurcation*

$$\begin{aligned}\dot{x} &= -y-z\\ \dot{y} &= x+ay\\ \dot{z} &= b+z(x-c).\end{aligned}$$

This system is known as Rössler's band [60]. The bifurcation diagram and some of the periodic solutions are shown on Figure 2.1 for $b=2, c=4$. The Hopf bifurcation point is located at $a = 0.1249676$ on the steady state solution branch (branch 1). The periodic solution branch 2 loses its stability at the periodic doubling point $a = 0.3348535$, a stable (period 2) solution branch appears (branch 3). This phenomenon repeats for all new periodic solution branches. In fact, the continuous period doubling bifurcation leads to chaos.

A period doubling bifurcation is also called a *flip bifurcation* or *subharmonic bifurcation*, and it typically happens when one of the Floquet multipliers crosses the unit circle at -1.

Example 2.7 *Bifurcation to Invariant Tori* [44]

$$\begin{aligned}\dot{x} &= (\lambda-3)x-4y+x[z+0.2(1-z^2)]\\ \dot{y} &= 4x+(\lambda-3)y+y[z+0.2(1-z^2)]\\ \dot{z} &= \lambda z-(x^2+y^2+z^2).\end{aligned}$$

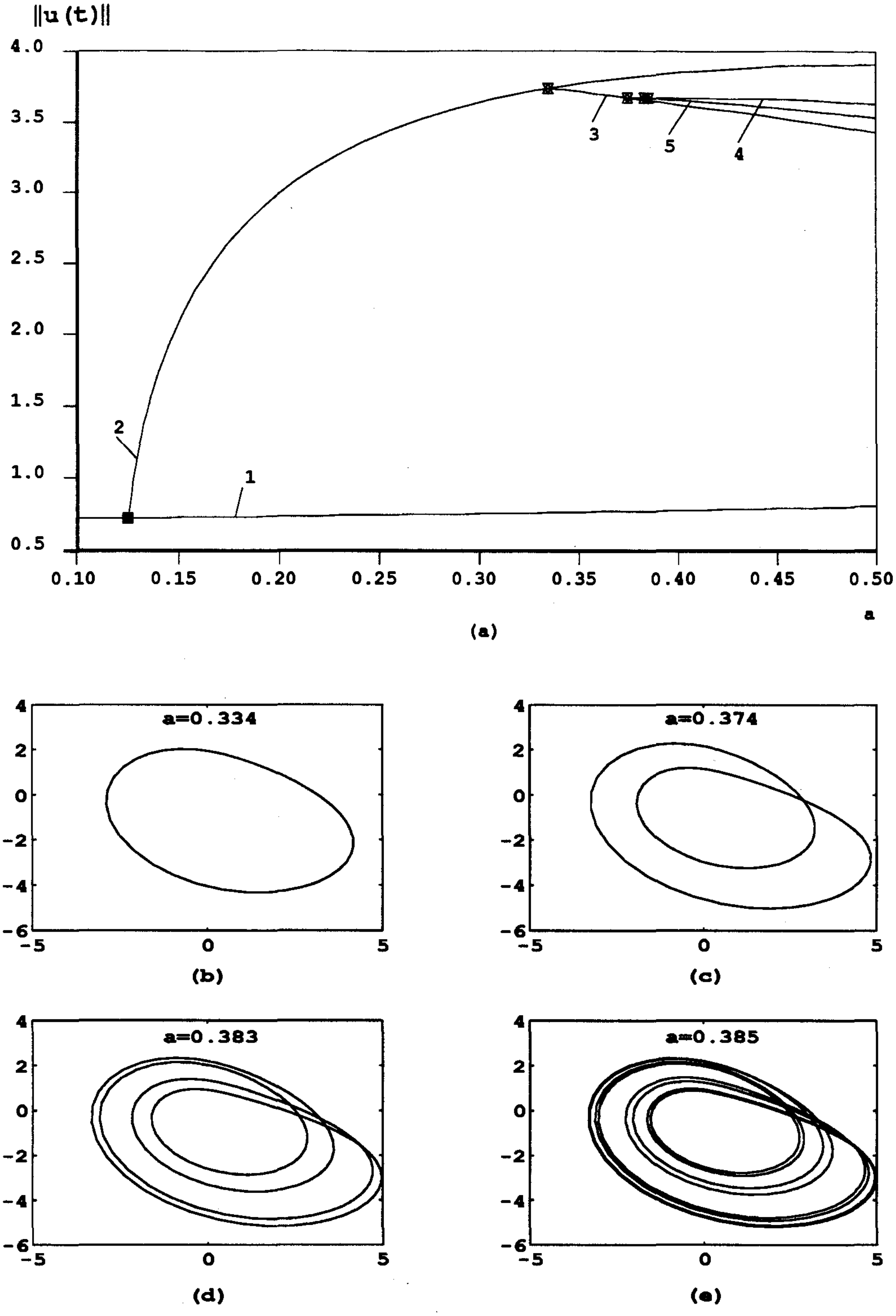


Figure 2.1: Period doubling bifurcation for Rössler's Band
(a) Period doubling bifurcation points are $a = 0.3348535, 0.3747523, 0.3833313, 0.3851549$;
(b) Periodic solutions on $x - y$ plane.

In this example, the steady state solution branch $(0, 0, \lambda)$ (branch B) loses its stability through a Hopf bifurcation at $\lambda = 1.683375$ (label 2 on Figure 2.2(a)). The periodic solution has the period $\frac{\pi}{2}$. On the periodic branch C, the torus bifurcation happens when $\lambda = 2.0$ (label 3 on Figure 2.2(a)). At this point, the stable solution has two frequencies, 4 and $\sqrt{1.2}$. Because $\frac{\sqrt{1.2}}{4}$ is an irrational number, this solution is quasi-periodic and stays on an invariant torus. Letting $x = r\cos\theta$ and $y = r\sin\theta$, we can reduce the above system to

$$\begin{aligned}\dot{r} &= (\lambda - 3)r + r[z + 0.2(1 - z^2)] \\ \dot{z} &= \lambda z - (r^2 + z^2) \\ \dot{\theta} &= 4.\end{aligned}$$

With the above transformation, the *periodic branch* in the Example 2.7 (branch C in Figure 2.2(a)) is the *steady state branch* for the (r, z) system (branch C in Figure 2.2(b)), the *tori bifurcation point* (label 3 in Figure 2.2(a)) becomes a *Hopf bifurcation point* (label 3 in Figure 2.2(b)), and the *invariant tori branch* corresponds to a *periodic solution branch* (branch D in Figure 2.2(b)).

Torus bifurcation has also been referenced as *Hopf bifurcation for periodic solutions*, and it usually happens when a pair of complex conjugate Floquet multipliers crosses the unit circle. In Example 2.7, they are $-0.14936 \pm 0.98878i$.

There is another type of bifurcation arising from periodic solutions. It is the limit of a periodic solution branch, where the period approaches infinity. Such a type of bifurcation is called a *homoclinic bifurcation* or *heteroclinic bifurcation* depending on the nature of the limiting solution. We will leave our discussion of this case to chapter 4. The period doubling bifurcation, the torus bifurcation and the homoclinic bifurcation are believed to be the routes to chaos in many applications.

2.1.4 Boundary Value Problems

For boundary value problems, we will use the word "conditioning" instead of "stability". Consider the linear BVP

$$\mathbf{u}'(t) = A(t)\mathbf{u}(t) + \mathbf{q}(t) \tag{2.11}$$

$$B_a\mathbf{u}(a) + B_b\mathbf{u}(b) = \mathbf{d} \tag{2.12}$$

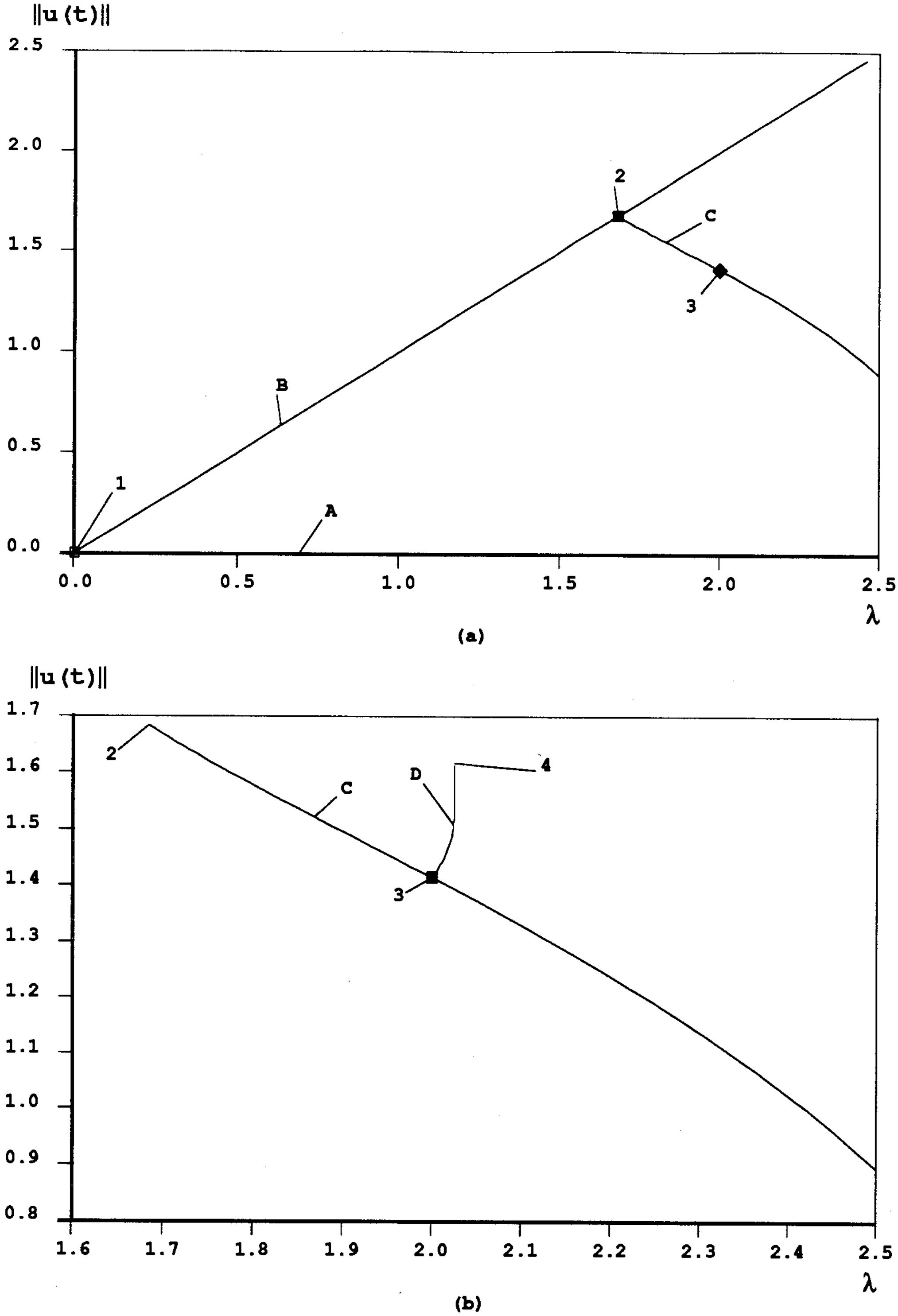


Figure 2.2: Bifurcation to tori

where $A(t) \in \mathbf{R}^{n\times n}$ is continuous in $t \in [a,b]$. In [4], the conditioning of the BVP (2.11) and (2.12) is studied by using the *dichotomy* conditions of the fundamental solutions for ODEs.

Let $Y(t),\ t \in [a,b]$ be a fundamental solution for the ODE (2.11) which satisfies

$$B_aY(a) + B_bY(b) = I. \tag{2.13}$$

Then

$$\mathbf{u}(t) = Y(t)\mathbf{d} + \int_a^b G(t,\tau)\mathbf{q}(\tau)d\tau \tag{2.14}$$

is the unique solution of (2.11) and (2.12) where

$$G(t;\tau) := \begin{cases} Y(t)B_aY(a)Y^{-1}(\tau) & \tau \le t \\ -Y(t)B_bY(b)Y^{-1}(\tau) & \tau > t \end{cases} \tag{2.15}$$

is called an $n \times n$ *Green's (matrix) function.*

The conditioning of the BVP (2.11) and (2.12) may be defined by the constant

$$\kappa = \max\{\kappa_1, \kappa_2\} \tag{2.16}$$

where

$$\kappa_1 = \|Y(t)\|, \quad \kappa_2 = \sup_{a\le t,\tau\le b} \|G(t,\tau)\|.$$

Definition 2.7 *Let* $Y(t)$ *be a fundamental solution of* (2.11). *The ODE has an exponential dichotomy if there exists an orthogonal matrix* $P \in \mathbf{R}^{n\times n}$ *of rank* p, $0 < p < n$, *and positive constants* K, λ, μ, *such that*

$$\|Y(t)PY^{-1}(\tau)\| \le Ke^{-\lambda(t-\tau)} \qquad t \ge \tau \tag{2.17}$$

$$\|Y(t)(I-P)Y^{-1}(\tau)\| \le Ke^{-\mu(\tau-t)} \qquad t < \tau \tag{2.18}$$

for $a \le t, \tau \le b$. *It is said to have an ordinary dichotomy if* $\lambda = 0$ *and/or* $\mu = 0$ *in the conditions* (2.17) *and* (2.18).

There is a close relationship between the conditioning of BVPs and dichotomy. A well-conditioned BVP must have a corresponding dichotomy; on the other hand, if the ODE has a dichotomy, then the associated BVP is well-conditioned provide κ_1 is of moderate size. In fact, κ_2 can be bounded in terms of κ_1 ([4], *Corollary* 3.104).

For the nonlinear BVP

$$\mathbf{u}'(t) = \mathbf{f}(\mathbf{u}(t), \boldsymbol{\lambda}), \qquad a \le t \le b,\ \mathbf{u} \in \mathbf{R}^n,\ \mathbf{f} \in \mathbf{R}^n,\ \boldsymbol{\lambda} \in \mathbf{R}^p \tag{2.19}$$

$$\mathbf{g}(\mathbf{u}(a), \mathbf{u}(b), \boldsymbol{\lambda}) = \mathbf{0}, \tag{2.20}$$

the conditioning results can be obtained by studying the *variational problem*

$$\mathbf{v}'(t) = A(t)\mathbf{v}(t), \qquad a < t < b \tag{2.21}$$

$$B_a\mathbf{v}(a) + B_b\mathbf{v}(b) = \mathbf{0} \tag{2.22}$$

where

$$A(t) = \mathbf{f}_{\mathbf{u}}(\mathbf{u}(t), \boldsymbol{\lambda}),\ B_a = \mathbf{g}_{\mathbf{u}(a)}(\mathbf{u}(a), \mathbf{u}(b), \boldsymbol{\lambda}),\ B_b = \mathbf{g}_{\mathbf{u}(b)}(\mathbf{u}(a), \mathbf{u}(b), \boldsymbol{\lambda}).$$

2.2 Numerical Continuation

In this section, we discuss the numerical methods for the continuation of the solutions of dynamical systems, the stability of the solution branch, the determination of the bifurcation points, and the branch switching techniques.

2.2.1 Steady State Solutions

Consider the steady state problem

$$\mathbf{f}(\mathbf{u}, \lambda) = \mathbf{0}, \qquad \mathbf{f} \in \mathbf{R}^n,\ \mathbf{u} \in \mathbf{R}^n,\ \lambda \in \mathbf{R} \tag{2.23}$$

and assume that $(\mathbf{u}_0, \lambda_0)$ is a known solution. Then, the continuation of a solution branch $(\mathbf{u}, \lambda)$ involves computing a series of solutions

$$(\mathbf{u}_1, \lambda_1),\ (\mathbf{u}_2, \lambda_2), \cdots,$$

from this known solution until a certain limit is reached. Many numerical methods for the continuation of solution branches have been developed. In [66], Seydel gives a complete review of these methods and divides them into four different categories: predictor, parameterization, corrector and step length control. In particular, the pseudo-arclength parameterization and continuation method is studied in [41] and implemented into the software AUTO.

(1) *Pseudo-arclength continuation*

Consider the $n+1$ dimensional system

$$\mathbf{F}(\mathbf{y}, s) = \begin{pmatrix} \mathbf{f}(\mathbf{u}, \lambda) \\ p(\mathbf{u}, \lambda, s) \end{pmatrix} = \mathbf{0} \tag{2.24}$$

where $\mathbf{y}(s) = \begin{bmatrix} \mathbf{u}(s) \\ \lambda(s) \end{bmatrix} \in \mathbf{R}^{n+1}$. If s represents the arclength along the solution branch, we choose the continuation equation

$$p(\mathbf{u}, \lambda, s) := \theta \|\dot{\mathbf{u}}(s)\|^2 + (1-\theta)|\dot{\lambda}(s)|^2 - 1 = 0 \tag{2.25}$$

where $0 < \theta < 1$. However, (2.25) is not practical in general as it involves evaluating the first derivatives with respect to the arclength of the solution branch. One solution is to use the linear approximation for $\dot{\mathbf{u}}(s)$ and $\dot{\lambda}(s)$ and thus, we obtain

$$p(\mathbf{u}, \lambda, s) := \theta \|\mathbf{u}(s) - \mathbf{u}(s_k)\|^2 + (1-\theta)|\lambda(s) - \lambda(s_k)|^2 - (s - s_k)^2 = 0 \tag{2.26}$$

where $(\mathbf{u}_k, \lambda_k) = (\mathbf{u}(s_k), \lambda(s_k))$ is the previous solution during the continuation. Another alternative is to use

$$p(\mathbf{u}, \lambda, s) := \theta [\mathbf{u}(s) - \mathbf{u}(s_k)]^T \dot{\mathbf{u}}(s_k) + (1-\theta)[\lambda(s) - \lambda(s_k)]\dot{\lambda}(s_k) - (s - s_k) = 0. \tag{2.27}$$

This approximation is obtained by the Taylor expansions

$$\mathbf{u}(s) - \mathbf{u}(s_k) = (s - s_k)\dot{\mathbf{u}}(s_k) + O(|s - s_k|^2),$$

$$\lambda(s) - \lambda(s_k) = (s - s_k)\dot{\lambda}(s_k) + O(|s - s_k|^2).$$

Equations (2.26) and (2.27) are called *pseudo-arclength continuation equations.* The advantage of this continuation method is that the Jacobian at a turning point is nonsingular.

Theorem 2.5 *Let* $(\mathbf{u}_k, \lambda_k)$ *be a solution of* (2.23) *with* $\operatorname{rank}(\mathbf{f}_{\mathbf{u}}(\mathbf{u}_k, \lambda_k) | \mathbf{f}_\lambda(\mathbf{u}_k, \lambda_k)) = n$ *and* $\mathbf{f} \in C^2$ *in neighborhood of* $(\mathbf{u}_k, \lambda_k)$. *Then there exists a unique solution* $(\mathbf{u}(s), \lambda(s))$ *for* (2.23) *and* (2.27) *on* $|s - s_k| < \rho$ *for sufficiently small* $\rho > 0$, *and the matrix*

$$\mathcal{A}(s) := \begin{bmatrix} \mathbf{f}_{\mathbf{u}}(\mathbf{u}(s), \lambda(s)) & \mathbf{f}_\lambda(\mathbf{u}(s), \lambda(s)) \\ \dot{\mathbf{u}}(s) & \dot{\lambda}(s) \end{bmatrix} \tag{2.28}$$

is nonsingular.

The continuation technique used in AUTO is based on (2.27). Note (2.27) is linear, but not (2.26). The algorithm implemented in the software solves the system

$$\mathbf{f}(\mathbf{u}_k, \lambda_k) = \mathbf{0} \tag{2.29}$$

$$\theta_{\mathbf{u}}^2(\mathbf{u}_k - \mathbf{u}_{k-1})^T\dot{\mathbf{u}}_{k-1} + \theta_\lambda^2(\lambda_k - \lambda_{k-1})\dot{\lambda}_{k-1} = \delta s \tag{2.30}$$

for $(\mathbf{u}_k, \lambda_k)$ by Newton's method where δs is the stepsize along the solution branch. $\theta_{\mathbf{u}}$ and θ_λ are the scaling parameters with

$$\theta_{\mathbf{u}}^2 + \theta_\lambda^2 = 1.$$

The direction vector $(\dot{\mathbf{u}}_{k-1}, \dot{\lambda}_{k-1})$ is approximated by extrapolating the two previous solutions

$$\dot{\mathbf{u}}_{k-1} \approx \frac{1}{\delta s}(\mathbf{u}_{k-1} - \mathbf{u}_{k-2}), \quad \dot{\lambda}_{k-1} \approx \frac{1}{\delta s}(\lambda_{k-1} - \lambda_{k-2}),$$

and then rescaling them so that

$$\theta_{\mathbf{u}}^2\|\dot{\mathbf{u}}_{k-1}\|^2 + \theta_\lambda^2\dot{\lambda}_{k-1}^2 = 1$$

to avoid instability from the continuation process.

(2) *Stability, bifurcation and branch switching*

The stability of the stationary solution $\mathbf{u}_k$ can be determined by calculating the eigenvalues of $\mathbf{f}_{\mathbf{u}}(\mathbf{u}_k, \lambda_k)$. If no eigenvalue has positive real part, then $\mathbf{u}_k$ is stable; otherwise, it is unstable. If there exists an eigenvalue with zero real part, it is a potential bifurcation point.

First, we consider the case where $\mathbf{f}_{\mathbf{u}}(\mathbf{u}_k, \lambda_k)$ is singular. If $[\mathbf{f}_{\mathbf{u}}(\mathbf{u}_k, \lambda_k)|\mathbf{f}_\lambda(\mathbf{u}_k, \lambda_k)]$ is of full rank (rank n), then $(\mathbf{u}_k, \lambda_k)$ is a turning point; otherwise, it is a bifurcation point. We will only discuss the *simple* bifurcation point and the *quadratic* turning point, where the matrix $\mathbf{f}_{\mathbf{u}}(\mathbf{u}_k, \lambda_k)$ is of rank $n-1$.

Let the function $d(s) = \det(\mathcal{A}(s))$. Since there is only one eigenvalue of $\mathbf{f}_{\mathbf{u}}(\mathbf{u}_k, \lambda_k)$ changing sign when the solution crosses the simple bifurcation points on the solution branch, $d(s)$ will also changes sign. In AUTO, such points are located accurately by using a secant iteration so that $d(\hat{s}) = 0$. After locating the bifurcation point $(\mathbf{u}(\hat{s}), \lambda(\hat{s}))$, we may perform *branch switching* from this point to trace out another steady state solution branch. In

AUTO, the method of doing branch switching is to obtain the first solution $(\hat{\mathbf{u}}_1, \hat{\lambda}_1)$ on the new steady state solution branch by solving the system

$$\mathbf{f}(\hat{\mathbf{u}}_1, \hat{\lambda}_1) = \mathbf{0}$$

$$\theta^2_{\mathbf{u}}(\hat{\mathbf{u}}_1 - \mathbf{u}_k)^T \mathbf{u}'_k - \theta^2_{\lambda}(\hat{\lambda}_1 - \lambda_k)\lambda'_k - \delta s = 0$$

where $(\mathbf{u}'_k, \lambda'_k)$ satisfies

$$\begin{bmatrix} \mathbf{f}_{\mathbf{u}}(\mathbf{u}_k, \lambda_k) & \mathbf{f}_{\lambda}(\mathbf{u}_k, \lambda_k) \\ \dot{\mathbf{u}}_k^T & \dot{\lambda}_k \end{bmatrix} \begin{bmatrix} \mathbf{u}'_k \\ \lambda'_k \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 0 \end{bmatrix}.$$

The quadratic turning points can also be located by using a secant iteration such that $\dot{\lambda}(s) = 0$ where $(\dot{\mathbf{u}}, \dot{\lambda})$ satisfies

$$\begin{bmatrix} \mathbf{f}_{\mathbf{u}}(\mathbf{u}, \lambda) & \mathbf{f}_{\lambda}(\mathbf{u}, \lambda) \\ \dot{\mathbf{u}}_k & \dot{\lambda}_k \end{bmatrix} \begin{bmatrix} \dot{\mathbf{u}} \\ \dot{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}.$$

The turning point branch can be continued by a pseudo-arclength continuation method in two parameters

$$\begin{bmatrix} \mathbf{f}(\mathbf{u}, \lambda, \mu) \\ \mathbf{f}_{\mathbf{u}}(\mathbf{u}, \lambda, \mu)\mathbf{v} \\ \|\mathbf{v}\|^2 - 1 \end{bmatrix} = \mathbf{0}. \tag{2.31}$$

For the Hopf bifurcation point, $\mathbf{f}_{\mathbf{u}}(\mathbf{u}_k, \lambda_k)$ is nonsingular. However, some eigenvalues have zero real parts. Again we consider the simplest case, i.e., only one pair of eigenvalues is purely imaginary. To compute the Hopf bifurcation points, we compute all eigenvalues of $\mathbf{f}_{\mathbf{u}}(\mathbf{u}_k, \lambda_k)$

$$\mu_i(s) = \alpha_i(s) + i\beta_i(s), \qquad i = 1, \cdots, n$$

along the solution branch. If there is a pair of eigenvalues with real parts, say $\alpha_j(s)$, changing sign, we may also use the secant method to calculate $\hat{s}$ such that $\alpha(\hat{s}) = 0$ and thus obtain the Hopf bifurcation point. To trace out the Hopf bifurcation branch with two-parameter continuation, a pseudo-arclength continuation method can be applied to solve the $3n + 2$ system

$$\begin{bmatrix} \mathbf{f}(\mathbf{u}, \lambda, \mu) \\ \frac{T}{2\pi}\mathbf{f}_{\mathbf{u}}(\mathbf{u}, \lambda, \mu)\boldsymbol{\xi} + \boldsymbol{\eta} \\ \frac{T}{2\pi}\mathbf{f}_{\mathbf{u}}(\mathbf{u}, \lambda, \mu)\boldsymbol{\eta} - \boldsymbol{\xi} \\ \|\boldsymbol{\xi}\|^2 + \|\boldsymbol{\eta}\|^2 - 1 \\ \boldsymbol{\eta}_0^T\boldsymbol{\xi} - \boldsymbol{\xi}_0^T\boldsymbol{\eta} \end{bmatrix} = \mathbf{0} \tag{2.32}$$

where $\mathbf{v}_0(t) = \sin(\frac{2\pi t}{T_0})\xi_0 + \cos(\frac{2\pi t}{T_0})\eta_0$ is the periodic solution at a previous solution on the Hopf bifurcation branch.

2.2.2 Boundary Value Problems

We now review the numerical methods for the continuation of solutions for boundary value problems. We defer a description of the numerical solutions for BVPs to the next chapter, but emphasize the numerical continuation method here. In general, all methods mentioned in section 2.2.1 can be extended to do the continuation for BVPs. In particular, a tangent vector continuation with step length control method is used in COLCON, while a pseudo-arclength continuation method is used in AUTO. We only discuss the latter.

(1) *Pseudo-arclength continuation*

Consider the first order ODE system

$$\mathbf{u}'(t) = \mathbf{f}(\mathbf{u}(t), \boldsymbol{\lambda}), \qquad t \in [0,1],\ \mathbf{f} \in \mathbf{R}^n,\ \mathbf{u} \in \mathbf{R}^n,\ \boldsymbol{\lambda} \in \mathbf{R}^p \tag{2.33}$$

subject to the non-separated boundary conditions

$$\mathbf{b}(\mathbf{u}(0), \mathbf{u}(1), \boldsymbol{\lambda}) = \mathbf{0}, \qquad \mathbf{b} \in \mathbf{R}^{n_b} \tag{2.34}$$

and the integral conditions

$$\int_0^1 \mathbf{q}(\mathbf{u}(t), \boldsymbol{\lambda})dt = \mathbf{0}, \qquad \mathbf{q} \in \mathbf{R}^{n_q}. \tag{2.35}$$

Assuming that there is only one free parameter in the BVP system (2.33), (2.34) and (2.35), then $p = n_b + n_q - n + 1$. To perform the continuation of the solution branch for this BVP system, AUTO uses a *pseudo-arclength continuation equation*

$$\theta_{\mathbf{u}}^2 \int_0^1 (\mathbf{u}(t) - \mathbf{u}_0(t))^T \dot{\mathbf{u}}_0(t)dt + \theta_{\boldsymbol{\lambda}}^2 (\boldsymbol{\lambda} - \boldsymbol{\lambda}_0)^T \dot{\boldsymbol{\lambda}}_0 = \delta s. \tag{2.36}$$

In (2.36), like the steady state case, the direction vector, $(\dot{\mathbf{u}}_0(t), \dot{\boldsymbol{\lambda}}_0)$, of the solution branch is approximated by extrapolating the previous two solutions.

Also similar to the steady state case, the turning point on the BVP solution branch can be accurately computed, and the turning point solution branch can be traced by solving a two-parameter continuation problem. Suppose that λ is the principal continuation parameter and $\boldsymbol{\lambda} = (\lambda, \boldsymbol{\mu})$. We use the secant method to obtain the zero of $\dot{\lambda}(s)$ on the ODE solution branch. Such a solution is the turning point. Adding the following equations to the system (2.33)–(2.35), we can carry out the two-parameter continuation for the turning point solutions:

$$\mathbf{v}'(t) = \mathbf{f_u}(\mathbf{v}(t), \boldsymbol{\mu}, \lambda)\mathbf{v}(t) + \mathbf{f}_{\boldsymbol{\mu}}(\mathbf{v}(t), \boldsymbol{\mu}, \lambda)\mathbf{w}, \tag{2.37}$$

$$\mathbf{b}_{\mathbf{u}(0)}(\mathbf{u}(0), \mathbf{u}(1))\mathbf{v}(0) + \mathbf{b}_{\mathbf{u}(1)}(\mathbf{u}(0), \mathbf{u}(1))\mathbf{v}(1) + \mathbf{b}_{\boldsymbol{\mu}}(\mathbf{u}(0), \mathbf{u}(1))\mathbf{w} = 0, \tag{2.38}$$

$$\int_0^1 [\mathbf{q_u}(\mathbf{u}(t), \boldsymbol{\mu}, \lambda)\mathbf{v}(t) + \mathbf{q}_{\boldsymbol{\mu}}(\mathbf{u}(t), \boldsymbol{\mu}, \lambda)\mathbf{w}]dt = 0, \tag{2.39}$$

$$\int_0^1 [\mathbf{v}^T(t)\mathbf{v}(t) + \mathbf{w}^T\mathbf{w} - 1]dt = 0 \tag{2.40}$$

where $\boldsymbol{\mu}, \mathbf{w} \in \mathbf{R}^{p-1}$.

(2) *Periodic solutions*

The problem for finding periodic orbits of (2.1) can be regarded as a boundary value problem with the boundary condition

$$\mathbf{u}(0) = \mathbf{u}(T) \tag{2.41}$$

where T is the minimum period. Equation (2.41) is called a *periodic boundary condition.* However, this condition does not give the unique periodic solution. In fact, if $\mathbf{u}(t)$ is a solution of (2.1) and (2.41), $\mathbf{u}(t+\tau)$ is also a solution for any constant τ. To form a well-posed problem, a certain *phase condition* is needed. We discuss the following two types of phase conditions which are often used in the numerical computations. As before, we refer to $\mathbf{u}_0(t)$ as the previous solution on the same periodic solution branch.

Classical phase condition The classical phase condition is defined by

$$[\mathbf{u}(0) - \mathbf{u}_0(0)]^T\mathbf{u}_0'(0) = 0. \tag{2.42}$$

This phase condition fixes $\mathbf{u}(0)$ in a hyperplane through $\mathbf{u}_0(0)$ and orthogonal to $\mathbf{u}_0'(0)$.

Integral phase condition Suppose $\tilde{\mathbf{u}}(t+\sigma)$ is a solution of (2.1) and (2.41). We choose σ^* such that the distance between $\mathbf{u}_0$ and $\tilde{\mathbf{u}}$,

$$G(\sigma) = \int_0^T ||\tilde{\mathbf{u}}(t+\sigma) - \mathbf{u}_0(t)||^2 dt, \tag{2.43}$$

is a minimum, or equivalently, $G'(\sigma^*) = 0$. Letting $\tilde{\mathbf{u}}(t + \sigma^*) = \mathbf{u}(t)$, then

$$\int_0^T [\mathbf{u}(t) - \mathbf{u}_0(t)]^T \mathbf{u}'(t)dt = 0. \tag{2.44}$$

After integration by parts for (2.44), we obtain the integral phase condition

$$\int_0^T \mathbf{u}^T(t)\mathbf{u}_0'(t)dt = 0. \tag{2.45}$$

Generally speaking, phase condition (2.45) is more reliable than (2.42) in numerical computation, but the classical phase condition is easy to program since it does not need to evaluate the integral constraints. AUTO uses the phase condition (2.45) due to its ability of handling the integral constraints directly.

During the continuation, the period T of the current solution is unknown. We will rescale the independent variable t to $[0, 1]$ and treat T as a free parameter and rewrite the boundary value problem as follows:

$$\mathbf{u}'(t) = T\mathbf{f}(\mathbf{u}, \lambda) \tag{2.46}$$

$$\mathbf{u}(0) = \mathbf{u}(1) \tag{2.47}$$

$$\int_0^1 \mathbf{u}^T(t)\mathbf{u}_0'(t)dt = 0 \tag{2.48}$$

$$\theta_{\mathbf{u}}^2 \int_0^1 [\mathbf{u}(t) - \mathbf{u}_0(t)]^T \dot{\mathbf{u}}_0(t)dt + \theta_T^2(T - T_0)\dot{T}_0 + \theta_\lambda^2(\lambda - \lambda_0)\dot{\lambda}_0 = \delta s. \tag{2.49}$$

Equations (2.46)–(2.49) consist of n ODEs, n boundary conditions and 2 integral constraints. There are two free parameters λ and T. This system is solved by a boundary value solver (a collocation method in AUTO). By computing the Floquet multipliers for the periodic solution, we can determine the stability. Branch switching methods for the steady state solution can be generalized for the (transcritical, pitch-fork and period doubling) bifurcations as well as two-parameter continuation for the turning points, period doubling points and tori bifurcation points. However, the current implementation for two-parameter continuation of the secondary periodic solutions (turning points, period doubling points and tori bifurcation points) is not efficient. The improvement is considered in [30].

Chapter 3

Linear System Solutions

In the last chapter, we discussed numerical methods for the continuation of the solutions for the boundary value ODEs. The essential part of each continuation step is to solve a nonlinear ordinary differential equation system with certain boundary value conditions and integral conditions. Several numerical methods are available for solving BVPs, among them, the finite difference method, the multiple shooting method and the spline collocation method are the most popular ones. All these methods involve solving linear systems with a similar sparse structure. In general, these linear systems are solved by a Gaussian elimination method with some type of pivoting. The cost of solving the linear systems is usually the most expensive part in the BVP solvers. Therefore, an efficient numerical method should be used if possible. However, the numerical stability of the linear system solution will affect the stability of the solution for boundary value problems. The trade off between the efficiency and the stability for the linear system solution should be balanced. For nonlinear boundary value problems, Newton's methods are usually used. Since each Newton iteration involves solving a linear system, we usually use the Newton-Chord methods to gain some efficiency, i.e., fix the Jacobian for several Newton iterations. In order to obtain the convergence, a stable numerical linear system solver is needed. For BVP continuation codes, one computes a series of solutions on one or several branches rather than one particular solution of a BVP. In this context, an efficient and stable linear system solver becomes even more important. In this chapter, we will discuss some efficiency and stability issues when solving block bidiagonal linear systems numerically. These types of systems are often the results of the discretization of the boundary value problems by the finite difference methods, the multiple shooting methods or the spline collocation methods. Our goal is to develop a stable and efficient numerical linear system solver, in particular to replace the current linear system solver for AUTO, where robustness is sacrificed to obtain efficiency.

3.1 Linear Two-Point Boundary Value Problems

In this section, we consider numerical methods for solving the linear two-point boundary value problem

$$\mathbf{y}'(t) = A(t)\mathbf{y}(t) + \mathbf{q}(t) \tag{3.1}$$

$$B_a\mathbf{y}(a) + B_b\mathbf{y}(b) = \mathbf{d} \tag{3.2}$$

where $t \in [a, b]$, $\mathbf{y}(t)$, $\mathbf{q}(t)$, $\mathbf{d} \in \mathbf{R}^n$ and $A(t)$, B_a, $B_b \in \mathbf{R}^{n \times n}$.

There are two classes of numerical methods for solving BVPs, initial value methods such as the *shooting* and *multiple shooting* methods, and the global methods such as the *finite difference* and *collocation* methods. In most of these methods, the discretization of the BVP produces a large sparse linear system of equations. The efficiency of the BVP solver mainly depends on the efficiency of the linear system solver; therefore, a stable and efficient linear system solver is essential for solving BVPs.

Our discussion of the numerical solutions for BVPs includes the *multiple shooting* methods, the *finite difference* method, and the *collocation* method. All of them lead to a similar linear system of equations, namely, (almost) block bi-diagonal linear systems.

The spectral and pseudo-spectral methods for solving boundary value problems are also studied by many other authors (e.g. [16, 17]) and these methods are very efficient in some applications. However, the linear systems generated by the spectral and pseudo-spectral methods have a different structure and may not have sparsity structure. We will not study these methods in this chapter. We will apply rational spectral methods to solve a certain type of boundary value problems (connecting orbits) in the next chapter.

3.1.1 Numerical Discretizations

Throughout our discussion, we assume that the BVP (3.1) and (3.2) is well-conditioned, i.e., the condition number κ defined in (2.16) is of moderate size. We now review the numerical algorithms for the multiple shooting method, the finite difference method and the spline collocation method, including their numerical stability and convergence properties.

(1) *Multiple shooting method*

Given a mesh $a = t_1 < t_2 < \cdots < t_{N+1} = b$, we compute a fundamental solution $Y_i(t) \in \mathbf{R}^{n\times n}$ and a particular solution $\mathbf{v}_i(t) \in \mathbf{R}^n$ on each mesh interval $[t_i, t_{i+1}]$, $1 \le i \le N$, such that

$$Y_i' = A(t)Y_i, \qquad Y_i(t_i) = F_i \tag{3.3}$$

$$\mathbf{v}_i' = A(t)\mathbf{v}_i + \mathbf{q}(t), \qquad \mathbf{v}_i(t_i) = \mathbf{e}_i. \tag{3.4}$$

We then find a vector $\mathbf{s}_i \in \mathbf{R}^n$, $1 \le i \le N$ such that

$$\mathbf{y}(t) = Y_i(t)\mathbf{s}_i + \mathbf{v}_i(t), \qquad t \in [t_i, t_{i+1}],\ i = 1, \cdots, N. \tag{3.5}$$

For the standard multiple shooting method, we choose $F_i = I$ and $\mathbf{e}_i = \mathbf{0}$. By satisfying the continuity conditions at the mesh-points and the boundary conditions to the end-points, we obtain

$$\begin{bmatrix} Y_1(t_2) & -I & & & \\ & Y_2(t_3) & -I & & \\ & & \ddots & \ddots & \\ & & & Y_N(t_{N+1}) & -I \\ B_a & & & & B_b \end{bmatrix} \begin{bmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \\ \vdots \\ \mathbf{s}_N \\ \mathbf{s}_{N+1} \end{bmatrix} = \begin{bmatrix} -\mathbf{v}_1(t_2) \\ -\mathbf{v}_2(t_3) \\ \vdots \\ -\mathbf{v}_N(t_{N+1}) \\ \mathbf{d} \end{bmatrix}. \tag{3.6}$$

Solving the above linear system, we obtain the numerical solution $\mathbf{y}(t_i) = \mathbf{s}_i$ for the BVP (3.1) and (3.2). Note on each mesh interval, we need to integrate the system of differential equations by an initial value solver.

If the shooting points (mesh points) are chosen so that all $||Y_i(t_{i+1})||$, $i = 1, \cdots, N$, are of moderate size (which implies that N is large enough), then the standard multiple shooting method is stable and the condition number of the coefficient matrix is $O(\kappa)$ [4, 46].

(2) *Finite difference method*

Similar to the multiple shooting case, we define a mesh $a = t_1 < t_2 < \cdots < t_{N+1} = b$ and let $h_i = t_{i+1} - t_i$, and $t_{i+\frac{1}{2}} = t_i + \frac{1}{2}h_i$. In the finite difference methods, we replace the derivates in the differential equations by the difference quotients and solve the resulting difference equations. To be more specific, we consider one step finite difference schemes, where the difference operator is based only on values over one subinterval $[t_i, t_{i+1}]$. We obtain

$$\begin{bmatrix} S_1 & R_1 & & & \\ & S_2 & R_2 & & \\ & & \ddots & \ddots & \\ & & & S_N & R_N \\ B_a & & & & B_b \end{bmatrix} \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \\ \mathbf{y}_N \\ \mathbf{y}_{N+1} \end{bmatrix} = \begin{bmatrix} \mathbf{q}_1 \\ \mathbf{q}_2 \\ \vdots \\ \mathbf{q}_N \\ \mathbf{d} \end{bmatrix} \tag{3.7}$$

where S_i, R_i are $n \times n$ matrices for all $1 \leq i \leq N$. For the trapezoidal scheme

$$S_i = -h_i^{-1} I - \frac{1}{2} A(t_i), \quad R_i = -h_i^{-1} I - \frac{1}{2} A(t_{i+1}), \quad \mathbf{q}_i = \frac{1}{2}[\mathbf{q}(t_{i+1}) + \mathbf{q}(t_i)] \tag{3.8}$$

and for the midpoint scheme

$$S_i = R_i = -h_i^{-1} I - \frac{1}{2} A(t_{i+\frac{1}{2}}) \qquad \mathbf{q}_i = \mathbf{q}(t_{i+\frac{1}{2}}). \tag{3.9}$$

Both schemes are numerically stable. The conditioning of the linear system is $O(\kappa N)$. There is a close relationship between the finite difference method and the multiple shooting method:

$$\Gamma_i := -R_i^{-1} S_i = Y_i(t_{i+1}) + O(h_i^2), \qquad i = 1, \cdots, N.$$

The local truncation error for the finite difference method is $O(h^2)$ where $h = \max_{1 \leq i \leq N} h_i$. In fact, the second order accuracy is the disadvantage of the finite difference method as it may not be efficient for many applications.

(3) *Collocation method*

Since our main purpose is to examine the linear system solver for the code AUTO, we only discuss the collocation method as it is implemented therein with Lagrange basis functions. We try to find

$$\mathbf{p}_j(t) = \sum_{i=0}^{m} w_{ji}(t) \mathbf{y}_{j+\frac{i}{m}} \tag{3.10}$$

where

$$w_{ji}(t) = \prod_{k=0, k \neq i}^{m} \frac{t - t_{j+\frac{k}{m}}}{t_{j+\frac{i}{m}} - t_{j+\frac{k}{m}}}, \quad t_{j+\frac{i}{m}} = t_j + \frac{i}{m}(t_{j+1} - t_j) \tag{3.11}$$

such that

$$\mathbf{p}_j'(z_{ji}) = A(z_{ji})\mathbf{p}_j(z_{ji}) + \mathbf{q}(z_{ji}) \tag{3.12}$$

$$B_a\mathbf{y}_1 + B_b\mathbf{y}_{N+1} = \mathbf{d} \tag{3.13}$$

where z_{ji}, $1 \leq i \leq m$, are the *Gauss-Legendre points* on the interval $[t_j, t_{j+1}]$ for $1 \leq j \leq N$. The above formulation results in $(mN+1)n$ linear equations. By using a local *parameter condensation* method, we can eliminate the unknowns at non-mesh points $\mathbf{y}_{j+\frac{i}{m}}$ for $1 < i < m$. The linear system is then reduced to

$$\begin{bmatrix} A_1 & C_1 & & & \\ & A_2 & C_2 & & \\ & & \ddots & \ddots & \\ & & & A_N & C_N \\ B_a & & & & B_b \end{bmatrix} \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \\ \mathbf{y}_N \\ \mathbf{y}_{N+1} \end{bmatrix} = \begin{bmatrix} \mathbf{q}_1 \\ \mathbf{q}_2 \\ \vdots \\ \mathbf{q}_N \\ \mathbf{d} \end{bmatrix}. \tag{3.14}$$

The collocation method is also numerically stable. Assuming that the solution is smooth enough, the above collocation method converges to the solution of the BVP at the rate of $O(h^m)$ globally. On the mesh points, the method has a *super-convergence* property where the rate of convergence is $O(h^{2m})$. In particular, if $m = 1$, this collocation method is the finite difference method using the mid-point scheme.

For the first-order system, the condition number of the linear system (3.14) is $O(\kappa N)$. For the higher order BVPs, depending on the actual implementation of the collocation algorithm, the condition number of the linear system number may be considerably larger than $O(\kappa N)$ [4]. Since we only deal with the first-order system in this thesis, this situation will not be discussed further.

3.1.2 Solution of Linear Systems

The three linear systems (3.6), (3.7) and (3.14) have a similar (almost block bi-diagonal) structure. Without loss of generality, we consider the equation (3.14). In general, it can be reliably solved with a standard linear system solver such as Gaussian elimination with partial pivoting or the QR factorization. However, in order to preserve the sparsity of the system, special linear system solvers are usually used. We discuss several of these below.

(1) *Compactification*

The compactification algorithm is a well-known but potentially unstable method for solving block bi-diagonal linear systems formed when solving two-point boundary value problems. To study this method, we consider the linear system (3.6) generated by the multiple shooting method. We compute a discrete fundamental solution $\{\Phi_i\}_{i=1}^{N+1}$ and particular solution $\{\mathbf{p}_i\}_{i=1}^{N+1}$ by

$$\mathbf{p}_{i+1} = \Phi_i \mathbf{p}_i + \mathbf{v}_i(t_{i+1}), \qquad \Phi_{i+1} = Y_i(t_{i+1})\Phi_i, \qquad i = 1, \cdots, N$$

with the initial condition

$$\mathbf{p}_1 = \mathbf{0}, \qquad \Phi_1 = I$$

and form

$$\mathbf{s}_i = \Phi_i \mathbf{s}_1 + \mathbf{p}_i, \qquad i = 2, \cdots N$$

where $\mathbf{s}_1$ satisfies

$$[B_a + B_b \Phi_{N+1}]\mathbf{s}_1 = \mathbf{d} - B_b \mathbf{p}_{N+1}.$$

The compactification algorithm, like the *single shooting* method, suffers from instability because the fastest growing (fundamental solution) mode dominates the others. A similar compactification technique can be applied for other methods by, for example, using

$$\mathbf{y}_{i+1} = \tilde{\mathbf{q}}_i - \tilde{A}_i \mathbf{y}_i, \quad \tilde{\mathbf{q}}_i = C^{-1}\mathbf{q}_i, \quad \tilde{A}_i = C^{-1}A_i \qquad i = 1, \cdots N$$

to eliminate all the interior mesh-point unknowns $\mathbf{y}_2, \cdots, \mathbf{y}_N$ and reduce the linear system to a $2n \times 2n$ dense system in the unknowns $\mathbf{y}_1$ and $\mathbf{y}_{N+1}$.

(2) *LU decomposition*

Gaussian elimination with various pivoting strategies has been used for solving the linear system (3.14). It is well-known that Gaussian elimination method with complete row pivoting is stable in practice, where the worst case error growth is $O(2^{(N+1)n})$. For the separated boundary condition case, the linear system is block bi-diagonal and Gaussian elimination with row pivoting can be used to efficiently solve the system

$$\begin{bmatrix} B_a & & & & \\ A_1 & C_1 & & & \\ & \ddots & \ddots & & \\ & & A_N & C_N & \\ & & & B_b \end{bmatrix} \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \\ \mathbf{y}_N \\ \mathbf{y}_{N+1} \end{bmatrix} = \begin{bmatrix} \mathbf{d}_a \\ \mathbf{q}_1 \\ \vdots \\ \mathbf{q}_N \\ \mathbf{d}_b \end{bmatrix}. \tag{3.15}$$

This method has been widely used (e.g. in the software SOLVEBLOK [13] used in COLSYS [3] and COLNEW [6]), and it is generally very stable. In fact the worst case error growth is exponential only in the bandwidth [12]. However, the non-separated boundary condition case is more delicate and more important to us as our interest is to modify AUTO's linear system solver. Although it is easy to convert non-separated boundary conditions to separated boundary conditions, this method usually doubles the size of the problem and is not efficient. Therefore, we will try to deal with the non-separated boundary conditions directly. A well-conditioned BVP has ordinary or exponential dichotomy. For the latter, the pivoting strategy should properly "decouple" the exponentially increasing and exponentially decreasing modes of the fundamental solution. With this idea in mind, Mattheij [55, 56] develops a stable block LU decomposition based on Gaussian elimination with row partial pivoting. Unfortunately, this method is not always practical because it requires knowing the exact number of increasing modes, and this may not be realistic to assume. Recently, Wright [76] considers applying Gaussian elimination to the columns of the reordered system

$$\begin{bmatrix} C_1 & & & & A_1 \\ A_2 & C_2 & & & \\ & \ddots & \ddots & & \\ & & A_N & C_N & \\ & & & B_b & B_a \end{bmatrix} \begin{bmatrix} \mathbf{y}_2 \\ \mathbf{y}_3 \\ \vdots \\ \mathbf{y}_{N+1} \\ \mathbf{y}_1 \end{bmatrix} = \begin{bmatrix} \mathbf{q}_1 \\ \mathbf{q}_2 \\ \vdots \\ \mathbf{q}_N \\ \mathbf{d} \end{bmatrix}. \tag{3.16}$$

The serial version of this method starts to use Gaussian elimination with row pivoting for the first $2n$ rows of (3.16) such that

$$\tilde{L}_{1,n}P_{1,n}\cdots\tilde{L}_{1,2}P_{1,2}\tilde{L}_{1,1}P_{1,1}\begin{bmatrix} C_1 \\ A_2 \end{bmatrix} = \begin{bmatrix} U_1 \\ 0 \end{bmatrix}$$

where $P_{1,i} \in \mathbf{R}^{2n\times 2n}$, $i = 1,\cdots,n$, are permutation matrices; $L_{1,i} \in \mathbf{R}^{2n\times 2n}$, $i = 1,\cdots,n$, are Gauss transformations; and U_1 is upper triangular. The transformed linear system of

(3.16) becomes

$$\begin{bmatrix} U_1 & E_1 & & & & G_1 \\ & \tilde{C}_2 & E_2 & & & \tilde{G}_2 \\ & A_3 & C_3 & & & \\ & & \ddots & \ddots & & \\ & & & A_N & C_N & \\ & & & & B_N & B_0 \end{bmatrix} \begin{bmatrix} \mathbf{y}_2 \\ \mathbf{y}_3 \\ \mathbf{y}_4 \\ \vdots \\ \mathbf{y}_{N+1} \\ \mathbf{y}_1 \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{q}}_1 \\ \tilde{\mathbf{q}}_2 \\ \mathbf{q}_3 \\ \vdots \\ \mathbf{q}_N \\ \mathbf{d} \end{bmatrix}. \tag{3.17}$$

If we apply a similar process to the blocks $\begin{bmatrix} \tilde{C}_k \\ A_{k+1} \end{bmatrix}$ for $k = 2, 3, \cdots, N-1$, we obtain the final system

$$\begin{bmatrix} U_1 & E_1 & & & & G_1 \\ & U_2 & E_2 & & & G_2 \\ & & \ddots & \ddots & & \vdots \\ & & & U_{N-1} & E_{N-1} & G_{N-1} \\ & & & & \tilde{C}_N & \tilde{A}_N \\ & & & & B_N & B_0 \end{bmatrix} \begin{bmatrix} \mathbf{y}_2 \\ \mathbf{y}_3 \\ \vdots \\ \mathbf{y}_N \\ \mathbf{y}_{N+1} \\ \mathbf{y}_1 \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{q}}_1 \\ \tilde{\mathbf{q}}_2 \\ \vdots \\ \tilde{\mathbf{q}}_{N-1} \\ \tilde{\mathbf{q}}_N \\ \mathbf{d} \end{bmatrix}. \tag{3.18}$$

The vector $\mathbf{y}_1$ and $\mathbf{y}_{N+1}$ can be obtained easily by solving the (dense) linear system

$$\begin{bmatrix} \tilde{C}_N & \tilde{A}_N \\ B_N & B_0 \end{bmatrix} \begin{bmatrix} \mathbf{y}_{N+1} \\ \mathbf{y}_1 \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{q}}_N \\ \mathbf{d} \end{bmatrix}, \tag{3.19}$$

and $\mathbf{y}_k$, $k = 2, \cdots, N$, can be recovered by back substitution from the equation (3.18). Wright [76] proves that under certain assumptions, this method is equivalent to Mattheij's block LU decomposition and that it is stable. However, it is not clear how often the required assumptions are satisfied. In fact, if no pivoting occurs between the blocks $\tilde{C}_k$ and A_{k+1}, then this method is equivalent to the compactification algorithm. The worst case error growth is $O(2^{(N+1)n})$.

(3) *QR factorization*

In [77], Wright also proposes applying a Householder QR factorization procedure to solve the linear system (3.16). The process is similar; instead of applying LU decompositions, we use the QR factorization to obtain the coefficient matrix (3.18). Wright has proved the stability of this algorithm. The serial version of the LU decomposition is about twice as fast as the QR factorization. Both methods have the advantage that they can be easily parallelized [76, 77].

To compare the stability properties of these methods, we now study two examples. In our numerical results, the multiple shooting method uses the ODE initial value code DVERK from NETLIB and the tolerance in this routine is set to 10^{-10}; the finite difference code uses the midpoint scheme.

Example 3.1

$$\begin{aligned} y''' &= 20y'' + y' - 20y \\ y(0) &= 0.1e^{-T} + e^{-20T} + 0.1 \\ y(T) &= 1.1 + 0.1e^{-T} \\ y'(T) &= 20.1 + 0.1e^{-T}. \end{aligned}$$

The exact solution is $y(t) = 0.1e^{t-T} + e^{20(t-T)} + 0.1e^{-t}$.

This example is constructed in [46] to show that the compactification algorithm can easily fail, which it does around $T = 2$. However, both the LU and QR algorithms solve the problem without any difficulty for much larger values of T (see Table 3.1).

	Multiple Shooting Method			Finite Difference Method		
T	Compact	LU method	QR method	Compact	LU method	QR method
1	3.7357 E-10	3.4657 E-10	3.4657 E-10	4.8981 E-03	4.8981 E-03	4.8981 E-03
2	7.4008 E-03	3.1327 E-09	3.1326 E-09	5.4491 E-02	2.0758 E-02	2.0758 E-02
5	4.2072 E 23	3.8738 E-09	3.8738 E-09	Fails	1.3534 E-01	1.3534 E-01
10	6.8387 E 63	1.2815 E-09	1.2815 E-09	2.6306 E 04	3.5170 E-01	3.5170 E-01

Table 3.1: Maximum absolute error for Example 3.1 (50 mesh intervals)

Example 3.2

$$\mathbf{y}' = \begin{bmatrix} -1 & 6 \\ 6 & -1 \end{bmatrix} \mathbf{y}$$

$$y_1(0) = 1 + e^{-5T}, \quad y_2(T) = 1 - e^{-7T}.$$

The exact solution is

$$\mathbf{y}(t) = \begin{bmatrix} e^{5(t-T)} + e^{-7t} \\ e^{5(t-T)} - e^{-7t} \end{bmatrix}.$$

This example is constructed by Wright [78] to show the instability for the LU algorithm applied to (3.16). We compute the solution for $T = 10$ and use an equal-spaced mesh, the compactification algorithm fails for all N, and the LU method fails for both multiple shooting and finite difference methods when the number of mesh points is big enough ($N > 177$). In fact, the fundamental solution for this example is

$$Y_i(t-t_i) = \frac{1}{2}\begin{bmatrix} e^{5(t-t_i)} + e^{-7(t-t_i)} & e^{5(t-t_i)} - e^{-7(t-t_i)} \\ e^{5(t-t_i)} - e^{-7(t-t_i)} & e^{5(t-t_i)} + e^{-7(t-t_i)} \end{bmatrix}, \qquad t \in [t_i, t_{i+1}].$$

For the multiple shooting method with equal-spaced mesh,

$$\begin{bmatrix} \tilde{C}_k \\ A_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -e^{5T/N} - e^{-7T/N} & -e^{5T/N} + e^{-7T/N} \\ -e^{5T/N} + e^{-7T/N} & -e^{5T/N} - e^{-7T/N} \end{bmatrix}$$

where N is the number of mesh points. Note, when $T = 10$, $e^{5T/N}+e^{-7T/N} < 1$ for $N > 177$. In this situation, there is no pivoting between blocks $\tilde{C}_k$ and A_{k+1} and the LU method is equivalent to the compactification algorithm. On the other hand, the QR method performs the pivoting between these two blocks, and the method is stable (see Table 3.2).

	Multiple Shooting Method			Finite Difference Method		
N	Compact	LU method	QR method	Compact	LU method	QR method
50	1.0000 E 00	4.3887 E-09	4.3887 E-09	1.0000 E 00	7.0126 E-02	7.0126 E-02
150	1.0000 E 00	5.9750 E-09	5.6750 E-09	1.0000 E 00	6.8273 E-03	6.8273 E-03
200	1.0000 E 00	1.0000 E 00	4.6708 E-09	3.9188 E 05	1.0000 E 00	3.8006 E-03
500	1.0000 E 00	1.0000 E 00	3.1246 E-11	1.0000 E 00	1.0000 E 00	6.0204 E-04

Table 3.2: Maximum absolute error for Example 3.2 (T=10)

3.2 Nonlinear Problems

Since our purpose is to implement a stable and efficient linear system solver for AUTO, we will consider the collocation method with Lagrange basis polynomials for solving first order nonlinear ordinary differential equations. We will also allow the integral constraints and scalar parameters to appear in our systems. The stability properties of the various linear system solvers when implemented in AUTO will be considered. There are several other factors which may affect the stability of the resulting codes, for example, the mesh selection and continuation strategies. We will investigate the role of these factors with a few examples.

Consider the ODE system

$$\mathbf{u}'(t) = \mathbf{f}(\mathbf{u}(t), \boldsymbol{\lambda}), \qquad 0 \le t \le 1 \tag{3.20}$$

with boundary conditions

$$b_i(\mathbf{u}(0), \mathbf{u}(1), \boldsymbol{\lambda}) = 0, \qquad 1 \le i \le n_b \tag{3.21}$$

and integral constraints

$$\int_0^1 q_i(\mathbf{u}(t), \boldsymbol{\lambda})dt = 0, \qquad 1 \le i \le n_q \tag{3.22}$$

where $\mathbf{u}$, $\mathbf{f} \in \mathbf{R}^n$, $\boldsymbol{\lambda} \in \mathbf{R}^p$ and $p = n_b + n_q - n$. We solve the boundary value problem (3.20), (3.21) and (3.22) by a spline collocation method with an appropriate mesh

$$\pi : \ 0 = t_0 < t_1 < \cdots < t_N = 1, \quad \delta t_j = t_{j+1} - t_j, \quad 0 \le j \le N-1$$

and basis functions

$$\omega_{ji}(t), \quad 0 \le j \le N-1, \quad 0 \le i \le m.$$

Using a spline collocation method with m Gauss points per subinterval, the differential equations are discretized by

$$\mathbf{p}_j'(z_{ji}) = \mathbf{f}(\mathbf{p}_j(z_{ji}), \boldsymbol{\lambda}), \qquad i = 1, \cdots, m, \quad j = 1, \cdots, N+1; \tag{3.23}$$

the boundary conditions become

$$b_i(\mathbf{u}_1, \mathbf{u}_{N+1}, \boldsymbol{\lambda}) = 0, \qquad 1 \le i \le n_b; \tag{3.24}$$

and the integral equations are approximated by a Gaussian quadrature

$$\sum_{j=1}^{N} \sum_{i=0}^{m} \omega_{ji} q_i(\mathbf{u}_{j+\frac{i}{m}}, \boldsymbol{\lambda}) = 0, \qquad 1 \le i \le n_q. \tag{3.25}$$

The above discretization gives $mnN + n_b + n_q$ nonlinear algebraic equations which are then solved by a Newton iteration. When solving the linearized system, all of the local variables $\mathbf{u}_{j+\frac{i}{m}}, 1 < i < m$, are eliminated by a condensation of parameters algorithm, which leaves a linear system in the following form:

$$A \begin{bmatrix} \mathbf{u} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} A_1 & C_1 & & & & D_1 \\ & A_2 & C_2 & & & D_2 \\ & & \ddots & \ddots & & \vdots \\ & & & A_N & C_N & D_N \\ B_0 & B_1 & \cdots & B_{N-1} & B_N & E \end{bmatrix} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \vdots \\ \mathbf{u}_{N+1} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{q}_1 \\ \mathbf{q}_2 \\ \vdots \\ \mathbf{q}_{N+1} \\ \mathbf{q}_{\boldsymbol{\lambda}} \end{bmatrix} \tag{3.26}$$

where $A_i, C_i \in \mathbf{R}^{n\times n}$, $B_i \in \mathbf{R}^{n_r\times n}, D_i \in \mathbf{R}^{n\times p}, E \in \mathbf{R}^{n_r\times p}$, $n_r = n_b + n_q$.

AUTO solves this linear system by a Gaussian elimination with row partial pivoting algorithm similar to Wright's LU decomposition. The pivoting is only applied to the first nN equations and the blocks of $B_k, 1 \le k \le N-1$, are eliminated without any pivoting. One advantage of this linear system solver is that the Floquet multipliers of the periodic solutions can be obtained with little extra work. However, as we have seen in the Example 3.2, this type of linear solver is potentially unstable. An alternative is to modify Wright's QR algorithm to replace the current linear system solver. It is also easy to perform partial pivoting during the condensation of parameters. While we have experimented extensively with this modification to the condensation process, it is usually not critical yet it is very expensive, and as it is not important to us here, we will not discuss it further.

We now give a brief description of our implementation of Wright's Householder QR algorithm. Let the Householder transformation $\hat{Q}_1 \in \mathbf{R}^{2n\times 2n}$ satisfy

$$\hat{Q}^T \begin{bmatrix} A_1 & C_1 & & D_1 \\ & A_2 & C_2 & D_2 \end{bmatrix} = \begin{bmatrix} G_1 & R_1 & F_1 & H_1 \\ \tilde{A}_2 & & \tilde{C}_2 & \tilde{D}_2 \end{bmatrix} \tag{3.27}$$

where $R_1 \in \mathbf{R}^{n\times n}$ is upper triangular. If $I_M \in \mathbf{R}^{M\times M}$ where $M = (N-2)n + n_r$, then

$$QA = \begin{bmatrix} \hat{Q}^T & \\ & I_M \end{bmatrix} A = \begin{bmatrix} G_1 & R_1 & F_1 & & & & H_1 \\ \tilde{A}_2 & & \tilde{C}_2 & & & & \tilde{D}_2 \\ & & A_3 & C_3 & & & D_3 \\ & & & \ddots & \ddots & & \vdots \\ & & & & A_N & C_N & D_N \\ B_0 & B_1 & B_2 & \cdots & B_{N-1} & B_N & E \end{bmatrix}. \tag{3.28}$$

Similarly, let $\hat{Q}_i \in \mathbf{R}^{2n \times 2n}$ satisfies

$$\hat{Q}_i^T \begin{bmatrix} \tilde{A}_i & \tilde{C}_i & & \tilde{D}_i \\ & A_{i+1} & C_{i+1} & D_{i+1} \end{bmatrix} = \begin{bmatrix} G_i & R_i & F_i & H_i \\ \tilde{A}_{i+1} & & \tilde{C}_{i+1} & \tilde{D}_{i+1} \end{bmatrix} \qquad (3.29)$$

where $1 \leq i \leq N-1$ and R_i is upper triangular. Let

$$Q_i = \begin{bmatrix} I_{M_1} & & \\ & \hat{Q}_i & \\ & & I_{M_2} \end{bmatrix} \qquad i = 1, \cdots, N-1 \qquad (3.30)$$

where I_{M_1} and I_{M_2} are $(i-1)n$ by $(i-1)n$ and $(N-i-1)n + n_r$ by $(N-i-1)n + n_r$ identity matrices. Applying these Householder transformations to A, we obtain

$$Q_{N-1}^T \cdots Q_1^T A = \begin{bmatrix} G_1 & R_1 & F_1 & & & & H_1 \\ G_2 & & R_2 & F_2 & & & H_2 \\ \vdots & & & \ddots & \ddots & & \vdots \\ G_{N-1} & & & & R_{N-1} & F_{N-1} & H_{N-1} \\ \tilde{A}_N & & & & & \tilde{C}_N & \tilde{D}_N \\ B_0 & B_1 & \cdots & \cdots & B_{N-1} & B_N & E \end{bmatrix}. \qquad (3.31)$$

After eliminating blocks $B_1, B_2, \cdots, B_{N-1}$, we obtain the same coefficient matrix structure as does AUTO's original linear system solver.

The original linear system solver in AUTO is virtually the same as the LU algorithm we discussed in the previous section and will thus be referred to below as simply the LU method. Similar to the linear case, the LU method is unstable in theory while the QR method is stable. As for the efficiency in the serial version of these two methods, the QR method roughly doubles the cost of the LU method. In practice, however, because the condensation of parameters takes most of the CPU time, the actual increase is not very significant. Our numerical tests indicate that the difference between the two algorithms is usually within 15% in CPU time. In fact, there are certain cases that the QR method out-performs the LU method due to its stability nature.

Both the LU and QR algorithms have similar characteristics. There is no column pivoting during the elimination, and boundary condition and integral condition blocks are eliminated without pivoting with the equations from the discretization of the differential equations. The advantage of these types of algorithm is that the Floquet multipliers for the periodic solution can be obtained easily with little extra work. In fact they are the eigenvalues of the matrix $\Phi = -\tilde{C}_N^{-1}\tilde{A}_N$.

Theorem 3.1 *Let the matrix*

$$A = \begin{bmatrix} A_1 & C_1 & & \\ & \ddots & \ddots & \\ & & A_N & C_N \end{bmatrix}$$

be the discretization of the Jacobian for the ODE (3.20). *If there exists a nonsingular matrix* $L \in \mathbf{R}^{nN \times nN}$ *such that*

$$LA = \begin{bmatrix} \cdot & \cdot & \cdots & \cdot & \cdot \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ P_0 & 0 & \cdots & 0 & P_1 \end{bmatrix}$$

where $P_0,\ P_1 \in \mathbf{R}^{n\times n}$, *then* $-P_1^{-1}P_0$ *is an approximate monodromy matrix.*

Proof If Φ_i, $i = 0, 1, \cdots, N$, are the discrete fundamental solutions of equation (2.10) at $t = t_i$, then $A_i\Phi_{i-1} + C_i\Phi_i = 0$, $\Phi_0 = I$. Therefore,

$$LA \begin{bmatrix} \Phi_0 \\ \vdots \\ \Phi_N \end{bmatrix} = \begin{bmatrix} \cdot & \cdot & \cdots & \cdot & \cdot \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ P_0 & 0 & \cdots & 0 & P_1 \end{bmatrix} \begin{bmatrix} \Phi_0 \\ \vdots \\ \Phi_N \end{bmatrix} = 0.$$

Hence $P_0\Phi_0 + P_1\Phi_N = 0$ which yields $\Phi_N = -P_1^{-1}(P_0\Phi_0) = -P_1^{-1}P_0$. □

In practice, the matrix $\tilde{C}_N$ may be ill-conditioned and the eigenvalues of Φ cannot be accurately computed. However, one can solve the generalized eigenvalue problem

$$\tilde{C}_N \mathbf{v} = -\lambda \tilde{A}_N \mathbf{v}$$

to recover more accurate approximations of the Floquet multipliers [31].

We now focus on the stability analysis of the two methods. First, we test both methods with Examples 3.1 and 3.2. Somewhat surprisingly, not only the QR solver but *also* the LU solver computes the solutions without any difficulty (see Table 3.3 and 3.4). In these tests, both fixed and adaptive meshes are used. One possible explanation for the success of AUTO (with the LU solver) for Example 3.2 is that the addition of the continuation parameter equation appears to help stablize the problem.

	Without Adaptive Mesh (IAD=0)		With Adaptive Mesh (IAD=1)	
T	LU method	QR method	LU method	QR method
1	4.9434 E-05	4.9434 E-05	2.6954 E-06	2.6954 E-06
2	5.9528 E-04	5.9528 E-04	1.1513 E-04	1.1513 E-04
5	1.0737 E-02	1.0737 E-02	1.6647 E-04	1.6647 E-04
10	5.8607 E-02	5.8607 E-02	6.2498 E-04	6.2498 E-04

Table 3.3: Maximum absolute error for Example 3.1 (use AUTO, NTST=50, NCOL=2)

	Without Adaptive Mesh (IAD=0)		With Adaptive Mesh (IAD=1)	
NTST	LU method	QR method	LU method	QR method
50	3.7409 E-03	3.7409 E-03	9.6359 E-05	9.6359 E-05
150	8.7258 E-05	8.7258 E-05	2.1765 E-06	2.1765 E-06
200	3.0056 E-05	3.0056 E-05	1.0281 E-06	1.0704 E-06
500	8.9945 E-07	8.9945 E-07	1.0833 E-07	1.0833 E-07

Table 3.4: Maximum absolute error for Example 3.2 (use AUTO, T=10, NCOL=2)

In order to further challenge the LU method for AUTO, we extend Example 3.2 to the following example:

Example 3.3

$$\mathbf{y}' = \begin{bmatrix} -1 & 6 & & \\ 6 & -1 & & \\ & & -1 & 8 \\ & & 8 & -1 \end{bmatrix} \mathbf{y}$$

$$y_1(0) = 1 + e^{-5T}, \qquad y_2(T) = 1 - e^{-7T}$$

$$y_3(0) = 1 + e^{-7T}, \qquad y_4(T) = 1 - e^{-9T}.$$

The exact solution is

$$\mathbf{y} = \begin{bmatrix} e^{5(t-T)} + e^{-7t} \\ e^{5(t-T)} - e^{-7t} \\ e^{7(t-T)} + e^{-9t} \\ e^{7(t-T)} - e^{-9t} \end{bmatrix}.$$

Not surprisingly, when using the multiple shooting and finite difference methods to solve this example, only the QR factorization is stable (see Table 3.5). We also run the problem with AUTO, using T as the continuation parameter and starting the solution from $T = 1$. As before, we turn off the adaptive mesh selection (set IAD=0). AUTO successfully computes the solution for at least $T = 20$ when using a small number of mesh intervals. However, when we increases the mesh number (NTST) to 300, AUTO fails at $T = 8.9893$. On the other hand, the QR solver is successful in all cases until at least $T = 20$ (see Table 3.6). This result seems to demonstrate that the continuation parameter equation in AUTO helps to stablize the problem when there is only one increasing mode, but with two increasing modes, instability may occur.

	Multiple Shooting Method			Finite Difference Method		
T	Compact	LU method	QR method	Compact	LU method	QR method
2	7.1638 E-11	5.0653 E-10	2.1455 E-13	1.1039 E-04	1.1039 E-04	1.1039 E-04
5	4.1914 E-02	1.3137 E-01	4.6527 E-11	7.1832 E-02	1.3137 E-01	6.9059 E-04
8	1.0000 E 00	1.0000 E 00	6.6350 E-10	1.1553 E 08	2.3920 E 00	1.7756 E-03
10	1.0000 E 00	1.0000 E 00	2.1705 E-09	1.3333 E 14	1.2665 E 00	2.7725 E-03

Table 3.5: Maximum absolute error for Example 3.3 (N=300)

	Without Adaptive Mesh (IAD=0)		With Adaptive Mesh (IAD=1)	
T	LU method	QR method	LU method	QR method
2	4.7525 E-08	4.7525 E-08	3.9444 E-08	3.9444 E-08
5	1.1764 E-06	1.1764 E-06	5.8461 E-09	5.8461 E-09
8	7.2077 E-06	7.2077 E-06	1.0866 E-08	1.0866 E-08
10	Fails at T=8.9893	1.6817 E-05	3.1209 E-08	3.1209 E-08

Table 3.6: Maximum absolute error for Example 3.3 (use AUTO, NTST=300, NCOL=2)

Interestingly, after we turn on the mesh selection (IAD=1), in all of the above cases, AUTO successfully computes the solution with the LU method. This shows how a good mesh selection strategy can help to stablize the problem. Presumably, this stablizing effect of the adaptive mesh selection would carry over for many other BVP solvers.

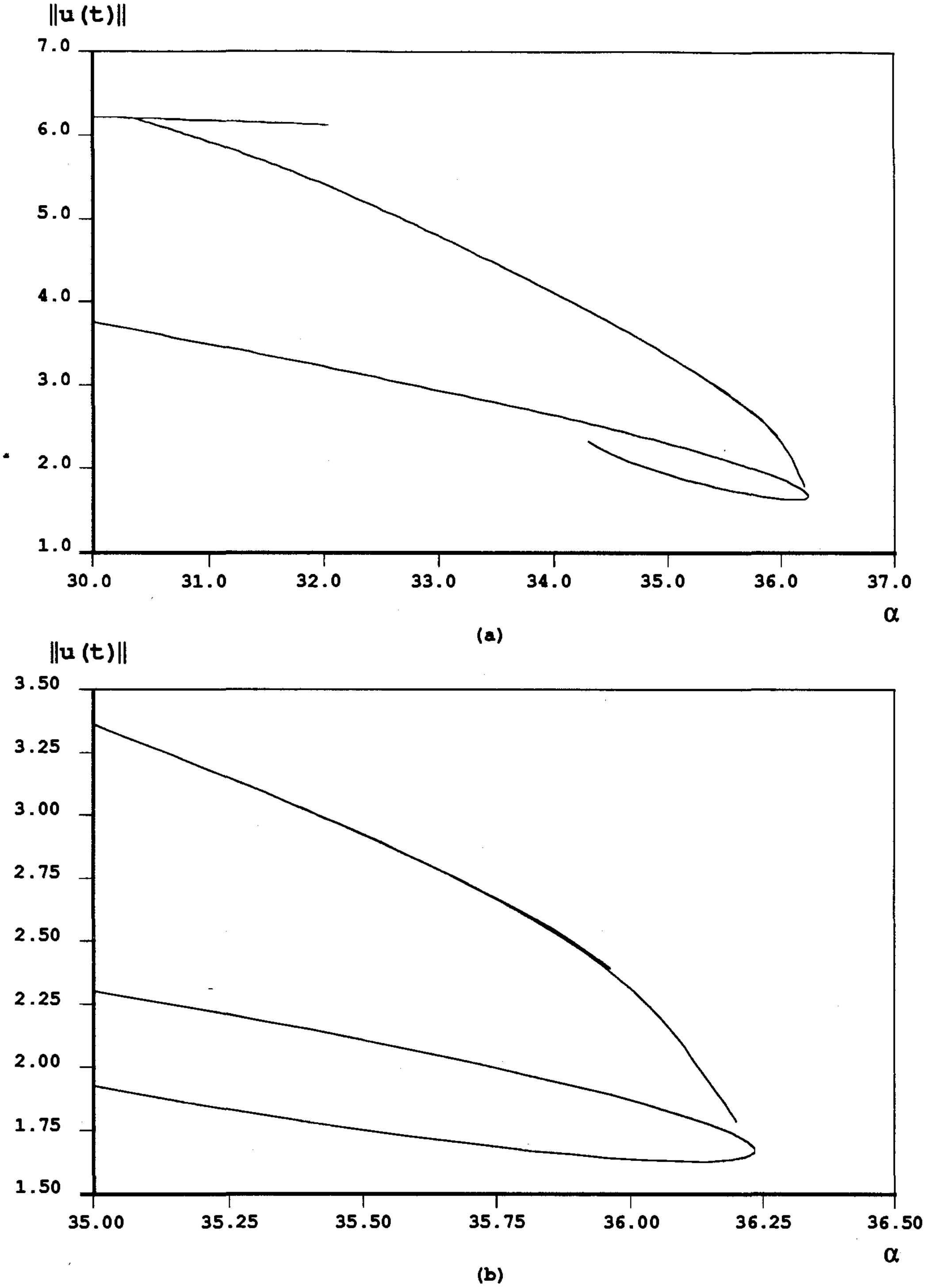
The following practical example shows that the advantage of the QR solver over the LU solver.

Example 3.4 *Kuramoto-Sivashinsky equations* [40]

$$\frac{\partial u}{\partial t} + 4\frac{\partial^4 u}{\partial x^4} + \alpha[\frac{\partial^2 u}{\partial x^2} + u\frac{\partial u}{\partial x}] = 0, \quad (x,t) \in \mathbf{R} \times \mathbf{R}^+,$$

$$u(x,t) = u(x+2\pi,t), \quad u(x,t) = -u(2\pi - x,t).$$

The detailed description of model and the numerical discretization for the K-S equations will appear in the next chapter. We use this example to test the stability of linear system solvers. We use AUTO to compute the main branch of periodic solutions which is born from the Hopf bifurcation point at $\alpha = 30.34522$ on the steady state solution branch (branch 1 in Figure 3.1). This periodic solution branch approaches a homoclinic cycle formed by two heteroclinic orbits (see discussion in Section 4.2). We use a 12-mode traditional Galerkin method with 10 mesh intervals and 4 collocation points on each interval to follow this periodic branch starting from this Hopf bifurcation point. Both the LU and QR methods fail after 115 continuation steps at $\alpha = 35.97086$. After increasing the number of mesh intervals to 40 and restarting the continuation from $\alpha = 35.37104$, the LU methods fails again after only 39 continuation steps when it reaches the turning point at $\alpha = 36.19939$. Using the QR method, we successfully continue this periodic solution branch for another 305 steps without difficulties (see Figures 3.1-3.2). As for the efficiency of both solvers, we compare the CPU time used for the continuation of the first 100 periodic solutions starting from the Hopf bifurcation point. Our test is performed on a SUN 4/670MP using the SUN Fortran 1.4 compiler. The LU solver takes 241 seconds while QR solver spends 254 seconds to complete the test. The difference is only about 5%. In fact, the total time spent on solving the block bi-diagonal systems is around 20 seconds, which is less than 10% of the total CPU time. Indeed, most of the CPU time is taken by the local condensation (more then 60%) and the linear system setup (more than 20%).



(a)

(b)

Figure 3.1: LU method for solving KS equation

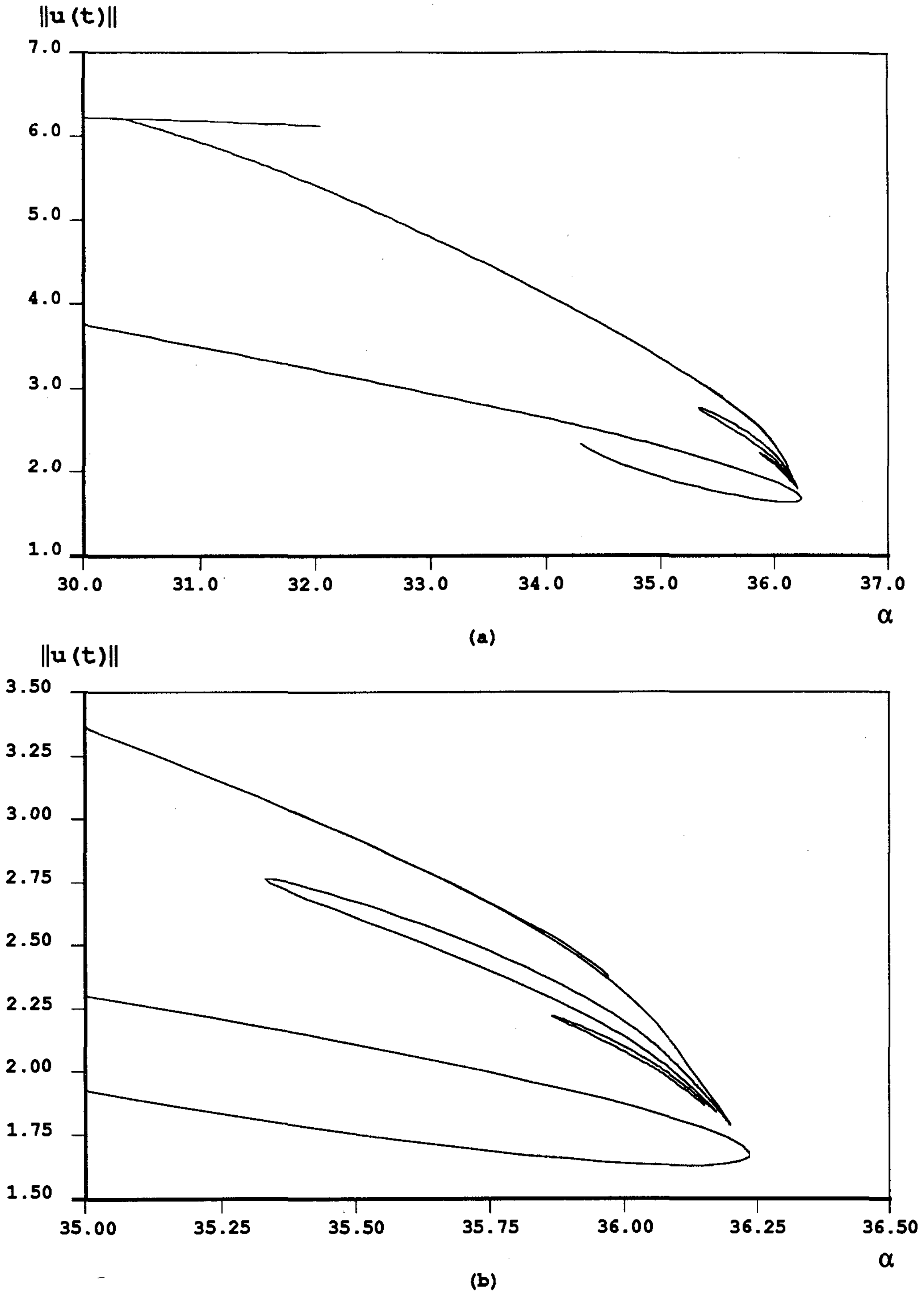


Figure 3.2: QR method for solving KS equation

Chapter 4

Computing Connecting Orbits

The appearance and disappearance of the solutions connecting one or several saddle points is one of the (extra) aspects in global bifurcation analysis for dynamical systems. In this chapter, we discuss numerical methods for computing these connecting orbits.

4.1 Connecting Orbits and Their Applications

Consider the dynamical system

$$\mathbf{u}'(t) = \mathbf{f}(\mathbf{u}(t), \boldsymbol{\lambda}), \qquad -\infty < t < \infty \tag{4.1}$$

where $\mathbf{u} \in \mathbf{R}^n$ and $\boldsymbol{\lambda} \in \mathbf{R}^p$. For certain $\boldsymbol{\lambda} = \bar{\boldsymbol{\lambda}}$, if there exists a non-constant solution $\bar{\mathbf{u}}(t)$ of (4.1) such that

$$\bar{\mathbf{u}}_- = \lim_{t \to -\infty} \bar{\mathbf{u}}(t), \quad \bar{\mathbf{u}}_+ = \lim_{t \to \infty} \bar{\mathbf{u}}(t), \quad \mathbf{f}(\bar{\mathbf{u}}_\pm, \bar{\boldsymbol{\lambda}}) = 0, \tag{4.2}$$

then the pair $(\bar{\mathbf{u}}(t), \bar{\boldsymbol{\lambda}})$ is called a *connecting orbit between stationary solutions* (COSS) or simply a *connecting orbit.* Moreover, if $\bar{\mathbf{u}}_- = \bar{\mathbf{u}}_+$, the orbit is called a *homoclinic orbit*; otherwise, it is called a *heteroclinic orbit.* A closed path formed by several heteroclinic orbits is called a *homoclinic cycle.*

To motivate the importance of connecting orbits, we begin this chapter with two applications of homoclinic and heteroclinic orbits: travelling wave problems for parabolic partial differential equations and limits of periodic solutions.

4.1.1 Travelling Waves in Parabolic Systems

Consider the one dimensional nonlinear parabolic partial differential equation

$$u_t = u_{xx} + f(u,\lambda), \quad t > 0, \quad -\infty < x < \infty. \tag{4.3}$$

If $u(x,t)$ is a travelling wave solution of the PDE (4.3) with a constant wave speed c, then $u(x,t)$ takes the form

$$u(x,t) = U(x+ct).$$

Substituting this expression into the PDE (4.3), we obtain the second order ODE

$$U'' - cU' + f(U,\lambda) = 0. \tag{4.4}$$

When u_1 and u_2 are the stationary solutions of the ODE (4.4), they are also the zeros of the function f. If $u_1 \neq u_2$, the *front wave* of the PDE (4.3) is a non-constant solution of (4.4) with

$$U(-\infty) = u_1, \quad U(\infty) = u_2.$$

Therefore U forms a heteroclinic orbit for the ODE (4.4). When $u_1 = u_2$, U is a homoclinic orbit of the ODE (4.4) which represents a *pulse wave* for the PDE (4.3).

Example 4.1 *The Nagumo equations*

$$\begin{aligned} u_t &= u_{xx} + f(u,a) \\ f(u,a) &= u(1-u)(u-a), \qquad 0 < a < 1. \end{aligned}$$

This equation is a special case of the *Fitzhugh-Nagumo equation* [32, 58],

$$\begin{aligned} u_t &= u_{xx} + u(1-u)(u-a) - v, \qquad 0 < a < 0.5 \\ v_t &= \epsilon(u - \gamma v), \qquad \epsilon, \gamma \geq 0, \end{aligned}$$

which is a conceptual model for nerve-membrane excitability. Here, t is dimensionless time, x is dimensionless distance along the "nerve", u is associated to membrane potential and v is the recovery variable.

The Nagumo equation has two branches of travelling wave solutions

$$u(x,t) = U(x+ct) = \frac{e^{\frac{1}{\sqrt{2}}(x+ct)}}{1 + e^{\frac{1}{\sqrt{2}}(x+ct)}}$$

with wave speed $c = \pm\sqrt{2}(a - \frac{1}{2})$. These two branches of heteroclinic orbits form a branch of homoclinic cycles on the phase plane (U, U') with stationary points $(0,0)$ and $(1,0)$ (see Figure 4.1).

4.1.2 Periodic Solutions

The appearance of homoclinic orbits and homoclinic cycles often occurs from the "birth" and "death" of periodic solutions where the periods tend to infinity. To demonstrate this phenomenon we present two examples: the Lorenz equations [53] and the Kuramoto-Sivashinsky equations [40, 62].

Example 4.2 *The Lorenz equations*

$$\begin{aligned} x' &= \sigma(y - x) \\ y' &= \lambda x - y - xz \\ z' &= xy - \mu z. \end{aligned}$$

The Lorenz equations provide a simple model for weather forecasting. It describes the dynamics of flow and temperature in a rectangle region, where there are only three modes of action and they are represented by x, y, z. In the equation, σ is the Prandtl number of the fluid, λ is a dimensionless Rayleigh number, and $\mu = 4/(1 + a^2)$ where a is the aspect ratio of the rectangle.

We assume that $\sigma = 10$, $\mu = \frac{8}{3}$ and λ is the continuation parameter. When $\lambda < 1$, the trivial stationary solution $(0,0,0)$ is stable. This solution branch goes to a pitchfork bifurcation at $\lambda = 1$, and the new steady state solutions

$$(\pm\sqrt{\mu(\lambda - 1)},\ \pm\sqrt{\mu(\lambda - 1)},\ \lambda - 1)$$

are stable. At the Hopf bifurcation point $\lambda_H = 24.73684$, the non-trivial steady state solution branches lose their stability. Two branches of periodic orbits born at Hopf bifurcation points converge to the homoclinic orbits at $\lambda = 13.92656$ (see Figure 4.2). Note both periodic branches exist only for $\lambda \leq \lambda_H$, and they are unstable. In fact, when $\lambda > \lambda_H$, the invariant manifold for the Lorenz equations is a strange attractor.

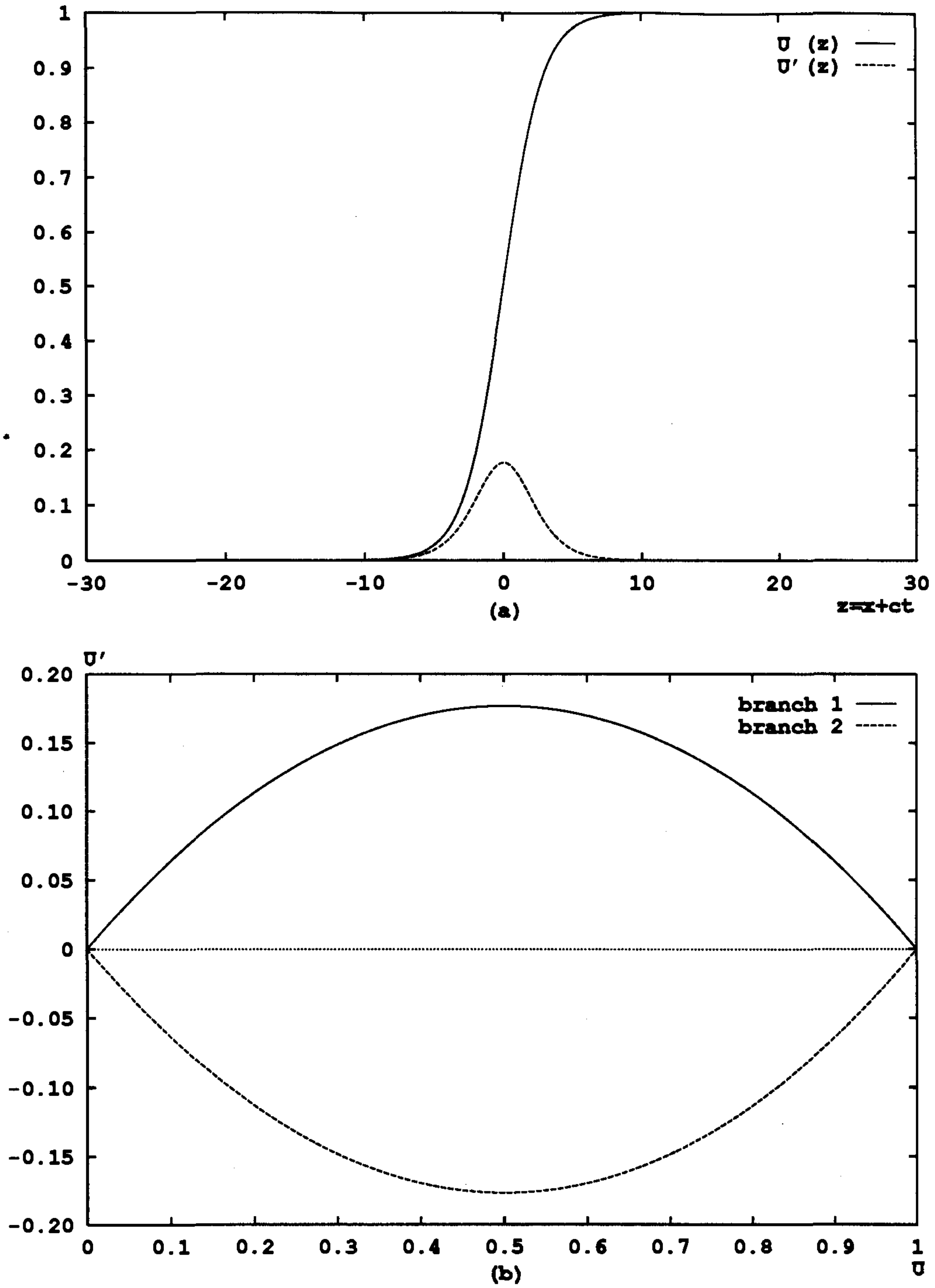
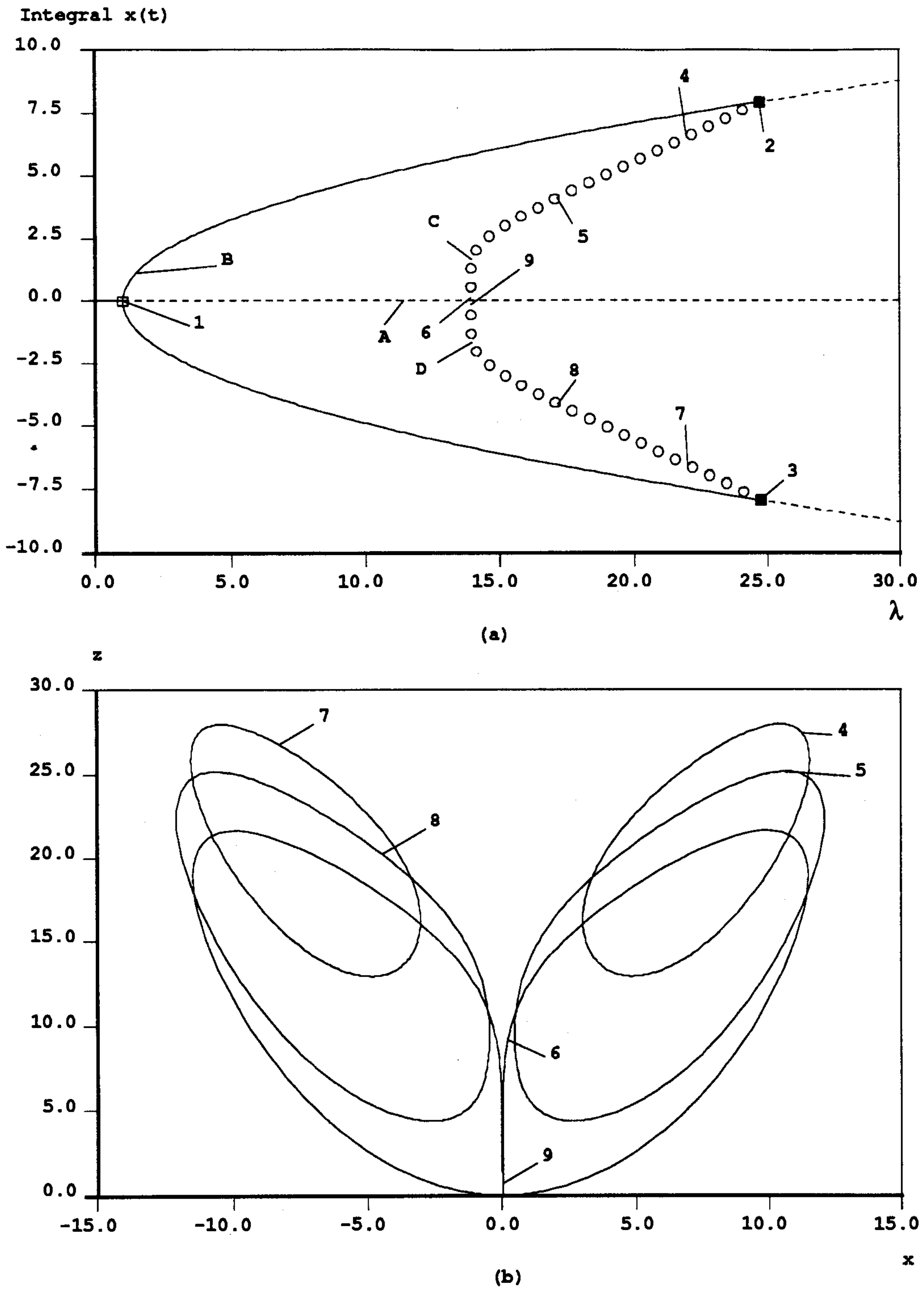


Figure 4.1: Travelling wave solutions in the Nagumo equation



(a)

(b)

Figure 4.2: Periodic solution branches approach to homoclinic orbits in the Lorenz equation

Example 4.3 *The Kuramoto-Sivashinsky equations*

$$\frac{\partial u}{\partial t}+4\frac{\partial^4 u}{\partial x^4}+\alpha[\frac{\partial^2 u}{\partial x^2}+u\frac{\partial u}{\partial x}]=0, \quad (x,t)\in \mathbf{R}\times\mathbf{R}^+,$$

$$u(x,t)=u(x+2\pi,t), \quad u(x,t)=-u(2\pi-x,t).$$

The K-S equations can arise in many applications, such as temporal phase evolution of reaction-diffusion chemical systems [43] and flame front propagation where $u(x,t)$ represents the perturbation of a plane flame front which propagates in a fuel-oxygen mixture [68]. When α is large enough, the flow tends to a turbulent state as $t \to \infty$.

It is known that there exists a *global attractor* for the K-S equation which is invariant, has finite Hausdorff dimension and attracts all solutions to it. However, the attractor may be topologically very complicated and attract the solutions at very slow rate. Therefore, in the numerical computation, it is more efficient to compute the *inertial manifold*, a finite dimensional invariant manifold which attracts all trajectories exponentially. Obviously, the global attractor is a subset of the inertial manifold. Spectral methods for computing inertial manifolds for K-S equations have been discussed by many authors recently, e.g., see [40, 62]. The numerical results in Figures 4.3–4.8 are obtained by using a 16 mode flat (traditional) Galerkin method as the spatial discretization of the PDE. We solve the resulting (16 dimensional) ODE system with AUTO. Figure 4.3 shows the bifurcation behavior of the K-S equation. When $\alpha = 30.34523$, a Hopf bifurcation occurs for the stationary solution branch 2, and a stable periodic solution branch 5 appears. This periodic branch goes to a secondary bifurcation at $\alpha = 32.85349$. The main branch becomes unstable and eventually converges to a homoclinic cycle (a pair of heteroclinic orbits) at about $\alpha = 36.12736$ (Figure 4.4). The secondary periodic solution branch 6 becomes unstable after a period doubling bifurcation at $\alpha = 32.97169$, and it converges to a homoclinic trajectory near $\alpha = 34.36606$ (Figure 4.5). Period doubling bifurcation repeats for branch 7 which also converges to a homoclinic orbit (Figure 4.6). At $\alpha = 34.29904$, a Hopf bifurcation occurs on the steady state solution branch 4, and the stable periodic branch 8 is born. This periodic branch also goes to a period doubling bifurcation, and both branch 8 and branch 9 terminate on the homoclinic orbits (Figure 4.7–4.8). Unlike the homoclinic orbits and heteroclinic orbits for the Lorenz equations and Nagumo equations, all the connecting orbits are are Šilnikov-type in which complex conjugate pair of eigenvalues exist for the Jacobian at the fixed points [67].

This type of connection is also called a *saddle-focus* bifurcation. All results were obtained by the author use'AUTO.

4.2 Well-Posed Problem for the Connecting Orbits

We start our discussion by considering a solution pair $(\mathbf{u}(t), \lambda)$ for the equation (4.1) and (4.2). We assume that both $\mathbf{u}_+$ and $\mathbf{u}_-$ are hyperbolic fixed points, and let $\mathcal{M}^S_\pm$ and $\mathcal{M}^U_\pm$ be the *stable* and *unstable manifolds* of the stationary solutions $\mathbf{u}_\pm$ defined by

$$\begin{aligned} \mathcal{M}^S_\pm &= \{\mathbf{u} \in \mathbf{R}^n : \lim_{t\to\infty} \mathbf{u}(t) = \bar{\mathbf{u}}_\pm\}, \\ \mathcal{M}^U_\pm &= \{\mathbf{u} \in \mathbf{R}^n : \lim_{t\to-\infty} \mathbf{u}(t) = \bar{\mathbf{u}}_\pm\}, \end{aligned}$$

respectively, with dimensions

$$\dim(\mathcal{M}^U_-) = n_-, \qquad \dim(\mathcal{M}^S_-) = n - n_-,$$

$$\dim(\mathcal{M}^S_+) = n_+, \qquad \dim(\mathcal{M}^U_+) = n - n_+.$$

Note that when $\mathcal{M}^U_-$ and $\mathcal{M}^S_+$ intersect, $\mathcal{M}^U_- \cap \mathcal{M}^S_+$ is at least one dimensional. Moreover, if $n_- + n_+ = n + 1$ there exists exactly one orbit between $\mathbf{u}_-$ and $\mathbf{u}_+$. When $n_- + n_+ > n + 1$, $\dim(\mathcal{M}^U_- \cap \mathcal{M}^S_+) > 1$, and further conditions are required to obtain the unique connecting orbit. When $n_- + n_+ < n + 1$, the connection solution is not structurally stable, and we need to add some free parameters to establish the stable connection between $\mathbf{u}_-$ and $\mathbf{u}_+$. In this case, the number of the free parameters is

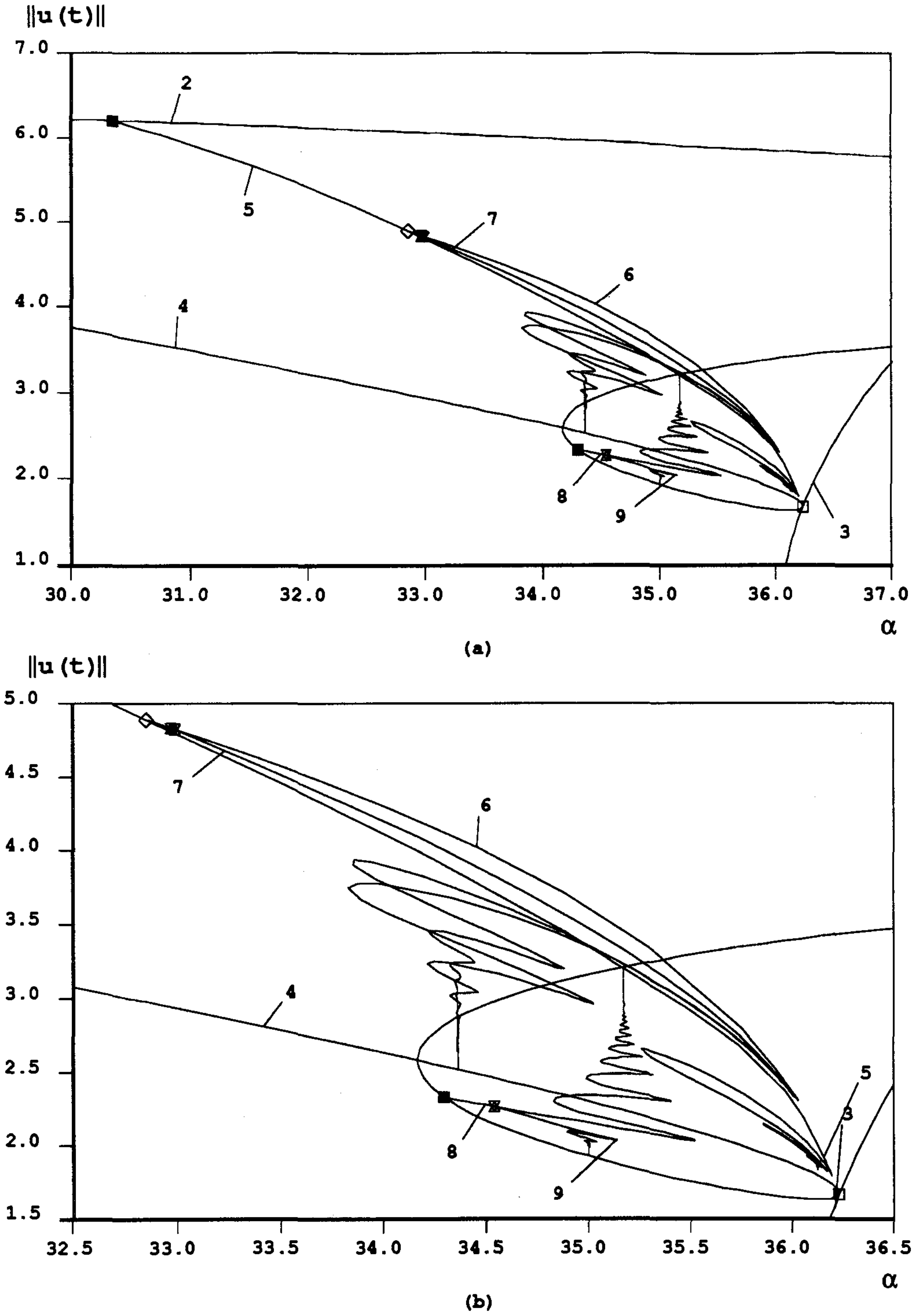
$$p = n + 1 - (n_- + n_+). \tag{4.5}$$

In particular, for homoclinic orbits, $p = 1$ or $n_- + n_+ = n$. In this thesis, we only consider the case for $n_- + n_+ \leq n + 1$. The following definition is given by Beyn [10].

Definition 4.1 *A connecting orbit between stationary solutions* $(\bar{\mathbf{u}}(t), \bar{\lambda})$ *is said to be non-degenerate if both* $\mathbf{u}_-$ *and* $\mathbf{u}_+$ *are hyperbolic fixed points,* $p = n + 1 - (n_- + n_+)$ *and the ODE system*

$$\mathbf{v}' = \mathbf{f}_\mathbf{u}(\bar{\mathbf{u}}, \bar{\lambda})\mathbf{v} + \mathbf{f}_\lambda(\bar{\mathbf{u}}, \bar{\lambda})\boldsymbol{\mu} \tag{4.6}$$

has only the trivial solution $\boldsymbol{\mu} = \mathbf{0}$, $\mathbf{v} = c\bar{\mathbf{u}}'$ *for all* $c \in \mathbf{R}$.



(a)

(b)

Figure 4.3: Periodic solution branches approach to connecting orbits in the K-S equation

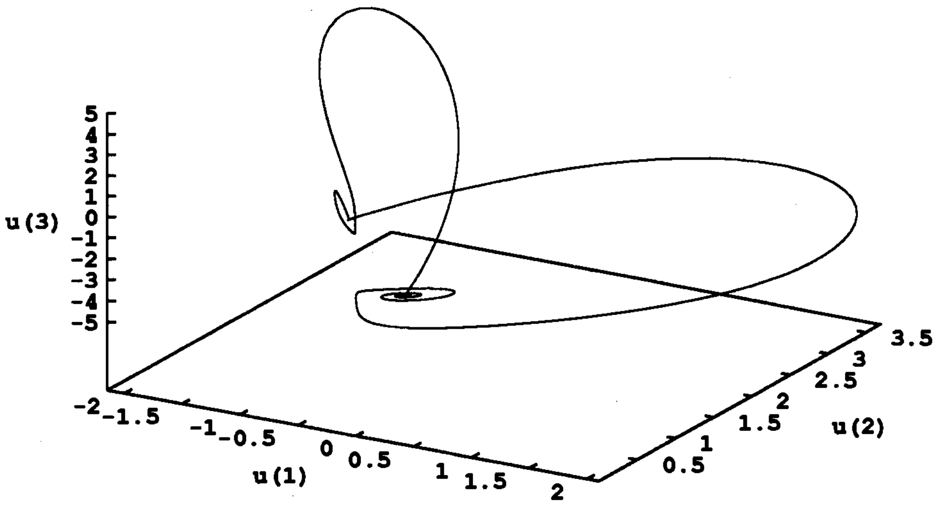


Figure 4.4: Periodic solution branch 5 converges to a homoclinic cycle

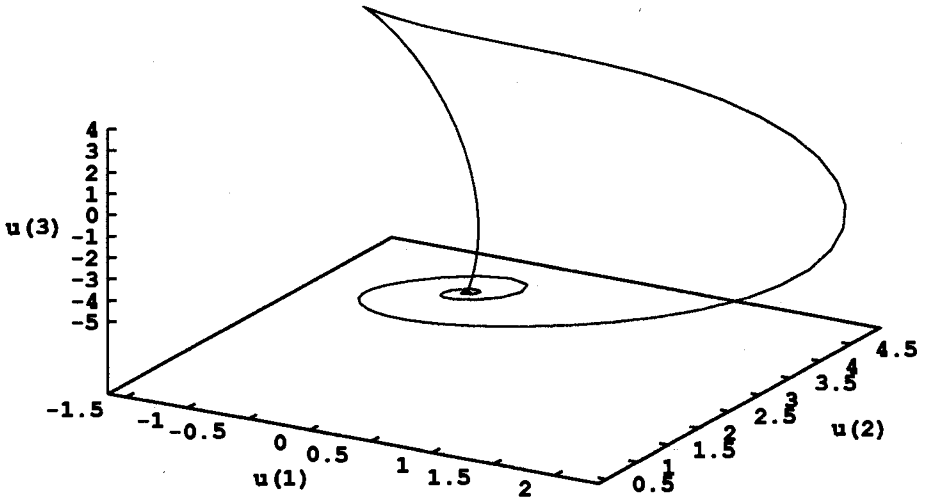


Figure 4.5: Periodic solution branch 6 converges to a homoclinic orbit

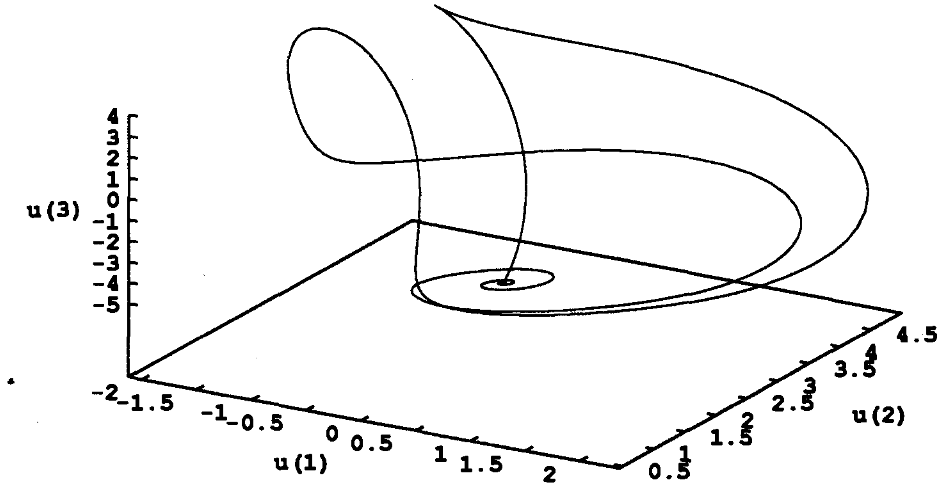


Figure 4.6: Periodic solution branch 7 converges to a homoclinic orbit

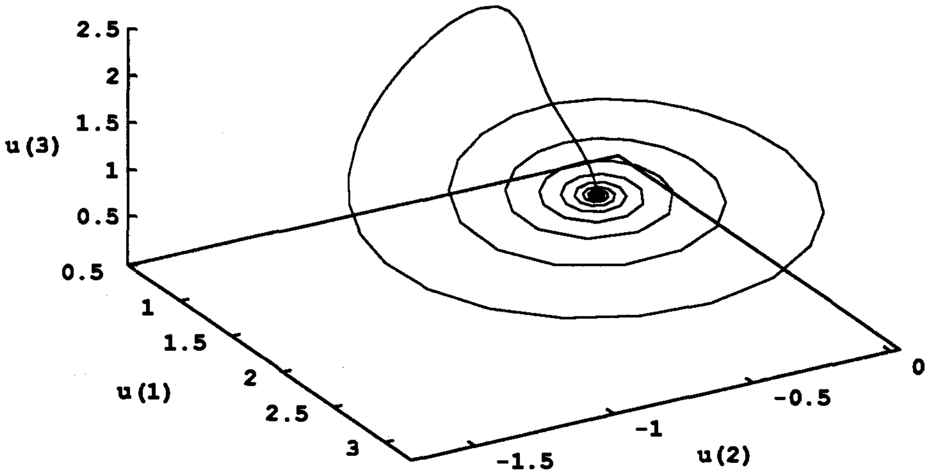


Figure 4.7: Periodic solution branch 8 converges to a homoclinic orbit

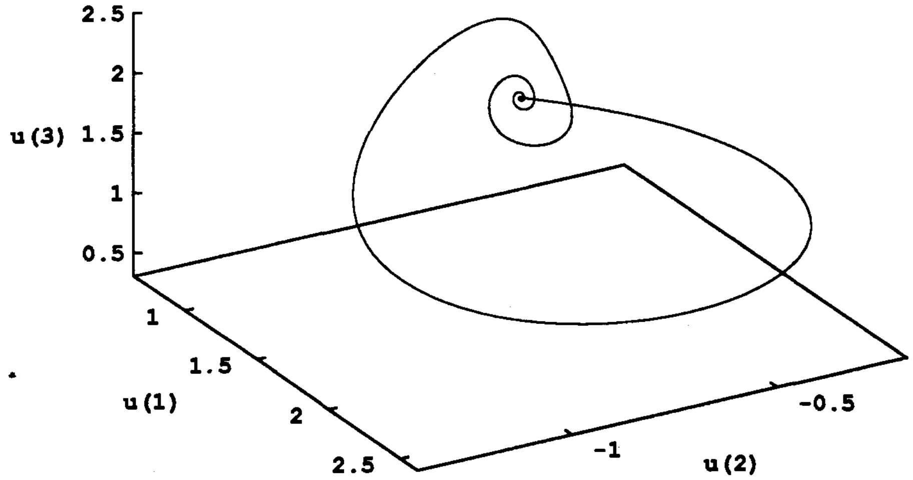


Figure 4.8: Periodic solution branch 9 converges to a homoclinic orbit

Let us consider the variational equation of (4.1) for a solution pair $(\mathbf{u}(t), \boldsymbol{\lambda})$

$$\mathbf{v}' = A(t)\mathbf{v} \tag{4.7}$$

where $A(t) = \mathbf{f}_{\mathbf{u}}(\mathbf{u}(t), \boldsymbol{\lambda})$ is continuous and satisfies

$$\lim_{t \to \pm\infty} A(t) = A_{\pm} := \mathbf{f}_{\mathbf{u}}(\mathbf{u}_{\pm}, \boldsymbol{\lambda}). \tag{4.8}$$

The theory of exponential dichotomy introduced in chapter 2 allows us to further investigate the properties of COSS. In the following theorem, we summarize the results obtained by Beyn [10, 11], Coppel [22] and Palmer [59].

Theorem 4.1 *Let $Y(t)$ be a fundamental solution for (4.7) with $Y(0) = I$. The ODE system (4.7) has an exponential dichotomy on both $[0, \infty)$ and $(-\infty, 0]$ if (4.8) is satisfied. Moreover, if P and Q are corresponding projectors, then*

$$\lim_{t \to \infty} Y(t)PY(t)^{-1} = P_{+}, \qquad \lim_{t \to -\infty} Y(t)(I - Q)Y(t)^{-1} = P_{-}$$

where P_{+} and P_{-} are the projections onto $\mathcal{M}_{+}^{S}$ and $\mathcal{M}_{-}^{U}$, respectively.

With the help of Theorem 4.1, Beyn further obtains the following results.

Theorem 4.2 *Define the Banach spaces*

$$\mathcal{X}^k = \{\mathbf{u} \in C^k(\mathbf{R}, \mathbf{R}^n) : \lim_{t \to \pm\infty} \mathbf{u}^{(i)} \textit{ exist for } 0 \leq i \leq k\},$$

with

$$\|\mathbf{u}\|_k = \sum_{i=0}^{k} \sup_{-\infty < t < \infty} \|\mathbf{u}^{(i)}\|.$$

Let $(\bar{\mathbf{u}}, \bar{\boldsymbol{\lambda}})$ *be a COSS of* (4.1) *with hyperbolic fixed points, and the linear operator*

$$\mathsf{L} : \mathcal{X}^1 \to \mathcal{X}^0, \quad \mathsf{L}\mathbf{u} = \mathbf{u}' - A(t)\mathbf{u}$$

with its adjoint

$$\mathsf{L}^* : \mathcal{X}^1 \to \mathcal{X}^0, \quad \mathsf{L}^*\mathbf{u} = \mathbf{u}' + A^T(t)\mathbf{u}.$$

Then $(\bar{\mathbf{u}}, \bar{\boldsymbol{\lambda}})$ *is nondegenerate if and only if*

(1) $\dim N(\mathsf{L}) = 1$ *and* $\dim N(\mathsf{L}^*) = p$*;*

(2) *the matrix* $E = \int_{-\infty}^{\infty} \Phi^T(t) \mathbf{f}_{\boldsymbol{\lambda}}(\bar{\mathbf{u}}(t), \boldsymbol{\lambda}) dt$ *is nonsingular, where the columns* $\phi_i \in \mathcal{X}^1$ *of* $\Phi = (\phi_1, \cdots, \phi_p)$ *form a basis of* $N(\mathsf{L}^*)$*.*

However, the condition for nondegeneracy does not include a phase condition

$$\Psi(\mathbf{u}, \boldsymbol{\lambda}) = 0,$$

and thus will not define a unique solution. To obtain a well-posed problem, this condition must be included. It can be shown that if the phase condition $\Psi \in C^1(\mathcal{X}^0 \times \mathbf{R}^p, \mathbf{R})$ satisfies

$$\Psi(\bar{\mathbf{u}}, \bar{\boldsymbol{\lambda}}) = 0, \qquad \Psi_{\mathbf{u}}(\bar{\mathbf{u}}, \bar{\boldsymbol{\lambda}})\bar{\mathbf{u}}' \neq 0,$$

$(\bar{\mathbf{u}}, \bar{\boldsymbol{\lambda}})$ is indeed a regular solution for (4.1). In the next section, we will discuss the numerical implementation for the phase conditions.

4.3 Numerical Methods for Computing Connecting Orbits

Numerical methods for computing homoclinic and heteroclinic orbits have been recently discussed in [8, 9, 10, 11, 28, 29, 33, 34, 51, 52, 57]. These methods use various discretizations for boundary value problems for ordinary differential equations on the infinite interval and generally are distinguished from one another by the way in which different types of boundary conditions and phase conditions are introduced. In [9, 10, 11], the domain is truncated to $[T_-, T_+]$ and the boundary value problem is solved by a multiple shooting method on this interval. The boundary conditions are taken by projecting $\mathbf{u}(T_-) - \mathbf{u}_-$ to the unstable manifold $\mathcal{M}_-^U$ at $\mathbf{u}_-$ and $\mathbf{u}(T_+) - \mathbf{u}_+$ to the stable manifold $\mathcal{M}_+^S$ at $\mathbf{u}_+$, and a classical phase condition is used. In AUTO, the homoclinic orbits and the homoclinic cycles are approached by computing the periodic solution branch to a very large period via a pseudo-arclength continuation process. With this software, the periodic boundary conditions and an integral phase condition are used and the differential equations are discretized by a spline collocation method. In [29, 33, 34], the end-points of the BVP are approximated by moving the fixed points slightly along their eigenvector directions and the phase condition is similar to that in [28]. Numerical results are obtained by using AUTO. In [8], projection boundary conditions are used with AUTO for computing heteroclinic solutions of parabolic partial differential equations. In [51, 52], rational spectral methods are applied to compute the connecting orbits, along with the implementation of special types of boundary conditions and a phase condition so that more collocation points are located near a variational region of the connecting orbits (away from the fixed points). In general, most of these methods truncate the infinite domain into a finite interval and rescale it to $[0, 1]$. Because the connecting orbit solution decays exponentially near the fixed points, the solution changes fast in a relatively small region, and the rescaling of the domain usually produces an interior layer (or possibly a boundary layer). These types of "artificial" interior layers or boundary layers may cause great difficulty in the numerical computation.

4.3.1 Periodic Solution Approach

Since homoclinic orbits and homoclinic cycles are the limiting cases of the periodic solutions in many applications, they can be approached by periodic solutions at very large periods. These (large period) solutions can be computed by the continuation method for the periodic solution branch. In AUTO, the method for the continuation is to solve the boundary value problem (2.46)–(2.49). Note, when a periodic solution branch approaches a connecting orbit, the period T tends to infinity. Therefore, there is exactly one free parameter λ in the system, which means that $p = 1$ and $n_- + n_+ = n$. As discussed in section 4.2, this condition always holds for homoclinic orbits. In fact, the relation between the periodic solutions and connecting orbits can be established by the *homoclinic bifurcation theorem* [36].

Theorem 4.3 *Let* $(\bar{\mathbf{u}}, \bar{\lambda})$ *be a nondegenerate homoclinic orbit pair of* (4.1) *with the fixed point* $\bar{\mathbf{u}}_0$, *then there exists a* T_0 *and a branch of* $2T$*-periodic solutions* $(\mathbf{u}_T(t), \lambda_T)$, $-T \leq t \leq T$, $T > T_0$ *of* (4.6) *such that for all* $t \in [-T, T]$,

$$\|\bar{\mathbf{u}}(t) - \mathbf{u}_T(t)\| \leq Ce^{-\alpha T}, \qquad |\bar{\lambda} - \lambda_T| \leq Ce^{-2\alpha T}$$

where $0 < \alpha < \min\{|Re(\mu)| : \mu$ *is the eigenvalue of* $\mathbf{f}_\mathbf{u}(\bar{\mathbf{u}}_0, \bar{\lambda})\}$.

Theorem 4.3 indicate that the periodic solutions exist near a homoclinic orbit and at least one branch of these periodic solutions converge to the orbit exponentially as $T \to \infty$. Unfortunately, in the case of heteroclinic orbits, p may not equal one; therefore, this approach may not be appropriate. Another disadvantage of this approach is when T becomes very large, $\mathbf{u}'(t)$ varies relatively fast (O(T)) only in a very small part of the domain $[0, 1]$. On the rest of the domain, $\mathbf{u}'(t) \approx 0$. In fact, it is similar to a singular perturbation problem where an interior layer or boundary layer can cause great numerical difficulty.

With the periodic solution approach, one can carry out numerical continuation for the solution branch of COSS. To do this, we fix the period T at a large value and free another parameter for the two-parameter continuation of the solution branches. Again, this method is not suitable for many applications such as the continuation of a heteroclinic orbit branch.

Recently, Moore [57] considered an arclength formulation for computing the periodic solutions. If L denotes the total arclength of the periodic solution $\mathbf{u}$ in phase space, then

$$L = T \int_0^1 \|\dot{\mathbf{u}}(t)\| dt. \tag{4.9}$$

Defining the arclength variable

$$\sigma(t) := T \int_0^t \|\dot{\mathbf{u}}(\tau)\| d\tau, \tag{4.10}$$

the function

$$\mathbf{v}(\sigma) := \mathbf{u}(t(\sigma)) \tag{4.11}$$

satisfies the following system:

$$\mathbf{v}'(\sigma) = L\frac{\mathbf{f}(\mathbf{v}(\sigma), \lambda)}{\|\mathbf{f}(\mathbf{v}(\sigma), \lambda)\|}, \qquad 0 \le \sigma \le 1 \tag{4.12}$$

$$\mathbf{v}(0) = \mathbf{v}(1) \tag{4.13}$$

$$\int_0^1 \mathbf{v}^T(\sigma)\mathbf{v}_0'(\sigma)d\sigma = 0 \tag{4.14}$$

$$\theta_{\mathbf{v}}^2 \int_0^1 (\mathbf{v}(\sigma) - \mathbf{v}_0(\sigma))^T \dot{\mathbf{v}}_0(\sigma)d\sigma + \theta_L^2(L - L_0)\dot{L}_0 + \theta_\lambda^2(\lambda - \lambda_0)\dot{\lambda}_0 = \delta s. \tag{4.15}$$

The BVP system (4.12)–(4.15) is similar to (2.46)–(2.49) and can be solved in the same manner. However, in the formulation, L is bounded as the period T goes to infinity. This method can be very efficient for computing the periodic solution branch with large period T. But it may not be suitable to compute the connecting orbits. One reason is that when the periodic solution approaches the connecting orbits, $\|\mathbf{f}(\mathbf{v}, \lambda)\|$ approaches zero for some values of $\sigma = \sigma_0$ and the right hand side of (4.12) becomes undefined. At these singular points, the solution $\mathbf{v}(\sigma)$ is continuous, but its first derivative is not. Therefore, numerical difficulty is expected.

4.3.2 Eigenvector Approach

In [29, 33, 34], Doedel and Friedman consider the following system:

$$\mathbf{u}'(t) - T\mathbf{f}(\mathbf{u}(t), \boldsymbol{\lambda}) = \mathbf{0}, \qquad 0 \le t \le 1 \tag{4.16}$$

$$\mathbf{f}(\mathbf{w}_0, \boldsymbol{\lambda}) = \mathbf{0}, \qquad \mathbf{f}(\mathbf{w}_1, \lambda) = \mathbf{0} \tag{4.17}$$

$$\mathbf{f_u}(\mathbf{w}_0, \boldsymbol{\lambda})\mathbf{v}_{0i} = \mu_{0i}\mathbf{v}_{0i}, \quad \|\mathbf{v}_{0i}\| = 1, \qquad i = 1, \ldots, n_0 \tag{4.18}$$

$$\mathbf{f_u}(\mathbf{w}_1, \boldsymbol{\lambda})\mathbf{v}_{1i} = \mu_{1i}\mathbf{v}_{1i}, \quad \|\mathbf{v}_{1i}\| = 1, \qquad i = 1, \ldots, n_1 \tag{4.19}$$

$$\int_0^1 [\mathbf{f}(\mathbf{u}, \boldsymbol{\lambda}) - \mathbf{f}(\mathbf{u}_0, \boldsymbol{\lambda}_0)]^T \mathbf{f_u}(\mathbf{u}, \boldsymbol{\lambda})\mathbf{f}(\mathbf{u}, \boldsymbol{\lambda})dt = 0 \tag{4.20}$$

$$\mathbf{u}(0) = \mathbf{w}_0 + \epsilon_0 \sum_{i=1}^{n_0} c_{0i}\mathbf{v}_{0i}, \quad \sum_{i=1}^{n_0} c_{0i}^2 = 1 \tag{4.21}$$

$$\mathbf{u}(1) = \mathbf{w}_1 + \epsilon_1 \sum_{i=1}^{n_1} c_{1i}\mathbf{v}_{1i}, \quad \sum_{i=1}^{n_1} c_{1i}^2 = 1. \tag{4.22}$$

With this approach, the independent variable t has been scaled to the interval $[0, 1]$ and the period T is treated as a free parameter. Two fixed points $\mathbf{w}_0, \mathbf{w}_1 \in \mathbf{R}^n$ are defined by (4.17). In (4.18) and (4.19), we assume that the Jacobian at $\mathbf{w}_0$ has n_0 distinct positive eigenvalues μ_{0i} with corresponding eigenvectors $\mathbf{v}_{0i}$ and the Jacobian at $\mathbf{w}_1$ has n_1 distinct negative eigenvalues μ_{1i} with corresponding eigenvectors $\mathbf{v}_{1i}$. The boundary conditions (4.21) and (4.22) are used to require that the end-point $\mathbf{u}(0)$ lies in the unstable tangent manifold of the fixed point $\mathbf{w}_0$ at the distance ϵ_0 away and that $\mathbf{u}(1)$ lies in the stable tangent manifold of the fixed point $\mathbf{w}_1$ at the distance ϵ_1 away. The phase condition (4.20) is obtained by minimizing the functional

$$\Phi(\sigma) := \int_0^1 \|\tilde{\mathbf{u}}'(t+\sigma) - \mathbf{u}_0'(t)\|_2^2 dt. \tag{4.23}$$

against the phase shift σ ($\Phi'(\sigma) = 0$).

In this formulation, there are $p + (n + 2)(n_0 + n_1) + 2$ free parameters, n differential equations and $2n + (n + 1)(n_0 + n_1) + 3$ constraints. Therefore, $p = n + 1 - (n_0 + n_1)$ is required to satisfy the requirements for the existence of connection orbits. If continuation of such solutions is needed, the pseudo-arclength continuation equation (2.49) can be used where $p = n + 2 - (n_0 + n_1)$.

4.3.3 Projection Approach

In [9, 10, 11], Beyn develops a numerical method for computing the connecting orbits based upon projecting the solution at the end points to the stable and unstable manifolds of fixed points. Choosing appropriate $J = [T_-, T_+]$ with $T_- < 0$ and $T_+ > 0$, he solves the differential equation

$$\mathbf{u}'(t) = \mathbf{f}(\mathbf{u}(t), \boldsymbol{\lambda}), \qquad t \in J \tag{4.24}$$

subject to the boundary conditions

$$B_-(\mathbf{u}(T_-), \boldsymbol{\lambda}) = 0, \quad B_+(\mathbf{u}(T_+), \boldsymbol{\lambda}) = 0, \tag{4.25}$$

and the phase condition

$$\Psi_J(\mathbf{u}, \boldsymbol{\lambda}) = 0, \tag{4.26}$$

where $\mathbf{f} \in C^2(\mathbf{R}^{n+p}, \mathbf{R}^n)$, $B_- \in C^1(\mathbf{R}^{n+p}, \mathbf{R}^{n-n_-})$, $B_+ \in C^1(\mathbf{R}^{n+p}, \mathbf{R}^{n-n_+})$, and $\Psi_J \in C^1(C^0(J, \mathbf{R}^n) \times \mathbf{R}^p, \mathbf{R})$. The BVP system (4.24)–(4.26) is solved by a multiple shooting method.

The implementation of the boundary conditions (4.25) is the *projection boundary conditions*

$$\mathbf{u}(T_-) - \mathbf{u}_- \in \mathcal{M}_-^U(\boldsymbol{\lambda}), \quad \mathbf{u}(T_+) - \mathbf{u}_+ \in \mathcal{M}_+^S(\boldsymbol{\lambda}), \tag{4.27}$$

i.e., to project the vector $\mathbf{u}(T_-) - \mathbf{u}_-$ and $\mathbf{u}(T_+) - \mathbf{u}_+$ to the unstable subspace $\mathcal{M}_-^U(\boldsymbol{\lambda})$ of $\mathbf{u}_-$ and stable subspace $\mathcal{M}_+^S(\boldsymbol{\lambda})$ of $\mathbf{u}_+$, respectively. However, in the numerical calculation, the projection conditions are presented in equation form

$$B_-(\mathbf{u}(T_-), \boldsymbol{\lambda}) := P_-(\boldsymbol{\lambda})[\mathbf{u}(T_-) - \mathbf{u}_-], \quad B_+(\mathbf{u}(T_+), \boldsymbol{\lambda}) := P_+(\boldsymbol{\lambda})[\mathbf{u}(T_+) - \mathbf{u}_+] \tag{4.28}$$

where the rows of matrices $P_-(\boldsymbol{\lambda}) \in \mathbf{R}^{(n-n_-)\times n}$ and $P_+(\boldsymbol{\lambda}) \in \mathbf{R}^{(n-n_+)\times n}$ span the stable subspace of $\mathbf{f}_\mathbf{u}^T(\mathbf{u}_-, \boldsymbol{\lambda})$ and the unstable subspace of $\mathbf{f}_\mathbf{u}^T(\mathbf{u}_+, \boldsymbol{\lambda})$, respectively. Note, (4.28) is equivalent to (4.27) when $\mathbf{u}_\pm$ are hyperbolic fixed points.

The standard choice for Ψ_J is the integral phase condition

$$\Psi_J(\mathbf{u}, \lambda) := \int_{T_-}^{T_+} [\mathbf{u}(t) - \mathbf{u}_0(t)]^T \mathbf{u}_0'(t) dt, \tag{4.29}$$

or the classical phase condition

$$\Psi_J(\mathbf{u}, \lambda) := [\mathbf{u}(0) - \mathbf{u}_0(0)]^T \mathbf{u}_0'(0) \tag{4.30}$$

introduced in chapter 2, where $\mathbf{u}_0$ is a nearby known solution. In Beyn's numerical computation, the classical phase condition is used.

The system (4.24)–(4.26) consist of n ODEs, $2n - (n_- + n_+)$ boundary conditions and one phase condition; therefore, the number of free parameters is $p = n + 1 - (n_- + n_+)$. The stability and convergence results are summarized in the following theorem.

Theorem 4.4 *Let $(\bar{\mathbf{u}}, \bar{\boldsymbol{\lambda}})$ be a nondegenerate connecting orbit between stationary solutions of (4.24). Assume that $\mathbf{f} \in C^2(\mathbf{R}^{n+p}, \mathbf{R}^n)$, $B_- \in C^1(\mathbf{R}^{n+p}, \mathbf{R}^{n-n_-})$, $B_+ \in C^1(\mathbf{R}^{n+p}, \mathbf{R}^{n-n_+})$ and $\Psi_J \in C^1(C^0(J, \mathbf{R}^n) \times \mathbf{R}^p, \mathbf{R})$, and the following conditions hold:*

A1: *$B_\pm(\bar{\mathbf{u}}_\pm, \bar{\boldsymbol{\lambda}}) = \mathbf{0}$, and $B_{-\mathbf{u}}(\bar{\mathbf{u}}_-, \bar{\boldsymbol{\lambda}}) E_-^S(\bar{\boldsymbol{\lambda}})$ and $B_{+\mathbf{u}}(\bar{\mathbf{u}}_+, \bar{\boldsymbol{\lambda}}) E_+^U(\bar{\boldsymbol{\lambda}})$ are nonsingular, where the columns of the matrices $E_-^S(\bar{\boldsymbol{\lambda}}) \in \mathbf{R}^{n\times(n-n_-)}$ and $E_+^U(\bar{\boldsymbol{\lambda}}) \in \mathbf{R}^{n\times(n-n_+)}$ form a basis of the stable subspace of $\mathbf{f}_\mathbf{u}(\bar{\mathbf{u}}_-, \bar{\boldsymbol{\lambda}})$ and the unstable subspace of $\mathbf{f}_\mathbf{u}(\bar{\mathbf{u}}_+, \bar{\boldsymbol{\lambda}})$, respectively;*

A2: $\Psi_J(\bar{\mathbf{u}}_J, \bar{\boldsymbol{\lambda}}_J) \to 0$ *where* $J \to \mathbf{R}$, $|\Psi_{J,\mathbf{u}}(\bar{\mathbf{u}}_J, \bar{\boldsymbol{\lambda}})\bar{\mathbf{u}}'_J| \geq c > 0$, *$c$ is constant, J is sufficiently large and* Ψ'_J *is uniformly bounded on*

$$K_\rho := \{(\mathbf{u}, \boldsymbol{\lambda}) : \|\mathbf{u} - \bar{\mathbf{u}}_J\|_1 + \|\boldsymbol{\lambda} - \bar{\boldsymbol{\lambda}}\| \leq \rho\}$$

for some $\rho > 0$.

Then the boundary value problem (4.24)–(4.26) *has a unique solution* $(\mathbf{u}_J, \boldsymbol{\lambda}_J)$ *in the tube* K_ρ. *Moreover, there is phase shift* $\tau_J \to 0$ *as* $J \to R$ *such that* $\bar{\mathbf{w}}(t) = \bar{\mathbf{u}}(t + \tau_J)$ *and*

$$\|\bar{\mathbf{w}}_J - \mathbf{u}_J\| \leq C(e^{d\beta_- T_-} + e^{-d\alpha_+ T_+}), \qquad \|\bar{\boldsymbol{\lambda}}_J - \boldsymbol{\lambda}_J\| \leq C(e^{\omega_- T_-} + e^{-\omega_+ T_+})$$

where

$$\alpha_\pm < \{\min |Re(\mu)| : \mu \textit{ is an eigenvalue of } \mathbf{f}_{\mathbf{u}}(\bar{\mathbf{u}}_\pm, \bar{\boldsymbol{\lambda}}) \textit{ with negative real part}\},$$

$$\beta_\pm < \{\min |Re(\mu)| : \mu \textit{ is an eigenvalue of } \mathbf{f}_{\mathbf{u}}(\bar{\mathbf{u}}_\pm, \bar{\boldsymbol{\lambda}}) \textit{ with positive real part}\},$$

$$\omega_- = \min(2d\beta_-, d\beta_- + \alpha_-), \quad \omega_+ = \min(2d\alpha_+, d\alpha_+ + \beta_+)$$

where $d = 1$ *or* 2.

Remarks:

(1) Using the projection boundary conditions (4.28) and either the phase condition (4.29) or (4.30), $d = 2$.

(2) Using the eigenvector boundary conditions (4.21) and (4.22) with the phase condition (4.20), $d = 2$.

(3) Using the periodic boundary conditions with the phase condition chosen by either (4.29) or (4.30), $d = 1$.

In [64], Schecter shows that the above estimate can be improved to

$$\|\bar{\boldsymbol{\lambda}}_J - \boldsymbol{\lambda}_J\| \leq C(e^{(\alpha_- + 2\beta_-)T_-} + e^{-(2\alpha_+ + \beta_+)T_+})$$

if the projection boundary conditions and appropriate phase condition is used. Beyn's numerical results also support this argument. In general, the projection approach cannot handle non-hyperbolic fixed points. Schecter [63, 64] shows that under certain conditions and with slight modification of the projection boundary conditions, this method can handle certain type of connecting orbits with semihyperbolic fixed points (quadratic turning points with $\beta_- = 0$). The rate of convergence is

$$\|\bar{\mathbf{w}}_J - \mathbf{u}_J\| \leq C(\frac{1}{|T_-|^r} + e^{2\alpha_- T_-}), \qquad \|\bar{\boldsymbol{\lambda}}_J - \boldsymbol{\lambda}_J\| \leq C(e^{\alpha_- T_-} + e^{-(2\alpha_+ + \beta_+)T_+}).$$

Unfortunately, the commonly used phase conditions may not satisfy the assumptions given by Schecter.

All three methods discussed in this chapter have a similar property. They all converge to the connecting orbits exponentially with respect to the truncation of the interval $(-\infty, \infty)$. The periodic solution approach may not be suitable for some problems, such as heteroclinic orbits and the rate of convergence is slower than the other two methods. For both eigenvector and projection approaches, the reduction of a BVP from an infinite domain to a finite domain is similar to [45], and the boundary conditions are the linear approximation of the stable and unstable subspace of the fixed points. These two methods have a close relation. In fact, $\mathbf{v}_{0i}$, $1 \leq i \leq n_0$, spans $\mathcal{M}_-^U(\boldsymbol{\lambda})$ and $\mathbf{v}_{1i}$, $1 \leq i \leq n_1$, spans $\mathcal{M}_+^S(\boldsymbol{\lambda})$, respectively. Therefore, (4.28) is equivalent to (4.21) and (4.22) when $\mathbf{u}_\pm$ are hyperbolic. However, because of the different implementation, both have their own advantages and disadvantages. The eigenvector approach can handle some cases with nonhyperbolic fixed points, and the phase condition in this method is more reliable. However, in this formulation, the eigenvalues and eigenvectors of $\mathbf{f}_\mathbf{u}(\bar{\mathbf{u}}_\pm, \bar{\boldsymbol{\lambda}})$ are introduced as new parameters, so it increases the dimension of the original problem and also needs certain knowledge of the eigen-structures at the fixed points. The projection approach can handle more general cases where one does not have the information about the eigen-structure. However, the projections for certain invariant subspaces is not unique; therefore, a smooth QR factorization is necessary for the convergence of the Newton iterations.

4.4 Arclength Parameterization

All three numerical methods for computing the connecting orbits discussed in the last section involve the truncation of the domain for the independent variable t from $(-\infty, \infty)$ to a finite interval J. In general, the eigenvector approach and the projection approach have better convergence properties than the periodic solution approach. When using a boundary value ODE solver, the domain $J = [T_-, T_+]$ is usually rescaled to $[0,1]$. If $T = T_+ - T_-$ is large, artificial interior layers or boundary layers often result since $\mathbf{u}'(t)$ is very large in a small region, then all three methods can suffer poor numerical stability. However, in the phase space the solution itself often remains bounded and smooth except at the fixed points. This suggests that a geometrical formulation may lead to a superior method. In [24], a geometrical method for computing invariant tori is developed and demonstrates the success of the geometrical approach. Moore's [57] recent geometrical approximation based upon an arclength parametrization for computing periodic orbits introduced in the previous section can be modified for computing connecting orbits.

4.4.1 Formulation

Let L be the total arclength of the solution $\mathbf{u}(t)$ in phase space and σ be the scaled arclength parametrization variable. Following (4.9)–(4.12), we obtain

$$L = \int_{-\infty}^{\infty} \|\mathbf{u}'(t)\| dt, \tag{4.31}$$

and

$$\sigma(t) = \frac{1}{L} \int_{-\infty}^{t} \|\mathbf{u}'(\tau)\| d\tau. \tag{4.32}$$

So

$$\mathbf{v}(\sigma) = \mathbf{u}(t(\sigma)), \tag{4.33}$$

satisfies the arclength parametrized differential equation

$$\mathbf{v}'(\sigma) = L\frac{\mathbf{f}(\mathbf{v}(\sigma), \boldsymbol{\lambda})}{\|\mathbf{f}(\mathbf{v}(\sigma), \boldsymbol{\lambda})\|}, \qquad 0 < \sigma < 1. \tag{4.34}$$

The equation (4.34) has singular points at the end points, and away from the end points $\|\mathbf{v}'(\sigma)\| = L$. We assume that the connecting orbit is not of Šilnikov type, i.e., all eigenvalues of $\mathbf{f}_{\mathbf{u}}(\bar{\mathbf{u}}_\pm, \bar{\boldsymbol{\lambda}})$ are real. In this case,

$$\lim_{\sigma \to 0} \mathbf{v}'(\sigma) \quad \text{and} \quad \lim_{\sigma \to 1} \mathbf{v}'(\sigma)$$

exist and they lie in the unstable and stable subspaces of $\mathbf{u}_-$, $\mathbf{u}_+$. To satisfy continuity conditions, we require

$$\|\mathbf{v}'(0)\| = \|\mathbf{v}'(1)\| = L. \tag{4.35}$$

Also, the boundary conditions

$$\mathbf{v}(0) = \lim_{t\to-\infty} \mathbf{u}(t) = \mathbf{u}_-, \qquad \mathbf{v}(1) = \lim_{t\to\infty} \mathbf{u}(t) = \mathbf{u}_+ \tag{4.36}$$

ensure that the fixed points $\mathbf{u}_-$ and $\mathbf{u}_+$ are the end point values for the solution. Note that a phase condition is not needed as we use the fixed points as the boundary points and L is bounded. The projection boundary conditions (4.25) become

$$P_-(\boldsymbol{\lambda})\mathbf{v}'(0) = 0, \qquad P_+(\boldsymbol{\lambda})\mathbf{v}'(1) = 0. \tag{4.37}$$

where the projection matrices $P_-(\boldsymbol{\lambda}) \in \mathbf{R}^{(n-n_-)\times n}$ and $P_+(\boldsymbol{\lambda}) \in \mathbf{R}^{(n-n_+)\times n}$ are defined as in section 4.3.3.

The boundary value problem (4.34)–(4.37) consists of n ODEs, $4n+2-(n_- + n_+)$ algebraic constraints, and $p+1$ parameters where $p = n+1-(n_- + n_+)$. Thus, we have $2n$ extra conditions, and a standard boundary value solver is not suitable to use without some modification.

In particular, we are interested in computing the connecting orbit branches; therefore, the continuation methods should be considered. It turns out that the psuedo-arclength continuation equation

$$\theta_{\mathbf{v}}^2 \int_0^1 (\mathbf{v}(\sigma) - \mathbf{v}_0(\sigma))^T \dot{\mathbf{v}}_0(\sigma) d\sigma + \theta_L^2 (L - L_0)\dot{L}_0 + \theta_{\boldsymbol{\lambda}}^2 (\boldsymbol{\lambda} - \boldsymbol{\lambda}_0)^T \dot{\boldsymbol{\lambda}}_0 = \delta s \tag{4.38}$$

can be easily implemented into the system (4.34)–(4.37). Therefore, the number of free parameters in our system should be $p = n+2-(n_- + n_+)$.

4.4.2 Numerical Discretization

We now consider the numerical discretization for the boundary value problem given by (4.34)–(4.38). This BVP system is a singular boundary value problem as the right hand side of (4.34) is undefined when $\sigma = 0$ and 1. The standard collocation methods cannot handle such problems. Therefore a special implementation of the collocation method is necessary.

In [57], Moore recommends solving the boundary value problem (4.34)–(4.37) by a Gauss-Lobatto collocation method, which is as follows:

Gauss-Lobatto Collocation Algorithm

- Choose C^1 piecewise m^{th} degree polynomial and a mesh $0 = \sigma_0 < \sigma_1 < \ldots < \sigma_N = 1$;
- Collocate (4.34) at all Gauss-Lobatto points apart from $\sigma = 0, 1$;
- $\mathbf{v}(0) = \mathbf{u}_-(\boldsymbol{\lambda}), \quad \mathbf{v}(1) = \mathbf{u}_+(\boldsymbol{\lambda})$;
- $P_-(\boldsymbol{\lambda})\mathbf{v}'(0) = \mathbf{0}, \quad P_+(\boldsymbol{\lambda})\mathbf{v}'(1) = \mathbf{0}$;
- $\|\mathbf{v}'(0)\| = \|\mathbf{v}'(1)\| = L$.

One of our goals is to adapt an efficient algorithm for the software AUTO. Gauss-Lobatto collocation involves different collocation points and continuity conditions than that used in AUTO. It is not easy to implement with the given basis function (Lagrange polynomials).

Our numerical implementation of the arclength formulation AUTO continues to use Lagrange polynomials as the basis functions and Gauss-Legendre collocation points in the interior subintervals but uses the Gauss-Radau collocation points on the first and last subintervals. The resulting numerical algorithm is as follows:

Gauss-Radau Collocation Algorithm

- Collocate (4.34) at all Gauss-Legendre points apart from the first and the last mesh intervals.
- Collocate (4.34) at the left Gauss-Radau points apart from $\sigma = 0$ in the first mesh interval;
- Collocate (4.34) at the right Gauss-Radau points apart from $\sigma = 1$ in the last mesh interval;
- $\mathbf{f}(\mathbf{v}(0), \boldsymbol{\lambda}) = \mathbf{0}, \quad \mathbf{f}(\mathbf{v}(1), \boldsymbol{\lambda}) = \mathbf{0}$;
- $P_-(\boldsymbol{\lambda})\mathbf{v}'(0) = \mathbf{0}, \quad P_+(\boldsymbol{\lambda})\mathbf{v}'(1) = \mathbf{0}$;
- $\|\mathbf{v}'(0)\| = \|\mathbf{v}'(1)\| = L$.

4.4.3 Software Implementation

In the Gauss-Radau Algorithm, there are only $(m-1)n$ collocation equations in the first and the last mesh interval, so n more equations are needed to replace the collocation equations at $\sigma = 0$ in the first subinterval and at $\sigma = 1$ in the last subinterval. In our implementation in AUTO, the equation

$$\mathbf{f}(\mathbf{v}(0), \boldsymbol{\lambda}) = 0 \tag{4.39}$$

is used in the first subinterval, and

$$\mathbf{f}(\mathbf{v}(1), \boldsymbol{\lambda}) = 0 \tag{4.40}$$

is used in the last. Other equations are treated as the boundary conditions, and they are placed in the bottom rows of the collocation matrix. Here $\mathbf{v}'(0)$ and $\mathbf{v}'(1)$ are approximated by an m^{th} order Langrage interpolation. The projection matrices $P_-(\boldsymbol{\lambda})$ and $P_+(\boldsymbol{\lambda})$ are obtained by using the LAPACK [1] subroutine DGEES. Note, when using the QR algorithm, these projection matrices are not unique and should be obtained smoothly. The final structure of the linear system is shown in Figure 4.9 for 2 differential equations, 4 mesh intervals, 3 collocation points and 3 free parameters.

Recall from the previous chapter that the numerical method for solving the linear system involves first a local condensation method to eliminate the unknowns at non-mesh points. This condensation does not use row pivoting and hence, would fail at the first block. The remedy is rather trivial: use row partial pivoting or place the equation (4.39) in the bottom rows of the first block. In order to maintain efficiency for the linear system solver, we implement the latter. The coefficient matrix profile is shown in Figure 4.10. The condensed linear system has a block bi-diagonal structure, as discussed earlier, and blocks C_1 and A_N in (3.26) are zero. After applying either a QR or LU algorithm, the coefficient matrix for the linear system has the structure

$$\begin{bmatrix} 0 & R_1 & F_1 & & & & & H_1 \\ 0 & & R_2 & F_2 & & & & H_2 \\ \vdots & & & \ddots & \ddots & & & \vdots \\ G_{N-1} & & & & 0 & & F_{N-1} & H_{N-1} \\ 0 & & & & & & \tilde{C}_N & \tilde{D}_N \\ B_0 & B_1 & \cdots & \cdots & & B_{N-1} & B_N & E \end{bmatrix}. \tag{4.41}$$

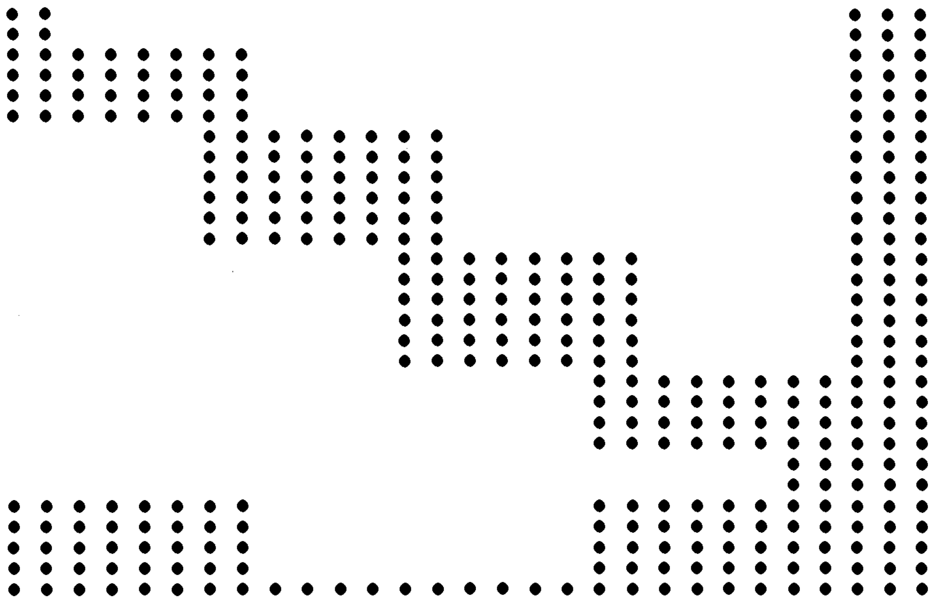


Figure 4.9: Structure of the linear system profile

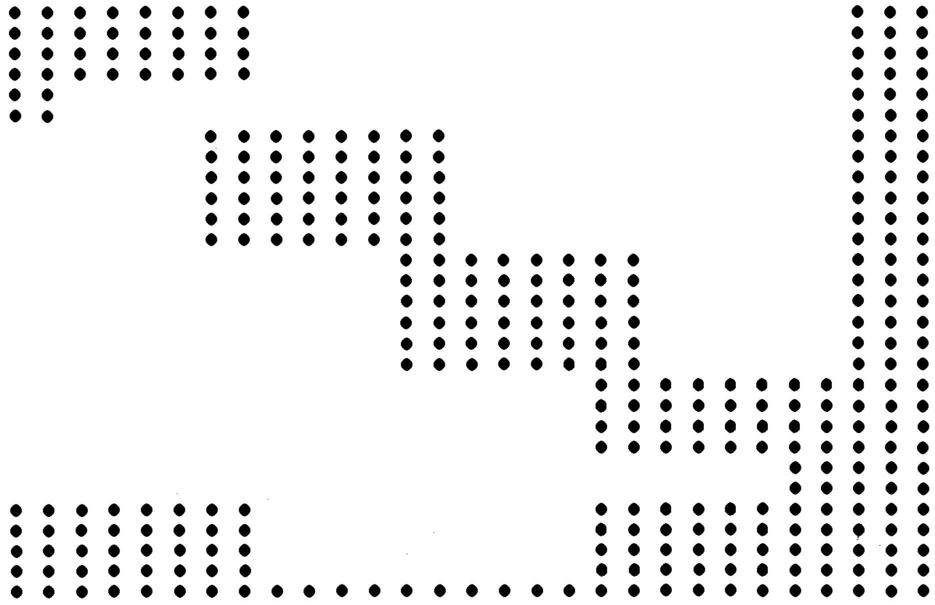


Figure 4.10: Structure of the linear system profile after reordering the equations

If not further modified, the linear system solver of AUTO would fail when eliminating the block B_{N-1}. However, it is possible to reorder the equations and unknowns in the linear system so that the bi-diagonal structure is kept and the QR or LU algorithm is able to solve the reordered system. In our modification, we reorder the unknowns in such a way that the linear system has the form

$$\begin{bmatrix} A_2 & C_2 & & & & & D_2 \\ & A_3 & C_3 & & & & D_3 \\ & & \ddots & \ddots & & & \vdots \\ & & & 0 & C_N & & D_N \\ & & & & 0 & A_1 & D_1 \\ B_1 & B_2 & \cdots & \cdots & B_N & B_0 & E \end{bmatrix}. \tag{4.42}$$

Because $\mathbf{v}(0)$ and $\mathbf{v}(1)$ are hyperbolic fixed points, the matrices A_1 and C_N are non-singular, and the linear system can be solved by the QR or LU factorization as before. The reduced linear system is

$$\begin{bmatrix} G_2 & R_2 & F_2 & & & & H_2 \\ G_3 & & R_3 & F_3 & & & H_3 \\ \vdots & & & \ddots & \ddots & & \vdots \\ 0 & & & & R_N & 0 & H_N \\ 0 & & & & & A_N & D_1 \\ B_1 & B_2 & \cdots & \cdots & B_N & B_0 & E \end{bmatrix}. \tag{4.43}$$

In order to be able to use AUTO with this arclength implementation, we have also modified several parts of the code to allow users to provide discrete solution data as the starting solution, and we implement the arclength formulation for the continuation of the periodic solution branches. It allows users to switch from the "time" formulation to the "arclength" formulation directly. However, several quantities, such as the solution norm, may have different meanings and numerical values. These appear to be minor problems and should be easy to fix. Also, we should mention that our current software implementation is not suitable for solving large dimensional problems. Some major changes for the basic data structure and subroutine inter-dependency structure in AUTO are necessary before this implementation can be realized.

4.4.4 Numerical Examples

To demonstrate the success for the arclength parameterization method, we give several numerical examples. We will simply refer to the arclength parameterization method as the *arclength method*, the eigenvector approach introduced in section 4.3.2 as the *eigenvector method*, implementation of projection boundary conditions discussed in 4.3.3 as the *projection method*, and the approach using the periodic solutions as the *periodic method*. In all of our computation, AUTO is used, with certain necessary modifications. Therefore, the discretization of the ODE system uses a Lagrange basis collocation method. For all cases, we choose 4 collocation points on each mesh subinterval (NCOL=4), which is the default for AUTO. Note, for the arclength method, the collocation points at two end-intervals are different from the middle ones. They are Gauss-Radau points. Our numerical test is performed on a Silicon Graphics Indigo running IRIX 4.0.5F with a MIPS FORTRAN 77 compiler. Ideally, "$-O$" (Optimization) option should be chosen to maximize the efficiency of the code; however, for debugging reason, the option "$-g$" is used in our test. Our numerical examples are the Nagumo equations, the Lorenz equations and Rössler's band.

Example 4.4 *The Nagumo equations*

$$\begin{aligned} u_1' &= u_2 \\ u_2' &= cu_2 - u_1(1-u_1)(u_1-a), \qquad 0 < a < 1. \end{aligned}$$

This system has been used in [10, 29, 52] as a test problem for computing heteroclinic orbits (a travelling wave solution branch). We test this example by the eigenvector method and the arclength method.

For the eigenvector method, we use 10 mesh intervals (NTST=10). As described in [29], we start with the exact solution

$$u_1(t) = \frac{\exp(\frac{\sqrt{2}}{2}t)}{1+\exp(\frac{\sqrt{2}}{2}t)}, \quad u_2(t) = u_1'(t)$$

at $(a,c) = (\frac{1}{2}, 0)$ and $T = 5$. We follows the solution by increasing T to 1000. Fixing T at this point and using a as the primary continuation parameter and c is the secondary continuation parameter, we compute both solution branches for $c = \pm\sqrt{2}(a - \frac{1}{2})$. The eigenvector method fails after only 6 steps. After increasing the number of mesh intervals to NTST=15, we successfully compute both branches. The absolute error for the lower half branch varies from $O(10^{-8})$ to $O(10^{-4})$. As for the arclength method, we successfully compute both branches with only 10 mesh intervals (NTST=10). The absolute error is $O(10^{-8})$ for all solutions (see Figure 4.11).

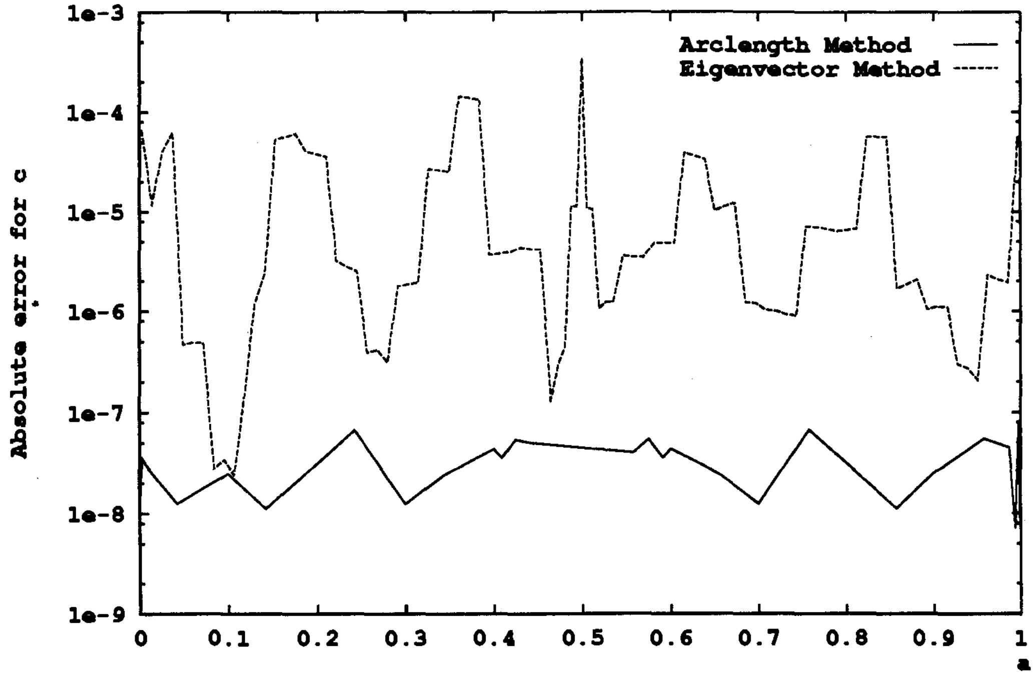


Figure 4.11: Absolute error for the Nagumo equation

In this computation, the tolerance for the Newton iteration is set to 10^{-6}. On Figure 4.12 and Figure 4.13, the distribution of the mesh points for the lower half branch solution is shown for the solution space (a) and the phase space (b). Note, there are 41 points for the arclength method (10 mesh intervals and 4 collocation points per interval) and 61 points for the eigenvector method (15 mesh intervals and 4 collocation points per interval). Clearly, the arclength method efficiently distributes the mesh points in phase space even though it uses less mesh points.

Example 4.5 *The Lorenz equations*

$$\begin{aligned} x' &= \sigma(y-x) \\ y' &= \lambda x - y - xz \\ z' &= xy - \mu z. \end{aligned}$$

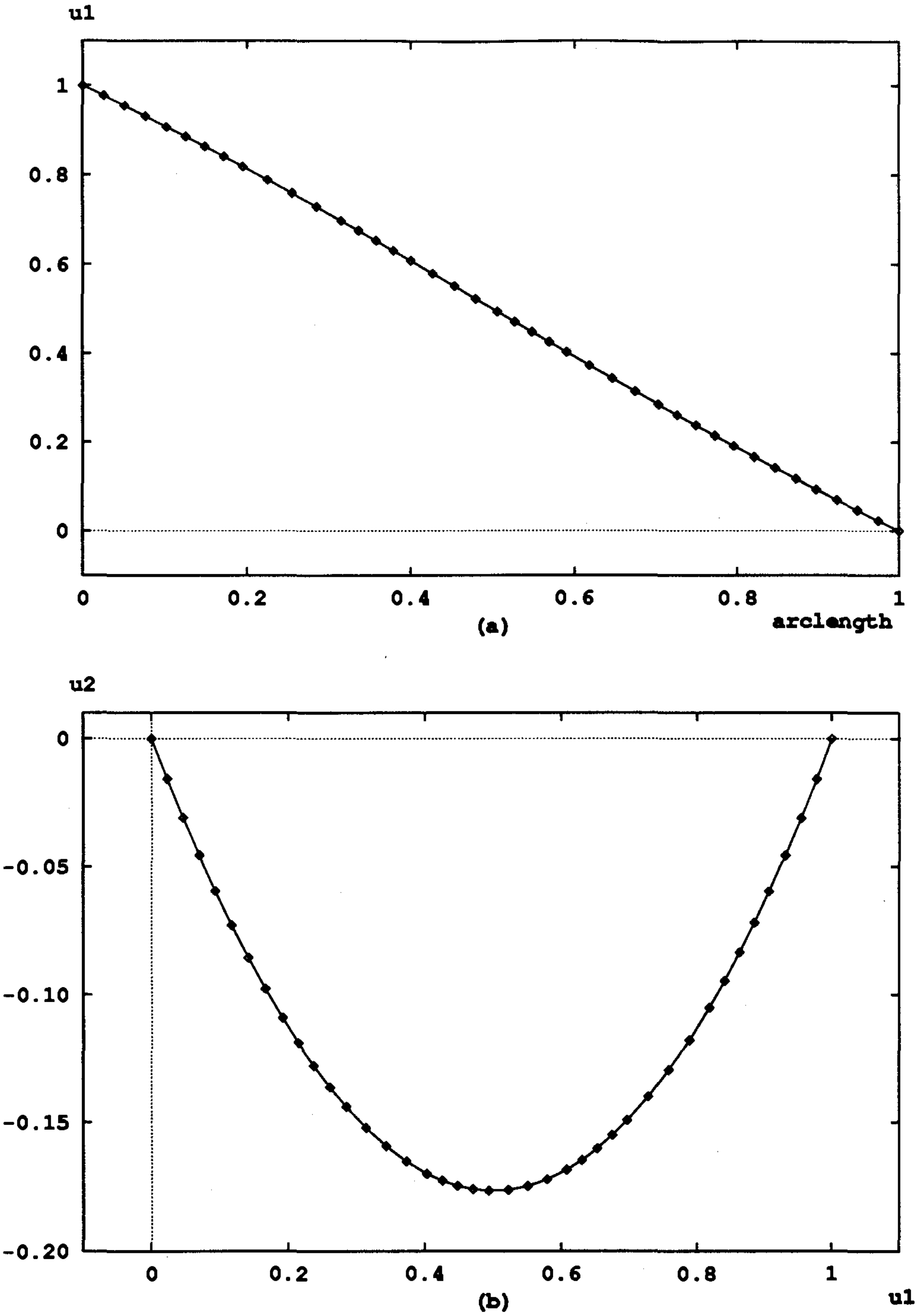


Figure 4.12: Distribution of the mesh points for the arclength method (a) arclength vs u; (b) u vs u' .

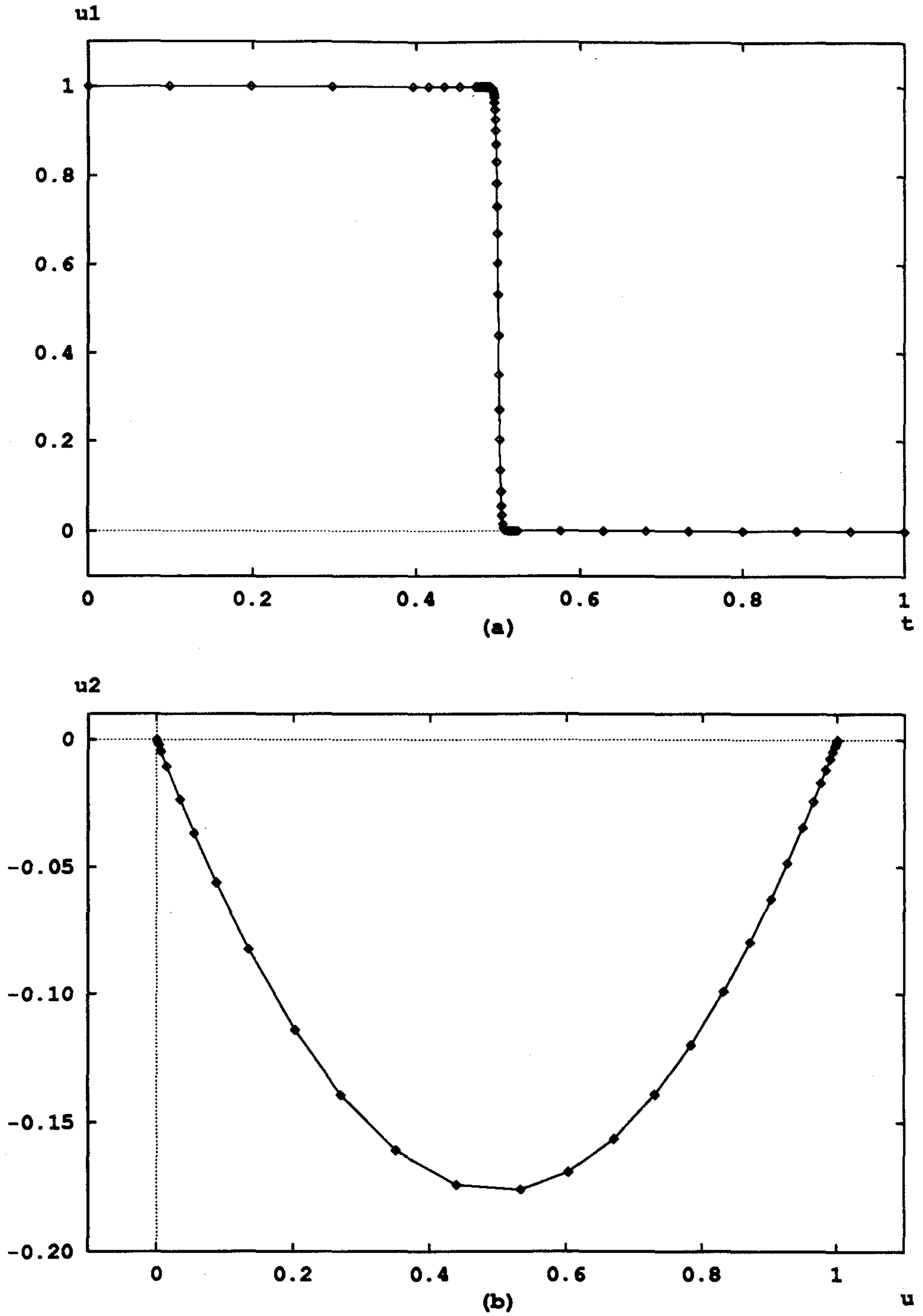


Figure 4.13: Distribution of the mesh points for the eigenvector method (a) t vs u; (b) u vs u'

At the Lorenz value $(\sigma, \lambda, \mu) = (10, 13.92656, \frac{8}{3})$, there exists a homoclinic orbit with $L = 64.89938$. We use AUTO to trace out the periodic solution branch for $(\sigma, \mu) = (10, \frac{8}{3})$ until close enough to this homoclinic orbit. Next we use several methods to continue the homoclinic orbit branch. The tolerance for the Newton iteration is set to 10^{-8}. The primary and secondary continuation parameters are μ and λ. Their relation is shown in Figure 4.14(a), and the solution in phase space is shown in Figure 4.14(b).

For the arclength parameterization method, using only 25 mesh intervals (NTST=25) we successfully continue the homoclinic orbit branch until $\mu = 14.376$. The final parameter value for λ is 95040.54 and the total arclength $L = 592401.13$. When we use NTST=50, we have no trouble in continuing the homoclinic branch to $\mu = 14.455$.

Next, we compute this branch of homoclinic orbit with the projection method and the periodic method. We start both computations from a periodic solution with the fixed period $T = 1000$ and use μ and λ as the primary and the secondary continuation parameters. Using 25 mesh intervals and setting the tolerance to 10^{-8}, we are unable to perform the continuation for both methods. Increasing the tolerance to 10^{-6}, we can continue the solution branch with both methods. The projection method fails at $\mu = 14.328$ and the periodic method fails at $\mu = 13.920$. However, the projection method cannot locate the solution at $\mu = 14$ during this continuation. In order to obtain the solution at $\mu = 14$, we have to further increase the tolerance to 10^{-5}. Both methods now are able to obtain the required solution.

To compare the accuracy for all three methods, we summarize the relative error for the parameter λ at $\mu = 14$ in Table 4.1. From these results, the superconvergence rate is clearly observed for the arclength method and the projection method. However, N=25 is inadequate for both the projection and periodic methods. For the latter, the superconvergence is not clear with these values. It suggests that $T = 1000$ may not be large enough for the periodic method. In this computation, we use the value of λ obtained using 200 mesh intervals for the arclength method and the projection method as the exact solution (they agree in all printed digits). In Figures 4.15–4.17, the distribution of the mesh points for $\mu = 14$ is shown for all three methods with NTST=50.

The above two examples have shown that the arclength parameterization method can be very efficient for computing the connecting orbits. However, there is one restriction in

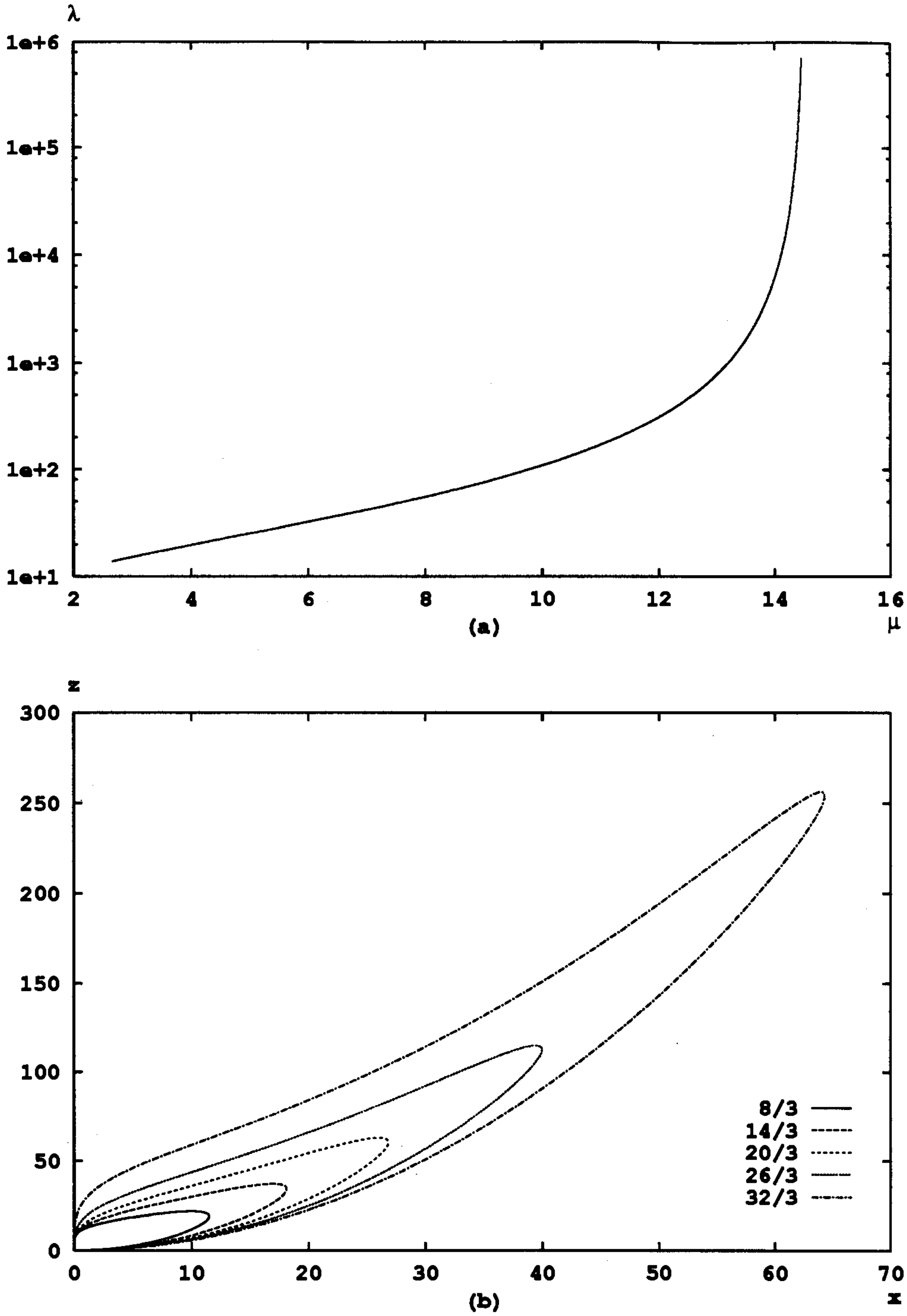


Figure 4.14: Homoclinic orbit branch for the Lorenz equations (a) Solution branch in parameter μ-λ space; (b) Homoclinic orbits in the $x-z$ plane. .

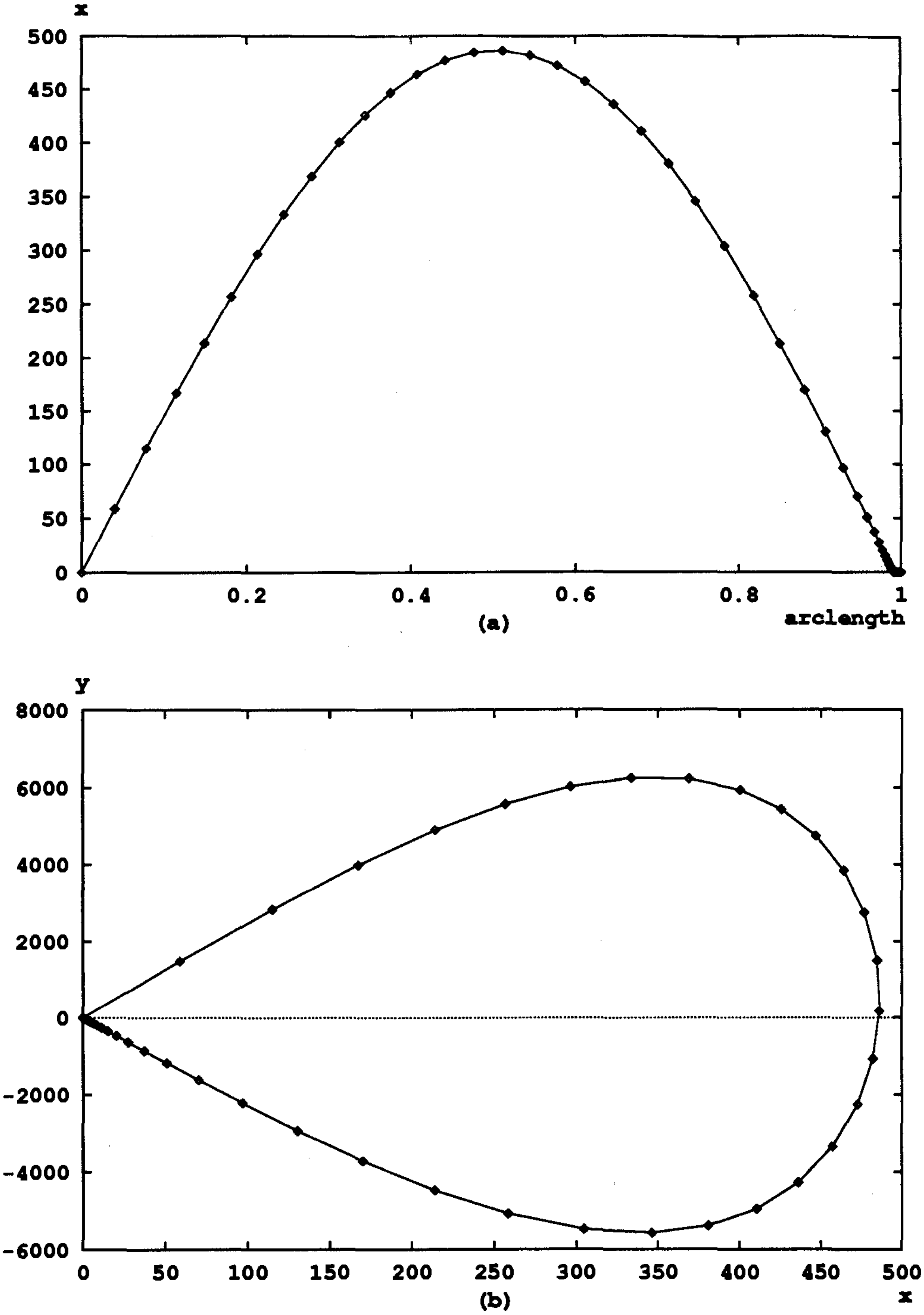


Figure 4.15: Distribution of the mesh points for the arclength method (a) arclength vs x; (b) x vs y

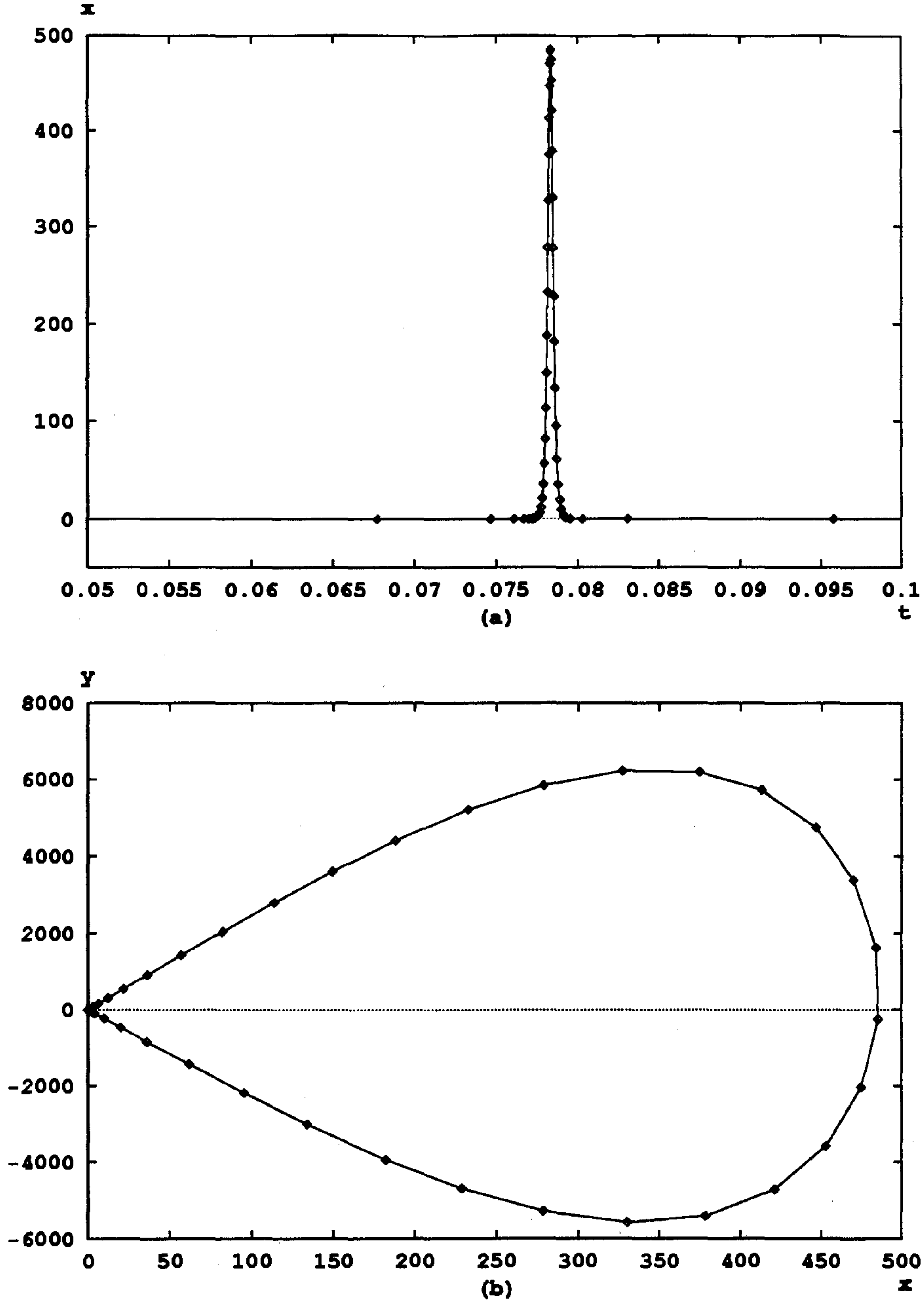


Figure 4.16: Distribution of the mesh points for the projection method (a) t vs x; (b) x vs y

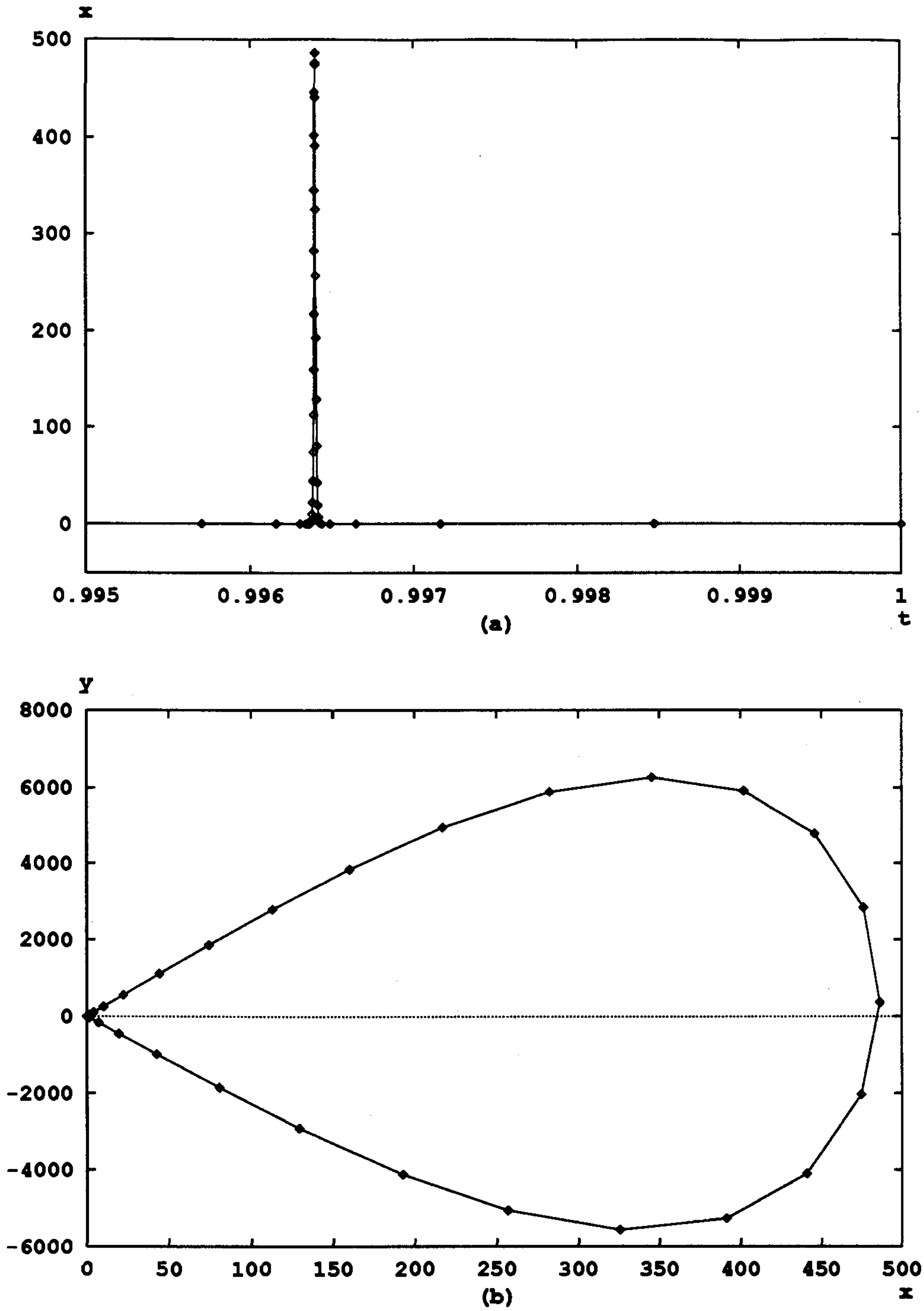


Figure 4.17: Distribution of the mesh points for the periodic method (a) t vs x; (b) x vs y

NTST	Projection method	Periodic method	Arclength method
12	–	–	8.0494 E-05
25	1.4231 E-05	1.6894 E-05	4.5898 E-07
50	8.3057 E-09	1.3579 E-09	6.0919 E-09
100	< 1.0 E-11	1.5836 E-09	8.0795 E-11

Table 4.1: Relative error for λ in the Lorenz equation

this method, i.e., the stationary points are not saddle-focus. In another word, there are no complex eigenvalues in the Jacobian at the fixed points. Indeed, this condition is crucial in both Gauss-Lobatto collocation algorithm and Gauss-Radau collocation algorithm. The next example shows that the Gauss-Radau algorithm would fail without this condition.

Example 4.6 *Rössler's band*

$$\begin{aligned} x' &= -y - z \\ y' &= x + ay \\ z' &= b + z(x - c). \end{aligned}$$

The bifurcation diagram is shown in Figure 4.18(a) for $b = 2$ and $c = 4$. In this figure, the main periodic branch (branch 2) converges to a saddle-focus type homoclinic orbit at $a = 0.6511$. First, we use T as the continuation parameter to compute the periodic solutions on branch 2. Using NTST=50, the periodic method takes 913 steps and fails at $T = 352.0895$; the projection method finishes 500 continuation steps with final $T = 858.8219$. Next, we use the solution at $T = 200$ as the starting point to compute the homoclinic orbit branch. The primary and the secondary continuation parameters are a and b. Both the periodic and projection methods successfully compute the homoclinic orbit branch (branch 3). The periodic method fails at $a = 0.6099257$ while the projection method fails at $a = 0.606665$. However, we are unable to continue this branch with the arclength method. Note that eigenvalues at the fixed points on branch 3 are $\lambda_1 > 0$ and $\lambda_{2,3} = \alpha \pm i\beta$ with $\alpha < 0$. Therefore, $\mathbf{v}'(1)$ does not exist and the projection condition

$$P_+\mathbf{v}'(1) = \mathbf{0}$$

is not appropriate to use in the computation. This problem a major drawback of the arclength method and is being studied.

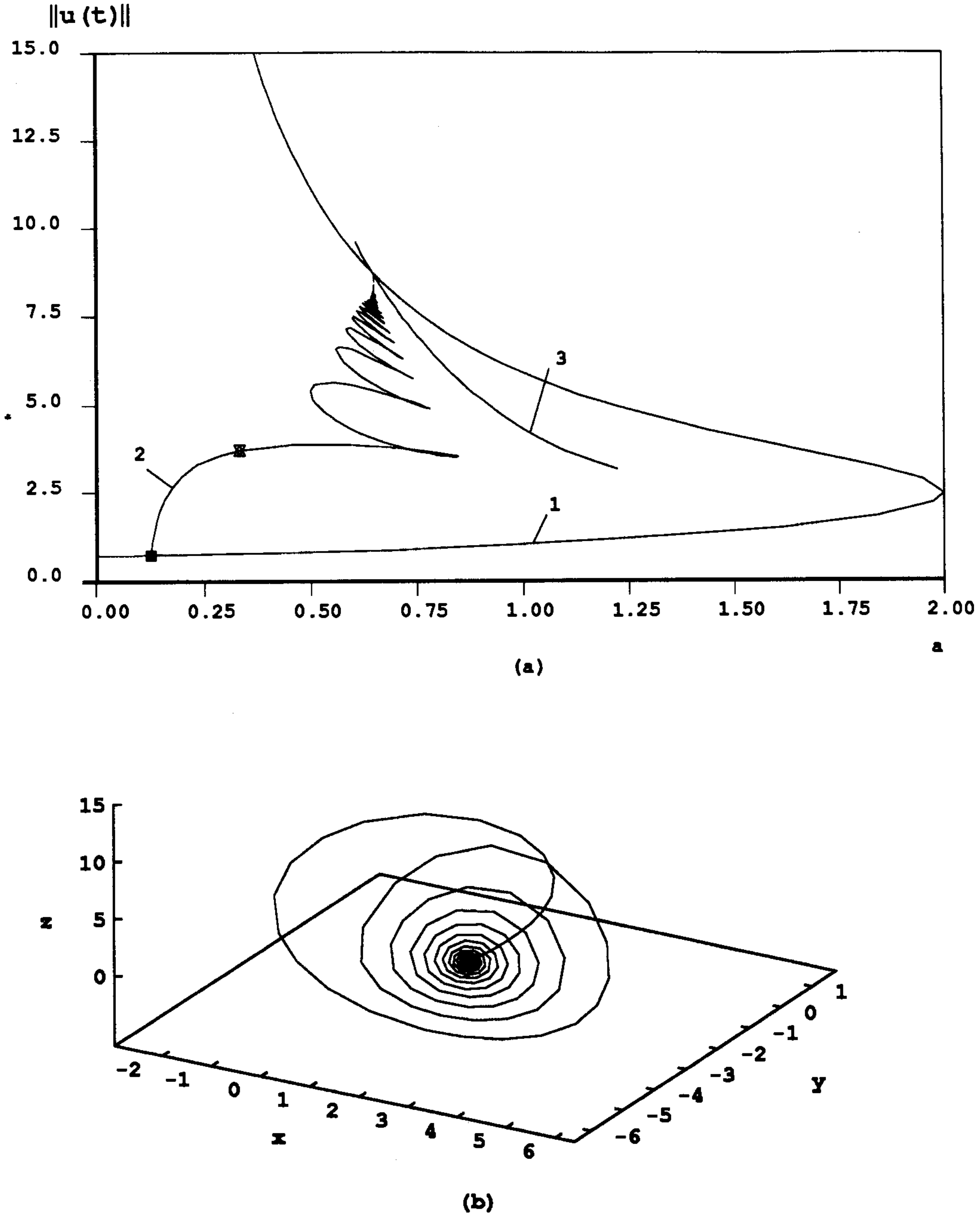


Figure 4.18: Homoclinic orbit branch in Rössler's band
(a) bifurcation diagram (b) homoclinic orbit at $a = 0.6511$

4.4.5 Analysis

Our numerical examples have shown that the arclength formulation can be very efficient for computing the connecting orbits. In this sub-section, we will discuss the convergence and uniqueness of the solutions for the system (4.34)–(4.37). In [57], Moore proves the following theorem:

Theorem 4.5 *Let $\mathcal{X}$ and $\mathcal{Y}$ be two real Banach spaces:*

$$\mathcal{X} := \left\{ x \in C^0[0,1] \cap C^1(0,1) : \sup_{\sigma\in(0,1)} \frac{|x(\sigma)|}{\sigma^{\alpha}(1-\sigma)^{\beta}} < \infty,\ \sup_{\sigma\in(0,1)} \frac{|x'(\sigma)|}{\sigma^{\alpha-1}(1-\sigma)^{\beta-1}} < \infty \right\},$$

$$\|x\| := \max\left\{ \sup_{\sigma\in(0,1)} \frac{|x(\sigma)|}{\sigma^{\alpha}(1-\sigma)^{\beta}},\ \sup_{\sigma\in(0,1)} \frac{|x'(\sigma)|}{\sigma^{\alpha-1}(1-\sigma)^{\beta-1}} \right\}$$

and

$$\mathcal{Y} := \left\{ f \in C^0(0,1) : \sup_{\sigma\in(0,1)} \frac{|f(\sigma)|}{\sigma^{\alpha-1}(1-\sigma)^{\beta-1}} < \infty \right\},$$

$$\|f\| := \sup_{\sigma\in(0,1)} \frac{|f(\sigma)|}{\sigma^{\alpha-1}(1-\sigma)^{\beta-1}}$$

where

$$\alpha := \left\{ \frac{\min|Re(\mu)|}{2\max|Re(\mu)|} : \mu \text{ is an eigenvalue of } \mathbf{f_u}(\bar{\mathbf{u}}_-, \bar{\boldsymbol{\lambda}}) \text{ with positive real part} \right\},$$

$$\beta := \left\{ \frac{\min|Re(\mu)|}{2\max|Re(\mu)|} : \mu \text{ is an eigenvalue of } \mathbf{f_u}(\bar{\mathbf{u}}_+, \bar{\boldsymbol{\lambda}}) \text{ with negative real part} \right\}.$$

Let the operator $\mathsf{H} : \mathcal{X}^n \times \mathbf{R}^p \times \mathbf{R} \to \mathcal{Y}^n$ satisfy equation

$$\mathsf{H}(\mathbf{w}, \boldsymbol{\lambda}, L) := (\mathbf{l} + \mathbf{w})' - L\frac{\mathbf{f}(\mathbf{l} + \mathbf{w}, \boldsymbol{\lambda})}{\|\mathbf{f}(\mathbf{l} + \mathbf{w}, \boldsymbol{\lambda})\|} = \mathbf{0} \tag{4.44}$$

where

$$\mathbf{l}(\sigma) = \sigma\mathbf{u}_+(\boldsymbol{\lambda}) + (1-\sigma)\mathbf{u}_-(\boldsymbol{\lambda})$$

and

$$\mathbf{w}(\sigma) := \mathbf{v}(\sigma) - \mathbf{l}(\sigma).$$

If $(\bar{\mathbf{v}}, \bar{\boldsymbol{\lambda}}, \bar{L})$ is a solution of (4.34), then $\mathsf{H}_{\mathbf{w}}(\bar{\mathbf{w}}, \bar{\boldsymbol{\lambda}}, \bar{L})$ is a Fredholm operator of index $-1-p$ with empty null space and $(\bar{\mathbf{w}}, \bar{\boldsymbol{\lambda}}, \bar{L})$ is an isolated solution of (4.44). Moreover,

$$\mathsf{H}_{\mathbf{w}}(\bar{\mathbf{w}}, \bar{\boldsymbol{\lambda}}, \bar{L}) + \mathsf{H}_{\boldsymbol{\lambda}}(\bar{\mathbf{w}}, \bar{\boldsymbol{\lambda}}, \bar{L}) + \mathsf{H}_{L}(\bar{\mathbf{w}}, \bar{\boldsymbol{\lambda}}, \bar{L})$$

is invertible.

Theorem 4.5 shows that $(\bar{\mathbf{w}}, \bar{\boldsymbol{\lambda}}, \bar{L})$ is an isolated solution for (4.44). Hence $(\bar{\mathbf{v}}, \bar{\boldsymbol{\lambda}})$ is an isolated connecting orbit solution for (4.1).

The system (4.34)–(4.37) is a boundary value problem with the singularity of first kind at both end points $\sigma = 0, 1$. Linear singular boundary value problems have been studied by many authors, e.g., Russell and Shampine [61], de Hoog and Weiss [37, 38], and Weinmüller [72, 73, 74, 75]. Our discussion of the convergence property is similar to that in [74].

The variational equation for (4.34) at the solution is $(\bar{\mathbf{v}}(\sigma), \bar{\boldsymbol{\lambda}})$

$$\frac{d\mathbf{y}}{d\sigma} - \bar{L}[I - \bar{P}(\sigma)]\frac{\mathbf{f_u}(\bar{\mathbf{v}}, \bar{\boldsymbol{\lambda}})}{\|\mathbf{f}(\bar{\mathbf{v}}, \bar{\boldsymbol{\lambda}})\|}\mathbf{y} = \mathbf{0}, \qquad 0 < \sigma < 1. \tag{4.45}$$

where $\bar{P}(\sigma)$ is an orthogonal projection onto $\bar{\mathbf{v}}'(\sigma)$. We further assume that there are no complex eigenvalues in both matrices $\mathbf{f_u}(\bar{\mathbf{u}}_\pm, \bar{\boldsymbol{\lambda}})$, i.e, the connection solution is not of Šilnikov type. Therefore,

$$\lim_{\sigma\to 0^+} \bar{\mathbf{v}}'(\sigma) \quad \text{and} \quad \lim_{\sigma\to 1^-} \bar{\mathbf{v}}'(\sigma)$$

exist. Taking into account of (4.35), we may assume that $\bar{\mathbf{v}}(\sigma)$ is at least $C^1[0,1]$. We can now rewrite (4.45) to

$$\frac{d\mathbf{y}}{d\sigma} - \frac{M(\sigma)}{\sigma(1-\sigma)}\mathbf{y} = \mathbf{0} \tag{4.46}$$

where

$$M(\sigma) = \sigma(1-\sigma)\bar{L}[I - \bar{P}(\sigma)]\frac{\mathbf{f_u}(\bar{\mathbf{v}}, \bar{\boldsymbol{\lambda}})}{\|\mathbf{f}(\bar{\mathbf{v}}, \bar{\boldsymbol{\lambda}})\|}. \tag{4.47}$$

It can be shown that

$$M(0) := \lim_{\sigma\to 0} M(\sigma) \quad \text{and} \quad M(1) := \lim_{\sigma\to 1} M(\sigma)$$

exist. We further assume that all eigenvalues of $M(0)$ and $M(1)$ are real with α and β being the smallest positive eigenvalue for $M(0)$ and largest negative eigenvalue for $M(1)$, respectively. Our goal is to prove the convergence of the Gauss-Radau collocation algorithm described in 4.4.2.

We approximate $\mathbf{y}(\sigma)$ by $\mathbf{p}(\sigma)$, a C^0 piecewise continuous polynomial of degree m, satisfying the collocation equations

$$\mathbf{p}'(\sigma_{ik}) - \frac{M(\sigma_{ik})}{\sigma_{ik}(1-\sigma_{ik})}\mathbf{p}(\sigma_{ik}) = \mathbf{0}, \quad 0 \le i \le N-1, \ 1 \le k \le m, \tag{4.48}$$

where $\sigma_{ik} = \sigma_i + \mu_k h_i$ are the collocation points in the interval $[\sigma_i, \sigma_{i+1}]$. Note, in the Gauss-Radau collocation algorithm, at $\sigma_{0,1} = 0$ and $\sigma_{N-1,m} = 1$, the equation (4.48) is not well defined. We may replace the collocation equations at $\sigma = 0$ and 1 by

$$M(0)\mathbf{p}(0) = M(1)\mathbf{p}(1) = \mathbf{0}.$$

Because $\bar{P}(\sigma)$ is a projection onto $\bar{\mathbf{v}}'(\sigma)$, we have

$$\mathbf{p}(0) = \mathbf{f_u}(\bar{\mathbf{v}}(0), \bar{\lambda})\bar{\mathbf{v}}'(0), \qquad \mathbf{p}(1) = \mathbf{f_u}(\bar{\mathbf{v}}(1), \bar{\lambda})\bar{\mathbf{v}}'(1). \tag{4.49}$$

Clearly, (4.49) satisfies the boundary conditions

$$P_-(\bar{\lambda})\mathbf{p}(0) = \mathbf{0}, \qquad P_+(\bar{\lambda})\mathbf{p}(1) = \mathbf{0}.$$

Theorem 4.6 *Let $M(\sigma) \in C^m[0,1]$, $\mathbf{y} \in C^1[0,1] \cap C^{m+1}(0,1)$ and $\mathbf{p}(\sigma)$ be a C^0 piecewise continuous polynomial of degree m. Moreover, let $\mathbf{p}$ satisfy (4.48) and (4.49). Then, for h sufficiently small*

$$\|\mathbf{y}(\sigma) - \mathbf{p}(\sigma)\| \leq \begin{cases} Ch^\gamma |\ln h|^{d-1}, & 1 < \gamma < m, \\ Ch^m |\ln h|^{d-1}, & m \leq \gamma < m+1, \\ Ch^m |\ln h|^d, & \gamma = m+1, \\ Ch^m, & \gamma > m+1 \end{cases}$$

where $\gamma = min\{\alpha, |\beta|\}$ and d is the dimension of the largest Jordan box associated with α and β.

The proof of this theorem follows from [37] *Lemma* 3.5 and [74] *Lemma* 3.8. Note in general, m is only one and we can only obtain the rate of convergence $O(h)$. One advantage of the collocation method is the property of superconvergence for suitably chosen $\mu_1, \cdots, \mu_m$. If the solution is smooth enough and

$$\int_0^1 s^k w(s) ds = 0, \qquad k = 0, \cdots, \nu < m,$$

then

$$\|\mathbf{y}(\sigma_i) - \mathbf{p}(\sigma_i)\| \leq Ch^{m+\nu+1}, \qquad i = 0, \cdots, N.$$

In particular, with Gauss-Legendre points, the rate of convergence is $O(h^{2m})$. With Gauss-Radau points, the optimal convergence rate is $O(h^{2m-1})$. However, we are not able to prove the superconvergence result at this stage since the solution $\mathbf{v}(\sigma)$ is only C^1 for the non-Šilnikov case. Our numerical results in Table 4.1 indicate that the rate of convergence is close to $O(h^{6.3})$ when $m = 4$. This rate is close to optimal as we use both Gauss-Legendre points and Gauss-Radau points. For the Šilnikov case, $\mathbf{v}(\sigma)$ is only C^0, and the Gauss-Radau collocation algorithm does not converge with the current implementation (as we showed in Example 4.6). The study for this type of problem will be carried out in [48].

4.5 Rational Spectral Methods

In [51, 52], rational spectral methods are applied to compute the connecting orbits for the dynamical system (4.1). These methods have several advantages; they do not require the truncation or translation of the infinite domain, have convergence $O(e^{-cN})$ and, with careful choice of collocation points, can be very efficient.

4.5.1 Rational Basis Functions

Rational spectral methods have been discussed by several authors with different choices of basis functions [14, 15, 20, 71]. In our computation, we use the basis functions

$$R_n(t) = \cos(n \operatorname{arccot}(t)), \qquad n = 0, 1, \cdots, \tag{4.50}$$

which can be obtained by mapping the Chebyshev polynomials

$$T_n(x) = \cos(n \arccos x)$$

from $[-1, 1]$ to $(-\infty, \infty)$ by $t = \cot(\arccos x)$. The first five basis functions are given by

$$\begin{aligned}
R_0(t) &= 1, \\
R_1(t) &= \frac{t}{\sqrt{t^2+1}}, \\
R_2(t) &= \frac{t^2-1}{t^2+1}, \\
R_3(t) &= \frac{t(t^2-3)}{(t^2+1)^{3/2}}, \\
R_4(t) &= \frac{t^4-6t^2+1}{(t^2+1)^2}.
\end{aligned}$$

The orthogonality relation for $\{R_n(t)\}_{n=0}^{\infty}$ follows immediately from that of $\{T_n(x)\}_{n=0}^{\infty}$, i.e.,

$$\int_{-\infty}^{\infty} \frac{1}{1+t^2} R_m(t) R_n(t) dt = \frac{\pi d_n}{2} \delta_{m,n}, \tag{4.51}$$

where $d_0 = 2$, $d_n = 1$ $(n \geq 1)$ and $\delta_{m,n}$ is the Kronecker delta. Like many other spectral basis functions, R_n also satisfy the recursive relations

$$R_{n+1}(t) = \frac{2t}{\sqrt{t^2+1}} R_n(t) - R_{n-1}(t),$$

$$tR_n'(t) = -\frac{n}{4}[R_{n+2}(t) - R_{n-2}(t)],$$

$$R_n''(t) = -\frac{n}{16}[(n+2)R_{n+4}(t) - 4(n+1)R_{n+2}(t) + 6n^2 R_n(t) - 4(n-1)R_{n-2}(t) + (n-2)R_{n-4}(t)].$$

These basis functions $R_n(t)$ are closely related to those used in [71],

$$\sigma_n = \left(\frac{it-1}{it+1}\right)^n, \qquad n = 0, \pm 1, \pm 2, \cdots. \tag{4.52}$$

In fact,

$$R_{2n}(t) = \frac{1}{2}[\sigma_n(t) + \sigma_{-n}(t)]. \tag{4.53}$$

With the help of (4.51), a function $f(t)$ can be expanded as

$$f(t) = \sum_{n=0}^{\infty} a_n R_n(t)$$

where

$$a_n = \frac{2}{\pi d_n} \int_{-\infty}^{\infty} \frac{1}{1+t^2} f(t) R_n(t) dt, \qquad n \geq 0.$$

4.5.2 Rational Spectral Methods

We now consider rational spectral methods for solving (4.1) and (4.2). Let $\mathbf{u} = (u_1, \cdots, u_n)^T$ and $\mathbf{f} = (f_1, \cdots, f_n)^T$. Approximating $u_i(t)$ by

$$u_i(t) = \sum_{k=0}^{N+1} c_{ik} R_k(t), \qquad 1 \leq i \leq n \tag{4.54}$$

(where the approximation is still denoted by $u_i(t)$) and substituting into (4.1), we obtain

$$\sum_{k=0}^{N+1} c_{ik} R_k'(t) = f_i\left(\sum_{k=0}^{N+1} c_{ik} R_k(t), \boldsymbol{\lambda}\right), \quad 1 \leq i \leq n, \tag{4.55}$$

where N is a given positive integer and the derivative of $R(t)$ can be directly calculated from (4.50). For the rational spectral method, (4.55) is required to hold at a set of collocation points. This gives the *collocation equations*

$$\sum_{k=0}^{N+1} c_{ik} R'_k(t_j) = f_i\left(\sum_{k=0}^{N+1} c_{ik} R_k(t_j), \boldsymbol{\lambda}\right), \quad 1 \le i \le n, \quad 1 \le j \le N, \tag{4.56}$$

where

$$t_j = \cot\left(\frac{j\pi}{N+1}\right). \quad 1 \le j \le N$$

are the *Gauss-Lobatto collocation points.*

In practice, it is often more convenient to use the solution values $u_i(t_j)$ at the collocation points than the coefficients c_{ik} as unknowns. From

$$u_i(t_j) = \sum_{k=0}^{N+1} c_{ik} \cos\left(\frac{kj\pi}{N+1}\right), \quad 0 \le j \le N+1, \tag{4.57}$$

we have

$$c_{ik} = \frac{2}{(N+1)\bar{c}_k} \sum_{m=0}^{N+1} \frac{1}{\bar{c}_m} u_i(t_m) \cos\left(\frac{mk\pi}{N+1}\right), \quad 0 \le k \le N+1 \tag{4.58}$$

where $\bar{c}_m = 2$ if $m = 0$ or $N+1$, $\bar{c}_m = 1$ if $1 \le m \le N$, and $t_0 := \infty$, $t_1 := -\infty$. Noticing that

$$R'_k(t_j) = k \sin^2\left(\frac{j\pi}{N+1}\right) \sin\left(\frac{kj\pi}{N+1}\right), \tag{4.59}$$

we have, for $1 \le j \le N$,

$$\begin{aligned} u'_i(t_j) &= \sum_{k=0}^{N+1} c_{ik} k \sin^2\left(\frac{j\pi}{N+1}\right) \sin\left(\frac{kj\pi}{N+1}\right) \\ &= \frac{2}{N+1} \sin^2\left(\frac{j\pi}{N+1}\right) \sum_{k,m} \frac{k}{\bar{c}_k \bar{c}_m} \cos\left(\frac{mk\pi}{N+1}\right) \sin\left(\frac{kj\pi}{N+1}\right) u_i(t_m). \end{aligned} \tag{4.60}$$

Substituting (4.60) into

$$u'_i(t_j) = f_i(\mathbf{u}(t_j), \boldsymbol{\lambda}), \quad 1 \le i \le n, \quad 1 \le j \le N, \tag{4.61}$$

yields nN equations for unknowns $u_i(t_j)$, $i = 1, \cdots, n$, $j = 0, \cdots, N+1$.

As noted in [14, 15, 69, 71], for the rational and the Hermite spectral methods, freedom exists to stretch the t variable by $t \to Lt$. The scaling factor L can be selected to optimize accuracy. A detailed analysis for choosing the scaling factor was given in [14] and [69] for the rational and Hermite spectral approximation, respectively. With the use of scaling factor L, (4.61) can be rewritten as

$$Lu_i'(t_j) = f_i(\mathbf{u}(t_j), \boldsymbol{\lambda}), \quad 1 \le i \le n, \quad 1 \le j \le N. \tag{4.62}$$

One may notice that the number of unknowns is $n(N+2)$ and (4.61) or (4.62) gives only nN equations. If p free parameters are needed to estabish the stable connecting orbit in the system, then $2n+p$ boundary and phase conditions must be employed together with (4.61) to determine $u_i(t_j)$ and $\boldsymbol{\lambda}$. The following two subsections are devoted to deriving these conditions.

4.5.3 Boundary Conditions

Generally speaking, the eigenvector or projection boundary conditions can be used for the computation. However, much simpler boundary conditions can be obtained when there is only one free parameter λ in the system, which is just the case for the computation of homoclinic orbits. In facting, substituting

$$\lim_{t\to-\infty} R_k(t) = (-1)^k, \qquad \lim_{t\to\infty} R_k(t) = 1, \qquad 0 \le k \le N+1 \tag{4.63}$$

into the boundary conditions (4.2), we obtain $2n$ equations as

$$(u_-)_i = \sum_{k=0}^{N+1} (-1)^k c_{ik}, \qquad (u_+)_i = \sum_{k=0}^{N+1} c_{ik}, \qquad 1 \le i \le n. \tag{4.64}$$

4.5.4 Phase Conditions

In the last section, we have seen that the connecting orbits usually change fast only in a relatively small region of the domain. For the rational spectral methods, the collocation points cluster around $t=0$. Thus, the rational spectral approximation can lead to highly accurate resolution of $\mathbf{u}(t)$. Unfortunately, the classical and integral phase conditions all employ a known nearby solution to fix the phase of the current solution. This may shift the high variation region of $\mathbf{u}(t)$ away from $t=0$. Here, we shall use a new phase condition which fixes the high variation region of $\mathbf{u}(t)$ close to $t=0$.

Notice that the function $||\mathbf{u}'(t)||$ measures the change of the solution. Requiring it to reach a maximum at $t = 0$ gives

$$\frac{d||\mathbf{u}'(t)||}{dt}\Big|_{t=0} = \frac{d||\mathbf{f}(\mathbf{u},\lambda)||}{dt}\Big|_{t=0} = 0. \tag{4.65}$$

From (4.65), we obtain the new phase condition as

$$\langle \mathbf{f}(\mathbf{u}(0),\lambda), \mathbf{f}_{\mathbf{u}}(\mathbf{u}(0),\lambda)\mathbf{u}'(0)\rangle = 0. \tag{4.66}$$

Note that

$$\begin{aligned} u_i(0) &= \sum_{k=0}^{[(N+1)/2]} (-1)^k c_{i,2k}, \qquad 1 \le i \le n, \\ u_i'(0) &= \sum_{k=0}^{[N/2]} (-1)^k (2k+1) c_{i,2k+1}, \qquad 1 \le i \le n \end{aligned}$$

where $[a]$ denotes the integral part of a. Hence, (4.66) gives an equation for c_{ik} which can be used to determine a unique solution of (4.1)–(4.2).

The equation (4.56) or (4.61), together with boundary conditions (4.64) and the phase condition (4.66), form a system of n(N+2)+1 nonlinear equations for n(N+2) unknowns $u_i(t_j)$ and one free parameter. This system can be solved by Newton's method.

4.5.5 Numerical Results

Again, we take the Nagumo equations and the Lorenz equations as our test problems. The computations are performed on a SUN 4/670 in double precision. The subroutine DNEQN from IMSL is employed to solve the related nonlinear system. We will refer to the rational spectral methods in the coefficient space and in the solution space as RSM-1 and RSM-2, respectively.

Example 4.7 *The Nagumo equations*

$$\begin{aligned} u_1' &= u_2 \\ u_2' &= cu_2 - u_1(1-u_1)(u_1-a), \qquad 0 < a < 1. \end{aligned}$$

As mentioned earlier, the scaling factor L can be very useful for improving computational accuracy. In Figure 4.19, we plot the numerical solution u_1 against u_2 with $N = 29$ for

$L = 2, 1, 0.5$ and 0.1. Clearly, if L is not carefully chosen then numerical oscillations occur. However, when $L \in [0.1, 0.5]$, there is no oscillation and the solution is fairly accurate. It was found in [14, 15] that the accuracy is not very sensitive to L in the neighborhood of the optimum L. Therefore, it is safe to use any L in this "trusted" region. Based on this observation, for any given N we can obtain a corresponding interval from which we can choose any value for L. The superconvergence property is demonstrated in Figure 4.20. In Figure 4.21, we compare numerical accuracy when using the rational spectral methods RSM-1, RSM-2, and the eigenvector method (EM). For the spectral methods, 29 collocation points are used. For the eigenvector method, 25 mesh intervals and 4 collocation points per interval are used.

Example 4.8 *The Lorenz equations*

$$\begin{aligned} x' &= \sigma(y - x) \\ y' &= \lambda x - y - xz \\ z' &= xy - \mu z. \end{aligned}$$

We use this example to demonstrate our phase condition. Our computation is carried out with 29 collocation points. For $(\sigma, \mu) = (10, \frac{8}{3})$, we obtain $\lambda = 13.926555$. The relative error is about 3.59×10^{-7}. In Figure 4.22, we plot $||\mathbf{u}'(t)||$ against t for $\sigma = 10$ and several values of μ. It is observed that our new phase condition fixes the maximum values of $||\mathbf{u}'||$ at the origin, which ensures the uniqueness of the solution.

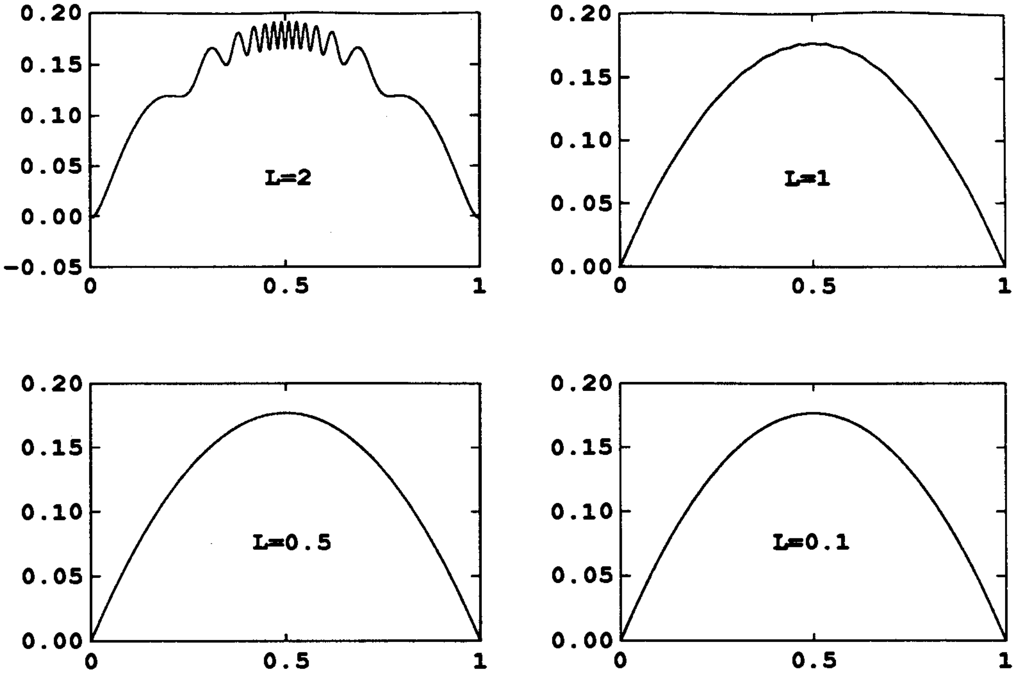


Figure 4.19: Numerical solution (u_1, u_2) obtained by RSM-1 with various values of L

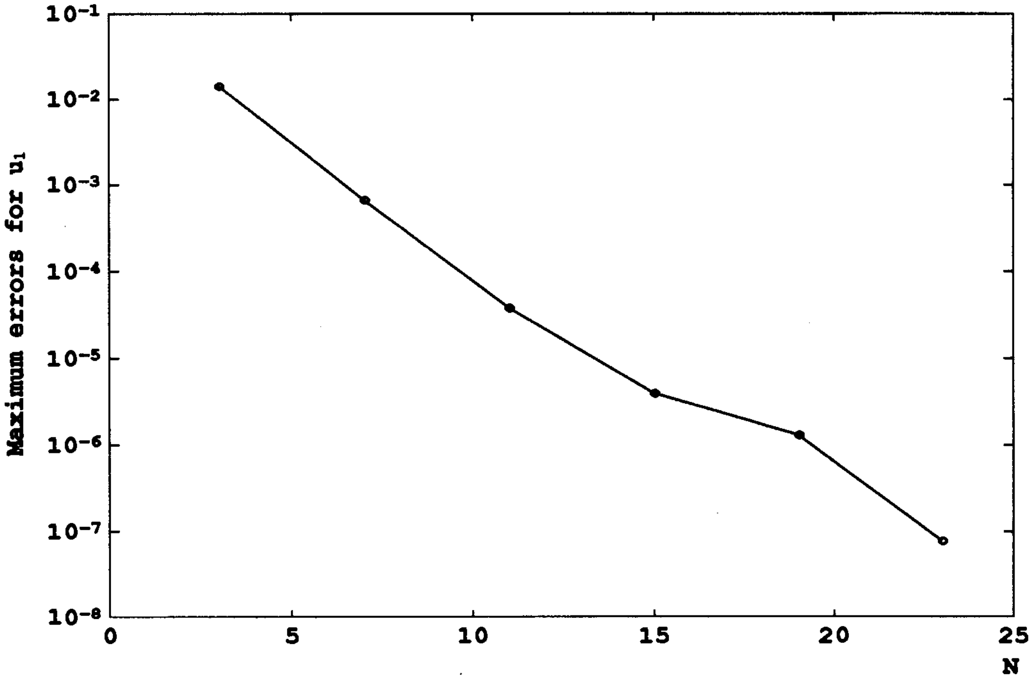


Figure 4.20: Convergence history for the rational spectral method RSM-1

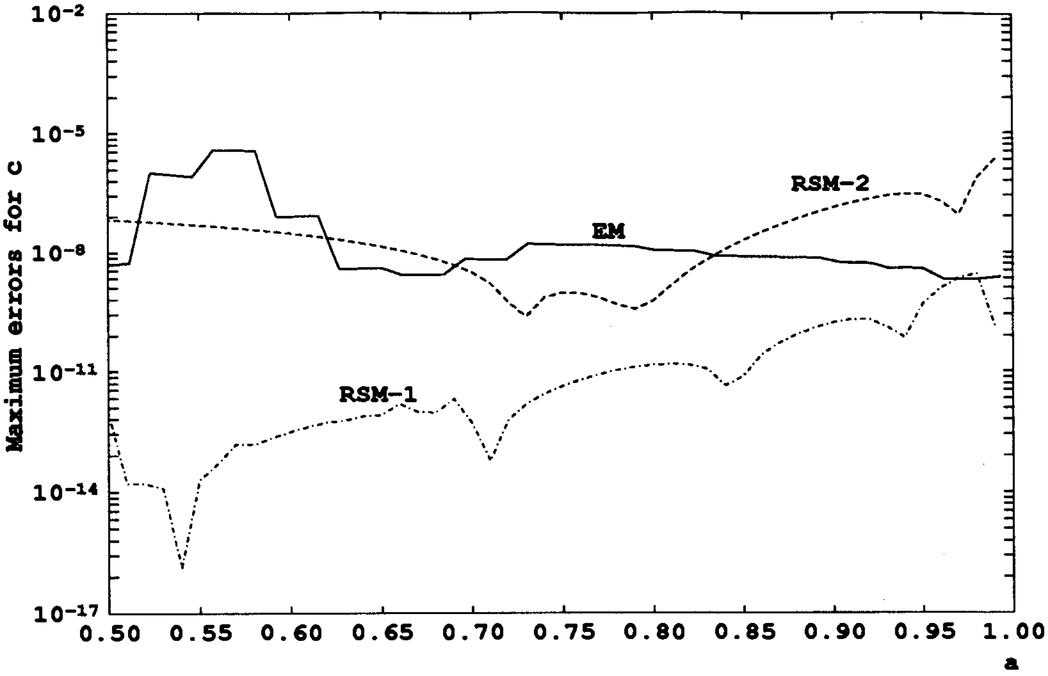


Figure 4.21: Comparison between RSM-1, RSM-2 and EM

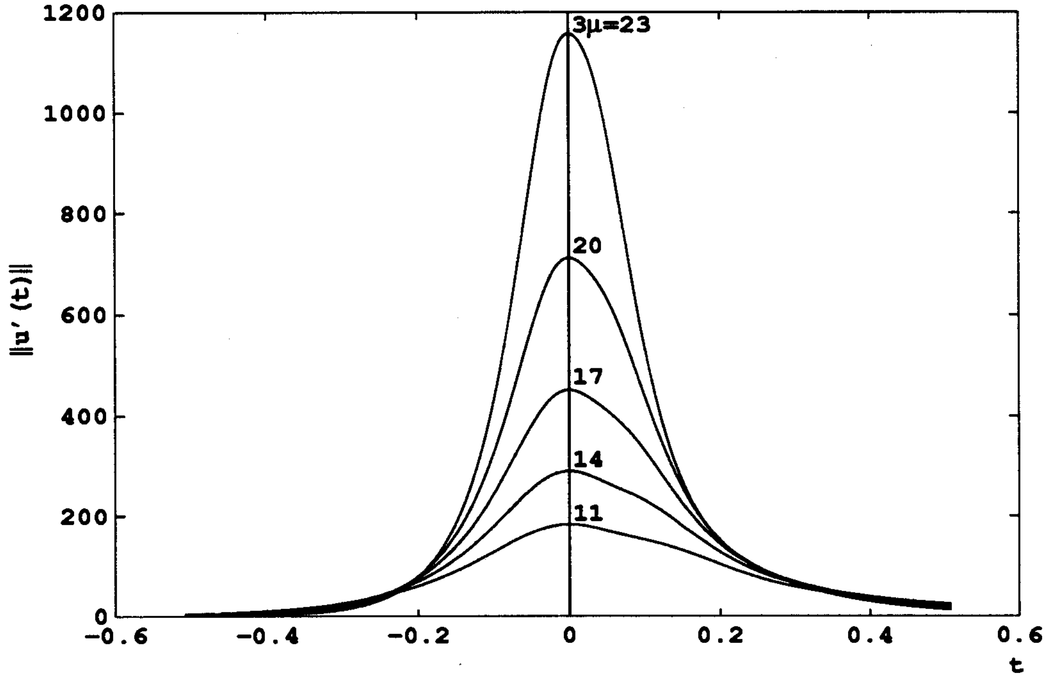


Figure 4.22: New phase condition fixes the maximum of $\|\mathbf{u}'(t)\|$ at $t = 0$

Chapter 5

Conclusions

This chapter contains our conclusions and a discussion of potential further research. Throughout this thesis, we have addressed many issues regarding numerical methods and algorithm for the study of bifurcations.

In chapter 2, we reviewed the stability and bifurcations for steady state problems and periodic problems, initial value problems and boundary value problems. Numerical methods for their computation and continuation were discussed. We especially focused on the implementation of various numerical algorithms in the software AUTO, which has been an extremely useful tool for doing bifurcation analysis. We also mentioned some of the efficiency and stability problems with this software and put our effort towards improving it. Our contributions for the improvement of AUTO include at least two major areas, and they were presented in chapters 3 and 4, respectively.

In chapter 3, we studied the stability problem for the linear system solver for boundary value problems. Several numerical methods such as the multiple shooting method, the finite difference method and the collocation method for solving boundary value ODEs involve solving almost block bi-diagonal linear systems. These systems usually have condition numbers of the same order as the conditioning constant of the BVP. However, without a proper numerical algorithm, solving these equations may be inefficient and unstable. Compactification is one of the well-known examples of an unstable linear solver of this type. As for the Gaussian elimination with partial pivoting, it is generally stable if the pivoting starts in blocks associated with the boundary conditions at the left endpoint. Numerical examples have shown that the instability can easily occur if the boundary conditions are not involved in the pivoting process until the end. Unfortunately, this type of linear solver is used in AUTO in order to obtain the Floquet multipliers of the periodic solution without significant extra cost. In the continuation code, the continuation equations may help to stablize the linear solver, but the potential instability still exists. In order to overcome the instability problems of Gaussian elimination and still preserve the structure of the Jacobian in AUTO to obtain the Floquet multipliers with little extra cost, we have implemented a linear system solver based on the QR factorization to replace the current LU decomposition based solver in AUTO. The extra cost of the QR factorization versus the LU decomposition is not very significant in AUTO since solving the block bi-diagonal linear system usually takes less than 15% of the total CPU time in the serial version. It appears that a good mesh selection strategy can also help to stablize the linear system solver. This phenomenon has been shown by an example. A close investigation into why the continuation and mesh selection strategies can stablize the linear system solvers will be carried out in the near future.

In chapter 4, we reviewed some of the theory and applications of connecting orbits of dynamical systems. These types of solution appear in many real-life applications. We have shown several examples related to the appearance of the connecting orbits. In many cases, such solutions indicate global bifurcations and the route to the chaotic motion. The Lorenz equations and the Kuramoto-Sivashinsky equations are just some of these examples. Traditional numerical methods for computing connecting orbits are based upon the truncation of the infinite domain to a finite one, and then the imposition of certain boundary conditions and phase conditions upon the problem. Although the methods introduced in [10, 11, 28, 29, 33, 34] are quite successful in many situations (such as the problem described in [8]), they all suffer a similar problem, i.e., the solutions change very fast in only in a small part of the domain. The solutions are similar to the ones in singular perturbation problems, and the numerical computation of such types of solution can be very inefficient. However, the solution is smooth and bounded in phase space. Therefore, a geometrical parameterization can be and should be considered. We have implemented an algorithm based upon the arclength parametrization of the solution introduced in [57]. The parametrization leads to a singular boundary value problem which needs to be handled differently from the standard ones. Our algorithm is to use Gauss-Radau collocation for the two end sub-intervals and

Gauss-Legendre collocation for the interior. The projection boundary conditions are used in our formulation. However, the implementation of this algorithm in AUTO is nontrivial, and many parts of the code need to be restructured. For example, the linear system solver needs to be adjusted. Nevertheless, our numerical examples show that, for the non-Šilnikov connecting orbits, this method has great advantages over the others. Numerical results indicate that the superconvergence properties are observed in spite the fact that the solution is usually not $C^m[0,1]$ for $m > 1$. At this stage, we have not been able to prove the superconvergence results. One of the problems in our current implementation is that Šilnikov connecting orbits cannot be handled. We believe that the lack of suitable continuity conditions is the main reason for the failure of this type of solution. In fact, Šilnikov connecting orbits are only $C^0[0,1]$ with respect to the arclength parametrization. However, the information about the eigen-structure for the fixed points may result in a better numerical approach for this case. The study for such problems is currently under investigation. Also, the current implementation for the arclength parametrization method in AUTO is not the best approach, and the efficiency needs to be improved. It is also not suitable for large systems. To solve these problems, a major re-structuring of the software is necessary, and we intend to carry out this change soon.

In the last part of the thesis, we studied rational spectral methods for computing connecting orbits. The advantage of these methods is that the infinite domains need not be truncated. In order to use this method more efficiently, we introduced special types of boundary conditions and phase conditions. The numerical results show that they can be very efficient. However, some of the problems should be carefully studied. For example, the scaling factor plays a very important role in achieving efficiency, but obtaining the optimal scaling factors is non-trivial. A better approach is to use the continuation technique for determining suitable scaling factors. Our current implementation does not have this facility, and it will be also considered in our further work.

Besides the linear system solver and the numerical computation of the connecting orbits, there are many other interesting issues. One of them is the numerical computation and continuation for other types of invariant manifolds, such as invariant tori, and there is the possibility for implementing these methods in mathematical software.such as AUTO. We would like to address some of these questions in the future.

Bibliography

[1] E. Anderson, et al., *LAPACK Users' Guide*, SIAM, Philadelphia, 1992.

[2] D. G. Aronson, E. J. Doedel, and H. G. Othmer, "An analytical and numerical study of the bifurcations in a system of linearly-coupled oscillators", *Physica D*, 25 (1987), pp. 20-104.

[3] U. Ascher, J. Christiansen, and R. D. Russell, "Collocation software for boundary value ODE's", *ACM. Trans. Math. Software*, 7 (1981), pp. 209-222.

[4] U. Ascher, R. M. M. Mattheij, and R. D. Russell, *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations*, Prentice-Hall, Englewood Cliffs, NJ, 1988.

[5] U. Ascher and R. D. Russell, "Reformulation of boundary value problems into 'standard' form", *SIAM Review*, 23 (1981), pp. 238-254.

[6] G. Bader and U. Ascher, "A new basis implementation for a mixed order boundary value solver", *SIAM J. Sci. Stat. Comput.*, 8 (1987), pp. 483-500.

[7] G. Bader and P. Kunkel, "Continuation and collocation for parameter-dependent boundary value problems", *SIAM J. Sci. Stat. Comput.*, 10 (1989), pp. 72-88.

[8] F. Bai, A. Spence, and A. M. Stuart, "The numerical computation of heteroclinic connections in systems of gradient partial differential equations", *SIAM J. Appl. Math.*, 53 (1993), pp. 743-769.

[9] W.-J. Beyn, "Global bifurcations and their numerical computation", in *Continuation and Bifurcations: Numerical Techniques and Applications*, D. Rossed, B. D. Dier and A. Spence eds., Kluwer, Dordrecht, 1990, pp. 169-181.

[10] W.-J. Beyn, "The numerical computation of connecting orbits in dynamical systems", *IMA J. Numer. Anal.*, 9 (1991), pp. 379-405.

[11] W.-J. Beyn, "Numerical methods for dynamical systems", in *Advances in Numerical Analysis*, W. Light, ed. Clarendon Press, Oxford, 1991-1992, pp. 175-236.

[12] Z. Bohte, "Bounds for rounding errors in Gaussian elimination for band systems", *J. Inst. Math. Appls.*, 16 (1975), pp. 790-805.

[13] C. de Boor and R. Weiss, "SOLVEBLOK: a package for solving almost block diagonal linear systems", *ACM Trans. Math. Software*, 6 (1980), pp. 80-87.

[14] J. P. Boyd, "The optimization of convergence for Chebyshev polynomial methods in an unbounded domain", *J. Comp. Phys.*, 45 (1982), pp. 43-79.

[15] J. P. Boyd, "Spectral methods using rational basis functions on an infinite interval", *J. Comp. Phys.*, 69 (1987), pp. 112-142.

[16] J. P. Boyd, *Chebyshev and Fourier Spectral Methods*, Lecture notes in engineering 49, Springer-Verlag, Berlin, New York, 1989.

[17] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang, *Spectral Methods in Fluid Dynamics*, Springer-Verlag, New York, Berlin, 1988.

[18] T. F. Chan and H. B. Keller, "Arclength continuation and multigrid techniques for nonlinear elliptic eigenvalue problems", *SIAM J. Sci. Stat. Comput.*, 3 (1982), pp. 904-913.

[19] T. N. Chan, *M.Sc. Thesis*, Department of Computer Science, Concordia University, Montreal, 1983.

[20] C. I. Christov, "A complete orthonormal system of functions in $L^2(-\infty,\infty)$ space", *SIAM J. Appl. Math.*, 42 (1982), pp. 1337-1344.

[21] E. A. Coddington and N. Levinson, *Theory of Ordinary Differential Equations*, McGraw-Hill, New York, NY, 1955.

[22] W. A. Coppel, *Dichotomies in Stability Theory*, Lecture Notes in Mathematics 629, Springer, New York, 1978.

[23] J. C. Diaz, G. Fairweather, and P. Keast, "FORTRAN packages for solving certain almost block diagonal linear systems by modified alternate row and column elimination", *ACM Trans. Math. Software*, 9 (1983), pp. 358-375.

[24] L. Dieci, J. Lorenz, and R. D. Russell, "Numerical calculation of invariant tori", *SIAM J. Sci. Stat. Comput.*, 26 (1991), pp. 607-647.

[25] E. J. Doedel, "AUTO: a program for the automatic bifurcation analysis of autonomous systems", *Congr. Numer.*, 30 (1981), pp. 265-284.

[26] E. J. Doedel, H. B. Keller, and J. P. Kernevez, "Numerical analysis and control of bifurcation problems, (i) bifurcation in finite dimensions" *Int. J. Bifurcation and Chaos*, 1 (1991), pp. 493-520.

[27] E. J. Doedel, H. B. Keller, and J. P. Kernevez, "Numerical analysis and control of bifurcation problems, (ii) bifurcation in infinite dimensions" *Int. J. Bifurcation and Chaos*, 1 (1991), pp. 745-772.

[28] E. J. Doedel and J. P. Kernevez, "AUTO: software for continuation and bifurcation problems in ordinary differential equations", *Technical Report*, Applied Mathematics, California Institute of Technology, Pasadena, CA, 1986.

[29] E. J. Doedel and M. J. Friedman, "Numerical computation of heteroclinic orbit", *J. Comp. Appl. Math.*, 26 (1989), pp. 155-170.

[30] T. F. Fairgrieve, *Ph.D. Thesis*, Department of Computer Science, University of Toronto, in preparation.

[31] T. F. Fairgrieve and A. D. Jepson, "OK Floquet multipliers", *SIAM J. Numer. Anal.*, 26 (1991), pp. 1446-1462.

[32] R. Fitzhugh, "Mathematical models of excitation and propagation in nerve", in *Biological Engineering*, H. P. Schwan ed., McGraw-Hill, New York, 1969, pp. 1-85.

[33] M. J. Friedman and E. J. Doedel, "Numerical computation and continuation of invariant manifolds connecting fixed points", *SIAM J. Numer. Anal.*, 26 (1991), pp. 789-808.

[34] M. J. Friedman and E. J. Doedel, "Computational methods for global analysis of homoclinic and heteroclinic orbits: a case study", *J. Dynamics Diff. Eqns.*, 5 (1993), pp. 59-87.

[35] G. H. Golub and C. F. van Loan, *Matrix Computations*, 2nd ed. John Hopkins University Press, Baltimore, MD, 1983.

[36] J. Guckenheimer and P. Holmes, *Nonlinear Oscillations Dynamical Systems, and Bifurcations of Vector Fields*, Applied MathematicSciences Vol. 42, Springer, New York, NY, 1983.

[37] F. de Hoog and R. Weiss, "Difference methods for boundary value problems with a singularity of first kind", *SIAM J. Numer. Anal.*, 13 (1976), pp. 775-813.

[38] F. de Hoog and R. Weiss, "Collocation methods for singular boundary value problems", *SIAM J. Numer. Anal.*, 15 (1978), pp. 198-217.

[39] F. de Hoog and R. Weiss, "An approximation theory for boundary value problems on infinite intervals", *Computing*, 24 (1980), pp. 227-239.

[40] M. S. Jolly, I. G. Kevrekidis, and E. S. Titi, *Approximate inertial manifolds for the Kuramoto-Sivashinsky equation: analysis and computations*, *Physica D*, 44 (1990), pp. 38-60.

[41] H. B. Keller, "Numerical solution of bifurcation and nonlinear eigenvalue problems", in *Proc. Applications of Bifurcation Theory*, P. H. Rabinowitz, ed., Academic Press, New York, 1977, pp. 359-384.

[42] H. B. Keller and M. Lentini, "Invariant imbedding, the box scheme and an equivalence between them", *SIAM J. Numer. Anal.*, 19 (1982), pp. 924-962.

[43] Y. Kuramoto and T. Tsuzuki, "Presistent propagation of concentration waves in dissipative media far from thermal equilibrium" *Prog. Theor. Phys.*, 55 (1976), pp. 356-369.

[44] W. F. Langford, "Unfolding of degenerate bifurcations", in *Chaos, Fractals, and Dynamics*, P. Fisher and W. Smith, eds, Marcel Dekker, New York, 1985, pp.87-103.

[45] M. Lentini and H. B. Keller, "Boundary value problems over semi-infinite intervals and their numerical solution", *SIAM J. Numer. Anal.*, 17 (1980), pp. 577-604.

[46] M. Lentini, M. R. Osborne, and R. D. Russell, "The close relationships between methods for solving two-point boundary value problems", *SIAM J. Numer. Anal.*, 22 (1985), pp. 280-309.

[47] M. Lentini and V. Pereyra, "An adaptive finite difference solver for nonlinear two-point boundary value problems with mild boundary layers", *SIAM J. Numer. Anal.*, 14 (1977), pp. 91-111.

[48] L. Liu, G. Moore, and R. D. Russell, "On the computation of homoclinic and heteroclinic orbits", in preparation.

[49] L. Liu and R. D. Russell, "Boundary value ODE algorithms for bifurcation analysis", in *Proc. 13th. IMACS World Congress on Comp. and Appl. Math.*, R. Vichnevetsky and J. J. H. Miller ed. Bail Press (1991), pp. 302-303.

[50] L. Liu and R. D. Russell, "Linear system solvers for boundary value ODEs", *J. Comp. App. Math.*, 45 (1993), pp. 1-15.

[51] Y. Liu and L. Liu, "Rational spectral methods for computing connecting orbits of dynamical systems", *Technical Report*, Centre for Systems Science, Simon Fraser University, October 1992.

[52] Y. Liu, L. Liu, and T. Tang, "The numerical computation of connecting orbits in dynamical systems: a rational spectral approach", *J. Comp. Phys.*, to appear.

[53] E. N. Lorenz, "Deterministic nonperiodic flow" *J. Atmos. Sci.*, 20 (1963), pp. 130-141.

[54] R. M. M. Mattheij, "The stability of LU-decomposition of block tridiagonal matrices", *Austr. Math. Soc.*, 29 (1984), pp. 177-205.

[55] R. M. M. Mattheij, "Stability of block LU-decomposition of matrices arising from BVP", *SIAM J. Alg. Disc. Meth.*, 5 (1984), pp. 314-331.

[56] R. M. M. Mattheij, "Decompling and stability of algorithms for boundary value problems", *SIAM Review*, 27 (1985), pp. 1-44.

[57] G. Moore, "Computation and parametrisation of periodic and connecting orbits", *Manuscript*, in preparation.

[58] J. S. Nagumo, S. Arimoto, and S. Yoshizawa, *Proc. IER*, 50 (1962), pp. 2061-2070.

[59] K. J. Palmer, "Exponential dichotomies and transversal homoclinic points", *J. Differential Equations*, 55 (1984), pp. 225-256.

[60] O. E. Rössler, "An equation for continuous chaos", *Phys. Lett.*, 57A (1976), pp. 397-398.

[61] R. D. Russell and L. F. Shampine, "Numerical methods for singular boundary value problems", *SIAM J. Numer. Anal.*, 12 (1975), pp. 13-36.

[62] R. D. Russell, D. M. Sloan, and M. R. Trummer, "Some numerical aspects of computing inertial manifolds", *SIAM J. Sci. Stat. Comp.*, 14 (1993), pp. 19-43.

[63] S. Schecter, "Numerical computation of saddle-node homoclinic bifurcation points", *SIAM J. Numer. Anal.*, 30 (1993), pp. 1155-1178.

[64] S. Schecter, "Rate of convergence of numerical approximations to homoclinic bifurcation points", *Manuscript*, 1993.

[65] R. Seydel, BIFPACK: *a program package for calculating bifurcation*, State University of New York at Buffalo (1983,1985).

[66] R. Seydel, *From equilibrium to chaos: practical bifurcation and stability analysis*, Elsevier, New York, 1988

[67] L. P. Šilnikov, "A case of the existence of a denumerable set of periodic motions", *Sov. Math. Dokl.*, 6 (1965), pp 163-166.

[68] G. Sivashinsky, "Nonlinear analysis of hydrodynamic instability in laminar flames, I. Derivation of basic equations", *Acta Astronautica*, 4 (1977), pp. 1117-1206.

[69] T. Tang, "The Hermite spectral method for Gaussian-type functions", *SIAM J. Sci. Comput.*, 14 (1993), pp. 594-606.

[70] M. Taylor, *Interactive* AUTO, Department of Chemical Enginerring, Princeton University, 1990

[71] J. A. C. Weideman, "The eigenvalues of Hermite and rational spectral differentiation matrices", *Numerische Mathematik*, 61 (1992), pp. 409-432.

[72] E. Weinmüller, "A difference Mathod for a singular boundary value problems of second order", *Math. Comp.*, 42 (1984), pp. 441-464.

[73] E. Weinmüller, "On the boundary value problem for systems of ordinary second order differential equations with a singularity of the first kind", *SIAM J. Math. Anal.*, 15 (1984), pp. 287-307.

[74] E. Weinmüller, "Collocation for singular boundary value problems of second order", *SIAM J. Numer. Anal.*, 23 (1986), pp. 1062-1095.

[75] E. Weinmüller, "Stability of singular boundary value problems and their discretization by finite differences", *SIAM J. Numer. Anal.*, 26 (1989), pp. 180-213.

[76] S. J. Wright, "Stable parallel elimination for boundary value ODEs", *Technical Report*, MCS-P229-0491, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, 1991.

[77] S. J. Wright, "Stable parallel algorithms for two-point boundary value problems", *SIAM J. Sci. Stat. Comput.*, 13 (1992), pp. 742-764.

[78] S. J. Wright, "A collection of problems for which Gaussian elimination with partial pivoting is unstable", *SIAM J. Sci. Comput.*, 14 (1993), pp. 231-238.