

University of Chester

This work has been submitted to ChesterRep – the University of Chester's
online research repository

<http://chesterrep.openrepository.com>

Author(s): Patricia M Lumb

Title: Delay differential equations: Detection of small solutions

Date: April 2004

Originally published as: University of Liverpool PhD thesis

Example citation: Lumb, P. M. (2004). *Delay differential equations: Detection of small solutions*. (Unpublished doctoral dissertation). University of Liverpool, United Kingdom.

Version of item: Submitted version

Available at: <http://hdl.handle.net/10034/68595>

Delay Differential Equations: Detection of Small Solutions

Thesis submitted in accordance with the requirements of the University of Liverpool for the degree of Doctor in Philosophy by Patricia Mary Lumb.

April 2004

Declaration

No part of the work referred to in this thesis has been submitted in support of an application for another degree or qualification of this or any other institution of learning. However some parts of the material contained herein have been previously published.

Acknowledgements

I would like to thank my colleagues in the Mathematics department at University College Chester for their support and encouragement during the period of the research presented in this thesis. In particular, I would like to acknowledge the advice and guidance of my colleague and supervisor Professor Neville J. Ford.

Abstract

This thesis concerns the development of a method for the detection of small solutions to delay differential equations. The detection of small solutions is important because their presence has significant influence on the analytical properties of an equation. However, to date, analytical methods are of only limited practical use. Therefore this thesis focuses on the development of a reliable new method, based on finite order approximations of the underlying infinite dimensional problem, which can detect small solutions.

Decisions (concerning the existence, or otherwise, of small solutions) based on our visualisation technique require an understanding of the underlying methodology behind our approach. Removing this need would be attractive. The method we have developed can be automated, and at the end of the thesis we present a prototype Matlab code for the automatic detection of small solutions to delay differential equations.

Contents

1	Introduction	1
1.1	Delay differential equations	1
1.1.1	Classification of DDEs	3
1.1.2	Applications of DDEs	5
1.2	Solving DDEs	6
1.2.1	What is meant by a solution of a DDE?	6
1.2.2	Existence and uniqueness of solutions	7
1.2.3	Stability of solutions of DDEs: Some definitions	7
1.2.4	The analytical solution of DDEs	8
1.2.5	The numerical solution of DDEs	12
1.3	Small solutions: An introduction	13
1.3.1	What do we mean by a small solution?	13
1.3.2	What is known about small solutions?	14
1.3.3	Why is their detection important?	15
1.4	Outline of the thesis	16
2	Background theory and information	19
2.1	Introduction and background theory	19
2.1.1	Exponential type calculus	20
2.1.2	Operator theory: A C_0 -semigroup	21
2.1.3	Relevant matrix theory	22
2.2	Different approaches to the theory of DDEs	28
2.2.1	Linear autonomous equations	28
2.2.2	The functional analytic approach	30
2.2.3	An alternative approach	33
2.3	Stability of the solutions of DDEs	33
2.4	Numerical methods for DDEs	35
2.4.1	Stability of the methods	36
2.5	Small solutions: Further background theory	39
2.5.1	Autonomous equations	39
2.5.2	Non-autonomous equations	41
2.6	An example from immunology	43

2.6.1	Introduction	43
2.6.2	The five models	43
2.6.3	Methodology	45
2.6.4	Some of the results	45
2.6.5	Observations from these results	45
3	Our method: Introduction and Justification	48
3.1	Introducing our numerical approach	49
3.1.1	Justification for our approach	49
3.2	Known analytical results about the existence of an equivalent au- tonomous problem	52
3.3	Using our numerical method to estimate the true eigenvalues . . .	54
4	Small solutions in one-dimension	57
4.1	Introduction	57
4.2	Known analytical results	57
4.3	An initial function that can give rise to small solutions	58
4.4	The discrete finite dimensional solution map	62
4.5	Results of applying the trapezium rule	67
4.5.1	Further examples	69
5	Choice of numerical scheme	74
5.1	Introduction	74
5.2	The Adams-Moulton method	74
5.3	Comparing five different numerical methods	78
5.3.1	Results	78
5.4	Further examples	84
5.4.1	Varying the function-type of $b(t)$	84
5.4.2	More complex forms of $b(t)$	84
5.4.3	Values close to a critical value of c_i	85
5.5	Conclusions for the one-dimensional case	86
6	Systems of delay differential equations	89
6.1	Introduction	89
6.1.1	The finite-dimensional solution map	92
6.2	Matrix $A(t)$ is diagonal with $\beta(t) \equiv 0, \gamma(t) \equiv 0$	95
6.2.1	The two-dimensional case	95
6.2.2	Extension to higher dimensions	99
6.3	Matrix $A(t)$ is triangular with $\gamma(t) \equiv 0$	101
6.3.1	The two-dimensional case	101
6.3.2	Extension to higher dimensions	104
6.4	The general real two-dimensional case	108

6.4.1	The eigenvalues of $A(t)$ are always real	108
6.4.2	$A(t)$ has complex eigenvalues	109
6.4.3	How does this relate to the scalar case?	109
6.4.4	Numerical results	110
6.4.5	Conclusions	118
7	Equations with multiple delays	120
7.1	Introduction and theoretical results	120
7.2	Known analytical results	120
7.3	Using our existing ideas directly	121
7.3.1	The case when $m = 1$ and $w = 1$	121
7.3.2	The more general case	122
7.3.3	Applying a numerical method	125
7.3.4	Numerical examples	126
7.3.5	Some observations	128
7.4	A more sophisticated approach using Floquet solutions	129
7.4.1	Developing the rationale	130
7.4.2	Numerical results	131
7.4.3	Some observations	132
7.5	Conclusion	133
8	Single delays revisited	135
8.1	The one-dimensional case	135
8.1.1	Using a transformation to remove the instantaneous term	135
8.2	Analytical results	136
8.3	Introductory background theory	139
8.4	Applying the trapezium rule	142
8.5	Numerical results	142
8.5.1	$p = 1, d \in \mathbb{N}$	143
8.5.2	A more general case	146
8.6	Extension to higher dimensions	150
8.6.1	The two-dimensional case	150
8.6.2	An example of the three-dimensional case	153
8.6.3	Conclusion	154
9	Can statistics help?	155
9.1	Which statistics? A reasoned choice.	156
9.2	Our initial approach: Using the cartesian form of the eigenvalues	157
9.2.1	Examples	157
9.3	Insight from visualisation: Consideration of the eigenvalues in polar form	163
9.3.1	Numerical results	164

10 Automating the process	168
10.1 Introducing ‘smallsolutiondetector1’	168
10.2 The Rationale behind the algorithm	168
10.2.1 The underlying methodology	169
10.3 A theoretical basis for the algorithm	171
10.4 Consideration of the reliability of the algorithm	172
10.5 Illustrative examples	173
10.6 Algorithm: Summary	174
10.7 Algorithm: Possible future developments	174
10.7.1 DDEs with delay and period commensurate	175
11 Complex-valued functions	182
11.1 Introduction	182
11.2 Known analytical results	183
11.3 Justification for our approach	184
11.4 Numerical results	189
11.4.1 The equation does not admit small solutions	189
11.4.2 A sufficient condition for small solutions is satisfied	194
11.4.3 The question of invertibility	201
11.4.4 Other observations and investigations	207
12 Summary and conclusions	213
12.1 Further commentary	214
13 Looking to the future	218
13.1 Is further automation possible?	218
13.2 Small solutions and other classes of DDE	222
A Matlab code for Smallsolutiondetector1	223
A.1 Smallsolutiondetector1	223
A.1.1 definefunctionb	224
A.1.2 smallsolutiondetection	224
A.1.3 modifiedalgorithm	225
A.1.4 reducingtolerance	228
A.1.5 eigenvaluecalculator	229
A.1.6 decisionchecker	231
B Matlab code for ‘findanswerchangeoint’	237
C Some relevant theorems	241
C.1 Theorem 3.2 from [27]	241
C.2 Theorem 3.1 from [33]	241

D Further examples of eigenspectra	243
E The first generation of the algorithm	246
F Preservation of the property of admitting small solutions	248

Chapter 1

Introduction

1.1 Delay differential equations

The study of delay differential equations (DDEs), that is equations of the form

$$y'(t) = f(t, y(t), y(t - \tau_1(t, y(t))), y(t - \tau_2(t, y(t))), \dots),$$

was originally motivated mainly by problems in feedback control theory [55]. The delays, $\tau_i, i = 1, 2, \dots$ are measurable physical quantities and may be constant, a function of t (the variable or time dependent case) or a function of t and y itself (the state dependent case). Examples of delays include the time taken for a signal to travel to the controlled object, driver reaction time, the time for the body to produce red blood cells and cell division time in the dynamics of viral exhaustion or persistence. In the life sciences delays are often introduced to account for hidden variables and processes which, although not well understood, are known to cause a time lag (see [8, 13] and the references therein).

‘Time delays are natural components of the dynamic processes of biology, ecology, physiology, economics, epidemiology and mechanics’ [37] and ‘to ignore them is to ignore reality’ [55].

Ordinary differential equations (ODEs) have been used as a fundamental tool of the mathematical modeller for many years. However, an ODE model formulation of a system ignores the presence of any delays. Formulation as a FDE, (a functional differential equation or differential equation with deviating argument), which includes all DDEs, enables both the current and all previous values of a function and/or its derivatives to be considered when determining the future behaviour of a system. This often leads to an improved model of a process since ‘an increase in the complexity of the mathematical models can lead to a better quantitative consistency with real data’, but at a cost [8]. The size of the

delay relative to the underlying time-scales influences the modellers’s decision about the choice of model formulation [6]. Systems for which a model based on a functional differential equation is more appropriate than one based on an ODE can be referred to as “problems with memory”. A delay differential equation model may also be used to approximate a high-dimensional model without delay by a lower dimensional model with delay, the analysis of which is more readily carried out. This approach has been used extensively in process control industry (see [54], p. 40-41).

There are many similarities between the theory of ODEs and that of DDEs and analytical methods for ODEs have been extended to DDEs when possible. However, their differences have necessitated new approaches. In Table 1.1 we highlight important differences between ODEs and DDEs, such as the need for an initial function and the infinite dimensionality of a DDE.

ODE Model	DDE Model
Assumes: effect of any changes to the system is instantaneous (A principle of causality, [41, 55])	Assumes: effect of any changes to the system is not instantaneous . i.e. past history is taken into account
Generates a system that is finite dimensional	Generates a system that is infinite dimensional
Needs an initial value (to determine a unique solution)	Needs an initial function (to determine a unique solution)
	Advantage Enables a more accurate reflection of the system being modelled
	Disadvantage: The analytical theory is less well developed

Table 1.1: Important Differences between ODEs and DDEs

Changes in the qualitative behaviour of the solution may be observed as a consequence of a delay term. In biological models the presence of delays is ‘a potent source of nonstationary phenomena such as periodic oscillations and instabilities’ [8, 13]. The delay can act as a stabiliser or a destabiliser of ODE models [11, 13, 37]. The following example from [7] provides a simple illustration.

Example 1.1.1 Consider the equation

$$(1.1) \quad x'(t) = \lambda x(t) + \mu x(t - \tau), \quad \tau \geq 0.$$

The zero solution of (1.1) is asymptotically stable if $\lambda + |\mu| \leq 0$.

If $\mu = 0$ we obtain an ODE whose zero solution is asymptotically stable if $\lambda < 0$.

However, positive values of λ exist which, with corresponding negative values of μ , give rise to asymptotic stability of (1.1). Hence the delay term can stabilise an unstable ODE. Alternatively, if $\tau = 0$, again leading to an ODE case, we have asymptotic stability if and only if $\lambda + \mu < 0$. However, if $\tau > 0$ then $\lambda + \mu < 0$ is insufficient to guarantee stability and in this case the introduction of a delay term can destabilise a stable solution.

The presence of an initial function, instead of an initial value, has several consequences:

1. In general it leads to a derivative jump discontinuity at the point t_0 , that is the right-hand derivative $y'(t_0)^+$ does not equal the left-hand derivative $\phi'(t_0)^-$. This propagates and leads to subsequent discontinuity points [11].
2. ‘Unlike the ordinary equations, there is no longer injectivity between the set of initial data and the set of solutions’ [11]. ‘A fundamental difference between DDEs and ODEs is that solutions corresponding to different initial function data can intersect’ [8, 13] (see also p. 312 in [6]). We illustrate this in example 1.2.2.
3. When the delay is state dependent the lack of regularity of the initial function may lead to a loss of uniqueness for the solution of the DDE, or to its termination after some bounded interval (see [11], p. 3-5 for further details and examples).

The dynamical structure exhibited by DDEs is richer than that of ODEs. Initial functions can be chosen in an infinite dimensional subspace. Hence, even a scalar problem can be infinite dimensional. According to [52] (page 123) initial functions should be functions that occur in practice - but for different real world processes there can be different admissible initial functions. Oscillatory and chaotic behaviour can arise even in the scalar case (see comments in [11] on the delay logistic and Mackey Glass equations). As a comparison we note that oscillatory behaviour of ODEs requires at least two components and that at least three components are needed for chaotic behaviour [4, 11].

1.1.1 Classification of DDEs

Delay differential equations can be classified as:-

- linear or non-linear,
- autonomous (invariant under the change $t \rightarrow t + T$ for all $T \in \mathbb{R}$) or non-autonomous,
- periodic with period T , $T > 0$, if invariant under the mapping $t \rightarrow t + T$.

They are also classified by their delay type and by their dependence on the state variable. Neutral delay differential equations (NDDEs) are characterised by the dependence of the derivative on previous derivatives, as in

$$y'(t) = F(t, y(t), y(\alpha(t)), y'(\beta(t))).$$

The reader is referred to [37, 54]. Formulation as a stochastic delay differential equation (SDDE) enables the effect of unknown disturbances or random processes to be taken into account in addition to the previous history. The reader is referred to [8, 53, 54] for further relevant theory and applications.

An equation can also be described as *stiff*. Various interpretations of the concept of stiffness in relation to ODEs can be found in the literature (see for example [2, 15, 56]). Section 3.1.2 in [57] cites several references relating to stiffness in ODEs. Reference to the stiffness of a DDE is found in [7], the authors of which state that “the delay term has an essential role to play”, and that it should not be ignored. Baker in [6] interprets stiffness in the context of DDEs, indicates the potential problem caused by the modification to the behaviour of the solution when delay terms are included and states that further work is needed in this area. In [47] in’t Hout defines stiffness and, giving several supporting references, comments on the fact that stiff initial value problems often arise in the field of immunology.

In this thesis we concentrate on DDEs with one or more fixed delays, that is, on equations of the form

$$(1.2) \quad \begin{aligned} \dot{y}(t) &= f(t, y(t), y(t - \tau)), \quad t > 0; \\ y(\theta) &= \phi(\theta), \quad -\tau \leq \theta \leq 0. \end{aligned}$$

or

$$(1.3) \quad \begin{aligned} \dot{y}(t) &= f(t, y(t), y(t - \tau_1), y(t - \tau_2), \dots, y(t - \tau_m)), \quad t > t_0; \\ y(\theta) &= \phi(\theta), \quad -\tau \leq \theta \leq 0. \end{aligned}$$

DDEs of the form (1.2) are said to be *retarded* if $\tau > 0$ and *advanced* if $\tau < 0$ (real-life examples of advanced delay equations can be found in economics [7]). A more general form of a DDE is given by

$$(1.4) \quad \dot{y}(t) = f(t, y(t), y(\alpha_1(t)), y(\alpha_2(t)), \dots, y(\alpha_m(t))).$$

Equations where $\alpha_\ell(t_*, y(t_*)) > t_*$, for some ℓ such that $1 \leq \ell \leq m$ and some $t_* > t_0$ are called advanced delay equations. Equation (1.4), with $m = 1$ and $\alpha_1(t) = t - \tau(t)$, is said to have fading memory if $\alpha_1(t) \rightarrow \infty$ as $t \rightarrow \infty$ and persistent memory if $\alpha_1(t) \not\rightarrow \infty$ as $t \rightarrow \infty$. The delay (or lag) is bounded if $\sup \tau(t) < \infty$, constant if $\alpha_1(t) = t - \tau_*$ with τ_* fixed, state dependent if $\alpha(t) = t - \tau(t, y(t))$ and vanishing if $\alpha(t_*) \rightarrow t_*$ as $t \rightarrow \infty$ [6, 53, 54]. A delay that depends on a continuum, possibly unbounded, set of past values is said to be distributed.

1.1.2 Applications of DDEs

Delay differential equation models have been considered as an alternative to ODE modes in a wide and diverse range of applications. Hutchinson, one of the first mathematical modellers to introduce a delay in a biological model, modified the classical model of Verhulst to account for hatching and maturation periods. Driver, in [23], gives several examples and cites references for earlier appearances of DDEs, for example in elasticity theory by Volterra in 1909.

Evidence of the wide-ranging application of DDEs is readily found in the literature. [8] and [5] report on the use of DDEs in numerical modelling in the biosciences and include applications in epidemiology, immunology, ecology and and in the study of HIV. The reader is referred to these and the references therein for further details and examples. [55] and [37] focus on applications of DDEs in population dynamics. Chapter 1 of [53] and chapter 2 of [54] detail the use of DDEs in a variety of general subject areas including viscoelasticity, physics, technical problems, biology, medicine, mechanics, the economy and immunology. Table 1.2 provides references to examples illustrating usage of several classes of DDE.

Class of DDE	Areas of application	Reference
Retarded DDE	Radiation damping	[18]
	Modelling tumour growth	[14]
Distributed DDE	Model of HIV infection	[62], p. 76
	Biomodelling	[8, 13] + included refs.
Neutral DDE	Distributed networks	[54], p. 32 and 191
Stochastic DDE	pupil light reflex	[13], p. 191, [8], p. 7
	immune responses	[8] + included refs.
	blood cell production	[8] + included refs.

Table 1.2: Examples of applications of types of DDEs

In the literature (to date) the majority of models employ state independent lag functions and constant delays are the most widely used delay-type [4]. This is possibly due to the analytical problems encountered if the problem is formulated using a more general equation [6]. However, applications of all types can be found and in Table 1.3 we provide references to illustrative examples.

Type of delay	Application	Ref.	Page
Single fixed delay	Nicholson blowflies model	[53]	27 + refs.
	Immunology	[9]	
	Immunology	[58]	
Multiple fixed delays	Cancer chemotherapy	[54]	74
	Lifespans in population models	[10]	
	Infectious disease modelling	[39]	347
	Enzyme kinetics	[39]	348
Varying delay (time dependent)	Transport delays	[54]	46
Varying delay (state dependent)	Combustion in the chamber of a turbojet engine	[54]	189

Table 1.3: Examples of applications of DDEs for different types of delay

1.2 Solving DDEs

1.2.1 What is meant by a solution of a DDE?

Two related concepts of the solution of a DDE are possible depending on whether the initial function is regarded as an independent object or as part of the solution. Attempts to utilise the tools of classical theory of ODEs in the study of DDEs were hindered by the first viewpoint which implies ‘the continuous joining of solution x with the initial function ϕ ’ (see [3]). This traditional joining, requiring $x(a) = \phi(a)$, was abandoned by many authors at the end of the 1960s which immediately allowed the theory of operators in Banach spaces to be used in the basic theory of DDEs (see discussions in [3] p. 3-5, 19-21; [53] p. 2-4 and [54] p. 12-14). We note that the later definition, thought to be more natural [3], does not contradict the traditional definition, in which the solution is understood “as a continuous continuation by virtue of the initial function ϕ ”, but complications caused by an extra condition $x(a) = \phi(a)$ are removed [3].

A widely-used version of a solvability theorem for the initial value problem for retarded delay equations is given in [54] as follows:

Consider

$$(1.5) \quad \dot{x}(t) = f(t, x_t), \quad x_t(\theta) := x(t + \theta), \quad -h \leq \theta \leq 0,$$

$$(1.6) \quad x_{t_0} = \phi,$$

where h is a positive constant, $x(t) \in \mathbb{R}^n$, $t_0 \in \mathbb{R}$, $\phi : [-h, 0] \rightarrow \mathbb{R}^n$, $n \geq 1$. A function $x \in C^1$ is said to be a solution of (1.5), (1.6), on an interval with left-hand end t_0 if it satisfies (1.5) along with $x_t(t + \theta) := \phi(t + \theta - t_0)$ for $t + \theta \leq t_0$

and $x(t_0) = \phi(0)$. The reader is referred to [12, 23, 41] for further details about uniqueness and existence theory for DDEs.

A solution of the form $x(t) = \bar{x}$ such that $F(\bar{x}, \bar{x}) = 0$ is known as the *steady state solution*. For example, the equation $\dot{x}(t) = Ax(t)[1 - x(t - 1)]$ has the steady state solution (or equilibrium solution) given by $x(t) = 1$.

1.2.2 Existence and uniqueness of solutions

For the delay equation $\dot{x}(t) = F(x(t), x(t - \tau)), t \geq 0$ the process referred to as ‘the method of steps’ guarantees a unique, globally defined solution on $[-\tau, \infty)$ (see [77]). The smoothness of the solution increases as t increases. In general solutions are only defined on $[-\tau, \infty)$. We note that additional smoothness of the initial function ϕ is required before backward continuation can be considered, and that uniqueness cannot be guaranteed [72, 77].

In [12] conditions for the existence and uniqueness of solutions are found in Theorem 3.1 for equations of the form $a_0 u'(t) + b_0 u(t) + b_1 u(t - w) = f(t)$, and in Theorem 6.1 for systems of constant coefficient DDEs. [23] includes existence and uniqueness theorems for more general DDEs and chapter 11 in [12] relates to those for non-linear DDEs (see also [11]).

1.2.3 Stability of solutions of DDEs: Some definitions

In this section we review definitions concerning the concept of the stability of a solution. The sensitivity of a particular solution to changes in the problem is an important characteristic [6]. These changes can be in the parameters of the model or in the initial conditions or due to persistent disturbances and result in different definitions of stability. Many complementary tools for analysing stability exist [6].

Consider equation

$$(1.7) \quad x'(t) = F(t, x_t), \quad x_t(\theta) = x(t + \theta), -h \leq \theta \leq 0$$

The study of the stability of a solution of (1.7) can be reduced to the study of the stability of the trivial (zero) solution (see, for example, [40] and p. 200 in [54]).

Definition 1.2.1 Stability of the trivial solution [54].

(see also, for example, [12, 23, 40, 41, 53, 55, 65])

Let $x(t)$ be a continuous function that satisfies (1.7) for $t > t_0$. The trivial solution of (1.7) is called:

stable for a given t_0 if for any $\epsilon > 0$ there exists a $\delta = \delta(\epsilon, t_0) > 0$ such that $|x(t)| \leq \epsilon$ for any initial function $\phi \in Q_\delta$ and $t \in [t_0, \infty)$;

uniformly stable if for any $\epsilon > 0$ there exists a $\delta(\epsilon) > 0$ independent of t_0 such that $|x(t)| \leq \epsilon$ for any initial function $\phi, t_0 \in \mathbb{R}, t \in [t_0, \infty)$;

asymptotically stable if for a given $t_0 \in \mathbb{R}$ it is stable and there exists a $\delta = \delta(t_0) > 0$ such that $\lim_{t \rightarrow \infty} x(t) = 0$ for any initial function ϕ ;

uniformly asymptotically stable if it is uniformly stable and there exists a $H > 0$ such that for any $\gamma > 0$ there is a $T(\gamma) > 0$ such that $|x(t)| \leq \gamma$ for any $t_0 \in \mathbb{R}, t \geq t_0 + T(\gamma)$ and $\phi \in Q_H$, where $Q_\delta := \{\mu \in C[-h, 0] : \|\mu\| < \delta\}$.

We note that uniform asymptotic stability is stronger than that of asymptotic stability and uniform stability is stronger than stability.

The stability of a solution can be dependent on the initial time t_0 . Example 1.3 in [54] shows stability of the trivial solution of $\dot{x}(t) = a(t)x(t - \frac{3\pi}{2})$, $a \in C(\mathbb{R})$, for $t_0 = 0$ but not for any $t_0 \geq 3\pi$. We review the stability of particular equations and of numerical methods in sections 2.3 and 2.4.

1.2.4 The analytical solution of DDEs

Real-life problems with delay are generally too complex for analytical solution [7]. However, for delay differential equations with constant coefficients several approaches to finding their analytical solution are possible. These include:

- the method of steps, developed by Bellman for constant delays and by Bellman and Cooke for variable delays (subject to stated hypotheses (see [11] and references therein)),
- a search for exponential solutions,
- using Laplace transforms.

The method of steps

In the method of steps we begin with the constant coefficient delay differential equation, defined for $t > t_0$ and the initial function defined on the interval $[t_0 - \tau, t_0]$ where τ is the delay. We are looking for a continuous extension into the future. We first consider the interval $[t_0, t_0 + \tau]$ on which the DDE reduces to an ODE. We find a solution valid on this interval and then use this solution as the initial function for the interval $[t_0 + \tau, t_0 + 2\tau]$. We then find a solution on $[t_0 + \tau, t_0 + 2\tau]$ and in this way the solution is extended forward from interval to interval. Continuing in this way yields a solution of the DDE, valid on $[t_0 - \tau, \infty)$, that becomes smoother in t as t increases. At each step in the process we are solving an ODE for which, under the hypothesis of uniform Lipschitz continuity of

the right hand side of the equation, we are guaranteed a unique solution [15, 39]. The process can be continued indefinitely but calculations become unwieldy very quickly. In addition it is not easy to determine properties of the solution such as the behaviour of the solution as $t \rightarrow \infty$. This method correctly suggests that properties of scalar DDEs are more similar to those of systems of ODEs rather than a scalar ODE [4]. Solving the DDE on an unbounded interval is an infinite dimensional problem. We illustrate the method in the following example.

Example 1.2.1 We seek a solution for the DDE $x'(t) = 2x(t-1), t \geq 1$ with the initial function given by $x(t) = t, 0 \leq t \leq 1$. We have computed the solution for $1 \leq t \leq 5$ and in Table 1.4 we give details of the solution over successive time intervals.

Time interval	ODE $x'(t)$	Initial condition	Solution $x(t)$
[1, 2]	$2(t-1)$	$x(1) = 1$	$1 + (t-1)^2$
[2, 3]	$2(t-2)^2 + 2$	$x(2) = 2$	$\frac{2}{3}(t-2)^3 + 2(t-1)$
[3, 4]	$\frac{4}{3}(t-3)^3 + 4(t-2)$	$x(3) = \frac{14}{3}$	$\frac{1}{3}(t-3)^4 + 2(t-2)^2 + \frac{8}{3}$
[4, 5]	$\frac{2}{3}(t-4)^4 + 4(t-3)^2 + \frac{16}{3}$	$x(4) = 11$	$\frac{2}{15}(t-4)^5 + \frac{4}{3}(t-3)^3 + \frac{16}{3}(t-2) - 1$

Table 1.4: Solution of DDE in example 1.2.1 for $1 \leq t \leq 5$ using the method of steps.

In the left-hand diagram of Figure 1.1 we show the solution computed using the method of steps (dotted line) and the solution obtained using the numerical code DDE23 (solid line). Discontinuities in the derivatives exist, for example, $x'(1)^- = 1$ and $x'(1)^+ = 0$. Further details and examples can be found in [12].

Example 1.2.2 We now present an example to illustrate the potential for solutions of the same DDE but with different initial functions to intersect. In the right-hand diagram of Figure 1.1 we present the solution of the DDE in example 1.2.1 and the solution of the same DDE but with a different initial function, $x(t) = (t-2)^2 - 1, 0 \leq t \leq 1$. The intersection of the two solution trajectories is evidence of a phenomenon that is possible for DDEs but not for ODEs. (Details of the computation of the second solution using the method of steps are presented in Table 1.5).

Searching for exponential solutions

Functions consisting of a linear combination of products $z^k e^{a_k z}$ (with integer $k \geq 0$), are known as quasipolynomials or exponential polynomials [12, 54].

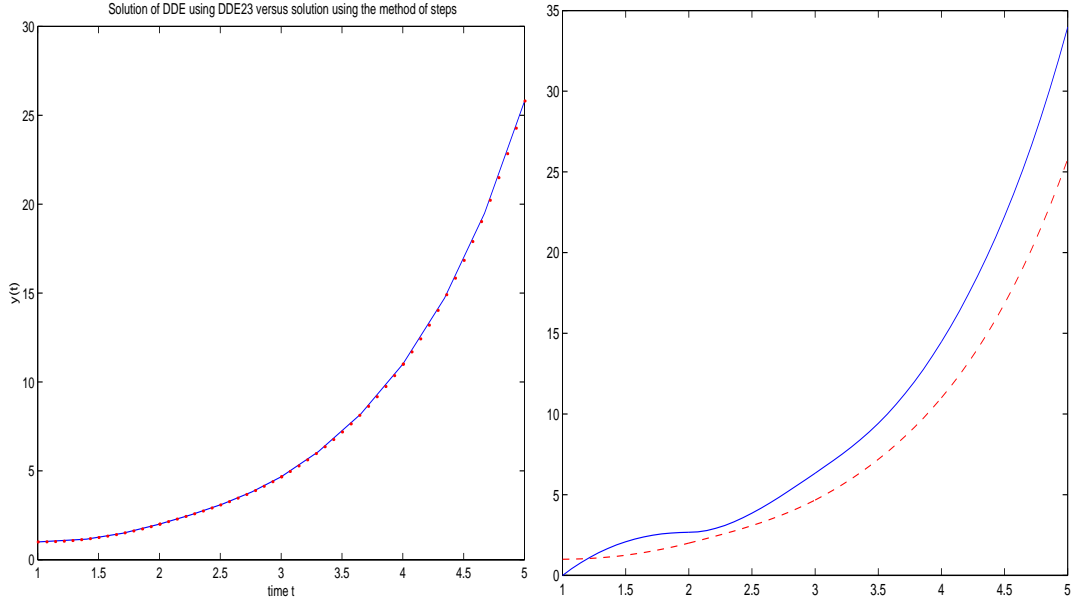


Figure 1.1: Left: Solutions of DDE in example 1.2.1 using DDE23 (solid line) and the method of steps (dotted line). Right: Example 1.2.2 illustrating that solutions for different initial functions can intersect.

Time int.	ODE $x'(t)$	Initial condition	Solution $x(t)$
[1, 2]	$2(t-3)^2 - 2$	$x(1) = 0$	$\frac{2}{3}(t-3)^3 - 2t + \frac{22}{3}$
[2, 3]	$\frac{4}{3}(t-4)^3 - 4(t-1) + \frac{44}{3}$	$x(2) = \frac{8}{3}$	$\frac{1}{3}(t-4)^4 - 2(t-1)^2 + \frac{44}{3}t - 30$
[3, 4]	$\frac{2}{3}(t-5)^4 - 4(t-2)^2 + \frac{88}{3}(t-1) - 60$	$x(3) = \frac{19}{3}$	$\frac{2}{15}(t-5)^5 - \frac{4}{3}(t-2)^3 + \frac{44}{3}(t-1)^2 - 60t + \frac{1999}{15}$
[4, 5]	$\frac{4}{15}(t-6)^5 - \frac{8}{3}(t-3)^3 + \frac{88}{3}(t-2)^2 - 120(t-1) + \frac{3998}{15}$	$x(4) = \frac{217}{15}$	$\frac{2}{45}(t-6)^6 - \frac{2}{3}(t-3)^4 + \frac{88}{9}(t-2)^3 - 60(t-1)^2 + \frac{3998}{15}t - \frac{8881}{15}$

Table 1.5: Solution of $x'(t) = 2x(t-1), t > 1; x(t) = (t-2)^2 - 1, 0 \leq t \leq 1$ for $1 \leq t \leq 5$ using the method of steps.

Searching for solutions of constant delay equations of the form $ce^{\lambda t}$ generally leads to the search for the infinitely many roots of a quasipolynomial, the characteristic equation. Using this approach for the equation in example 1.2.1 leads to the search for solutions of the equation $\lambda = 2e^{-\lambda}$ which has infinitely many complex roots. Linear combinations of known solutions are also solutions and hence there

are infinitely many exponential solutions. The reader is referred to [12, 54] for further discussion.

Laplace transform methods

Applying the Laplace transform to a linear constant coefficient DDE results in a quasipolynomial in the parameter of the transform. Although it is not usually possible to determine explicitly all the zeros of the quasipolynomial knowledge of their location is an aid in the analysis of stability [65].

Laplace transform techniques are used in the solution of DDEs arising in control theory by rational approximation methods (see reference in [4]).

We illustrate the solution of a DDE by Laplace transform methods in the following example:

Example 1.2.3 Consider again the equation $x'(t) = 2x(t - 1)$. Taking Laplace transforms leads to

$$\int_1^{\infty} x'(t)e^{-st} dt = 2 \int_1^{\infty} x(t - 1)e^{-st} dt,$$

from which we obtain, (using a change of variable on the right-hand side),

$$[x(t)e^{-st}]_1^{\infty} - \int_1^{\infty} x(t)(-se^{-st})dt = 2 \int_0^{\infty} x(u)e^{-s(1+u)} du.$$

Assuming that $x(t)e^{-st} \rightarrow 0$ as $t \rightarrow \infty$ this leads to

$$-e^{-s}x(1) + s \int_1^{\infty} e^{-st}x(t)dt = 2e^{-s} \int_0^1 x(u)e^{-su} du + 2e^{-s} \int_1^{\infty} x(u)e^{-su} du.$$

Hence, assuming that $s - 2e^{-s} \neq 0$ then we obtain

$$\int_1^{\infty} x(t)e^{-st} dt = \frac{x(1)e^{-s} + 2e^{-s} \int_0^1 x(u)e^{-su} du}{s - 2e^{-s}}.$$

Assuming the inversion formula can be applied we obtain

$$x(t) = \int_{(C)} \frac{x(1)e^{-s} + 2e^{-s} \int_0^1 x(u)e^{-su} du}{s - 2e^{-s}} .e^{st} ds.$$

Hence, provided that all the steps can be rigorously justified, the solution can be expressed in terms of the initial values of $x(t)$ over $[0, 1]$ by means of a contour integral. We note that it is rare for the resulting contour integral to be expressible in terms of elementary functions. However, we are able to use it to deduce useful information about the solution. For further details and proofs of the appropriate results see [12].

1.2.5 The numerical solution of DDEs

Given a DDE to solve, one option is to reduce it to a system of ODEs to enable solution using an ODE numerical code. The elimination of the lag-terms from the DDE is achieved by the introduction of additional variables. In Bellman's method of steps (see section 1.2.4) a DDE, with initial data on $[-\tau, 0]$, is represented 'on successive intervals $[0, \tau]$, $[\tau, 2\tau]$, ..., $[(N-1)\tau, N\tau]$ by successive systems of ODEs with increasing dimension' [7]. Authors of [7] also refer to the use of 'gearing up' variables to model the effect of the time lag and to the 'introduction of intermediate stages using an ODE system to mimic the transition through the stages'. However, replacing a scalar DDE by a system of ODEs is felt to be a risky strategy by authors of [4, 24] and authors of [13] note that, although this approach has appeal, "the long-term dynamics of DDEs and of approximating finite-dimensional ODEs differ substantially". They advise that the use of a purpose-built numerical code for DDEs may prove advantageous.

The classical approach to numerical calculations involves designing algorithms suitable for a wide range of problems. Authors of [7] regard "the temptation to try for a code that is optimal for all classes of DDE" as a major problem and authors of [48] refer to "a new paradigm for numerical analysis". Qualitative numerical analysis aims, when possible, to embed known qualitative information about the system under consideration into the numerical method, resulting in algorithms which cater for small collections of similar problems. The advantages of the classical approach are clear. Users of numerical mathematics need to be aware of a narrower range of computational tools. The reader is referred to the discussions in section 1 of [48] and in the first section of [49]. We note here that in chapter 10 we adopt the second approach and present an algorithm for a particular class of DDE.

Faced with these two different approaches, designers of codes for solving DDEs have to decide whether the code being developed is to handle general DDEs or particular classes of DDEs. In addition, users of codes need to be aware of their applicability to ensure that a suitable code is selected. It would be unwise, for example, to attempt to use a code specifically designed to solve 'stiff' problems if the problem is known not to be stiff. The bibliography in [5] introduces the reader to papers and technical reports involving the numerical solution of DDEs.

Discussions about the issues involved in the numerical solution of evolutionary delay differential equations can be found in the literature (see, for example, [4, 7, 63, 64]). The four main issues to be addressed during the design of an efficient and robust code are raised, discussed and stated in [4] to be:

1. the control of error - whether or not to track discontinuities,
2. the choice of dense-output,

3. possible difficulties in solving vanishing lag DDEs, and
4. correct simulation of the qualitative behaviour of the solution.

DDE solvers frequently rely upon a robust ODE solver with dense output. Most one-step numerical codes are based on explicit Runge-Kutta methods due to the ease with which they can be implemented.

What software is available?

Authors of [11] cite Tavernini’s package CTMS (continuous-time model simulation) as the first software package for solving DDEs that was ‘based on a firm mathematical foundation’. Many codes for the numerical integration of DDEs are now available. These include ARCHI, DDE23, DDE-STRIDE, DDVERK, DELH, DESOL, DIFSUB-DDE, DMRODE, DRLAG6, RADAR5, RETARD and SNDDLM. The scope of the DDEs for which each is applicable varies. For example ARCHI by Paul [64] solves delay differential equations, including those with time-dependent delay, and neutral differential equations. It is also designed to solve vanishing-lag DDEs and a limited class of delay-integro-differential equations. It tracks any discontinuities and includes them as mesh points. DDE23 by Shampine and Thompson [66] is written to solve DDEs with many constant delays. The reader is referred to [11] and the references therein for further details of the codes and their applications.

1.3 Small solutions: An introduction

1.3.1 What do we mean by a small solution?

Definition 1.3.1 A solution $x(t)$ is said to be small if $\lim_{t \rightarrow \infty} e^{kt}x(t) = 0$, for all $k \in \mathbb{R}$ [22, 25, 34, 41].

The zero solution is called the trivial small solution. We focus our interest upon the existence, or otherwise, of non-trivial small solutions. Such solutions are not identically zero but approach zero faster than any exponential function. Cao in [17] refers to a *superexponential solution* as a small solution which is not identically zero for all large t . Using this definition we may regard the set of superexponential solutions as a subset of the set of small solutions.

The decay rate of a solution

Cao in [16, 17] considers the equation $\dot{x}(t) = f(x(t), x(t-1), t)$. If x is a solution of the equation defined for all $t \in [-1, \infty)$ then the exponential decay rate of x , $\bar{\alpha}(x)$ is defined by $\bar{\alpha}(x) = \inf\{\alpha \leq 0 \text{ such that } \lim_{t \rightarrow \infty} \|x(t)\|e^{-\alpha t} = 0\}$. When

$\bar{\alpha}(x) = -\infty$ then x_t is a small solution. If x_t is not identically zero on any interval of length one then it is also referred to as a superexponential solution (see also [53]).

We illustrate the concept of a small solution with the following examples:

Example 1.3.1 (From [75]) The ODE $\dot{x} = -2tx(t)$ admits the small solution $x(t) = e^{-t^2}$.

Example 1.3.2 The DDE $x'(t) = -2ate^{a(1-2t)}x(t-1)$ admits the small solution $x(t) = e^{-at^2}$ on $[-1, \infty)$, since $x'(t) = -2ate^{-at^2} = -2ate^{-a(t-1)^2+a(1-2t)} = -2ate^{a(1-2t)}x(t-1)$.

Remark: By a similar argument we can show that $x'(t) = -3kt^2e^{k(-3t^2+3t-1)}x(t-1)$ admits small solutions of the form $x(t) = e^{-kt^3}$ and that, more generally, other equations can be formed which admit small solutions of the form $x(t) = e^{-kt^n}$.

Example 1.3.3 The DDE $x'(t) = \frac{(b-2at-2bt^2)}{(a+bt-b)}e^{(1-2t)}x(t-1)$, $t \neq \left(\frac{b-a}{b}\right)$, admits the small solution $x(t) = (a+bt)e^{-t^2}$.

Remark 1.3.1 We note that alternative uses of the term *small solution* can be found in the literature and include the following two illustrative examples.

Example 1.3.4 Let a, b, c be squarefree (numbers that do not have any repeated prime factors) and pairwise relatively prime.

For the Legendre equation given in normal form, defined by $ax^2 + by^2 - cz^2 = 0$, a solution is called *small* in [19] if it satisfies Holzer's bound, namely $|x| \leq \sqrt{bc}$, $|y| \leq \sqrt{ac}$ and $|z| \leq \sqrt{ab}$.

Example 1.3.5 In [42] a *small solution* of the second order differential equation $x''(t) + a^2x(t) = 0$ with random coefficients is defined as a function $t \rightarrow x_0(t)$ satisfying the equation and such that $\lim_{t \rightarrow \infty} x_0(t) = 0$.

The definition in example 1.3.4 is clearly different from the definition of a small solution given in Definition 1.3.1 and adopted throughout our work. The definition in example 1.3.5 differs in that it refers to a second order ODE and does not involve a solution that decays to zero faster than any exponential.

1.3.2 What is known about small solutions?

The existence of small solutions depends upon specific properties of the coefficients and the detection of them for general delay differential equations is difficult [34].

We can view the presence of small solutions with reference to the completeness, or otherwise, of the eigenvectors and generalised eigenvectors of the solution

map. When an equation does not admit small solutions the eigenvalues and generalised eigenvectors span the solution space. However, for equations admitting small solutions this is not the case [41, 69, 73]. We present further details about the connection between small solutions and completeness in section 2.5. Using a conventional approach (such as seeking an expansion in terms of eigenfunctions and generalised eigenfunctions) to understand the behaviour of the solution to an equation admitting small solutions will fail. Some aspects of the behaviour of the true solutions are lost, possibly leading to misleading conclusions. For further details see [33, 34, 70, 71].

The possible existence of nontrivial small solutions is important because it is a truly infinite dimensional concept. In later sections we will analyse delay differential equations using a finite dimensional approximation, in which small solutions do not occur. We are thus using a finite dimensional approximation to attempt to identify an infinite dimensional property, namely that of possessing small solutions.

1.3.3 Why is their detection important?

The detection of small solutions, when present, is a key tool for the mathematical analyst (see [25, 41, 43, 70, 71]). In the theory of control for linear functional differential equations the theory of existence of small solutions has important applications [41]. Questions about the existence, or otherwise, of small solutions play an important role in the qualitative theory of functional differential equations (see [21] and the references therein for three examples, and [33]).

Mallet-Paret in [59] refers to the existence or non-existence of superexponential solutions as ‘perhaps the most challenging of issues which are peculiar to infinite dimensional systems’. He indicates some potential problems for the analyst working with delay differential equations which may admit superexponential solutions. These include [59]:

1. The existence of a uniform lower bound on the $|\mu|$ in solutions satisfying $x(t + T) = \mu x(t)$ to delay differential systems in feedback form is related to questions involving superexponential solutions.
2. Obtaining certain uniform bounds on the exponent α in solutions of the form $x(t) = e^{\alpha t} q(t)$ in the general time-periodic case is related to the occurrence of superexponential solutions.

Alboth in [1] states that “another important reason for the study of small solutions” is that, unless the semigroup generator T in $x' = Tx$, $x(0) = x_0$, generates a group, then the backward equation $x' = -Tx$, $x(0) = x_0$, is not well-posed for all x_0 . The components of the solution which are small give rise

to transient behaviour making it impossible to reconstruct the history after the transient behaviour has vanished.

The non-existence of small solutions plays a crucial role in parameter identifiability [77]. Under the assumption of perfect data parameter identifiability questions whether knowledge about certain solutions enables the parameters of a specific model to be identified. Relating to Theorem 2.1 in [76], we find “an important ingredient in the proof is a result about the completeness of the set of eigenvectors and generalised eigenvectors ...”. Theorems 2.1 and 2.8 in [78] both involve the assumption that an operator has a complete set of eigenvectors and generalised eigenvectors, that is, an assumption that the equation does not admit small solutions. Verduyn Lunel in [76] states that if the condition $E(\det \Delta_m(z)) = nh$ is omitted then “no information is obtained about the unknown parameters”. (This condition is equivalent to saying that the equation does not admit small solutions - see section 2.5). The assumption that $E(\det \Delta(z) = nh$ can also be found in, for example, Theorem 4.1 and Lemma 4.1 in [76] and Theorem 3.2 in [78].

In [69] we find ‘.. in order to control the behaviour of all solutions one needs completeness of the system of eigenfunctions and generalised eigenfunctions’.

Fiagbedzi in [26] considers the state delayed system $\dot{x}(t) = A_0x(t) + A_1x_1(t - r) + B_0u(t), x_0 = \phi$ and constructs a finite-dimensional system which, *in the absence of small solutions* to $\dot{q}(t) = A_0q(t) + A_1q(t - r)$, will “replicate exactly the response of the state-delayed system”.

The afore-mentioned quotes from, and reference to, current literature emphasise the importance of being able to detect small solutions, providing evidence that research in this area is of genuine practical and theoretical interest.

1.4 Outline of the thesis

Detecting small solutions is the focus of our work. We concentrate on delay differential equations with constant delays, that is, on equations of the form

$$(1.8) \quad y'(t) = f(t, y(t), y(t - \tau_1), y(t - \tau_2), \dots).$$

In chapter 2 we include elements of both matrix theory and operator theory that are relevant to the research presented in this thesis. We refer to the adaptation of numerical methods for ODEs to DDEs, briefly indicate problems encountered and refer to current codes specifically written for DDEs. We state results concerning stability of the solutions of DDEs and of the numerical methods used to solve DDEs. An illustrative example from the field of immunology is included. In section 2.5, following the introduction to the concept of a small solution in section 1.3, we outline further known theory relating to small solutions. We state results

about small solutions which arise out of Laplace Transform methods and/or from the application of operator theory. In chapter 3 we introduce the methodology that underpins our work.

We begin our own investigations by considering the one dimensional problem represented by the equation $x'(t) = b(t)x(t-1), t \geq 0; \quad x(\theta) = \phi(\theta), -1 \leq \theta \leq 0$. In fact, chapters 4 to 11 all contain original work. In chapter 4 we demonstrate our successful detection of small solutions to this equation using the trapezium rule as our numerical method. In chapter 5 we justify our choice of the trapezium rule. We apply several different numerical methods to the same one-dimensional problems and compare the ease and clarity with which small solutions can be detected.

In chapter 6 we move on to consider the detection of small solutions for higher dimensional systems of DDEs.

DDEs with multiple delays are the focus of our attention in chapter 7. We begin by adopting the approach used in earlier chapters directly. We then consider a more sophisticated approach using Floquet solutions which, as we demonstrate, leads to a significant reduction in the computational time needed.

In chapter 8 we consider DDEs in which the delay and period are commensurate and include an example of a three-dimensional case.

In each of chapters 4 to 8 and 11 we demonstrate successful detection of small solutions using numerical discretisation in accordance with known theory, with a view to gaining insight into the detection of small solutions in cases where the analytical theory is less well developed. Known analytical results that refer to the existence, or otherwise, of small solutions for the class of equations under consideration are stated, with references to literature where the reader can find further details.

In chapter 9 we consider the use of statistics to detect the presence of small solutions. This novel approach led to the development of an algorithm, ‘Small-solutiondetector1’, to automate the detection of small solutions to a particular class of DDE. Details of the algorithm and the underlying methodology are presented in chapter 10. We include illustrative examples, consider its reliability and extend the algorithm to the class of multi-delay differential equations considered in chapter 7. In addition we indicate the possibility of adapting our algorithm to other classes of DDE.

Chapter 11 returns to one-dimensional problems but considers the case when $b(t)$ is a complex-valued function. Published theory relating to this case is less readily available. A result concerning the instability of the trapezium rule for this case encourages us to consider an alternative numerical method. We compare the results of applying both the trapezium rule and the backward Euler method to several problems and begin to develop an insight into this case using the

approach developed in earlier chapters.

In chapter 12 we summarise our results and present our conclusions. Finally, in chapter 13 we indicate some potential questions that we can consider in future research in this area.

Conference presentations and publications

This thesis contains material which has been the subject of conference and journal papers, the details of which are given below:

1. Some of the material from chapters 4 and 5 was presented at a seminar day on problems with memory and after-effect, organised by the MCCM.
2. Material from chapters 4 and 5 was presented at the HERCMA conference, Athens 2001 and appeared in the proceedings [28].
3. [29] is based on material from chapter 6 and a related presentation was given at Algorithms for Approximation IV, Huddersfield, 2001.
4. The material from chapter 7 forms the basis for the paper [30] which has been submitted for publication.
5. Part of the material in chapter 8 was presented at the Conference on Scientific Computation, Geneva 2002, and a paper relating to this chapter is in preparation [32].
6. Material from chapters 9 and 10 was presented at the 20th Biennial Conference on Numerical Analysis, Dundee 2003, and a paper has been submitted for publication.

Chapter 2

Background theory and information

2.1 Introduction and background theory

In later chapters of this thesis we consider the following classes of equations:

- the one-dimensional equation $x'(t) = b(t)x(t - \tau), b(t + \tau) = b(t)$, where $b(t)$ is a real or complex-valued function,
- the multi-delay equation $x'(t) = \sum_{i=0}^n b_i(t)x(t - \tau_i)$,
- the one-dimensional equation $x'(t) = b(t)x(t - \tau), b(t + p) = b(t)$ with p and τ commensurate,
- the system $y'(t) = A(t)y(t - 1), A(t + 1) = A(t)$.

In this chapter we include known theory relevant to, and underpinning, the research presented in this thesis. Several analytical results concerning small solutions are stated using exponential type calculus. Hence, to assist the reader we include an introduction to this terminology in section 2.1.1. An introduction to relevant elements of operator theory is included in section 2.1.2 and we develop some results using matrix theory in section 2.1.3. We review stability criteria for the equations considered in the thesis in section 2.3 and state results concerning the stability of numerical methods employed in section 2.4.1. In section 2.5 we present further known background theory and results concerning small solutions, particularly those relating to autonomous problems. Known analytical results for non-autonomous problems will be introduced in the relevant chapter. Section 2.6 introduces an example involving parameter estimation, from the field of mathematical immunology, where the non-existence of small solutions is important. In section 3.1 we introduce the methodology behind our numerical detection of small solutions.

2.1.1 Exponential type calculus

Let X be a complex Banach space and let $F : \mathbb{C} \rightarrow X$ be an entire function.

Let

$$M(r) = \max_{0 \leq \theta \leq 2\pi} \{|F(re^{i\theta})|\}.$$

F is of order ρ if and only if

$$\limsup_{r \rightarrow \infty} \frac{\log \log M(r)}{\log r} = \rho.$$

An entire function of order at most 1 is of exponential type if and only if

$$\limsup_{r \rightarrow \infty} \frac{\log M(r)}{r} = E(F),$$

where $0 \leq E(F) < \infty$ (see [22, 41, 69, 72]). $E(F)$ is called the exponential type of F . If F is a vector-valued function, say $F = (f_1, f_2, \dots, f_n) : \mathbb{C} \rightarrow \mathbb{C}^n$, then, provided that the components f_j are entire functions of order 1 that are of exponential type, the exponential type of F is defined by $E(F) = \max_{1 \leq j \leq n} E(f_j)$ [41, 71]. We illustrate the case when F is a scalar-valued function with the following examples.

Example 2.1.1 Let $F(x) = e^{3x}$.

$$M(r) = \max_{0 \leq \theta \leq 2\pi} \{|e^{3re^{i\theta}}|\} = e^{3r}.$$

$$\limsup_{r \rightarrow \infty} \frac{\log(\log e^{3r})}{\log r} = \limsup_{r \rightarrow \infty} \frac{\log(3r)}{\log r} = \limsup_{r \rightarrow \infty} \left\{ \frac{\log 3}{\log r} + 1 \right\} = 1$$

Hence $F(x)$ is of order 1.

$$\limsup_{r \rightarrow \infty} \frac{\log M(r)}{r} = \limsup_{r \rightarrow \infty} \left(\frac{3r}{r} \right) = 3.$$

Hence $F(x)$ is of exponential type 3.

Example 2.1.2 Let $F(x) = e^{x^2}$. $F(re^{i\theta}) = e^{r^2 e^{2i\theta}}$. $M(r) = e^{r^2}$.

Hence,

$$\limsup_{r \rightarrow \infty} \frac{\log(\log e^{r^2})}{\log r} = \limsup_{r \rightarrow \infty} \frac{\log(r^2)}{\log r} = 2.$$

Hence $F(x)$ is not of order 1.

$$\limsup_{r \rightarrow \infty} \frac{\log M(r)}{r} = \limsup_{r \rightarrow \infty} \left(\frac{r^2}{r} \right)$$

which is infinite.

2.1.2 Operator theory: A C_0 -semigroup

Let $X = C([-h, 0], \mathbb{C})$ provided with the supremum-norm. We adopt the standard notation $x_t(\theta) := x(t + \theta)$ for $t \geq 0$ and $-h \leq \theta \leq 0$, so that $x_t \in X$ is the state at time t . When the solution $x(t)$ depends upon the initial function ϕ we adopt the notation $x = x(\cdot; \phi)$.

Let $T = \{T(t)\}_{t \geq 0}$ be a family of bounded linear operators on a Banach space X . A C_0 -semigroup generated by a bounded operator A is an exponential operator-function

$$e^{At} = \sum_{k=1}^{\infty} \frac{A^k t^k}{k!}$$

(see [61] for example). The properties of a strongly continuous semi-group of operators (a C_0 -semigroup) are given as (see for example [22, 51, 72]):

1. $T(0) = I$ (the identity)
2. $T(t)T(s) = T(t + s)$ for $t, s \geq 0$
3. for any $\phi \in X$, $\|T(t)\phi - \phi\| \rightarrow 0$ as $t \downarrow 0$

The abstract differential equation $\frac{d}{dt}(T(t)\phi) = A(T(t)\phi)$ can be associated with such a semi-group.

By definition,

$$A\phi = \lim_{t \downarrow 0} \frac{1}{t}(T(t)\phi - \phi) \text{ for every } \phi \in D(A)$$

with

$$D(A) = \left\{ \phi \mid \lim_{t \downarrow 0} \frac{1}{t}(T(t)\phi - \phi) \text{ exists} \right\}.$$

A is a linear operator and is called the infinitesimal generator of the semi-group $\{T(t)\}$. By definition A is the derivative at $t = 0$ [22] (see also Definition 1.1.2 in [61]). Further details about C_0 -semigroups can be found in, for example, [22, 41, 61].

Example 2.1.3 Consider the scalar equation

$$\begin{aligned} \dot{x}(t) &= 0 \text{ for } t \geq 0 \\ x(\theta) &= \phi(\theta) \text{ for } -h \leq \theta \leq 0. \end{aligned}$$

For $t \geq 0$ we define the bounded linear operator $T_0(t) : X \rightarrow X$ by:

$$T_0(t)\phi(\theta) = \begin{cases} \phi(t + \theta) & \text{if } -h \leq t + \theta \leq 0 \\ \phi(0) & \text{if } t + \theta \geq 0. \end{cases}$$

Hence, $T_0(t)$ maps the initial state ϕ at time zero onto the state x_t at time t (see [22]).

T_0 as defined above is a C_0 -semigroup, with generator given by

$$\begin{aligned} D(A_0) &= \{\phi | \dot{\phi} \in C([-h, 0], \mathbb{C}), \dot{\phi}(0) = 0\}, \\ A_0\phi &= \dot{\phi}. \end{aligned}$$

Example 2.1.4 (Example 1.5 in [51]) The infinitesimal generator associated with

$$\dot{x}(t) = x(t) - x(t-1), t \geq 0$$

is given by

$$\begin{aligned} D(A) &= \{\phi \in C[-1, 0] : \phi \in C^1[-1, 0], \dot{\phi}(0) = \phi(0) - \phi(-1)\} \\ A\phi &= \dot{\phi}. \end{aligned}$$

The scalar function $\Delta(z) = z - 1 + e^{-z}$ is a characteristic matrix for A . (see Theorem 1.2 in [51]).

Remark 2.1.1 Let E be a Banach space and let $(e^{tT})_{t \geq 0}$ be a C_0 -semigroup of operators such that $\|e^{tT}\| \leq Me^{\omega_0 t}$. $E_\infty(T)$ denotes the set of small solutions by $E_\infty(T)$. Proposition 1 in [1] asserts that (i) $E_\infty(T)$ is invariant under e^{tT} for $t \geq 0$ and (ii) $\dim E_\infty(T) = 0$ or $\dim E_\infty(T) = \infty$.

Completeness

The operator A has a complete span of eigenvectors and generalised eigenvectors if the linear space spanned by all eigenvectors and generalised eigenvectors is dense in C . In this case each solution can be approximated by a linear combination of elementary solutions [76].

2.1.3 Relevant matrix theory

Eigenvalues of a matrix

Consider the matrix A such that $A = \begin{pmatrix} \alpha(t) & \beta(t) \\ \gamma(t) & \delta(t) \end{pmatrix}$.

The eigenvalues of $A(t)$ are the values of λ such that $\begin{vmatrix} \alpha(t) - \lambda & \beta(t) \\ \gamma(t) & \delta(t) - \lambda \end{vmatrix} = 0$.

This gives

$$(2.1) \quad \lambda^2 - [\alpha(t) + \delta(t)]\lambda + [\alpha(t)\delta(t) - \beta(t)\gamma(t)] = 0.$$

which can be expressed in the form

$$(2.2) \quad \lambda^2 - [Tr(A(t))]\lambda + |A(t)| = 0.$$

From (2.2) we can see that $A(t)$ has real eigenvalues if $[Tr(A(t))]^2 - 4|A(t)| \geq 0$, or, alternatively, if $[\alpha(t) - \delta(t)]^2 + 4\beta(t)\gamma(t) \geq 0$. The roots of (2.2) are complex with real part equal to zero if $Tr(A(t)) = \alpha(t) + \delta(t) = 0$, $|A(t)| > 0$ and $[Tr(A(t))]^2 - 4|A(t)| < 0$.

The characteristic polynomial of the $n \times n$ matrix A is defined by the degree n polynomial $\rho_A(z) = \det(zI - A)$ and λ is an eigenvalue if and only if $\rho_A(\lambda) = 0$. Hence, if $\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_n$ are the n eigenvalues of A then

$$\rho_A(z) = (z - \lambda_1)(z - \lambda_2)(z - \lambda_3)\dots\dots\dots(z - \lambda_n).$$

The set of these roots is called the spectrum of A , denoted by $\lambda(A)$. We note that $\det(A) = |A| = \prod_{j=1}^n \lambda_j$ and $Tr(A) = \sum_{j=1}^n \lambda_j$ (see [36]).

What is a companion matrix?

If $Ax = \lambda x$ then the characteristic equation of the matrix A is given by

$$(-1)^n \lambda^n - p_{n-1} \lambda^{n-1} - p_{n-2} \lambda^{n-2} - \dots - p_0 = 0$$

where the left-hand side of the equation is the characteristic polynomial of A .

The characteristic equation of the matrix C ,

where $C = \begin{pmatrix} p_{n-1} & p_{n-2} & \dots & \dots & p_1 & p_0 \\ 1 & 0 & \dots & \dots & 0 & 0 \\ 0 & 1 & \ddots & & \vdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & & \ddots & \ddots & 0 & \vdots \\ 0 & \dots & \dots & 0 & 1 & 0 \end{pmatrix}$, can be shown to be identical to

the characteristic equation of A . C is called the companion matrix of the characteristic polynomial of A (see, for example, [81]).

Some preliminary results

Here we establish some results that will be used in sections 6.1.1 and 6.2.1.

We begin by defining four different matrix forms.

Definition 2.1.1 Let P, Q, F and $G \in \mathbb{R}^{(n+1) \times (n+1)}$. Let $p(t), q(t), g(t)$ and $f(t)$ be continuous functions and write $p_n = p(nh), q_n = q(nh), f_n = f(nh), g_n = g(nh)$.

Define P, Q, F and G as follows:

$$P(p_k) = \begin{pmatrix} 1 & 0 & \cdots & 0 & \frac{h}{2}p_{k+1} & \frac{h}{2}p_k \\ 1 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & 1 & \ddots & & & \vdots \\ \vdots & \ddots & 1 & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & 0 & 1 & 0 \end{pmatrix},$$

$$Q(q_k) = \begin{pmatrix} 0 & \cdots & \cdots & 0 & \frac{h}{2}q_{k+1} & \frac{h}{2}q_k \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ \vdots & & & & & \vdots \\ \vdots & & & & & \vdots \\ \vdots & & & & & \vdots \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 \end{pmatrix}.$$

For $k = 1, 2, \dots, n-1$

$$G(g_k) = \begin{pmatrix} 1 & 0 & \cdots & 0 & \frac{h}{2}g_{k+1} & hg_k & hg_{k-1} & \cdots & \cdots & hg_2 & \frac{h}{2}g_1 \\ 1 & 0 & \cdots & \cdots & 0 & \frac{h}{2}g_k & hg_{k-1} & \cdots & \cdots & hg_2 & \frac{h}{2}g_1 \\ \vdots & \vdots & & & & \ddots & \frac{h}{2}g_{k-1} & \ddots & & \vdots & \vdots \\ \vdots & \vdots & & & & \ddots & \ddots & \ddots & & \vdots & \vdots \\ \vdots & \vdots & & & & & & \ddots & \ddots & hg_2 & \frac{h}{2}g_1 \\ \vdots & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 & \frac{h}{2}g_2 & \frac{h}{2}g_1 \\ 1 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & 1 & \ddots & & & & & & & & \vdots \\ \vdots & \ddots & \ddots & \ddots & & & & & & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & & & & & & \vdots \\ 0 & \cdots & \cdots & 0 & 1 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \end{pmatrix}.$$

For $k = n$

$$G(g_k) = \begin{pmatrix} 1 + \frac{h}{2}g_{k+1} & hg_k & hg_{k-1} & \cdots & \cdots & hg_2 & \frac{h}{2}g_1 \\ 1 & \frac{h}{2}g_k & hg_{k-1} & \cdots & \cdots & hg_2 & \vdots \\ 1 & 0 & \frac{h}{2}g_{k-1} & \ddots & & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & & \ddots & \ddots & hg_2 & \vdots \\ \vdots & \vdots & & & \ddots & \frac{h}{2}g_2 & \frac{h}{2}g_1 \\ 1 & 0 & \cdots & \cdots & \cdots & 0 & 0 \end{pmatrix}.$$

For $k = 1, 2, \dots, n$

$$F(f_k) = \begin{pmatrix} 0 & \cdots & \cdots & 0 & \frac{h}{2}f_{k+1} & hf_k & hf_{k-1} & \cdots & \cdots & hf_2 & \frac{h}{2}f_1 \\ \vdots & & & & 0 & \frac{h}{2}f_k & hf_{k-1} & \cdots & \cdots & hf_2 & \frac{h}{2}f_1 \\ \vdots & & & & & \ddots & \frac{h}{2}f_{k-1} & \ddots & & \vdots & \vdots \\ \vdots & & & & & & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & & & & & & & \ddots & \ddots & hf_2 & \frac{h}{2}f_1 \\ \vdots & & & & & & & & 0 & \frac{h}{2}f_2 & \frac{h}{2}f_1 \\ \vdots & & & & & & & & & 0 & 0 \\ \vdots & & & & & & & & & & \vdots \\ \vdots & & & & & & & & & & \vdots \\ 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \end{pmatrix}.$$

Proposition 2.1.2 is referred to in chapter 6. We begin by establishing results in proposition 2.1.1 which we will need in the proof of proposition 2.1.2, the more important proposition with relation to our future work.

Proposition 2.1.1 If P, Q, F and G are as defined in definition 2.1.1 then for $n > 1$

- (i) $Q(q_i) \times Q(q_j) = 0$ for all i and j ,
- (ii) $Q(q_i) \times F(f_j) = 0$ for all $j \leq (n - 1)$,
- (iii) $P(p_{k+1}) \times G(p_k) = G(p_{k+1})$, for $k = 1, 2, \dots, (n - 1)$,
- (iv) $Q(\gamma_{k+1}) \times G(\alpha_k) + P(\delta_{k+1}) \times F(\gamma_k) = F(\gamma_{k+1})$, for $k \leq (n - 1)$.

Proof. Result (i) follows directly from the product of the two matrices, noting that all entries in both the last two rows and the first two columns of each matrix are zero.

Result (ii) holds since all entries in the last $(n + 1 - i)$ rows of the matrix $F(f_j)$ are zero.

Result (iii) also follows easily from the matrix product. The effect of pre-multiplying $G(p_k)$ by $P(p_{k+1})$ is to displace all rows downwards by one, discard the last row, and replace the first row

$$\left(1 \ 0 \ \cdots \ \cdots \ \cdots \ 0 \ \frac{h}{2}p_{k+1} \ hp_k \ \cdots \ \cdots \ hp_2 \ \frac{h}{2}p_1 \right)$$

by

$$\left(1 \ 0 \ \cdots \ \cdots \ 0 \ \frac{h}{2}p_{k+2} \ hp_{k+1} \ hp_k \ \cdots \ \cdots \ hp_2 \ \frac{h}{2}p_1 \right).$$

We prove result (iv) as follows:

$$Q(\gamma_{k+1}) \times G(\alpha_k) = \begin{pmatrix} 0 & D_1 \\ 0 & 0 \end{pmatrix}$$

where $D_1 \in \mathbb{R}^{1 \times (k+2)}$ and $D_1 = \left(\frac{h}{2}\gamma_{k+2} \ \frac{h}{2}\gamma_{k+1} \ 0 \ \cdots \ \cdots \ 0 \right)$.

$$P(\delta_{k+1}) \times F(\gamma_k) = \begin{pmatrix} 0 & D_2 \\ 0 & 0 \end{pmatrix}$$

$$\text{where } D_2 \in \mathbb{R}^{(k+1) \times (k+1)} \text{ and } D_2 = \begin{pmatrix} \frac{h}{2}\gamma_{k+1} & h\gamma_k & \cdots & \cdots & h\gamma_2 & \frac{h}{2}\gamma_1 \\ \frac{h}{2}\gamma_{k+1} & h\gamma_k & \cdots & \cdots & h\gamma_2 & \frac{h}{2}\gamma_1 \\ 0 & \frac{h}{2}\gamma_k & \ddots & & \vdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & & \ddots & \ddots & h\gamma_2 & \vdots \\ 0 & \cdots & \cdots & 0 & \frac{h}{2}\gamma_2 & \frac{h}{2}\gamma_1 \end{pmatrix}.$$

Hence

$$Q(\gamma_{k+1}) \times G(\alpha_k) + P(\delta_{k+1}) \times F(\gamma_k) = \begin{pmatrix} 0 & D_3 \\ 0 & 0 \end{pmatrix} = F(\gamma_{k+1}),$$

where $D_3 \in \mathbb{R}^{(k+1) \times (k+2)}$

$$\text{and } D_3 = \begin{pmatrix} \frac{h}{2}\gamma_{k+2} & h\gamma_{k+1} & \cdots & \cdots & \cdots & h\gamma_2 & \frac{h}{2}\gamma_1 \\ 0 & \frac{h}{2}\gamma_{k+1} & h\gamma_k & \cdots & \cdots & h\gamma_2 & \frac{h}{2}\gamma_1 \\ \vdots & \ddots & \ddots & \ddots & & \vdots & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & & & \ddots & \ddots & h\gamma_2 & \vdots \\ 0 & \cdots & \cdots & \cdots & 0 & \frac{h}{2}\gamma_2 & \frac{h}{2}\gamma_1 \end{pmatrix}.$$

□

Proposition 2.1.2 Let $C_n = \prod_{i=1}^n A(i)$ where $A(i) = \begin{pmatrix} P(\alpha_i) & Q(\beta_i) \\ Q(\gamma_i) & P(\delta_i) \end{pmatrix}$. The $(2n+2) \times (2n+2)$ matrix C_n can be considered as four $(n+1) \times (n+1)$ blocks in a 2×2 formation and there is no *pollution* of the blocks from the neighbouring functions.

Proof. For $k = 1, 2, \dots, n-1$ let $C_k = \prod_{i=1}^k A(i) = \begin{pmatrix} G(\alpha_k) & F(\beta_k) \\ F(\gamma_k) & G(\delta_k) \end{pmatrix}$.

$$C_2 = A(2) \cdot A(1) = \begin{pmatrix} P(\alpha_2) & Q(\beta_2) \\ Q(\gamma_2) & P(\delta_2) \end{pmatrix} \begin{pmatrix} P(\alpha_1) & Q(\beta_1) \\ Q(\gamma_1) & P(\delta_1) \end{pmatrix}$$

which, using block matrix operations, gives

$$C_2 = \begin{pmatrix} P(\alpha_2)P(\alpha_1) + Q(\beta_2)Q(\gamma_1) & P(\alpha_2)Q(\beta_1) + Q(\beta_2)P(\delta_1) \\ Q(\gamma_2)P(\alpha_1) + P(\delta_2)Q(\gamma_1) & Q(\gamma_2)Q(\beta_1) + P(\delta_2)P(\delta_1) \end{pmatrix}.$$

Using result (i) from proposition 2.1.1

$$Q(\beta_2)Q(\gamma_1) = 0 = Q(\gamma_2)Q(\beta_1).$$

Since $P(g_1) = G(g_1)$ using result (iii) from proposition 2.1.1 gives

$$P(\alpha_2)P(\alpha_1) = P(\alpha_2)G(\alpha_1) = G(\alpha_2).$$

Similarly,

$$P(\delta_2)P(\delta_1) = G(\delta_2).$$

$$Q(\gamma_2)P(\alpha_1) = \begin{pmatrix} 0 & D_4 \\ 0 & 0 \end{pmatrix} \text{ where } D_4 \in \mathbb{R}^{1 \times 3} \text{ and } D_4 = \begin{pmatrix} \frac{h}{2}\gamma_3 & \frac{h}{2}\gamma_2 & 0 \end{pmatrix}.$$

$$P(\delta_2)Q(\gamma_1) = \begin{pmatrix} 0 & D_5 \\ 0 & 0 \end{pmatrix}, \text{ where } D_5 \in \mathbb{R}^{2 \times 2} \text{ and } D_5 = \begin{pmatrix} \frac{h}{2}\gamma_2 & \frac{h}{2}\gamma_1 \\ \frac{h}{2}\gamma_2 & \frac{h}{2}\gamma_1 \end{pmatrix}.$$

Hence

$$Q(\gamma_2)P(\alpha_1) + P(\delta_2)Q(\gamma_1) = \begin{pmatrix} 0 & D_6 \\ 0 & 0 \end{pmatrix} = F(\gamma_2),$$

where

$$D_6 \in \mathbb{R}^{2 \times 3} \text{ and } D_6 = \begin{pmatrix} \frac{h}{2}\gamma_3 & h\gamma_2 & \frac{h}{2}\gamma_1 \\ 0 & \frac{h}{2}\gamma_2 & \frac{h}{2}\gamma_1 \end{pmatrix}.$$

Similarly, we can show that

$$P(\alpha_2)Q(\beta_1) + Q(\beta_2)P(\delta_1) = F(\beta_2).$$

Hence

$$C_2 = \begin{pmatrix} G(\alpha_2) & F(\beta_2) \\ F(\gamma_2) & G(\delta_2) \end{pmatrix}.$$

There is clearly no pollution in $A(1)$. We have shown that there is no pollution of blocks from neighbouring functions resulting from the product of the first two matrices $A(2)$ and $A(1)$. Hence there is no pollution in C_k for $k = 1, 2$.

We now assume that there is no pollution from neighbouring functions for the product of the first k matrices.

Hence we assume that $C_k = \begin{pmatrix} G(\alpha_k) & F(\beta_k) \\ F(\gamma_k) & G(\delta_k) \end{pmatrix}$ and consider the product of the first $(k+1)$ matrices.

$$\begin{aligned} A(k+1).C_k &= \begin{pmatrix} P(\alpha_{k+1}) & Q(\beta_{k+1}) \\ Q(\gamma_{k+1}) & P(\delta_{k+1}) \end{pmatrix} \begin{pmatrix} G(\alpha_k) & F(\beta_k) \\ F(\gamma_k) & G(\delta_k) \end{pmatrix} \\ &= \begin{pmatrix} P(\alpha_{k+1})G(\alpha_k) + Q(\beta_{k+1})F(\gamma_k) & P(\alpha_{k+1})F(\beta_k) + Q(\beta_{k+1})G(\delta_k) \\ Q(\gamma_{k+1})G(\alpha_k) + P(\delta_{k+1})F(\gamma_k) & Q(\gamma_{k+1})G(\beta_k) + P(\delta_{k+1})G(\delta_k) \end{pmatrix} \end{aligned}$$

Using results (ii), (iii) and (iv) from proposition 2.1.1 gives

$$C_{k+1} = \begin{pmatrix} G(\alpha_{k+1}) & F(\beta_{k+1}) \\ F(\gamma_{k+1}) & G(\delta_{k+1}) \end{pmatrix}.$$

Hence, if there is no pollution in the product of k matrices then there is no pollution in the product of $k+1$ matrices. Since there is no pollution for $k = 1, 2$ then the result is also true for $k = 3, 4, \dots, (n-1)$ and the proposition is proven. \square

2.2 Different approaches to the theory of DDEs

The theory relating to linear autonomous DDEs can be developed using Laplace transform theory. Laplace transforms cannot be used for non-linear systems [22]. Hence, to further the development of the theory of DDEs, an alternative approach was needed. We begin by considering possible approaches to linear autonomous equations and then move on, in section 2.2.2, to outline the functional analytic approach, an approach that can be used with autonomous equations but which has a much wider application. In section 2.2.3 we make reference to an algebraic approach.

2.2.1 Linear autonomous equations

Characteristic equations

For a system of n linear homogeneous ODEs with constant coefficients there are n linearly independent solutions and the general solution can be expressed as a linear combination of these n solutions [12]. If we adopt a similar approach with the system

$$(2.3) \quad y'(t) = \sum_{j=1}^m A_j y(t - \tau_j)$$

then we seek exponential solutions (elementary solutions) of the form $y(t) = ce^{\lambda t}$, where c is constant. This leads to an equation of the form

$$(2.4) \quad (\lambda I - \sum_{j=1}^m A_j e^{-\lambda \tau_j})c = 0.$$

which has non-zero solutions if and only if λ satisfies the characteristic equation,

$$(2.5) \quad \det(\lambda I - \sum_{j=1}^m A_j e^{-\lambda \tau_j}) = 0.$$

We include the following examples for illustration.

Example 2.2.1 The characteristic equation for $\dot{y}(t) = Ay(t) + By(t - \tau)$ is

$$\lambda - A - Be^{-\lambda \tau} = 0.$$

Example 2.2.2 The characteristic equation for $\dot{x}(t) = b_0x(t) + b_1x(t - 1) + b_2x(t - 2)$ is given by $\lambda - b_0 - b_1e^{-\lambda} - b_2e^{-2\lambda} = 0$.

Example 2.2.3 Consider the equation $x'(t) = b_0x(t) + b_1x(t - 1)$.

The characteristic polynomial is given by $h(\lambda) = \lambda - b_0 - b_1e^{-\lambda}$.

The zeros of this polynomial lie asymptotically along the curve given by

$$\Re(\lambda) + \log |\lambda| = \log |b_1|.$$

Figure 4.1 in [12] indicates the general appearance of the curve.

For finite delays the characteristic equations are functions of delays and hence the roots of the characteristic equation are functions of delays. Stability of the trivial solution depends upon location of the roots of the associated characteristic equation [55]. The steady state solution will be asymptotically stable if all the zeros of the characteristic equation have strict negative real part [24]. A numerical algorithm to compute the right-most zero of the characteristic equation is presented in [24]. A change in the length of the delay can lead to a change in the stability of the trivial solution, a phenomenon known as a stability switch [55]. Reliance on locating zeros of the characteristic function is a step in proofs of fundamental theorems on expansions in series of exponentials. The nature of the solution (for large t) is closely related to the distribution of the characteristic roots (see [12]).

In general, equation (2.5) has infinitely many complex roots, each of which has a certain multiplicity. Linear combinations of the exponential solutions are also solutions of equation (2.4) provided that the series converges and admits term-by-term differentiation [12, 23, 41].

The Laplace transform approach

We illustrate the Laplace transform approach using the equation

$$(2.6) \quad \dot{y}(t) = B_0 y(t) + B_1 y(t-1), \quad t \geq 0.$$

We define $\mathcal{L}(y)$ to be the Laplace transform of the function y , that is

$$\mathcal{L}(y)(z) = \int_0^{\infty} e^{-zt} y(t) dt.$$

Applying the Laplace transform to (2.6), with initial data $y(\theta) = \phi(\theta)$ for $-1 \leq \theta \leq 0$, gives

$$\Delta(z)\mathcal{L}(y)(z) = \phi(0) + B_1 \int_0^1 e^{-zt} \phi(t-1) dt,$$

where $\Delta(z)$, the characteristic matrix, is given by

$$(2.7) \quad \Delta(z) = zI - B_0 - B_1 e^{-z}.$$

An explicit representation of y is sought using the inverse Laplace transform, the Cauchy theorem and a residue calculus (see [22, 77] for further details and references). Theory relating to characteristic equations of the type (2.7) is readily found in the literature [12]. We include the following lemma.

Lemma 2.2.1 (Lemma 1.1 from [77]. See also Theorem 4.4 in [22])
The roots of the transcendental equation

$$(2.8) \quad \det \Delta(z) = \det(zI - B_0 - B_1 e^{-z}) = 0$$

have the following properties:

- (i) There exists a right half-plane $\{z \in \mathbb{C} | \operatorname{Re}(z) > \gamma_0\}$ without roots of (2.8).
- (ii) The number of roots of (2.8) on a given strip $\{z \in \mathbb{C} | \gamma_- < \operatorname{Re}(z) \leq \gamma_+\}$ is finite.
- (iii) The roots of (2.8) in the left half-plane, necessarily satisfy

$$|\operatorname{Im}(z)| \leq C e^{-\operatorname{Re}(z)},$$

where C is a constant determined by B_0 and B_1 .

Theorem 1.2 in [77] concerns the asymptotic expansion of the solution of (2.6) in the form

$$y(t) = \sum_{j=1}^m p_j(t) e^{\lambda_j t} + O(e^{\gamma t}) \text{ for } t \rightarrow \infty$$

which includes a term of $O(e^{\gamma t})$. Questions concerning the convergence of the infinite series as $\gamma \rightarrow -\infty$ and the possible existence of small solutions are raised (see [77] for details).

2.2.2 The functional analytic approach

In the functional analytic approach we analyse solution operators acting on function spaces of initial data. We associate with the equation a semi-flow, defined by the time evolution of segments of solutions, acting on the space of initial data. We denote by C the Banach space of continuous functions defined on the interval $[-1, 0]$ with values in \mathbb{C}^n , provided with the sup-norm, $\|\phi\| := \sup_{-1 \leq \theta \leq 0} |\phi(\theta)|$ for $\phi \in C$. If we denote the solution of $\dot{x}(t) = F(x(t), x(t - \tau))$, $t \geq 0$; $x(\theta) = \phi(\theta)$, $-1 \leq \theta \leq 0$ by $x(\cdot; \phi)$ and define the state of the solution $x(\cdot; \phi)$ by $x_t(\theta; \phi) = x_t(t + \theta; \phi)$, $-1 \leq \theta \leq 0$ then the semiflow $\Sigma(t; \cdot) : C \rightarrow C$ is defined by $\Sigma(t; \phi) = x_t(\cdot; \phi)$, $t \geq 0$. (see [77])

Linear autonomous equations

Consider equation

$$(2.9) \quad \dot{x}(t) = B_0 x(t) + B_1 x(t - 1), \quad x_0 = \phi, \quad \phi \in \mathbb{C}.$$

Define the strongly continuous semi-group $T(t)$ by

$$T(t)\phi = x_t = x(t + \theta)$$

where x is the solution of (2.9). The solutions of (2.9) are in one-to-one correspondence with the solutions of the equation

$$\frac{du}{dt} = Au, \quad u(0) = \phi, \quad \phi \in \mathbb{C}$$

where $A(\mathbb{C} \rightarrow \mathbb{C})$ is the unbounded operator defined by

$$\left\{ \begin{array}{l} A\phi = \frac{d\phi}{d\theta} \\ D(A) = \{ \phi \in \mathbb{C} \mid A\phi \in \mathbb{C} \text{ and } \frac{d\phi}{d\theta}(0) = B_0\phi(0) + B_1\phi(-1) \}, \end{array} \right.$$

the correspondence being given by $u(t)(\theta) = x(t + \theta)$.

More generally, for each $\lambda \in \sigma(A)$, the spectrum of A , the eigenfunctions are elements of the nullspace $\ker(I\lambda - A)$ and are given by

$$\phi(\theta) = \phi^0 e^{\lambda\theta}, \quad \theta \in [-h, 0], \quad \Delta(\lambda)\phi^0 = 0.$$

If there are multiple eigenvalues then

$$M_\lambda = \ker(I\lambda - A)^m$$

are known as the generalised eigenfunctions of A , involving linear combinations of $\theta^k e^{\lambda\theta}$ (see [60] for example).

Let M_λ denote the generalised eigenspace corresponding to an eigenvalue λ of A .

Let M denote the linear subspace generated by M_λ , so that $M = \bigoplus_{\lambda \in \sigma(A)} M_\lambda$. M is called the generalised eigenspace of A and is the smallest subspace that contains all $M(\lambda)$, $\lambda \in \sigma(A)$.

If the closure \bar{M} of M equals the whole space X then the system of generalised eigenfunctions of A is said to be complete [71, 72, 74]. In this case then each solution of the equation can be approximated by a linear combination of elementary solutions of the form $x(t) = p(t)e^{\lambda t}$ [77].

Theorem 1.1 in [74] concerns the expansion of the state $x_t = x(t + \theta)$ into a linear combination of eigenvectors and generalised eigenvectors. Verduyn Lunel in [72] proves necessary and sufficient conditions for completeness of the system of generalised eigenfunctions of the infinitesimal generator A of a C_0 -semigroup. Manitius gives necessary and sufficient conditions for completeness of generalised eigenfunctions associated with systems of linear autonomous delay differential equations in [60]. He introduces the concept of F -completeness of the generalised eigenfunctions of A and ‘links F -completeness with the problem of “small solutions”.’

The connection between the operator A , defined as in section 2.1.2, and the matrix function Δ is described in greater detail in [51]. It is shown that they are related through an equivalence relation. Δ is called a characteristic matrix for A whenever the equivalence relation holds. The spectrum of the infinitesimal generator is given by the roots of the characteristic equation $\det \Delta(z) = 0$ (see [41]).

Periodic linear equations

We first concern ourselves with the equation

$$(2.10) \quad \dot{y}(t) = B(t)y(t-1), \quad t \geq 0,$$

where $B(t+w) = B(t)$. If the initial function $y(\theta) = \phi(\theta)$, $-1 \leq \theta \leq 0$, is given then a unique solution to (2.10) exists. From [77] we have that the evolutionary system associated with (2.10) is given by translation along the solution

$$T(t, s)\phi = x_t(s, \phi).$$

Using the periodicity of $B(t)$ we can define the period map

$$\Pi(s)\phi = T(s+w, s)\phi, \quad \phi \in C.$$

Denoting the spectrum of $\Pi(s)$ by $\sigma(\Pi(s))$ we have that if $\mu \in \sigma(\Pi(s))$ and $\mu \neq 0$ then there exists a $\phi \in \mathbb{C}$, $\phi \neq 0$ such that $\Pi(s)\phi = \mu\phi$. In this case μ is called a characteristic multiplier of (2.10) and if λ is such that $\mu = e^{\lambda w}$ then λ is called a characteristic exponent of (2.10).

The generalised eigenspace $M_\mu(s)$ of $\Pi(s)$ at μ is defined to be $N((\mu I - \Pi(s))^{k_\lambda})$ where k_λ is the smallest integer such that

$$M_\mu(s) = N((\mu I - \Pi(s))^{k_\lambda}) = N((\mu I - \Pi(s))^{k_\lambda+1}).$$

Solutions of (2.10) with initial value in $M_\mu(s)$ are of the form $x(t) = e^{\lambda t}p(t)$ where $\mu = e^{\lambda w}$ and $p(t+w) = p(t)$, that is, they are of the Floquet type. Hence we can see that a periodic transformation, such that the periodic equation is transformed to an autonomous equation, exists on each generalised eigenspace.

In a similar way to that for autonomous equations, the Floquet solutions, corresponding to initial data $\phi \in M(s)$, where $M(s)$ denotes the generalised eigenspace of $\Pi(s)$ and is given by $M(s) = \bigoplus_{\mu \in \sigma(\Pi(0)) \setminus \{0\}} M_\mu(s)$, can be related to an arbitrary solution $x_t(s; \phi)$. Verduyn Lunel in [77] shows that when the delay is an integer multiple of the period then exponents of (2.10) coincide with the spectrum corresponding to the autonomous equation $\dot{y}(t) = \hat{b}y(t-1)$, where $\hat{b} = \frac{1}{w} \int_0^w B(s)ds$.

Remark 2.2.1 [77] If w is irrational little is known about the spectral data of $\Pi(s)$.

2.2.3 An alternative approach

We note that delay differential equations can also be studied using an algebraic approach. In [35] Gluesing-Luerssen adopts the behavioural approach to systems theory, (the behaviour is the space of all possible trajectories of a system), and shows that linear autonomous DDEs with commensurate point delays can be studied from a behavioural point of view. Serious obstacles prevent a similar approach being used when the delays are not commensurate (see [35] p. 9). The approach adopted in this thesis is not an algebraic one. For further details about the behavioural approach to the study of DDEs the reader is referred to [35] and the references therein.

2.3 Stability of the solutions of DDEs

Two possible approaches to the stability analysis of DDEs are possible, delay independent stability and delay dependent stability. The first approach involves finding conditions such that the problem is stable for all or for some classes of delay. In the second approach weaker conditions are found such that the problem is stable for the specific given delay, generally constant [11]. The delays in the problems analysed in this thesis are fixed, hence the second approach is appropriate. Asymptotic stability analysis for a fixed delay is more difficult [11]. Authors of [11] describe delay dependent asymptotic stability of numerical methods for systems as ‘a real challenge’ (see p. 355).

For linear non-autonomous DDEs we can talk of the stability (asymptotic stability) of the equation. In Table 2.1 we state conditions for the asymptotic stability of equations relevant to our work. In relation to other equations of

Equation	Sufficient condition for stability	Ref.
$y'(t) = \lambda y(t) + \mu y(t - \tau), \lambda, \mu \in \mathbb{R}$	$ \mu \leq -\lambda$	[6]
$y'(t) = \lambda y(t) + \mu y(t - \tau), \lambda, \mu \in \mathbb{C}$	$ \mu \leq -\Re(\lambda)$	[6, 11]
$y'(t) = \mu y(t - \tau), \mu \in \mathbb{R}$	$0 < -\tau\mu < \frac{\pi}{2}$	[11]
$y'(t) = \mu y(t - \tau), \mu \in \mathbb{C}$	$\Re(\mu) < 0$ and $0 < \tau \mu < \min\{\theta_1, \theta_2\}$ where $\theta_1 = \frac{3\pi}{2} - \arg(\mu)$ and $\theta_2 = \arg(\mu) - \frac{\pi}{2}$.	[11]
$y'(t) = \lambda(t)y(t) + \mu(t)y(t - \tau)$	$ \mu(t) < -\Re(\lambda(t)), t \geq t_0$	[6]
$\dot{x}(t) = Ax(t) + \sum_{j=1}^m B_j x(t - \tau_j)$	$\mu(A) + \sum_{j=1}^m \ B_j\ < 0,$ $\mu(A) = \text{logarithmic norm of A}$	[45]

Table 2.1: Conditions for the stability of some DDEs

interest to our work we note the following results concerning stability.

- The trivial solution of $y'(t) = \sum_{j=1}^m A_j y(t - \tau_j)$ is asymptotically stable if all roots of the characteristic equation have negative real parts (see [23], p.363).
- All solutions of $y'(t) = \lambda_* y(t) + \mu_* y(t - \tau_*)$, $t \geq t_0$, with $y(t) = \psi_*(t)$, $t \in [t_0 - \tau_*, t_0]$ are stable with respect to perturbed initial conditions when the point $(\lambda, \mu) := (\lambda_* \tau_*, \mu_* \tau_*)$ lies in the stability region Σ which is the region of the (λ, μ) -plane that includes the half-line $\lambda < 0$, $\mu = 0$ and whose boundary $\partial\Sigma = \partial\Sigma_1 \cup \partial\Sigma_2$ is formed by the loci $\partial\Sigma_1 := \{\mu = -\lambda\}$, $\partial\Sigma_2 := \{(\lambda = \omega \cot \omega, \mu = -\omega \operatorname{cosec}(\omega)); 0 < \omega < \pi\}$ (see [6]) .
- Let A and B be constant $m \times m$ matrices. If the matrices A and B are simultaneously diagonalisable then the asymptotic stability of the system $y'(t) = Ay(t) + By(t - \tau)$, $t \geq t_0$; $y(t) = \phi(t)$, $t \leq t_0$, with τ constant, is completely described by the eigenvalues of the two matrices [11, 82].
- Conditions for stability of pure delay equations can only be obtained if we take the particular delay into account [82], that is, we can only obtain delay dependent conditions [11].
- The solutions of $y'(t) = \mu(t)y(t - \tau)$, $t \geq t_0$; $y(t) = \phi(t)$, $t \leq t_0$, for real $\mu(t)$, satisfy $|y(t)| \leq 2 \max_{x \leq t_0} |\phi(x)|$, $t \geq t_0$, whenever $\frac{-1}{\pi} \leq \mu(t) < 0$, $t \geq t_0$ (see [11] and included reference).
- Sufficient conditions for the asymptotic stability of equations of the form $x'(t) = ax(t) + \sum_{j=1}^m b_j x(t - \tau_j)$, $t > 0$; $x(t) = \phi(t)$ are given in [44] as $\Re(a) < 0$ and $\sum_{j=1}^m |b_j| < -\Re(a)$.
- Authors of [11], published in 2003, state that (at the time of writing) explicit conditions suitable to describe the stability region of $y'(t) = Ly(t) + My(t - \tau)$, $t \geq t_0$; $y(t) = \phi(t)$, $t \leq t_0$ for fixed delay are unknown. In the case when $L = 0$ and M is constant the whole spectrum of M must lie in the stability region of the scalar equation $y'(t) = \mu y(t - \tau)$, $t \geq t_0$; $y(t) = \phi(t)$, $t \leq t_0$ [11].
- Stability analysis for non-linear delay differential equations of the form $y'(t) = f(t, y(t), y(t - \tau(t)))$, $t \geq t_0$; $y(t) = \phi(t)$, $t \leq t_0$ (f and ϕ such that there is a unique solution) can be found in [68].

For non-linear equations the stability properties attach to a particular solution [6]. For further discussions on stability the reader is referred to [6, 11, 23, 54].

2.4 Numerical methods for DDEs

An introduction to numerical methods for ODEs is given in [57] using source material such as [2, 56]. In this section we give a brief introduction to the numerical solution of DDEs. We concentrate on issues relevant to this thesis but include some references to further material when appropriate. In line with the thesis we focus on results related to DDEs with constant delay. Results concerning stability are included but we choose not to refer to other issues such as error control strategies. We refer the reader to publications such as Hairer, Norsett and Wanner [39], Zennaro [82], Bellen and Zennaro [11], and Baker, Paul and Willé [4, 5] for more detailed treatments.

Numerical methods are sought that preserve the asymptotic stability property under the same conditions as those guaranteeing asymptotic stability of the exact solution. Chapter 1 of [11] includes examples that illustrate the destruction of some desirable accuracy and stability properties, such as order failure and stability failure, when an underlying ODE method is applied to a DDE.

Two types of schemes have been developed:

Adaptations of ODE schemes: The standard approach, detailed in [11], uses continuous ODE methods, based on the method of steps (see section 1.2.4). These combine an ODE numerical method with an interpolation scheme. Theta-methods with linear interpolation are the most widely studied Runge-Kutta methods for DDEs [82].

Specialised methods: These are generally inexpensive but remain accurate.

θ -methods

Applying the θ -method to equation

$$(2.11) \quad \begin{aligned} \dot{x}(t) &= b(t)x(t - \tau), b(t + \tau) = b(t) \text{ with } h = \frac{\tau}{N}, \\ x(0) &= \phi(0) \end{aligned}$$

gives

$$(2.12) \quad \begin{aligned} x_{n+1} &= x_n + h\{(1 - \theta)b_n x_{n-N} + \theta b_{n+1} x_{n+1-N}\}, n \geq 0 \\ x_n &= \phi(nh), \quad -N \leq n \leq 0. \end{aligned}$$

For equation (2.11), with $b(t)$ replaced by \hat{b} , equation (2.12) becomes

$$(2.13) \quad \begin{aligned} x_{n+1} &= x_n + h\hat{b}\{(1 - \theta)x_{n-N} + \theta x_{n+1-N}\}, n \geq 0 \\ x_n &= \phi(nh), \quad -N \leq n \leq 0. \end{aligned}$$

$\theta = 0$ gives the forward Euler method.
 $\theta = \frac{1}{2}$ gives the trapezium rule.
 $\theta = 1$ gives the backward Euler method.

Adams methods

Applying the Adams-Bashforth method of order 2 to (2.11) gives

$$x_{n+1} = x_n + h \left\{ \frac{3}{2} b_n x_{n-N} - \frac{1}{2} b_{n-1} x_{n-1-N} \right\}.$$

Applying the Adams-Moulton method of order 3 to (2.11) gives

$$x_{n+1} = x_n + \frac{h}{12} \{ 5b_{n+1}x_{n+1-N} + 8b_nx_{n-N} - b_{n-1}x_{n-1-N} \}.$$

2.4.1 Stability of the methods

Existing stability theory is almost invariably concerned with constant stepsize formulae [4]. We begin with the test equation 2.14 whose solution is known to be asymptotically stable for all initial functions ϕ and all delays τ if $\Re(\lambda) + |\mu| < 0$. A numerical method is said to be P -stable if it preserves the asymptotic stability of equation (2.14). Definitions of P -stability and GP -stability for numerical methods for DDEs relate to (2.14), with condition 2.15, the stability region for the equation, where $\lambda, \mu \in \mathbb{C}$ and τ is a constant delay. P -stability is a delay independent property. We note the constrained mesh in the definition of a P -stable method.

$$(2.14) \quad \begin{aligned} y'(t) &= \lambda y(t) + \mu y(t - \tau), & t \geq t_0, \\ y(t) &= \phi(t), & t \leq t_0. \end{aligned}$$

$$(2.15) \quad \Re(\lambda) + |\mu| < 0.$$

Definition 2.4.1 (Definition 10.2.1 in [11]) The P -stability region of a numerical method for DDEs is the set S_P of pairs of complex numbers (α, β) , $\alpha = h\lambda, \beta = h\mu$, such that the discrete numerical solution $\{y_n\}_{n \geq 0}$ of (2.14), obtained with constant stepsize h under the constraint $h = \frac{\tau}{m}, m \geq 1, m$ integer, satisfies

$$(2.16) \quad \lim_{n \rightarrow \infty} y_n = 0$$

for all constant delays τ and all initial functions $\phi(t)$.

Definition 2.4.2 (Definition 10.2.2 in [11]) A numerical method for DDEs is P -stable if $S_p \supseteq \{(\alpha, \beta) \in \mathbb{C}^2 \mid \Re(\alpha) + |\beta| < 0\}$.

Definition 2.4.3 (Definition 10.2.3 in [11]) The GP -stability region of a numerical method for DDEs is the set S_{GP} of pairs of complex numbers (α, β) , $\alpha = h\lambda, \beta = h\mu$, such that the discrete numerical solution $\{y_n\}_{n \geq 0}$ of (2.14), obtained with constant stepsize h satisfies (2.16) for all constant delays τ and all initial functions $\phi(t)$.

Definition 2.4.4 (Definition 10.2.4 in [11]) A numerical method for DDEs is GP -stable if $S_{GP} \supseteq \{(\alpha, \beta) \in \mathbb{C}^2 \mid \Re(\alpha) + |\beta| < 0\}$.

The concepts of PN -stability and GPN -stability are similarly defined in [11] for the more general test equation (see also definitions 10.4.1 and 10.4.2 in [68])

$$(2.17) \quad y'(t) = \lambda(t)y(t) + \mu(t)y(t - \tau), t \geq t_0,$$

$$(2.18) \quad y(t) = \phi(t), t \leq t_0$$

(where $\lambda(t)$ and $\mu(t)$ are continuous complex valued functions and τ is a constant delay) with the condition $\Re(\lambda(t)) + |\mu(t)| \leq 0, t \geq t_0$. Being based on a more general test equation these are stronger concepts of stability.

Definitions of P_M -stability and GP_M -stability, which relate to the the test equation

$$y'(t) = ay(t) + \sum_{j=1}^m b_j y(t - \tau_j), t > 0; y(t) = \phi(t), t \leq 0,$$

where $a, b_j \in \mathbb{C}$, can be found in [44].

We note that GP -stability implies P -stability, GPN -stability implies PN -stability etc. For variable stepsize methods we find the definitions of fully P -stable and fully PN -stable methods in chapter 10 of [11]. Other definitions of stability relating to different test equations can be found in the literature, for example NP -stability and GNP -stability for neutral DDEs (see [11]). The reader is referred to [11, 82] for further general discussion about the stability of numerical methods for DDEs.

The trapezium rule is the numerical method predominantly used in the research presented in this thesis. Other methods employed include Backward Euler method, Forward Euler method, Adams-Bashforth method of order 2 and Adams-Moulton method of order 3. We concentrate on aspects of stability pertaining to these methods. The following results can be found in the literature.

Linear multistep methods are known to be GP_M -stable if and only if the method is A -stable for ODEs ([44]). The order of an A -stable linear multistep method is at most 2 ([2, 56]).

Forward Euler method is zero-stable for ODEs with a very small region of absolute stability [2].

Adams-Bashforth methods and Adams-Moulton methods have small regions of absolute stability for ODEs [2].

The linear θ -method with piecewise linear interpolation is GP_M -stable [44] if and only if $\frac{1}{2} \leq \theta \leq 1$ and is GP -stable [82] if and only if $\frac{1}{2} \leq \theta \leq 1$.

Runge-Kutta methods are P -stable if the underlying method is A -stable [11] and can be GP -stable [82].

Backward Euler method is known to be GP -stable and GPN -stable [68].

The reader is referred to [39, 56] for details of stability of methods for ODEs. A collection of results concerning stability of numerical methods for ODEs, together with detailed references to appropriate literature, is presented in [57].

We can see that, as a Runge-Kutta method which is A -stable for ODEs, the trapezium rule is P -stable for DDEs and, as a θ -method with $\theta = \frac{1}{2}$, it is also GP -stable and GP_M -stable. In chapter 4 we make an informed choice of numerical method, based on our experimental results. The results presented in this section concerning stability of numerical methods, along with the test equations considered in our work lend further credence to our choice.

Guglielmi in [38] regards the trapezium rule as “a good method for solving real DDEs” since it provides a “good compromise between stability and order requirements” and computational efficiency. However, we are alerted by the heading “Instability of the trapezoidal rule” to the fact that the trapezium rule is not τ -stable (see [38] for the proof). $\tau(0)$ -stability and τ -stability relate to equation (2.14) but with a fixed value of the delay τ and with $\lambda, \mu \in \mathbb{R}$ and $\lambda, \mu \in \mathbb{C}$ respectively. These concepts are stronger than that of P -stability, a property holding for all delays.

Definition 2.4.5 (Definition 2.1 in [38]) The $\tau(0)$ -stability region of a numerical method for DDEs is the set

$$S_{\tau(0)} = \bigcap_{m \geq 1} S_m,$$

where, for a given positive integer m , S_m is the set of the pairs of *real* numbers (λ, μ) such that the discrete numerical solution $\{y_n\}_{n \geq 0}$ of (2.14) with constant stepsize $h = \frac{1}{m}$, satisfies $\lim_{n \rightarrow \infty} y_n = 0$ for all initial functions $\phi(t)$.

Definition 2.4.6 (Definition 2.2 in [38]) A numerical step-by-step method for DDEs is $\tau(0)$ -stable if $S_{\tau(0)} \supseteq \Sigma_*$ where Σ_* is the asymptotic stability region for equation (2.14). (See Table 2.1).

Definition 2.4.7 (Definition 5.1 in [38]) The τ -stability region of a numerical step-by-step method for DDEs is the set

$$Q_{\tau(0)} = \bigcap_{m \geq 1} Q_m,$$

where, for a given positive integer m , Q_m is the set of the pairs of *complex* numbers (λ, μ) such that the discrete numerical solution $\{y_n\}_{n \geq 0}$ of (2.14) with constant stepsize $h = \frac{1}{m}$, satisfies $\lim_{n \rightarrow \infty} y_n = 0$ for all initial functions $\phi(t)$.

Definition 2.4.8 (Definition 5.2 in [38]) A numerical step-by-step method for DDEs is τ -stable if $Q_{\tau(0)} \supseteq \Xi_*$, where Ξ_* is the stability region for equation (2.14). (See Table 2.1).

The following results, relating to delay dependent stability, can be found in the literature:

Backward Euler method is conjectured to be τ -stable in [38] and said to be D -stable in [11].

Theta-methods are $D(0)$ -stable for $\frac{1}{2} \leq \theta \leq 1$ [11] and $\tau(0)$ -stable for $\theta \geq \frac{1}{2}$ (see [38]).

The trapezium rule is $\tau(0)$ -stable but not τ -stable [6, 38]. Similarly, the trapezium rule is said to be $D(0)$ -stable but not D -stable in [11].

Definitions of the D -stability region of a numerical method and of a D -stable numerical method can be found in [11].

2.5 Small solutions: Further background theory

2.5.1 Autonomous equations

Autonomous ODEs of the form

$$(2.19) \quad \dot{x}(t) = Bx(t), \quad x(0) = x_0 \in \mathbb{C}^n,$$

where B denotes an $n \times n$ matrix cannot have non-trivial small solutions (see [75] for a proof). Solutions of (2.19) can be represented by a sum of elementary solutions of the form $x(t) = p(t)e^{\lambda t}$, that is, they are of the form $x(t) = \sum_{j=1}^n p_j(t)e^{\lambda_j t}$, where λ_j is a zero of $\det(zI - B)$ and p_j is a polynomial [75].

However, autonomous DDEs of the form

$$(2.20) \quad \dot{x}(t) = Bx(t-1)$$

can admit small solutions (see example on page 532 of [75]). From Theorem 2.1 in [75] we know that (2.20) has non-trivial small solutions if and only if $\det(B) = 0$. Completeness of the elementary solutions is obtained if and only if the zero solution is the only small solution of (2.20) (see [75]). Henry in [43] showed that small solutions of autonomous DDEs are identically zero after finite time. In [70] Verduyn Lunel presents a formula for the smallest possible time T_0 with the property that all small solutions are identically zero on $[T_0, \infty)$. The situation is clear for autonomous delay equations. Necessary and sufficient conditions for the existence of small solutions are known for a very general class of delay equations including both retarded and neutral equations [75].

The Laplace transform approach

The following results relating to the Laplace transform approach can be found in the literature:

1. Small solutions are in one-to-one correspondence with entire solutions of an algebraic equation for the Laplace transform of the solution [21].
2. The Laplace transform of a small solution converges everywhere. Hence a small solution is an entire function. (See [69] or p. 83 in [41]).

Some analytical results

In relation to completeness and small solutions we find the following results in the literature.

1. The system of eigenvalues and generalised eigenvectors of the generator A is complete if and only if the exponential type of $\det \Delta = nh$, that is $E(\det \Delta) = nh$ (Theorem 3.13 in [22], Corollary 4.7 in [69], Theorem 3.1 in [73], Corollary 3.3 in [70]).
2. The associated solution operator $T(t)$ is one-to-one if and only if $E(\det \Delta) = nh$ (Theorem 3.3 in [41], Corollary 4.8 in [69]).
3. The system has no small solutions if and only if $E(\det \Delta) = nh$ (Theorem 4.1 in [69], Theorem 2.3 in [70], Theorem 4.4 in [73]).
4. Completeness of the system of generalised eigenfunctions fails if the semi-group is not one-to-one [72].
5. Completeness holds if and only if there are no small solutions [70].
6. Theorem 1.20 in [72] encapsulates points 1-4 above in a single theorem in which four statements are said to be equivalent.

7. Theorem 4.1 in [73] states that the system of eigenvectors and generalised eigenvectors of the generator of the semigroup is complete if and only if the semigroup is one-to-one and that the semigroup is one-to-one if and only if $E(\det \Delta) = nh$.
8. All small solutions are in the null space of the C_0 -semigroup [71].
9. If the infinitesimal generator has an empty spectrum then for every ϕ the solution $t \rightarrow T(t)\phi$ is a small solution [69].
10. The multi-delay equation $\dot{x}(t) = Ax(t) + Bx(t-1) + Cx(t-2)$ (where A , B and C are $n \times n$ matrices) has a complete set of eigenfunctions and generalised eigenfunctions if and only if $\det C \neq 0$ (see [22, 73]).

The existence of small solutions to an equation is closely related to the question of whether or not a solution of the equation has a convergent series expansion in characteristic solutions $p_j(t)e^{\lambda_j t}$, where $p_j(t)$ is a polynomial and λ_j is a root of the characteristic equation [41]. Small solutions have a series expansion in which all terms are zero [41, 71]. Section 3.3 in [77] includes further results relating to series expansions for autonomous equations. For the particular case of equation $\dot{x}(t) = B_0x(t) + B_1x(t-1)$ we find in [77] that if B_1 is non-singular then the solution to the equation has a convergent series expansion and that the system of eigenvectors and generalised eigenvectors is complete.

2.5.2 Non-autonomous equations

Non-autonomous ODEs can admit small solutions, for example $x(t) = e^{-t^2}$ is a small solution of $\dot{x}(t) = -2tx(t)$ [75]. However, it is known that $\dot{x}(t) = b(t)x(t)$, $t \geq s$; $x(s) = x_0$, cannot have non-trivial small solutions if there exists a positive constant m_0 such that $-m_0 \leq b(t) \leq 0$ for $t \geq 0$ [75].

DDEs of the form $\dot{x}(t) = b(t)x(t-1)$ can admit non-trivial small solutions (see examples 1.3.2 and 1.3.3 in chapter 1), even if $b(t)$ is bounded below on $[0, \infty)$. In the more general case the system $\dot{x}(t) = A(t)x(t) + B(t)x(t-h)$, with $A(t)$ and $B(t)$ bounded, real analytic $n \times n$ matrix-valued functions, has no non-trivial small solutions if $|\det B(t)| > 0$ (see [21]). (This is restated as Theorem 3.1 in [75] with $h = 1$.) If $A(t)$ and $B(t)$ are 1-periodic and h is a positive integer then the assumption that $A(\cdot)$ and $B(\cdot)$ are real analytic can be omitted [21]. Authors of [20] regard the conditions of this theorem as sharp. They show that boundedness and analyticity of the coefficients on the whole line cannot be omitted and present examples of equations admitting nontrivial small solutions which have coefficients that are bounded on all of \mathbb{R} but which are not analytic.

We note that, in this case, the Laplace transform of the solution no longer satisfies an algebraic equation [21].

Equations with periodic coefficients

Analytical theory for equations with periodic coefficients has developed using the concept of the period map (see section 2.2.2). Here we state results concerning small solutions, found in the literature, pertaining to particular equations in this class.

1. $\dot{x}(t) = a(t)x(t) + b(t)x(t - 1), t \geq s$ where $a(t)$ and $b(t)$ are 1-periodic functions.
 - (a) Theorem 4.1 in [75] states that if the zeros of $b(t)$ are isolated then the system has small solutions if and only if $b(t)$ has a sign change (see also [33]).
 - (b) From Theorem 3.4 in [21] if $|b(t)| > 0$ then the equation has no small solutions and the system of Floquet solutions is complete.
 - (c) The Floquet solutions are dense in $C(= C([-1, 0], C^n))$ if and only if the equation has no non-trivial small solutions [75].
2. $\dot{x}(t) = b_0(t)x(t) + b_1(t)x(t - \omega), t \geq s$ where $b_0(t)$ and $b_1(t)$ are ω -periodic functions.
 - (a) Theorem 6.1 in [73] states that, supposing that the zeros of b_1 are isolated, the system of eigenvectors and generalised eigenvectors is complete if and only if b_1 has no sign change. A proof is included in the paper. We note that Verduyn Lunel states (in [73]) that the theorem holds if the delay is an integer multiple of the period but that the appropriate conditions for the theorem to hold in the matrix case are not yet known.
3. $\dot{x}(t) = B(t)x(t - 1)$ where $B(t)$ is a real, continuous matrix function with minimal period w .
 - (a) If the linear space is dense in C then each solution can be approximated by a linear combination of Floquet type solutions [77].
4. $\dot{x}(t) = b(t)x(t - 1), b(t + 1) = b(t), t \geq 0$ where $b(t)$ is a non-zero scalar-valued function.
 - (a) Theorem 4.3 in [77] states that a convergent series expansion exists if $b(t)$ has constant sign.

- (b) If $b(t)$ has constant sign and isolated zeros then the equation has no non-trivial small solutions ([77] Cor. 4.4) and the monodromy operator has a complete set of eigenvectors and generalised eigenvectors ([77] Cor. 4.5).
- (c) The direct sum of the generalised eigenspaces M_λ , $\lambda \in \sigma(\pi(s))$ (λ belongs to the spectrum of the period map $\pi(s)$), is not dense in C “if and only if there exist non-trivial small solutions if and only if the coefficient b changes sign” [33].

Remark 2.5.1 We note here that the property of possessing, or not possessing, small solutions is preserved by a similarity transformation. The reader is referred to appendix F for an explanation and an example.

2.6 An example from immunology

2.6.1 Introduction

This example, from the field of immunology, involves the computational implementation of an information-theoretic approach to modelling the viral kinetics of LCMV-WE virus in C57/BL/6 mice. We include this example to illustrate:

- a better fit with the (real) data can result from the inclusion of a delay term (see comment in the introduction to chapter 1).
- an application in the area of parameter estimation where the non-existence of small solutions is essential (see comments in section 1.3.3).
- the application of the code DDE23 to solve a system of DDEs.

The experiment: Brief details

A batch of genetically identical C57/BL/6 mice were infected with 200 pfu (plaque forming units) of LCMV (WE strain). Data concerning the viral titer in the spleen and the clonal expansion of CTL cells specific for the gp33 epitope in spleens was collected after discrete intervals of time.

2.6.2 The five models

In [9] a hierarchy of five models is considered, including two ODE formulations and three DDE formulations of the problem. Values of biologically significant parameters are estimated using real data, one of the objectives being to develop a best-fit mathematical model to the given data. (The data was obtained by several of the authors of the paper.) In the equations for the models $V(t)$ denotes

the virus, measured in pfu, $V(t)$ denotes the number of virus-specific activated *CTL* and $E_m(t)$ denotes the virus-specific memory *CTL*. The equation for the rate of change of $V(t)$ is the same for each model but the equations describing the immune response differ. The reader is referred to [9] for the data used and the biological interpretation of the parameters involved.

Model 1: (simplest consideration of the CTL dynamics)

$$(2.21) \quad \frac{d}{dt}V(t) = \beta.V(t) \cdot \left(1 - \frac{V(t)}{K}\right) - \gamma.V(t).E(t)$$

$$(2.22) \quad \frac{d}{dt}E(t) = b_1.V(t).E(t) - \alpha_E.E(t)$$

Model 2:(virus-dependent with saturation CTL proliferation)

$$(2.23) \quad \frac{d}{dt}V(t) = \beta.V(t) \cdot \left(1 - \frac{V(t)}{K}\right) - \gamma.V(t).E(t)$$

$$(2.24) \quad \frac{d}{dt}E(t) = \underbrace{b_2.V(t).E(t)/(\theta_{Sat} + V(t))}_{\text{A modification of model 1}} - \alpha_E.E(t)$$

Model 3: (virus-dependent with saturation CTL proliferation with time lag)

$$(2.25) \quad \frac{d}{dt}V(t) = \beta.V(t) \cdot \left(1 - \frac{V(t)}{K}\right) - \gamma.V(t).E(t)$$

$$(2.26) \quad \frac{d}{dt}E(t) = \underbrace{b_3.V(t - \tau).E(t - \tau)/(\theta_{Sat} + V(t))}_{\text{As in model 2 but incorporating delay}} - \alpha_E.E(t)$$

Model 4: (primary CTL homeostasis)

$$(2.27) \quad \frac{d}{dt}V(t) = \beta.V(t) \cdot \left(1 - \frac{V(t)}{K}\right) - \gamma.V(t).E(t)$$

$$(2.28) \quad \frac{d}{dt}E(t) = b_4.V(t - \tau).E(t - \tau)/(\theta_{Sat} + V(t)) - \underbrace{\alpha_E.E(t) + T^*}_{\text{includes additive term}}$$

Model 5: (Additional equation for the population of memory CTL)

$$(2.29) \quad \frac{d}{dt}V(t) = \beta.V(t) \cdot \left(1 - \frac{V(t)}{K}\right) - \gamma.V(t).E(t)$$

$$(2.30) \quad \frac{d}{dt}E(t) = b_5.V(t - \tau).E(t - \tau)/(\theta_{Sat} + V(t)) - \alpha_E.E(t) - \mu.E(t) + T^*$$

$$(2.31) \quad \frac{d}{dt}E_m(t) = \mu.E(t) - \alpha_m.E_m(t)$$

The general initial data:

$$V(t) = 0, t \in [-\tau, 0), V(0) = V_0;$$

$$E(t) = E_0, t \in [-\tau, 0];$$

$$E_m(0) = 0.$$

2.6.3 Methodology

Relying on the argument that, since the mice are genetically identical, large numbers of mice are unnecessary, we proceed, for each time t , to use the average of the two pieces of available data to calculate estimates of the parameters of the model, (under the assumption of reliable/perfect data [76]). Least squares data fitting involves selecting an appropriate least squares objective function. In [9] three types of objective function were considered:- ordinary least-squares, weighted least-squares and log-least squares. Best-fit estimates of the parameters were obtained for each type of objective function. This was achieved using ARCHI-L and Matlab, with contour plots proving to be a valuable tool in the process.

2.6.4 Some of the results

Results from each method are included in the paper and the parsimony of the models is considered. In this thesis we choose to refer only to the use of the ordinary least-squares objective function and the resulting parameter estimates. In Table 2.2 estimates of the parameters using an ordinary least-squares objective function are presented.

In Figure 2.1, produced using DDE23, the upper two diagrams originate from the ODE models 1 and 2 and the lower three from the DDE models, 3, 4, and 5 (the ordering of the diagrams is sequential). In each case we have plotted the original data and the solution to each set of equations after the parameter estimation process has been completed.

2.6.5 Observations from these results

We observe an improvement in the modelling process, evidenced by a more accurate representation of the qualitative behaviour of the solution (see Figure

Parameter	Model 1	Model 2	Model 3	Model 4	Model 5
β	4.44	4.36	4.52	4.52	4.50
K	3.99×10^6	3.23×10^6	3.17×10^6	3.17×10^6	3.19×10^6
γ	3.02×10^{-6}	3.48×10^{-6}	3.45×10^{-6}	3.48×10^{-6}	3.63×10^{-6}
b_i	1.23×10^{-6}	1.92	2.52	2.41	2.40
α_E	0	0.0914	0.0862	0.0910	0.0931
θ_{Sat}	-	2.46×10^4	1.34×10^5	1.31×10^5	1.15×10^5
τ	-	-	0.0717	0.0898	0.0954
T^*	-	-	-	124	140
α_m	-	-	-	-	0.255
μ	-	-	-	-	0.00517
Residual	3.240×10^6	2.119×10^6	2.010×10^6	1.977×10^6	1.943×10^6

Table 2.2: Estimates of the parameters of the model (to 3 s.f.) and the resulting residual (to 4 s.f.)

2.1) and a reduction in the least squares residual (see Table 2.2), when delay differential equation formulations of increasing complexity are used. (The same improvements were also observed when a weighted least-squares objective function was used). The interested reader is referred to [9] for full details about the theory, experiment and methodology, and for a complete set of results arising from the different objective functions used and the conclusions reached from the research.

Remark 2.6.1 We note that the equations used in this model are autonomous and consequently questions concerning the existence, or otherwise, of small solutions do not arise. However, should a modeller feel that a non-autonomous equation would be more appropriate then it would be necessary to identify whether or not the equation could admit small solutions. We anticipate that the algorithm presented in chapter 10, along with any future modifications to it, or extensions of it, will be of assistance to the modeller.

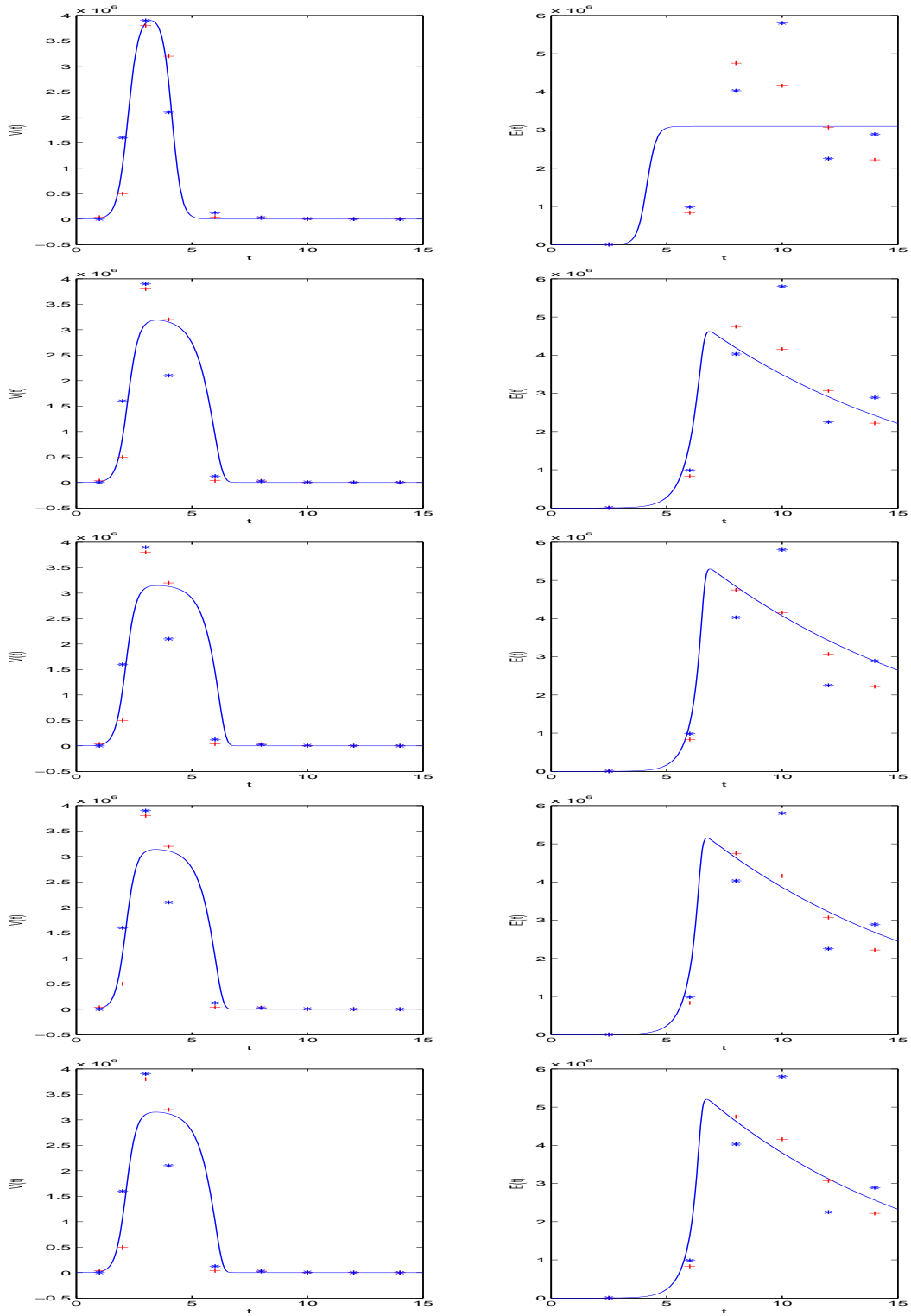


Figure 2.1: Ordinary least squares objective function: The fitted model for the viral load, $V(t)$, and the number of CTL cells, $E(t)$ and the original data sets.

Chapter 3

Our method: Introduction and Justification

In chapters 1 and 2 we have established a need for research into the detection of small solutions. However, it is not usually easy to determine by direct analysis whether or not an equation admits small solutions [31, 34]. Therefore we are prompted to turn to numerical methods. One role of the numerical analyst is to provide insight into analytical theory. (The reader is referred to section 1 in [48] and to the introduction to [49] for a discussion about the authors' viewpoints on the relationship between "analysis and computation: the quest for quality and the quest for quantity" [49].) In this chapter we introduce, and justify, the methodology behind our approach to the numerical detection of small solutions.

Our interest lies in the ability to detect the existence of small solutions to DDEs by studying the behaviour of the spectrum of the finite dimensional approximation to them. Testing our method using equations for which the analytical theory is known enables identification of characteristics of the eigenspectra that are indicative of the existence, or otherwise, of small solutions. Hence, through our numerical discretisation we hope to gain further insight into analytical theory.

Analytical methods of detection: An example

Cao in [16] considers the scalar DDE equation $\dot{x}(t) = f(t, x(t), x(t-1)), x(\theta) = \phi(\theta), \theta \in [-1, 0]$. The discrete Lyapunov function (the number of zeros of $x(t)$ on the unit interval, not counting multiplicities) is used to determine whether a solution $x(t)$, where $x(t)$ is a solution defined for all $t \in [-1, \infty)$ and such that $\lim_{t \rightarrow \infty} x(t) = 0$, is a superexponential solution. Cao shows that $x(t)$ is a superexponential solution if and only if the Lyapunov function of x tends to infinity as t tends to positive infinity.

3.1 Introducing our numerical approach

Our approach generally involves a comparison of the eigenspectra arising from a non-autonomous problem to that arising from an autonomous problem. The underlying theoretical justification for using eigenspectra derived from a numerical approximation to give information about the exact eigenspectra is given in [27]. We adopt the approach used in [34].

The dynamics of some periodic DDEs can be described by an autonomous DDE [33]. In a discussion relating to the analytic theory of periodic delay equations authors of [33] state that “the non-existence of nontrivial small solutions is a necessary condition to make a transformation of variables to an autonomous delay differential equation”.

In general we assume, for possible contradiction, that an equivalent autonomous problem exists. We calculate the eigenspectrum for the solution operator of that equation and compare it with that arising from the non-autonomous problem. When the equation does not admit small solutions the (exact) characteristic values all lie on one curve [79] and we expect the two trajectories to lie close to each other. When the non-autonomous equation admits small solutions this is not the case. We observe whether differences exist between the eigenspectra and use known analytical theory to identify characteristics of the eigenspectrum that indicate the presence of small solutions. Hence, we are able to make progress with the interpretation of eigenspectra for equations where analytical theory is less well developed.

However, not knowing the equivalent autonomous problem is not critical to our numerical detection of small solutions [79]. We are trying to detect multiple chains of roots, or trends in the chains of roots, to provide evidence for the existence, or otherwise, of small solutions. Eigenspectra displaying more than one asymptotic trend or curve are indicative of the presence of small solutions and hence it may not be necessary to ‘match’ the non-autonomous problem with an autonomous problem.

3.1.1 Justification for our approach

We follow the approach used in section 4 of [33], with the backward Euler method, but here with the trapezium rule. The characteristic values of the equation $x'(t) = \hat{b}x(t-1)$ lie on the locus $|\lambda| = |\hat{b}e^{-\lambda}|$. By Theorem 3.2 of [27] (see appendix C) the corresponding characteristic values of the discrete solution should provide a close approximation to the true characteristic values.

In Figure 3.1, with $h = \frac{1}{128}$ and $\hat{b} = 1.4$, we show:

(solid line) $y = \pm\sqrt{e^{-2x\hat{b}^2} - x^2}$, the true locus of the true eigenvalues of the autonomous problem,

- (***) $\frac{1}{h} \times$ the natural logarithm of the eigenvalues arising from use of the trapezium rule for the autonomous problem,
- (+++)
 $\frac{1}{h} \times$ the natural logarithm of the eigenvalues arising from use of the trapezium rule for a non-autonomous problem that is known not to admit small solutions and for which the illustrated autonomous problem is equivalent.

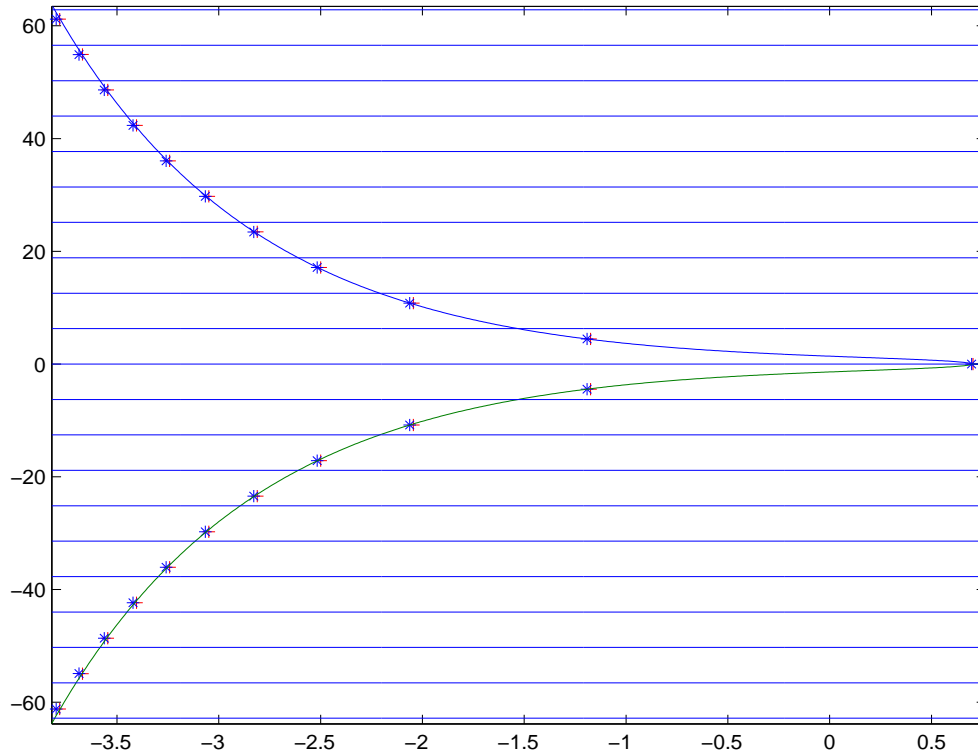


Figure 3.1: Approximation to characteristic values using the trapezium rule

The trajectories of eigenvalues arising from both the autonomous and non-autonomous problem lie very close to the true trajectory and the known property that there is one characteristic root in each horizontal band of width 2π , (see [33]), is also visualised.

To remove any ambiguity caused by an incorrect choice of the branch of the complex logarithm of the eigenvalues of ΠA_n , in Figure 3.2 we plot e^λ for each characteristic value λ . To clarify the picture nearer to the origin we zoom in in Figure 3.3. We note that, although our choice of scale is also a factor, the equivalence of the non-autonomous and autonomous problems is clearly demonstrated by the *invisibility* of the (+++) in Figure 3.2 and their poor visibility in Figures 3.1 and 3.3.

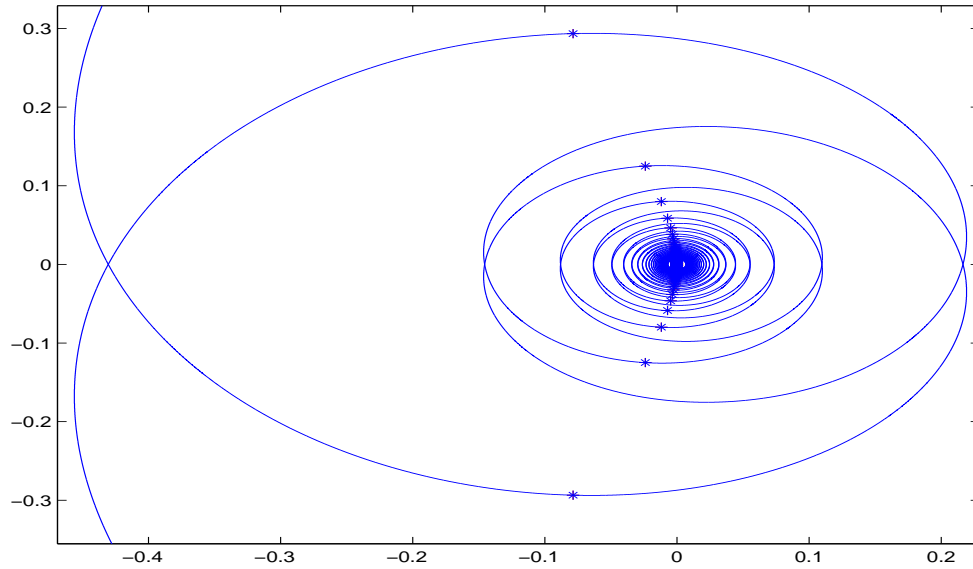


Figure 3.2: Locus of exponentials of true characteristic values of $x'(t) = \hat{b}x(t-1)$ with $\hat{b} = 1.4$ (solid line).

Exponentials of approximations to the true eigenvalues arising from discretisation of $x'(t) = \hat{b}x(t-1)$ using the trapezium rule (***)

Exponentials of approximations to the true eigenvalues arising from discretisation of $x'(t) = b(t)x(t-1)$, $b(t) = \sin(2\pi t) + 1.4$ using the trapezium rule (+++).

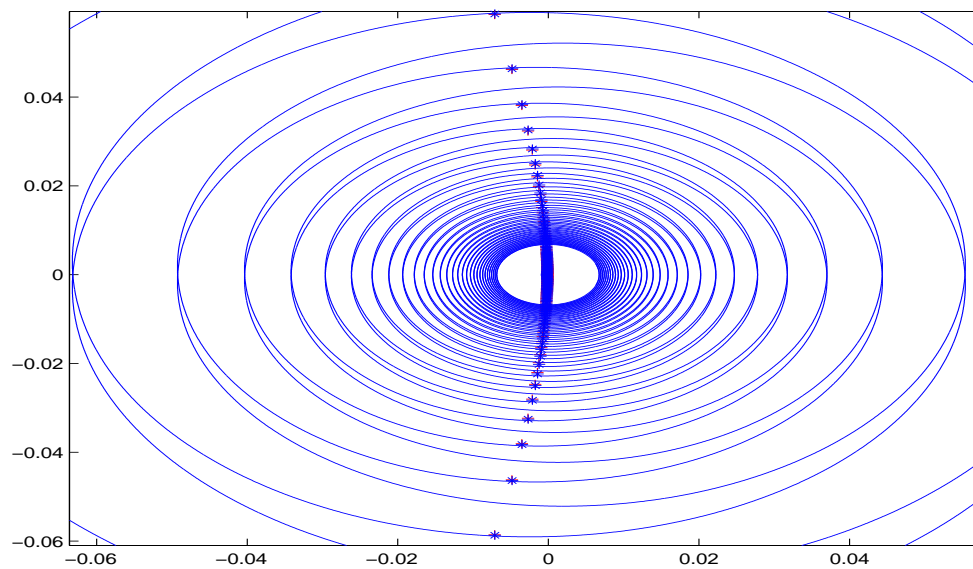


Figure 3.3: A zoomed-in version of Figure 3.2

In Figure 3.4 we illustrate the clear difference in the graphic when the non-autonomous problem admits small solutions. We note the similarity in scale to the earlier figures but the increase in visibility of the trajectory denoted by (+++).

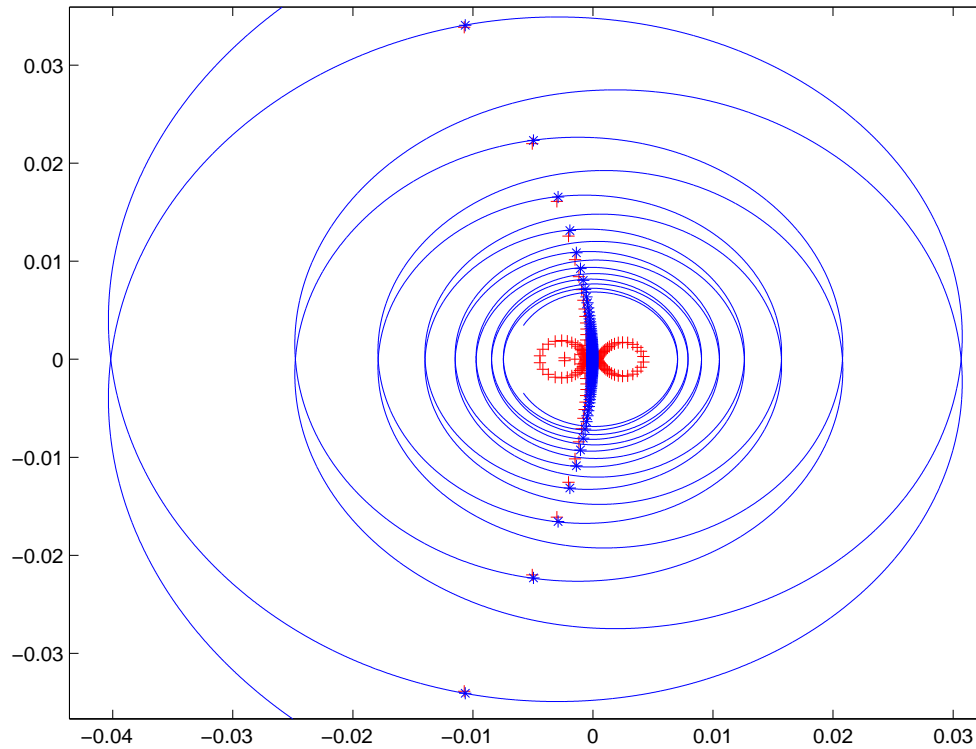


Figure 3.4: An illustration when small solutions are present. $b(t) = \sin(2\pi t) + 0.4$.

In Figures 3.1 to 3.4 we have illustrated how known theoretical behaviour of the solution map is characterised in our eigenspectra.

3.2 Known analytical results about the existence of an equivalent autonomous problem

In the scalar case the equivalent autonomous problem is only known analytically if the delay and period are equal. It is not clearly defined when p and d are not equal [79].

For an autonomous DDE such as $\dot{x}(t) = Ax(t-1)$ with A a constant matrix, the asymptotic roots are on one single curve when $\det A \neq 0$, that is when the equation does not admit small solutions. If $\det A = 0$ then there is no $e^{-n\lambda}$ term in the characteristic polynomial equation $\lambda^n + \dots + \det Ae^{-n\lambda} = 0$ and the

characteristic roots are asymptotically not on a single exponential curve [79]. We expect this fact to be visualised in our eigenspectra.

The equivalent autonomous problem is not known analytically in the matrix case. Floquet theory provides the underlying autonomous system. However, if we consider the ODE $\dot{y}(t) = A(t)y(t)$ with $A(t+p) = A(t)$ then we know from theory that a constant matrix B exists such that the solution to the equation is given by $y(t) = e^{Bt}p(t)$ with $p(t)$ a periodic function, but, in general, it is unknown how to compute B without computing the solutions to the equation. The Floquet theory holds for the DDE case but the computation of B is again the difficulty. Theory for autonomous systems implies that the characteristic values are on a single exponential curve.

Remark 3.2.1 When appropriate we will state that the equivalent autonomous problem is not known analytically. In this case we take the presence of more than one asymptotic curve, such as the presence of *closed loops*, in the eigenspectra to be characteristic of equations that admit small solutions. Where we appear to have successfully ‘matched’ the non-autonomous problem with an autonomous problem then it is possible that it may be correct up to leading order [79].

Example 3.2.1 In this example we show the equivalence between the non-autonomous problem $x'(t) = b(t)x(t-1)$, $b(t+1) = b(t)$ and the autonomous problem $y'(t) = \hat{b}y(t-1)$, where $\hat{b} = \int_0^1 b(s)ds$.

For the periodic equation $\dot{x}(t) = b(t)x(t-1)$, with $b(t+1) = b(t)$, the spectrum of the monodromy operator T , defined by

$$(T\phi)(\theta) = \int_{-1}^{\theta} b(s)\phi(s)ds + \phi(0), \quad -1 \leq \theta \leq 0,$$

determines whether or not the equation admits small solutions.

If

$$T\phi = \lambda\phi$$

then

$$\phi(0) + \int_{-1}^{\theta} b(s)\phi(s)ds = \lambda\phi.$$

Differentiating gives

$$b(\theta)\phi(\theta) = \lambda\dot{\phi}(\theta).$$

Hence

$$\frac{\dot{\phi}}{\phi} = \frac{1}{\lambda}b(\theta)$$

which leads to

$$\phi(\theta) = \phi(-1)e^{\int_{-1}^{\theta} \frac{1}{\lambda}b(s)ds}.$$

Hence

$$\phi(0) = \phi(-1)e^{\frac{1}{\lambda}\hat{b}}, \text{ where } \hat{b} = \int_{-1}^0 b(s)ds.$$

Using the non-local condition $\phi(0) = \lambda\phi(-1)$ this leads to

$$\lambda\phi(-1) = \phi(-1)e^{\frac{1}{\lambda}\hat{b}}$$

so that

$$e^{\frac{1}{\lambda}\hat{b}} - \lambda = 0.$$

If we let $\eta = \frac{1}{\lambda}$ then $e^{-\eta\hat{b}} = \eta$ which is the characteristic equation of $\dot{y}(t) = \hat{b}y(t-1)$.

A key to our eigenspectra

In our diagrams we choose to show the eigenvalue trajectory arising from the autonomous problem by * and that arising from the non-autonomous problem by +. To enable reliable conclusions to be drawn we vary the magnification of the eigenspectra near to the origin to suit the equation under consideration.

Some notation

We note here that, from this point of the thesis onwards, we will, in general, denote the solution to an equation by $x(t)$ in the scalar case and by $y(t)$ in the matrix case.

3.3 Using our numerical method to estimate the true eigenvalues

The characteristic equation for the autonomous DDE

$$x'(t) = \beta x(t-1)$$

is

$$\lambda - \beta e^{-\lambda} = 0.$$

This equation has infinitely many complex roots of the form $\lambda = x + iy$.

Let $\hat{\lambda} = \hat{x} + i\hat{y}$ be the approximation of the eigenvalue λ . In our eigenspectra we plot approximations to e^λ , that is $e^{\hat{\lambda}}$. Hence in our diagrams we have plotted (X, Y) where $X = e^{\hat{x}} \cos \hat{y}$, $Y = e^{\hat{x}} \sin \hat{y}$.

The true eigenvalues lie on the curves

$$x^2 + y^2 = \beta^2 e^{-2x} \text{ and } \tan y = \frac{-y}{x}.$$

The points (X, Y) on the trajectory of the autonomous problem satisfy

$$X^2 + Y^2 = e^{2\hat{x}} \text{ and } \tan \hat{y} = \frac{Y}{X}$$

leading to

$$\hat{x} = \frac{1}{2} \ln(X^2 + Y^2) \text{ and } \hat{y} = \tan^{-1} \left(\frac{Y}{X} \right) + n\pi.$$

Question: Do we have enough information to find values of \hat{y} corresponding to each value of \hat{x} ?

We note that it would be inappropriate to use $\hat{y}^2 = \frac{\beta^2}{(X^2+Y^2)} - \left[\frac{1}{2} \ln(X^2 + Y^2)\right]^2$ to estimate the value of \hat{y} since we would then be ‘forcing it’ to lie on the correct curve. However, if no small solutions are present this would give us the value that \hat{y} should take.

If we consider $\hat{y} = n\pi + \tan^{-1} \left(\frac{Y}{X} \right)$ can we use the fact that there is only one eigenvalue in each horizontal strip of width 2π (see [33]) to obtain \hat{y} ? We can order the values of \hat{x} , and add on the vector $(-2n\pi, 2n\pi, \dots, -4\pi, 4\pi, -2\pi, 2\pi, 0)^T$ to the vector for y . This would guarantee at most one eigenvalue in each horizontal strip but we need to ask the question: ‘Which of the true eigenvalues do we have estimates for?’

A possible approach: Decreasing the step size increases the number of eigenvalues estimated, and hence the likelihood that we have estimated the first N eigenvalues correctly will increase as $h = \frac{1}{N} \rightarrow 0$.

We present evidence of this approach. The diagram in Figure 3.5 shows the true trajectory of eigenvalues and the trajectories formed using the above approach for different values of N . The equation used is $x'(t) = \sin(2\pi t) + 1.6$ for $N = 200, 300, 400, 500$. The diagrams are different when small solutions are present. See Figure 3.6 for $x'(t) = \sin(2\pi t) + 0.6$, again for $N = 200, 300, 400, 500$. We also find eigenvalues on the boundaries of the strips which is not possible in the autonomous case [22].

Another possible approach: Taking logarithms to base e of $e^{\hat{\lambda}}$ gives $\hat{\lambda} = \hat{x} + i\hat{y}$. This is potentially an estimate of the true eigenvalue. However, this step involves taking logarithms of complex numbers. If $\hat{x} + i\hat{y} = \hat{r}e^{i\hat{\theta}}$ then $\ln(\hat{x} + i\hat{y}) = \ln(\hat{r}) + i\hat{\theta}$, and again the value of $\hat{\theta}$ is only the value of the argument in the range $[-2\pi, 2\pi]$. Again we are unable to ‘extract’ estimates of both x and y directly.

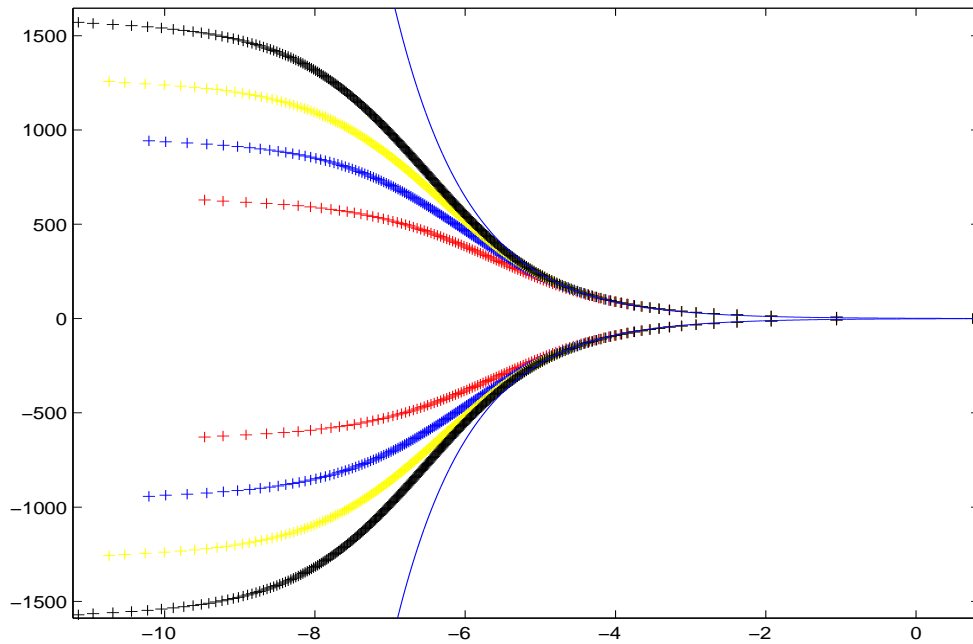


Figure 3.5: No small solutions. Trajectories approach true curve as the step size decreases
 Key: Red(N=200), Blue(N=300), Yellow(N=400), Black(N=500)

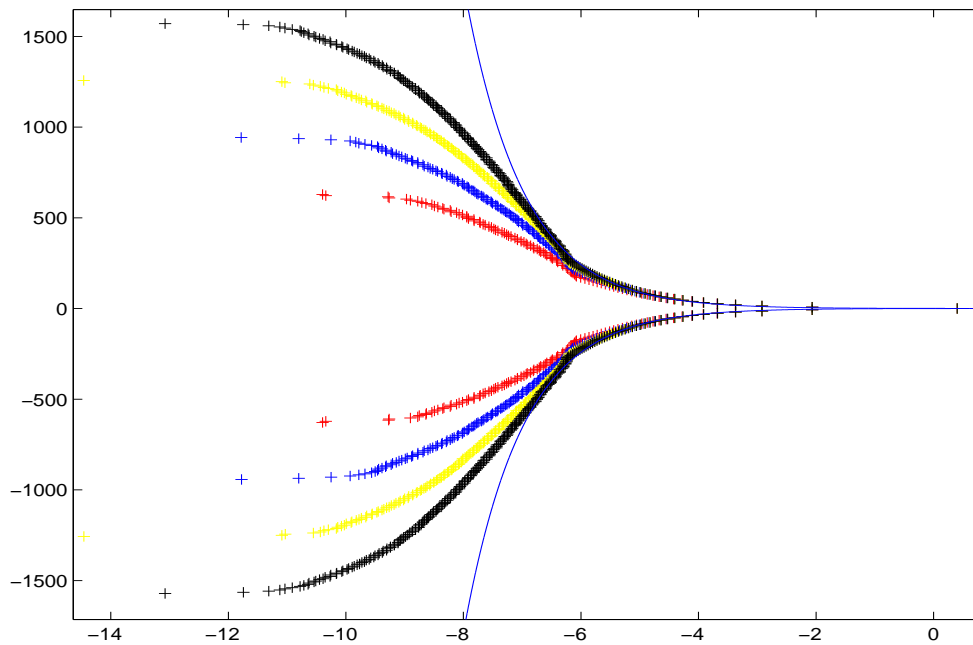


Figure 3.6: Small solutions exist.
 Red(N=200), Blue(N=300), Yellow(N=400), Black(N=500)

Chapter 4

Small solutions in one-dimension

4.1 Introduction

In this chapter we consider equations of the form $x'(t) = b(t)x(t - \tau)$, $b(t + \tau) = b(t)$. Since only one time delay is involved we can normalise the delay, τ , to unity using a simple change of variable. Hence we are able to restrict our investigations to problems where both the time delay and the period of $b(t)$ are equal to unity and consider simple delay differential equations of the form

$$(4.1) \quad x'(t) = b(t)x(t - 1), \quad t \geq 0, \quad x(\theta) = \phi(\theta), \quad -1 \leq \theta \leq 0$$

where $b(t)$ is a bounded, real, continuous, periodic function such that $b(t + 1) = b(t)$, for all $t \geq 0$. We begin by referring to known analytical results relating to equation (4.1) and give an example of an initial function that gives rise to small solutions for an equation of this class. The chapter then focuses on the use of the trapezium rule to show that, for this class of problem, it is indeed possible to identify the presence of small solutions through a numerical approximation.

4.2 Known analytical results

The analysis of (4.1) in which $b(t)$ does not change sign on $[0, 1]$ is well understood [22, 34, 41]. In this case (4.1) is equivalent, in the sense that the long term behaviour of the solution is the same whenever the initial function is the same, to the autonomous problem

$$(4.2) \quad x'(t) = \hat{b}x(t - 1), \quad \text{where } \hat{b} = \int_0^1 b(t)dt,$$

(see example 3.2.1). If $b(t)$ changes sign on $[0, 1]$ it is known analytically that (4.1) has small solutions. In this case there is no autonomous DDE whose dynamical

system is equivalent to that of the non-autonomous DDE (4.1). (See section 2.5.1). The (analytically) known trajectory of the true eigenvalues of equation (4.1) is given in section 3.1.1. In section 3.3 we considered the use of the multi-valued logarithmic function in following the trajectory of the true eigenvalues.

Remark 4.2.1 With reference to earlier examples of equations admitting small solutions: The function $b(t)$ in example (1.3.2) is not periodic and does not change sign on $[0, 1]$ and consequently we will not refer to it again.

In example (1.3.3) the function $b(t)$ is not periodic but using suitable values of a and b we can induce a change of sign on $[0, 1]$. We find that suitable values are such that a and b are of opposite sign and $|\frac{b}{a}| > 2$.

4.3 An initial function that can give rise to small solutions

Consider equation (4.1) when $b(t)$ changes sign on $[-1, 0]$. In this case values of α and β can be found such that $\int_{\alpha}^{\beta} b(s)ds = 0$. Verduyn Lunel in [77] shows how to construct a small solution by iterating the period map, defined by

$$(\Pi\hat{\phi})(\theta) = \hat{\phi}(0) + \int_{-1}^{\theta} \hat{\phi}(s)b(s)ds.$$

The following initial function gives rise to small solutions.

$$\hat{\phi}(\theta) = \begin{cases} 0 & \text{for } -1 \leq \theta \leq \alpha \\ \int_{\alpha}^{\theta} b(s)ds & \text{for } \alpha \leq \theta \leq \beta \\ \int_{\alpha}^{\beta} b(s)ds = 0 & \text{for } \beta \leq \theta \leq 0 \end{cases}.$$

Iterating the period map gives

$$(\Pi^n\phi)(\theta) = \begin{cases} 0 & \text{for } -1 \leq \theta \leq \alpha \\ \frac{1}{(n+1)!} \left[\int_{\alpha}^{\theta} b(s)ds \right]^n & \text{for } \alpha \leq \theta \leq \beta \\ 0 & \text{for } \beta \leq \theta \leq 0 \end{cases}$$

We illustrate with the following example.

Example 4.3.1 Consider $\dot{x}(t) = \sin(2\pi t)x(t-1)$ with initial data given by $\hat{\phi}(\theta)$ with $\hat{\phi}(\theta) = \int_{-1}^{\theta} \sin(2\pi s)ds = \frac{1}{2\pi}[1 - \cos(2\pi\theta)]$, $-1 \leq \theta \leq \alpha$.

We observe that $\hat{\phi}(0) = 0$.

We compute

$$\begin{aligned} (\Pi\hat{\phi})(\theta) &= \int_{-1}^{\theta} \sin(2\pi s) \int_{-1}^s \sin(2\pi\tau)d\tau ds \\ &= \frac{1}{2} \left[\int_{-1}^{\theta} \sin(2\pi\tau)d\tau \right]^2. \end{aligned}$$

We observe that $(\Pi\hat{\phi})(0) = 0$ and again iterate the period map Π .

$$\begin{aligned}\Pi\left(\Pi\hat{\phi}\right)(\theta) &= (\Pi\hat{\phi})(0) + \int_{-1}^{\theta} \sin(2\pi s)\Pi\hat{\phi}(s)ds \\ &= \int_{-1}^{\theta} \sin(2\pi s)\Pi\hat{\phi}(s)ds \\ &= \int_{-1}^{\theta} \sin(2\pi s) \cdot \frac{1}{2} \left[\int_{-1}^s \sin(2\pi\tau)d\tau \right]^2 ds.\end{aligned}$$

Hence,

$$(\pi^2\hat{\phi})(\theta) = \frac{1}{3!} \left[\int_{-1}^{\theta} \sin(2\pi\tau)d\tau \right]^3,$$

and, by induction we can show that

$$(\pi^n\hat{\phi})(\theta) = \frac{1}{(n+1)!} \left[\int_{-1}^{\theta} \sin(2\pi\tau)d\tau \right]^{n+1},$$

which $\rightarrow 0$ faster than any exponential function of the form e^{kt} , $k \in \mathbb{R}$. In Figure 4.1 we show the solution using DDE23.

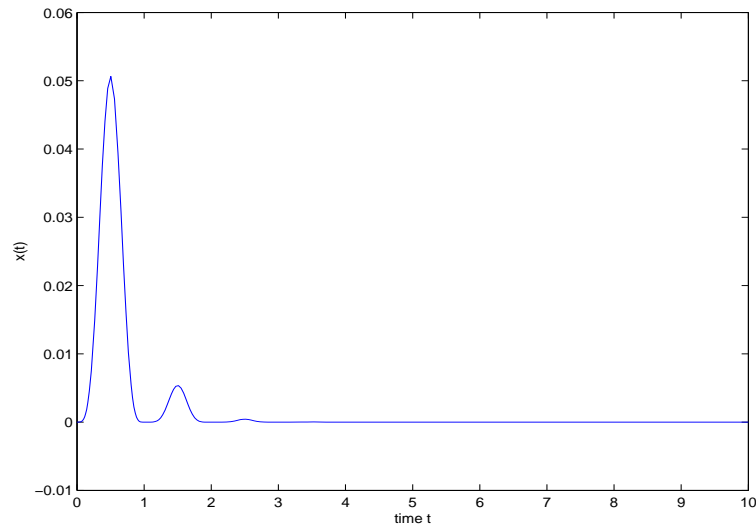


Figure 4.1: Solution of equation in example 4.3.1 using DDE23

Remark 4.3.1 In Figures 4.2 and 4.3 we have used the same initial function as in example 4.3.1, where it gave rise to small solutions, but with equation $b(t) = (\sin(2\pi t) + c)x(t-1)$. Comparing Figure 4.1 with the diagram in Figure 4.2 we observe very different behaviour of the solution. We see that although

both equations admit small solutions (due to the sign change of $b(t)$) this initial function has not induced them. In Figure 4.3 the functions used for $b(t)$ are ‘close’ to that in the example. The solution shown in the right-hand diagram appears to be oscillatory and does not appear to be approaching zero. Clearly there are difficulties in judging whether or not a solution is small by this method.

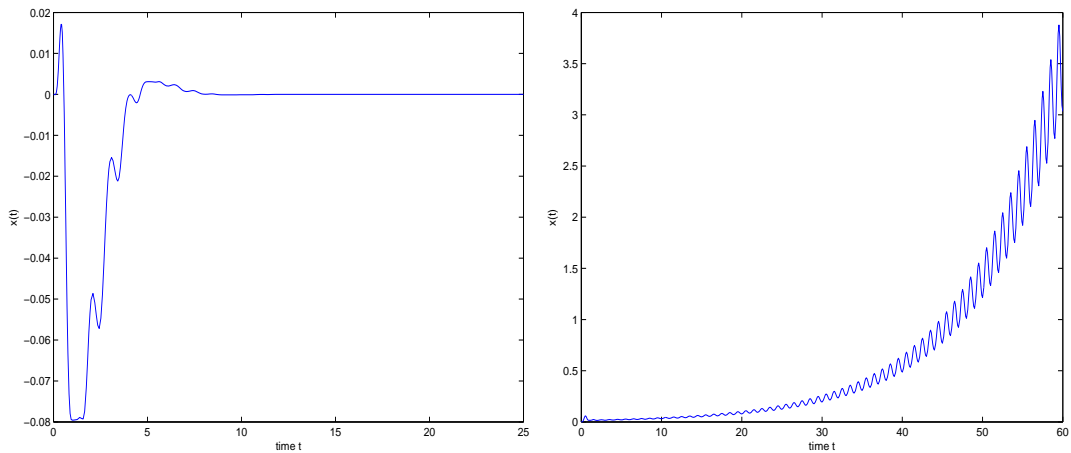


Figure 4.2: Solutions of $\dot{x}(t) = (\sin(2\pi t) + c)x(t - 1)$. Initial function as in example 4.3.1.

Left: $c = -0.5$, Right: $c = 0.1$

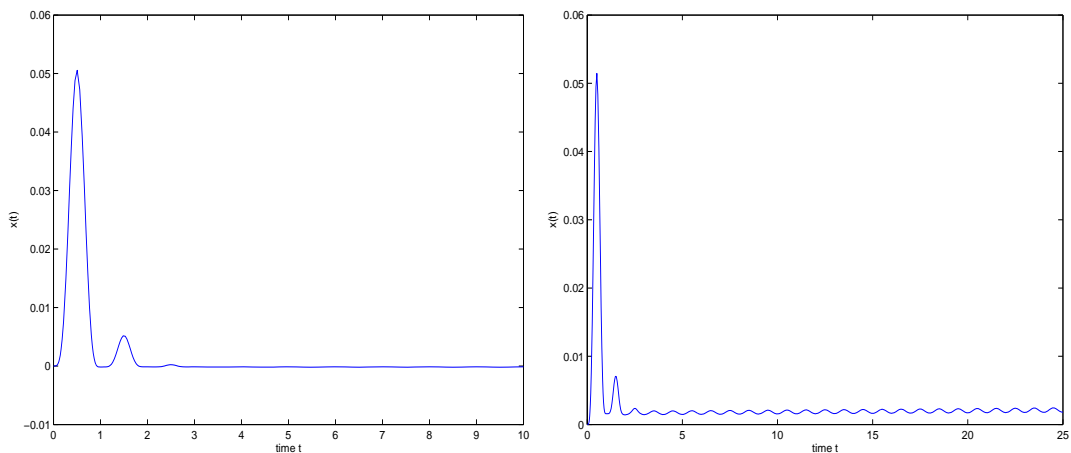


Figure 4.3: Solutions of $\dot{x}(t) = (\sin(2\pi t) + c)x(t - 1)$ with initial function as in example 4.3.1. Left: $c = -0.001$, Right: $c = 0.01$

Remark 4.3.2 In Figure 4.4 the same DDE has been used as in example 4.3.1 but with different initial functions. Here we observe oscillatory behaviour and

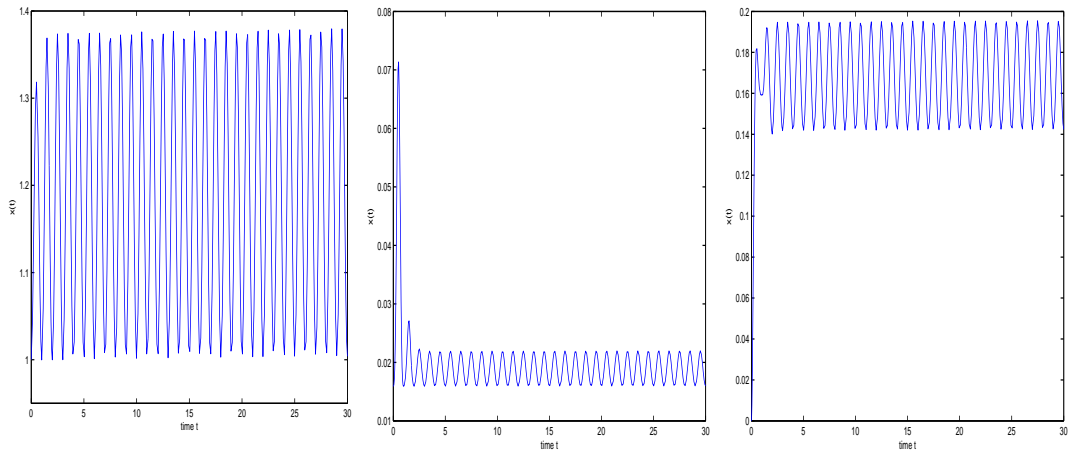


Figure 4.4: Initial functions: Left: $\phi(t) = 1$;
 Centre: $\phi(t) = \frac{1}{2\pi}[1.1 - \cos(2\pi t)]$; Right: $\phi(t) = t$

the solutions are clearly not tending to zero. Again, the equation admits small solutions but an appropriate initial function must be chosen.

Remark 4.3.3 In Figure 4.5 we solve $x'(t) = (\sin(2\pi t) - 1.5)x(t-1)$ with initial function as in example 4.3.1. The left-hand diagram shows the solution for up to $t = 100$ and the right-hand diagram shows the behaviour of the solution for later values of t . This equation does not admit small solutions. For $t \leq 100$

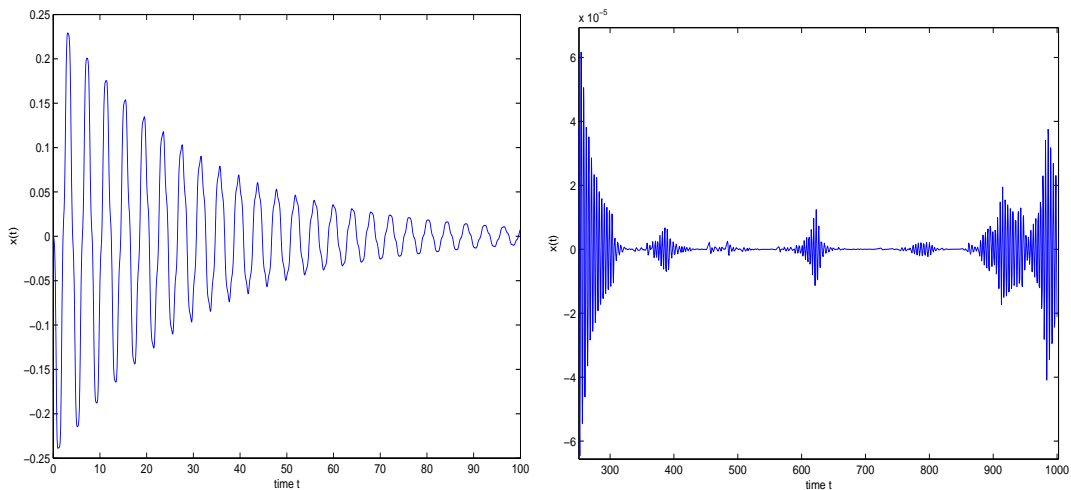


Figure 4.5: A potential problem: Extrapolation into the future.

the solution appears to follow a definite 'pattern' (oscillating and decreasing).

For the problems we are considering we can use the periodicity of $b(t)$ to deduce that $A(n) = A(n - N)$ for all $n > N$. For $n = 1, N + 1, 2N + 1, \dots$ we can then write

$$(4.8) \quad y_{n+N} = Cy_n,$$

where

$$(4.9) \quad C = \prod_{i=1}^N A(N - i).$$

Similar expressions could be derived for $n = j, N + j, 2N + j, \dots$ with a suitable rearrangement of the order of multiplication of the matrices. C is the discrete analogue of the solution map in the continuous case. Theorem 3.2 of [27] (see appendix C) guarantees that the eigenvalues of matrix C can be used as a good approximation to the eigenvalues of the solution operator of equation (4.4). We are now able to use the eigenvalues of C to investigate the dynamical behaviour of the solution to (4.6), since (4.8) is an autonomous problem which is equivalent to (4.6) in the sense that the solutions of (4.8) coincide with every N th term in the solution of (4.6) whenever the initial vector is the same. Thus the dynamical behaviour of (4.6) is identical to that of (4.8).

We are dealing with an infinite dimensional problem. Applying a numerical method has the effect of reducing the problem to a finite dimensional case. The dimension of C is specified by the choice of step-length. The behaviour of some of the functions $b(t)$ in the continuous case may not be fully evident in the discrete version.

Example 4.4.1 To illustrate the general principle of applying a numerical method to solve (4.3) we use the trapezium rule which gives us

$$(4.10) \quad x_{n+1} = x_n + \frac{h}{2}b_n x_{n-N} + \frac{h}{2}b_{n+1} x_{n+1-N}$$

We thus obtain

$$(4.11) \quad y_{n+1} = \begin{pmatrix} x_{n+1} \\ x_n \\ \vdots \\ \vdots \\ \vdots \\ x_{n+1-N} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \cdots & 0 & \frac{h}{2}b_{n+1} & \frac{h}{2}b_n \\ 1 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & 1 & \ddots & \ddots & & \vdots \\ \vdots & \ddots & 1 & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x_n \\ x_{n-1} \\ \vdots \\ \vdots \\ \vdots \\ x_{n-N} \end{pmatrix}$$

and find that the matrix C , as defined in (4.9), takes the form

$$(4.12) \quad C = \begin{pmatrix} 1 + \frac{h}{2}b_{n+1} & hb_n & hb_{n-1} & \dots & \dots & hb_2 & \frac{h}{2}b_1 \\ 1 & \frac{h}{2}b_n & hb_{n-1} & \dots & \dots & hb_2 & \vdots \\ 1 & 0 & \frac{h}{2}b_{n-1} & \ddots & & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & & & \ddots & \frac{h}{2}b_3 & hb_2 & \vdots \\ \vdots & \vdots & & & 0 & \frac{h}{2}b_2 & \frac{h}{2}b_1 \\ 1 & 0 & \dots & \dots & \dots & 0 & 0 \end{pmatrix}$$

The choice of step size

The finite dimensional numerical scheme can reproduce the dynamical behaviour of the infinite dimensional problem only in the limit as $h \rightarrow 0$ (that is, as the dimension $\rightarrow \infty$) [34]. We can, of course, decrease h indefinitely towards the limit of 0. However, this course of action is impractical. In Figure 4.6 we illustrate the effect of increasing the value of N , thus decreasing the step size. We use, as illustration, the eigenspectra arising from the discretisations of (4.3) with $b(t) = \sin(2\pi t) + 1.5$ and $b(t) = 1.5$ using four different step sizes. (We note that $b(t)$ does not change sign and hence the equation does not admit small solutions.) We observe, as expected, that the approximation of the the non-autonomous system by the equivalent autonomous system improves as h decreases.

We choose to use fixed step-length schemes with the step size h given by $h = \frac{1}{N}$. The discussion in the previous paragraph indicates that the use of an algorithm for selecting the step-length to improve the approximation would, in the presence of small solutions, result in the selection of smaller and smaller step-lengths until the lower limit of machine accuracy was reached. Hence we do not anticipate any advantage in adopting a variable step-length scheme in our numerical discretisation of equation (4.3) (see [34]) .

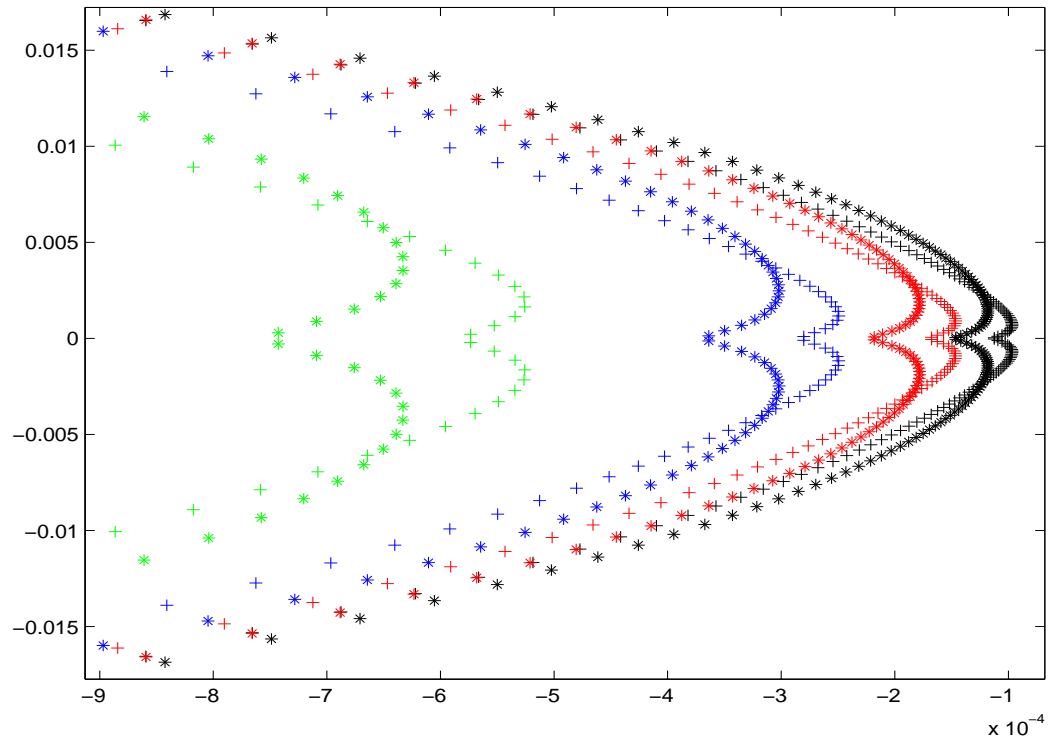


Figure 4.6: The approximation improves as the step size $h = \frac{1}{N}$ decreases. (Green: $N=60$, Blue: $N=90$, Red: $N=120$, Black: $N=150$)

A range of values of h have been used in our experiments. In Figures 4.7 and 4.8 we display the eigenspectra arising from discretisation of $x'(t) = (\sin 2\pi t + 0.4)x(t-1)$ using the trapezium rule with $N=32, 64, 128$ and 256 . ($b(t)$ changes sign and the equation admits small solutions).

Eigenvalue trajectories for different values of N ; $b(t) = \sin 2\pi t + 0.4$

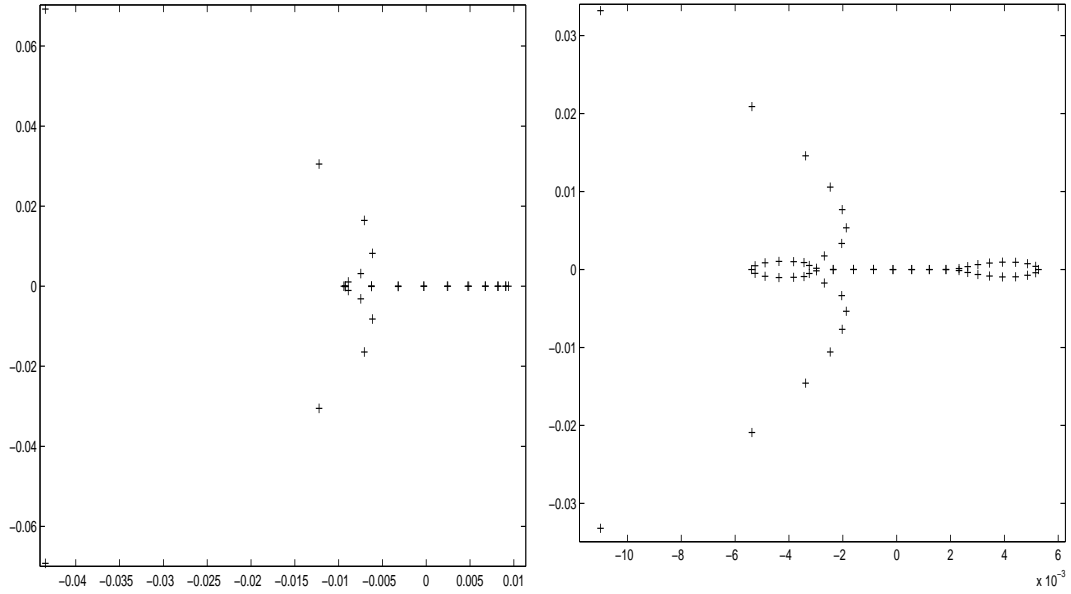


Figure 4.7: Left: $N = 32$ Right: $N = 64$

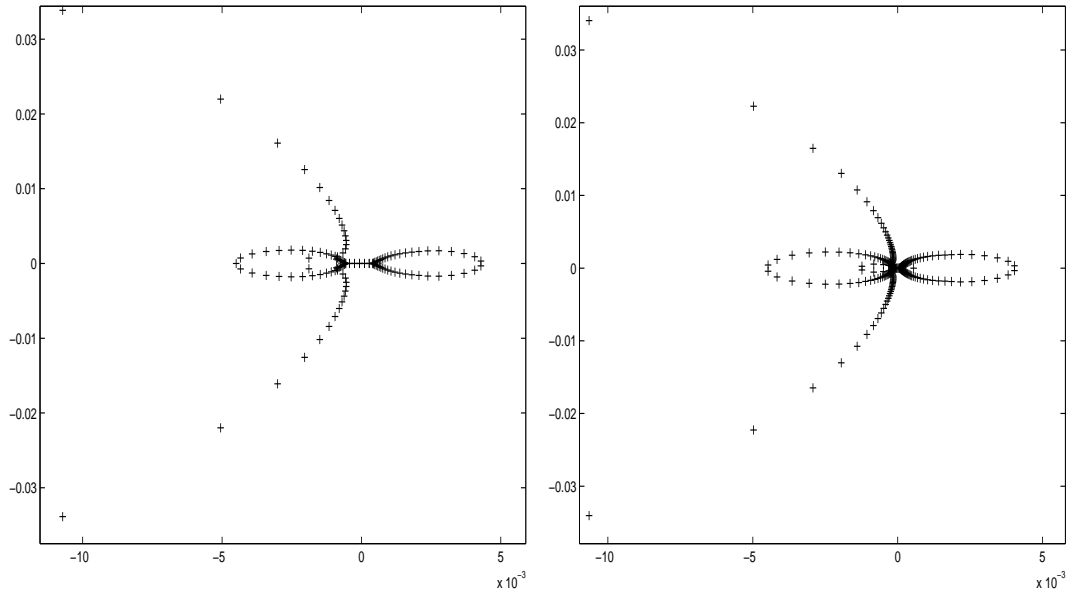


Figure 4.8: Left: $N = 128$ Right: $N = 256$

After due consideration we feel that using $N = 128$ provides a good compromise between clarity and speed. Hence, all future diagrams of eigenspectra in chapters 4 and 5 are illustrative of the case when $N = 128$.

In a finite dimensional scheme small solutions, as defined in section 1.3.1, do not exist. However, we expect the presence of small solutions in the continuous problem to be indicated by the presence of small non-zero eigenvalues in the discrete scheme [34]. We anticipate that as $h \rightarrow 0$ the eigenvalues corresponding to small solutions will tend to the limit 0, with all other eigenvalues approaching non-zero limits equal to an eigenvalue of the continuous scheme.

4.5 Results of applying the trapezium rule

We adopt the approach taken in [33]. We calculate the eigenspectrum for the solution operator of (4.2) and compare this with the eigenspectrum of the solution operator of the non-autonomous scheme (4.1). When $b(t)$ does not change sign we find that the trajectory of the eigenvalues calculated in the numerical solution of (4.1) follows closely that of the eigenvalues calculated in the numerical solution to (4.2) and we illustrate this in example 4.5.1.

Example 4.5.1 We apply the trapezium rule to (4.1) with $b(t) = \sin 2\pi t + 1.5$, from which it follows that $\hat{b} = 1.5$. The results of applying the numerical approximation are shown in Figure 4.9 and we note that although Figure 4.9 focuses on the eigenvalues near the origin, the proximity of the two trajectories to each other is clearly evident.

When $b(t)$ changes sign we observe not only a trajectory close to that from the autonomous problem, but also an additional trajectory, passing through the origin and including two ‘circles’. We take the appearance of the additional trajectory as visual evidence of the non-equivalence of the two problems and evidence that the equation admits small solutions. We illustrate this in example 4.5.2.

Example 4.5.2 We apply the trapezium rule to (4.1) with $b(t) = \sin 2\pi t + 0.4$. In this case $b(t)$ changes sign on $[0, 1]$ and $\hat{b} = 0.4$. We again focus our attention on eigenvalues lying near to the origin. Results of our numerical approximation are shown in Figure 4.10. We note the clear evidence of an additional trajectory, indicating that the non-autonomous problem admits small solutions.

The characteristic shape of the eigenspectrum, arising from discretisation using the trapezium rule of an autonomous problem of the form (4.2), is that indicated by the $*$ in Figures 4.9 and 4.10. In our discussions concerning the evidence for the existence, or otherwise, of small solutions we are comparing the

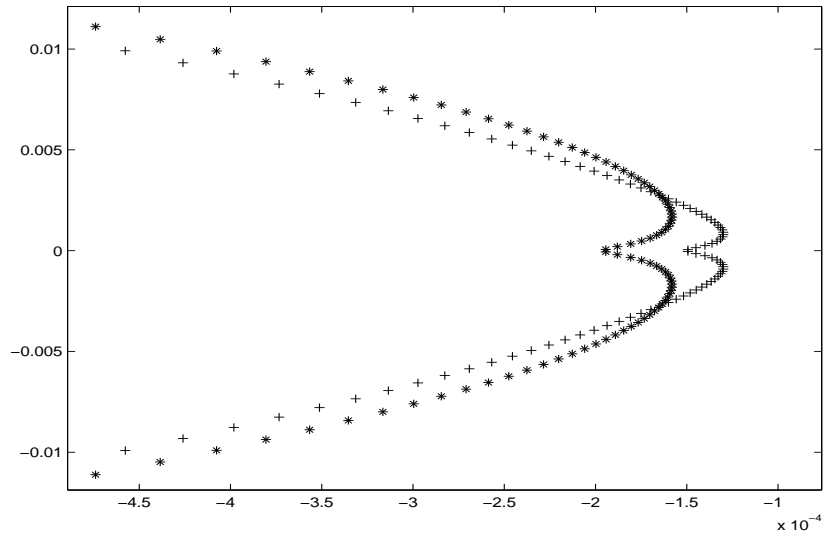


Figure 4.9: Comparison of eigenspectrum for C with $b(t) = (\sin 2\pi t + 1.5)$ with that when $b(t) = 1.5$.

The eigenspectra are very similar. The equation does not admit small solutions. The two problems are equivalent

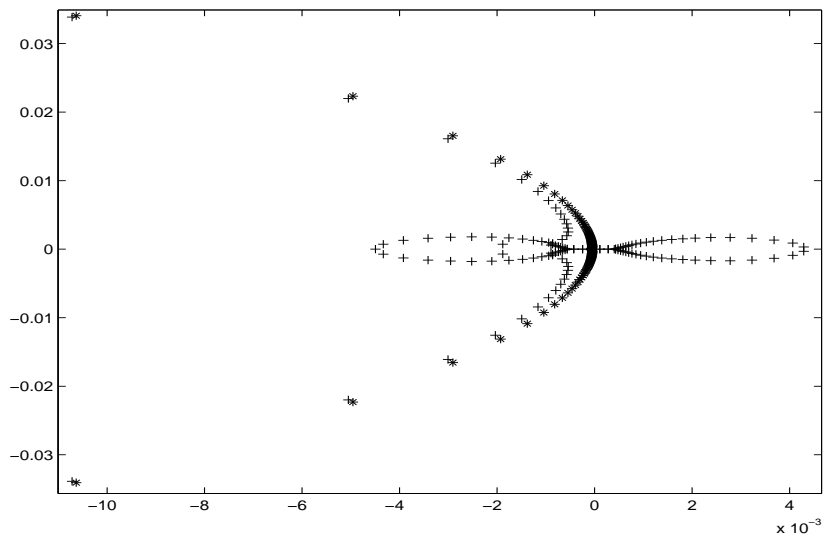


Figure 4.10: Comparison of eigenspectra for C with $b(t) = (\sin 2\pi t + 0.4)$ and C with $b(t) = 0.4$

Clear differences in the eigenspectra are visible. The equation admits small solutions. An equivalent autonomous problem does not exist

eigenvalue trajectories illustrated in this chapter with this shape, that is with the eigenvalue trajectory arising from the discretisation of the autonomous problem as defined in (4.2).

4.5.1 Further examples

We now consider the results of applying the trapezium rule to (4.1) in which $b(t)$ takes one of the following simple forms:

$$(4.13) \quad b_1(t) = \sin 2\pi t + c_1$$

$$(4.14) \quad b_2(t) = t - \frac{1}{2} + c_2 \text{ for } t \in [0, 1], \quad b_2(t) = b_2(t - 1) \text{ for } t > 1.$$

$$(4.15) \quad b_3(t) = t(t - \frac{1}{2})(t - 1) + c_3 \text{ for } t \in [0, 1], \quad b_3(t) = b_3(t - 1) \text{ for } t > 1.$$

$$(4.16) \quad b_4(t) = \begin{cases} -1 + c_4 & \text{for } t \in (0, \frac{1}{2}] \\ 1 + c_4 & \text{for } t \in (\frac{1}{2}, 1] \end{cases}, \quad b_4(t) = b_4(t - 1) \text{ for } t > 1.$$

Each of these functions has period equal to 1. For each function, putting the appropriate constant, c_i , equal to zero produces a function $b_i(t)$ such that $\int_0^1 b_i(t) dt = 0$.

We separate the diagrams for each function $b_i(t)$ into three categories, defined as follows:

Category A: $b_i(t)$ does not change sign on $[0, 1]$.

Category B: $b_i(t)$ changes sign on $[0, 1]$ and $\int_0^1 b_i(t) dt = 0$.

Category C: $b_i(t)$ changes sign on $[0, 1]$ and $\int_0^1 b_i(t) dt \neq 0$.

We hope to identify characteristics of the eigenspectra that indicate the presence, or otherwise, of small solutions.

In the following diagrams we again focus our attention on eigenvalues which lie close to the origin. In our experiments we observed that the eigenspectrum differed according to which of the three categories A, B or C , $b_i(t)$ belongs, rather than on whether $b_i(t)$ was a trigonometric function, a linear function, etc. To illustrate these differences we include one example from each category from each of the four functions $b_i(t)$, for $i = 1, 2, 3, 4$. and present the eigenvalue trajectories for $c_1 = 1.6, 0, 0.5$, $c_2 = 1, 0, 0.2$, $c_3 = 0.25, 0, \frac{1}{64}$ and $c_4 = 1.5, 0, 0.5$.

Category A: Eigenspectra when $b_i(t)$ does not change sign on $[0, 1]$.
 The equation does not admit small solutions

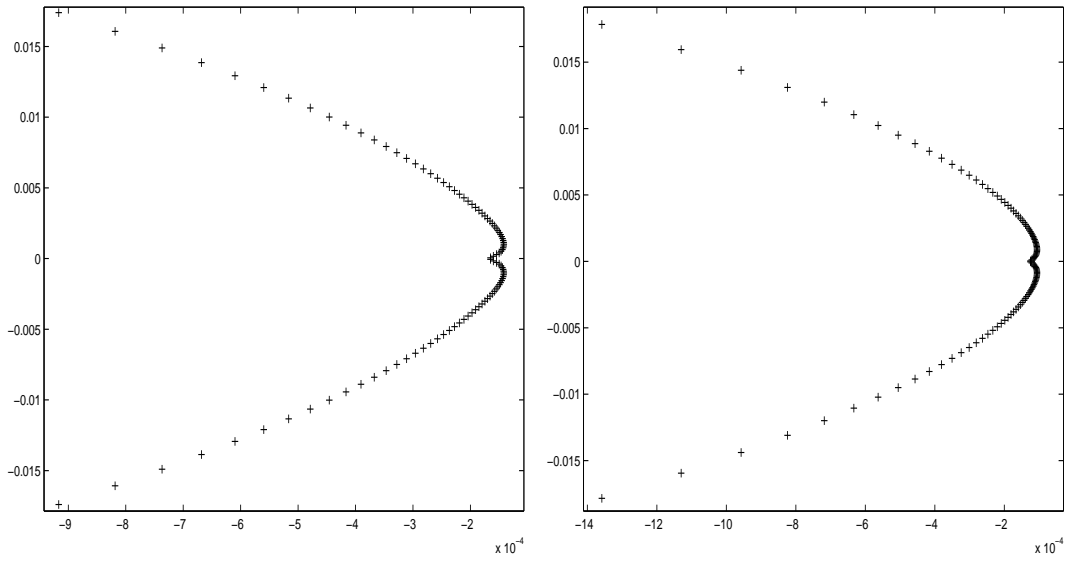


Figure 4.11: Eigenspectra for C from: Left: Equation (4.13) with $c_1 = 1.6$
 Right: Equation (4.14) with $c_2 = 1$

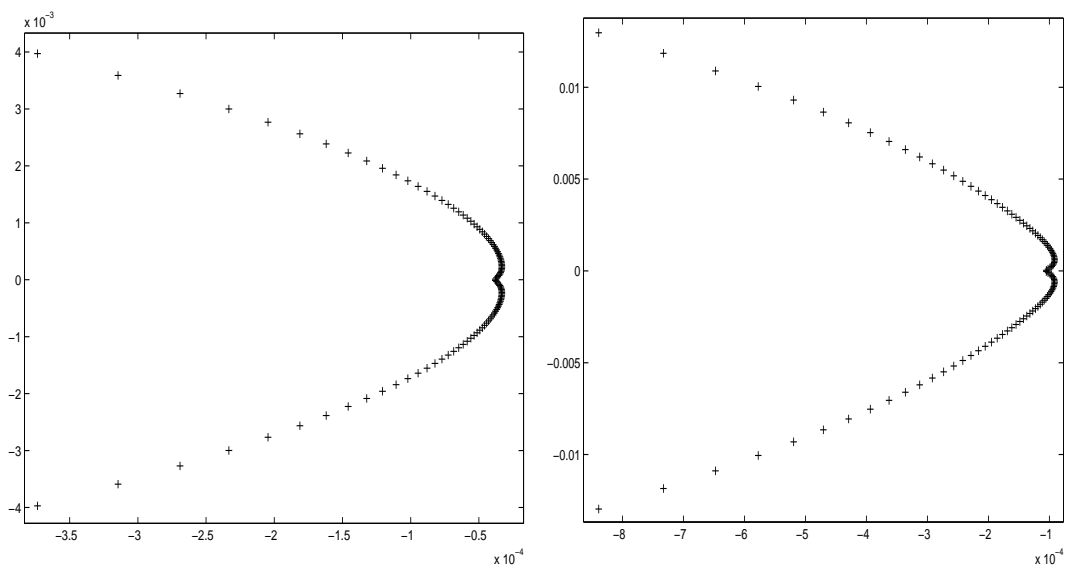


Figure 4.12: Eigenspectra for C from: Left: Equation (4.15) with $c_3 = 0.25$
 Right: Equation (4.16) with $c_4 = 1.5$

Category B: Eigenspectra when $b_i(t)$ changes sign on $[0, 1]$ and $\int_0^1 b_i(t) = 0$.
 The majority of solutions are small

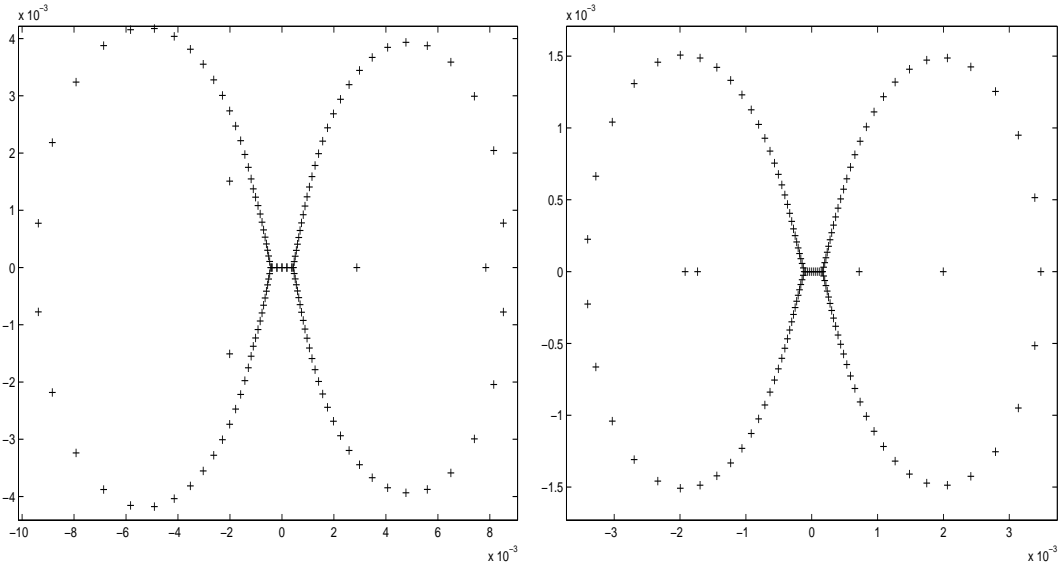


Figure 4.13: Eigenspectra for C from: Left: Equation (4.13) with $c_1 = 0$
 Right: Equation (4.14) with $c_2 = 0$

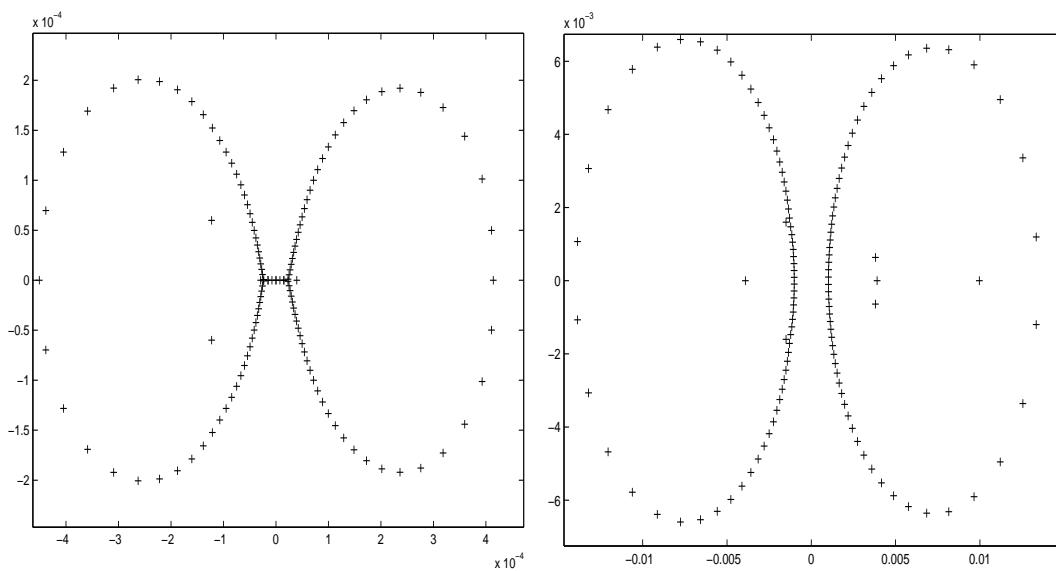


Figure 4.14: Eigenspectra for C from: Left: Equation (4.15) with $c_3 = 0$
 Right: Equation (4.16) with $c_4 = 0$

Category C: Eigenspectra when $b_i(t)$ changes sign on $[0, 1]$ and $\int_0^1 b_i(t) \neq 0$.
 The equation admits small solutions

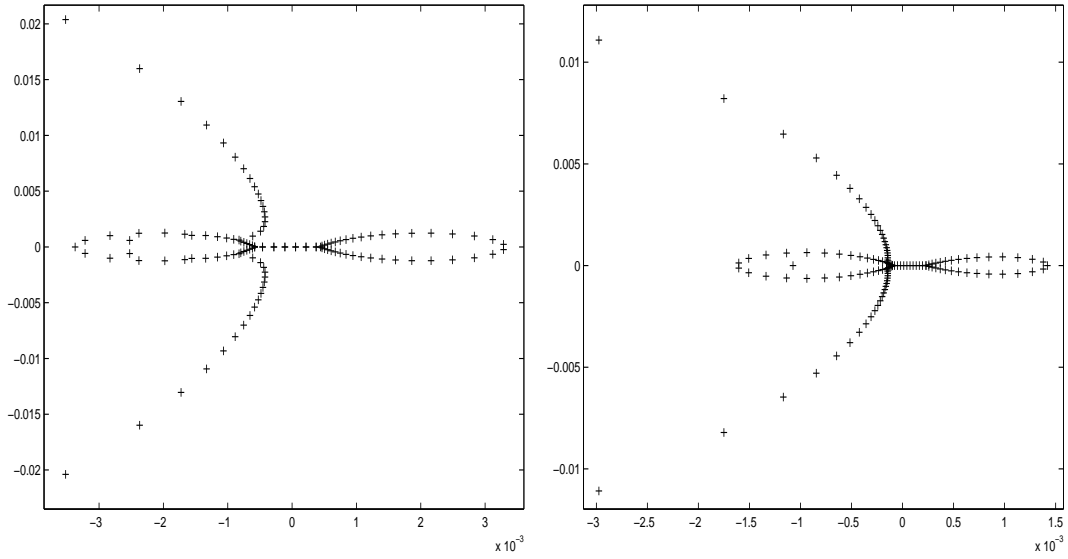


Figure 4.15: Eigenspectra for C from: Left: Equation (4.13) with $c_1 = 0.5$
 Right: Equation (4.14) with $c_2 = 0.2$

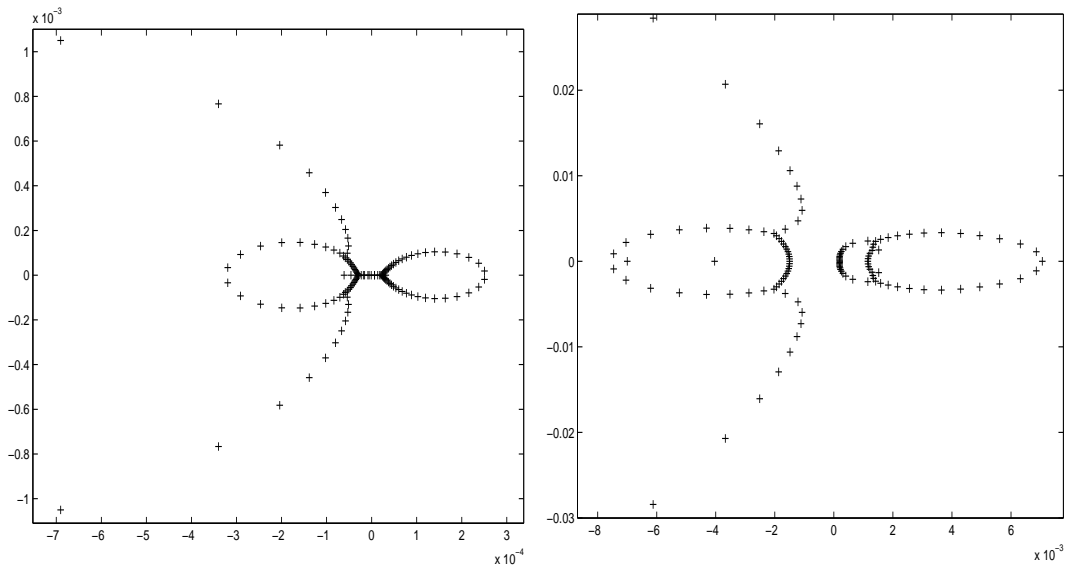


Figure 4.16: Eigenspectra for C from: Left: Equation (4.15) with $c_3 = \frac{1}{64}$
 Right: Equation (4.16) with $c_4 = 0.5$

We observe that in the one-dimensional case when the eigenvalue is necessarily real, then the characteristic shape of the eigenspectrum arising from the non-autonomous problem depends upon whether or not $b_i(t)$ changes sign on $[0, 1]$, that is on whether or not the equation admits small solutions.

For category A, (see Figures 4.11 and 4.12), when $b_i(t)$ does not change sign on $[0, 1]$, we notice the absence of small solutions.

For category B, when $b_i(t)$ changes sign on $[0, 1]$ but $\int_0^1 b_i(t)dt = 0$, then Figures 4.13 and 4.14 indicate that most of the solutions are small.

For category C, when $b_i(t)$ changes sign on $[0, 1]$ and $\int_0^1 b_i(t)dt \neq 0$, we observe, in Figures 4.15 and 4.16, a combination of the trajectories seen in the previous two cases. The equation admits small solutions.

The clear indication of the presence of small solutions which we observed in our diagrams is consistent with the known theory. Experiments with other choices of 1-periodic functions $b_i(t)$, including logarithmic, exponential and rational functions, and other choices of constant c_i produced similar results. We conclude that we can, using a numerical approximation, reliably identify whether or not an equation of the form (4.1) admits small solutions. We now question whether insight from our experimental work would be enhanced by the use of an alternative numerical method, possibly one of higher order. In the next chapter we assess the reliability, ease and clarity with which the detection of small solutions to equation (4.1) can be achieved by considering other numerical schemes.

Chapter 5

Choice of numerical scheme

5.1 Introduction

In chapter 4 we established that the presence of small solutions to linear non-autonomous delay differential equations of the form (4.1) can be identified through the use of the trapezium rule. Earlier work in [33] used the backward Euler method to discretise the equation. We now consider whether the use of an alternative numerical discretisation scheme might improve the ease and clarity with which the phenomenon of small solutions can be detected. As in section 4.5.1 we show only the trajectory arising from the non-autonomous problem in each diagram and take the presence of more than one asymptotic trajectory as an indication that small solutions are admitted.

5.2 The Adams-Moulton method

The trapezium rule, used in chapter 4, is a second order method. In order to assess whether any advantage could be gained by using a higher order method we solved similar problems using the Adams-Moulton method of order 3. As an illustration of our results we present eigenspectra arising from the following problems using the Adams-Moulton method of order 3.

$$b_1(t) = \sin 2\pi t + c_1$$

$$b_2(t) = t - \frac{1}{2} + c_2 \text{ for } t \in [0, 1], \quad b_2(t) = b_2(t - 1) \text{ for } t > 1.$$

$$b_3(t) = t(t - \frac{1}{2})(t - 1) + c_3 \text{ for } t \in [0, 1], \quad b_3(t) = b_3(t - 1) \text{ for } t > 1.$$

$$b_4(t) = \begin{cases} -1 + c_4 & \text{for } t \in (0, \frac{1}{2}] \\ 1 + c_4 & \text{for } t \in (\frac{1}{2}, 1] \end{cases}, \quad b_4(t) = b_4(t - 1) \text{ for } t > 1.$$

To enable easier comparison of the two methods we include, by way of illustration, the results from using the same values of c_i . Hence we present the

eigenvalue trajectories for $c_1 = 1.6, 0, 0.5$, $c_2 = 1, 0, 0.2$, $c_3 = 0.25, 0, \frac{1}{64}$ and $c_4 = 1.5, 0, 0.5$ and again separate the diagrams for each function $b_i(t)$ into the three categories A, B and C , defined in section 4.5.1. We again observe clear evidence of the presence of small solutions in Figures 5.3, 5.4, 5.5 and 5.6. However, using this higher order method has not improved upon the clarity with which we detected small solutions using the trapezium rule.

Eigenspectra when $b_i(t)$ does not change sign on $[0, 1]$.

The equation does not admit small solutions. Eigenvalues lie on one characteristic curve.

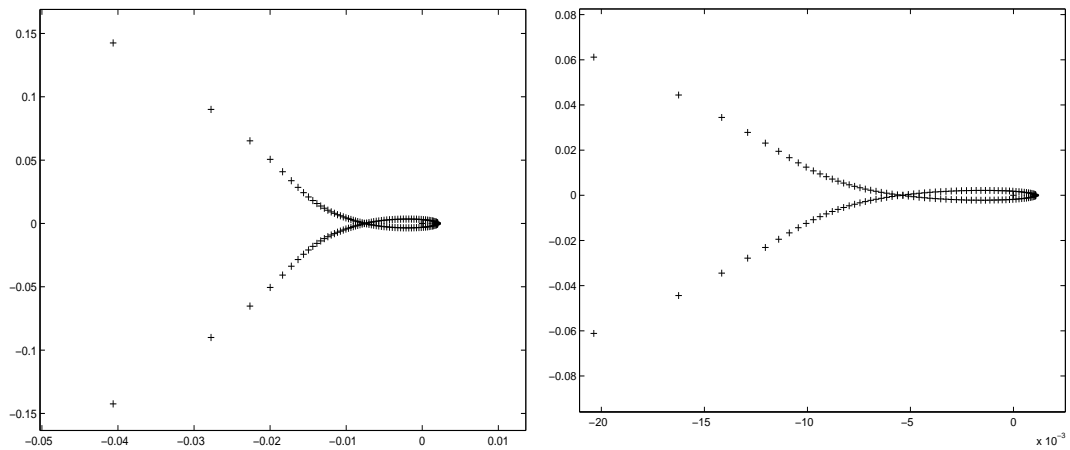


Figure 5.1: Left: $c_1 = 1.6$ Right: $c_2 = 1$

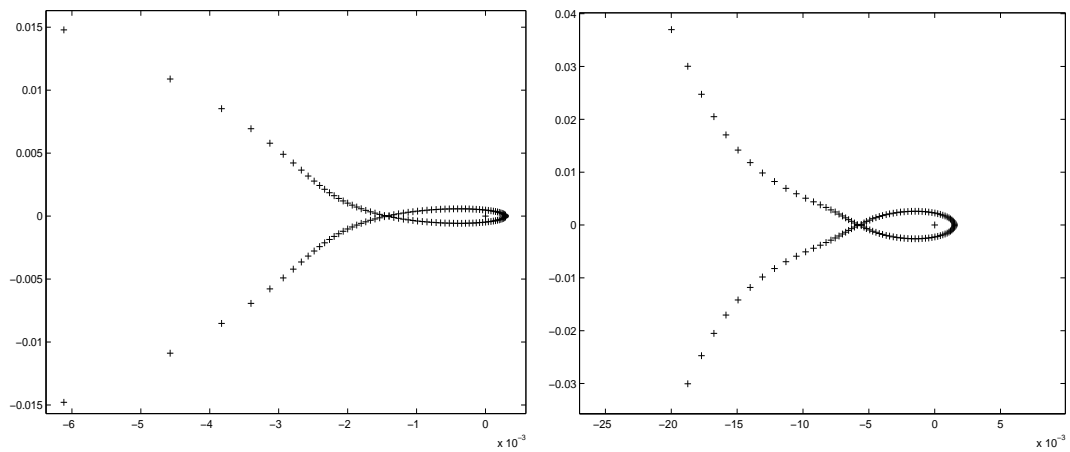


Figure 5.2: Left: $c_3 = 0.25$ Right: $c_4 = 1.5$

Eigenspectra when $b_i(t)$ changes sign on $[0, 1]$ and $\int_0^1 b_i(t) = 0$.
 The majority of solutions are small. More than one asymptotic curve is present.

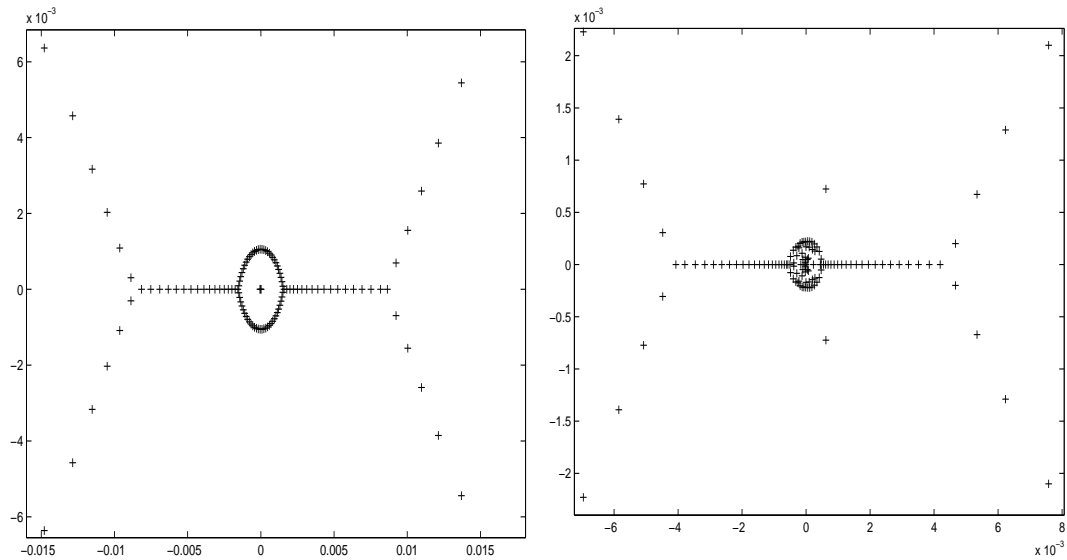


Figure 5.3: Left: $c_1 = 0$ Right: $c_2 = 0$

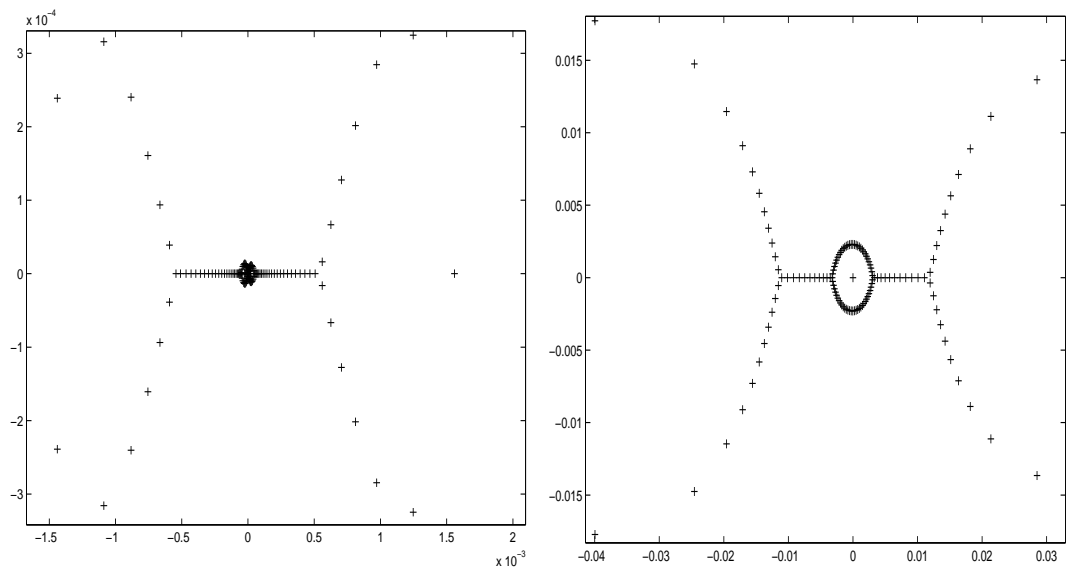


Figure 5.4: Left: $c_3 = 0$ Right: $c_4 = 0$

Eigenspectra when $b_i(t)$ changes sign on $[0, 1]$ and $\int_0^1 b_i(t) \neq 0$.
 The equation admits small solutions. The eigenvalues do not all lie on the same asymptotic curve.

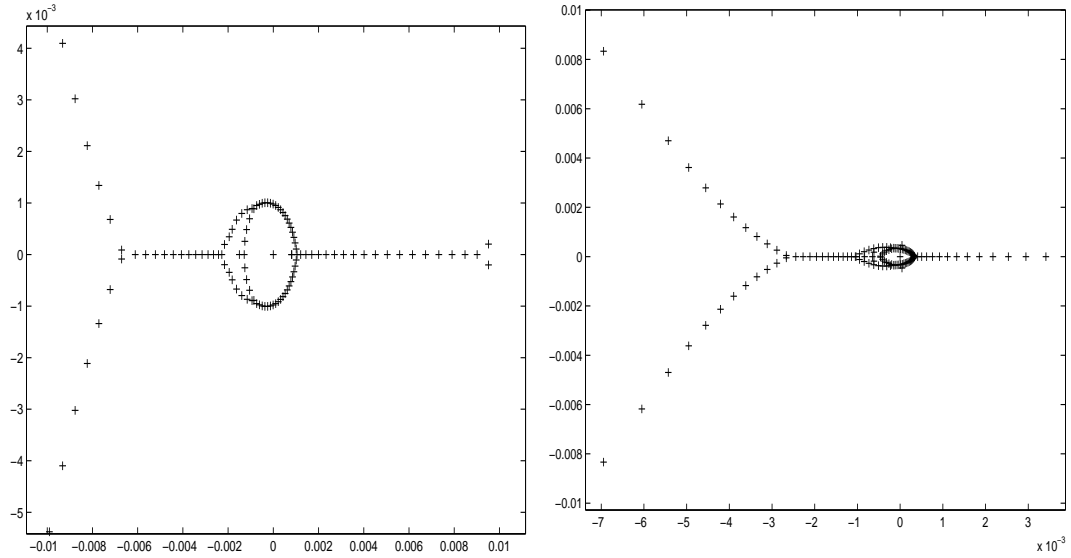


Figure 5.5: Left: $c_1 = 0.5$ Right: $c_2 = 0.2$

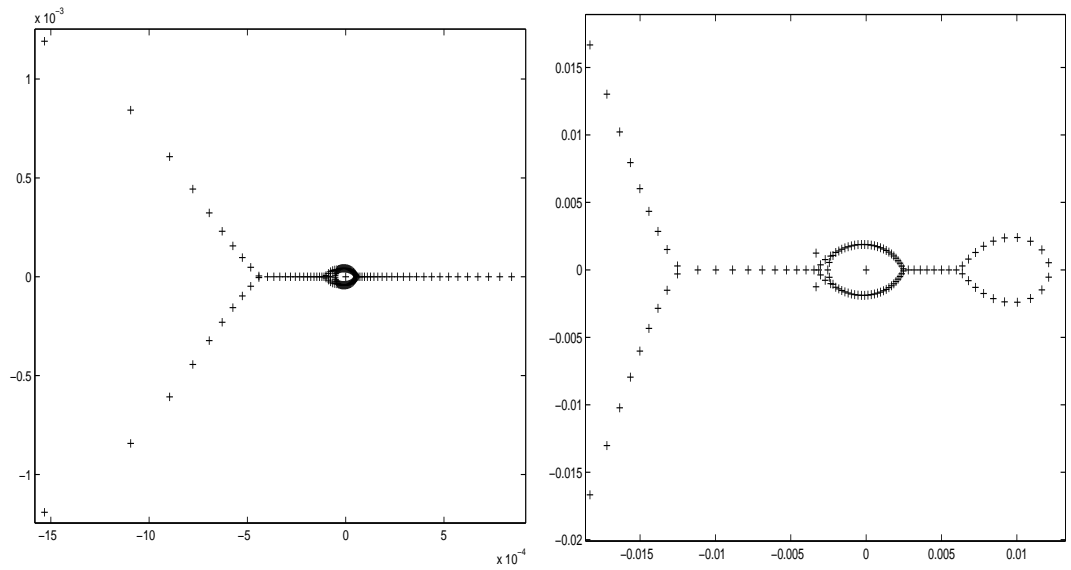


Figure 5.6: Left: $c_3 = \frac{1}{64}$ Right: $c_4 = 0.5$

5.3 Comparing five different numerical methods

We are looking for a numerical method which will clearly and reliably indicate the presence of small solutions. In our search for a method that might improve upon the results of using the trapezium rule we considered other numerical approximation schemes. We present some results of using the following schemes:

Method 1: The Forward Euler Method.

Method 2: The Backward Euler Method.

Method 3: The Trapezium Rule.

Method 4: The Adams-Bashforth method of order 2.

Method 5: The Adams-Moulton method of order 3.

Methods 1 and 2 are also θ -methods. Method 4 is of the same order as the trapezium rule and method 5 is of higher order. To continue our interest in the comparison of numerical schemes we compare the relative merits of using the above five methods to solve equations (4.13) and (4.15).

5.3.1 Results

We illustrate some of the comparisons for each of the three categories A , B and C , already established in section 4.5.1, in the following diagrams in which, as usual, we focus our attention on eigenvalues that lie close to the origin. To enable easier comparison of the five methods we choose to repeat the eigenspectra arising from the application of the trapezium rule and the Adams-Moulton method of order 3.

Figures 5.7, 5.8, 5.9, 5.10 and 5.11 illustrate the case when small solutions are not present.

Figures 5.12, 5.13, 5.14, 5.15 and 5.16 illustrate the case when most of the solutions are small solutions.

Figures 5.17, 5.18, 5.19, 5.20 and 5.21 illustrate the case when the solutions to the equation include small solutions.

The values of the c_i have been chosen arbitrarily within the constraints imposed on our functions $b_i(t)$ for each category. Similar diagrams resulted when other values satisfying the relevant constraints on $b_i(t)$ were used.

Eigenspectra when $b_i(t)$ does not change sign on $[0, 1]$.

The equation does not admit small solutions.

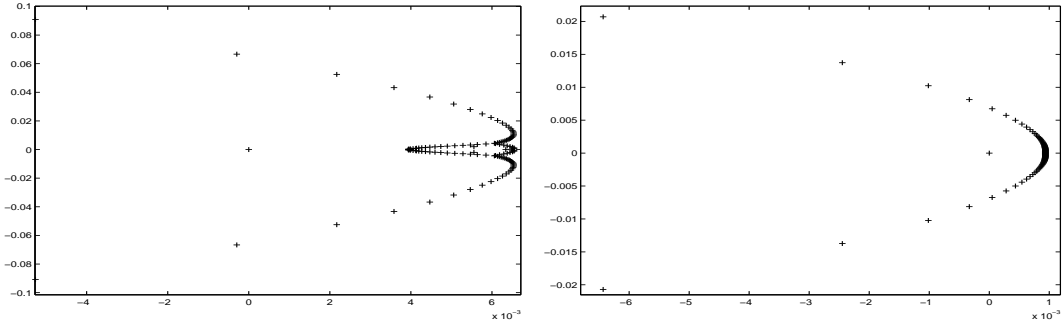


Figure 5.7: Forward Euler : Left: $b_1 = \sin 2\pi t + 1.6$

Right: $b_3 = t(t - \frac{1}{2})(t - 1) + 1/4$

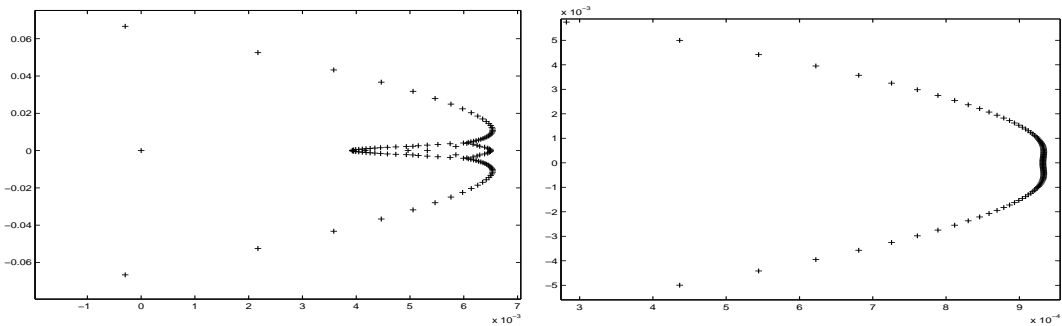


Figure 5.8: Backward Euler: Left: $b_1 = \sin 2\pi t + 1.6$

Right: $b_3 = t(t - \frac{1}{2})(t - 1) + 1/4$

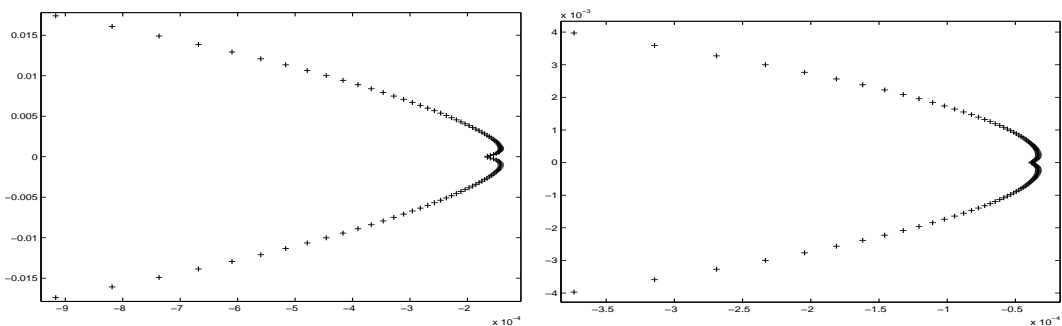


Figure 5.9: Trapezium rule: Left: $b_1 = \sin 2\pi t + 1.6$

Right: $b_3 = t(t - \frac{1}{2})(t - 1) + 1/4$

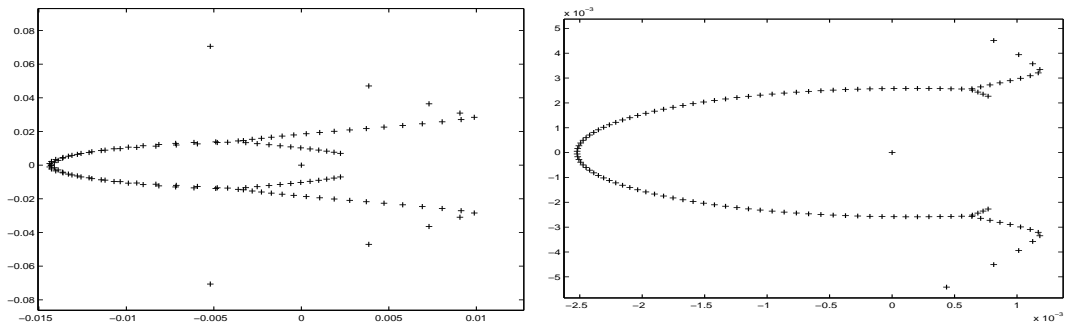


Figure 5.10: Adams-Bashforth (order 2):
 Left: $b_1 = \sin 2\pi t + 1.6$ Right: $b_3 = t(t - \frac{1}{2})(t - 1) + 1/4$

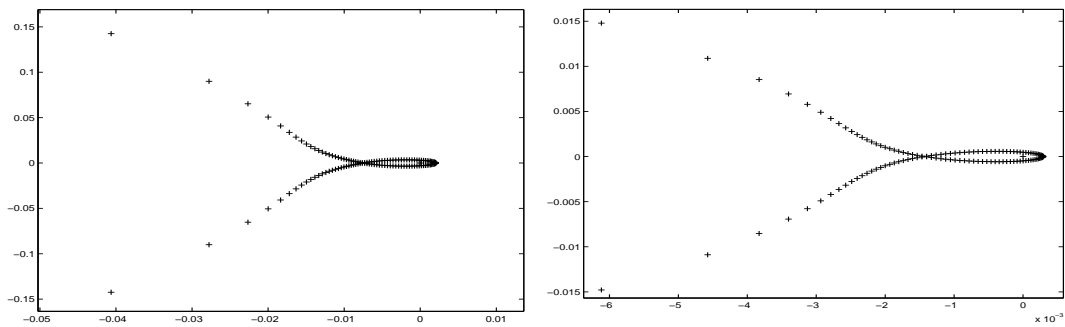


Figure 5.11: Adams-Moulton (order 3):
 Left: $b_1 = \sin 2\pi t + 1.6$ Right: $b_3 = t(t - \frac{1}{2})(t - 1) + 1/4$

Eigenspectra when $b_i(t)$ changes sign on $[0, 1]$ and $\int_0^1 b_i(t) = 0$.

Most of the solutions are small solutions.

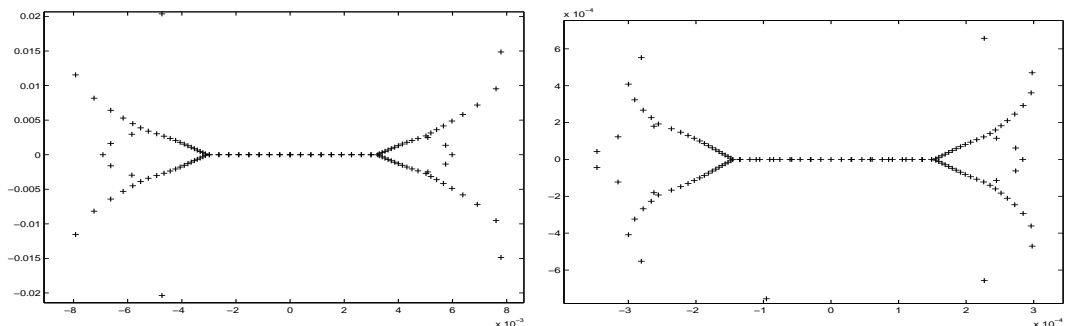


Figure 5.12: Forward Euler : Left: $b_1 = \sin 2\pi t$ Right: $b_3 = t(t - \frac{1}{2})(t - 1)$

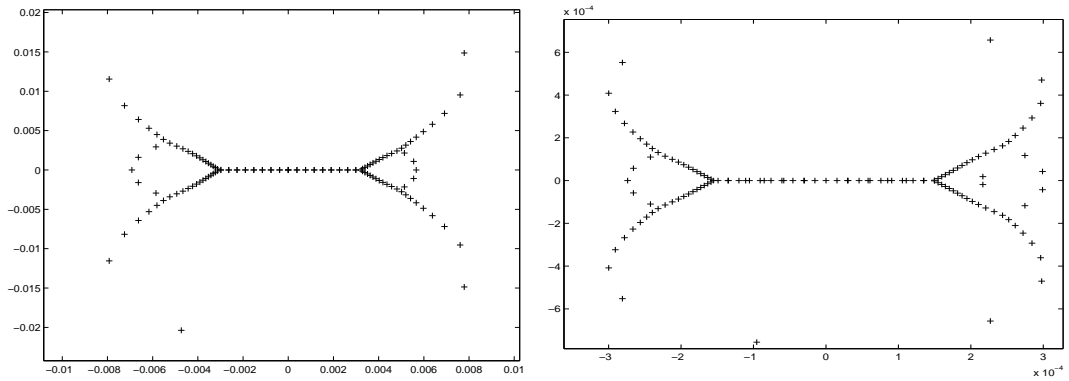


Figure 5.13: Backward Euler: Left: $b_1 = \sin 2\pi t$ Right: $b_3 = t(t - \frac{1}{2})(t - 1)$

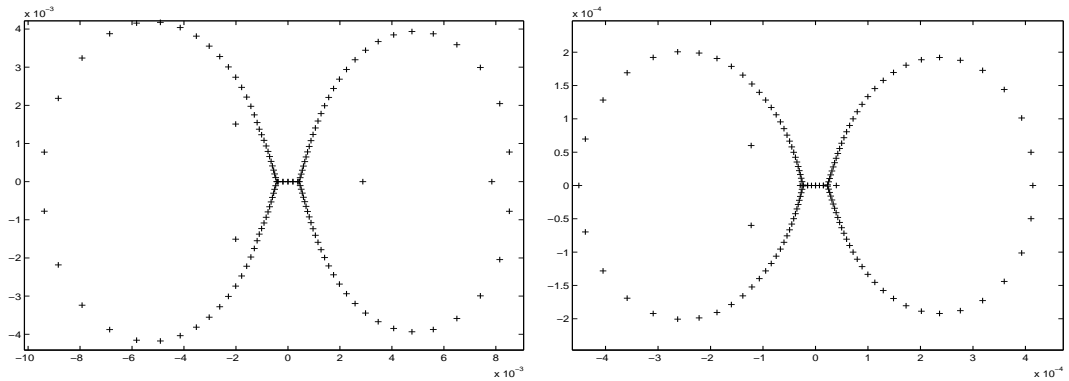


Figure 5.14: Trapezium rule: Left: $b_1 = \sin 2\pi t$ Right: $b_3 = t(t - \frac{1}{2})(t - 1)$

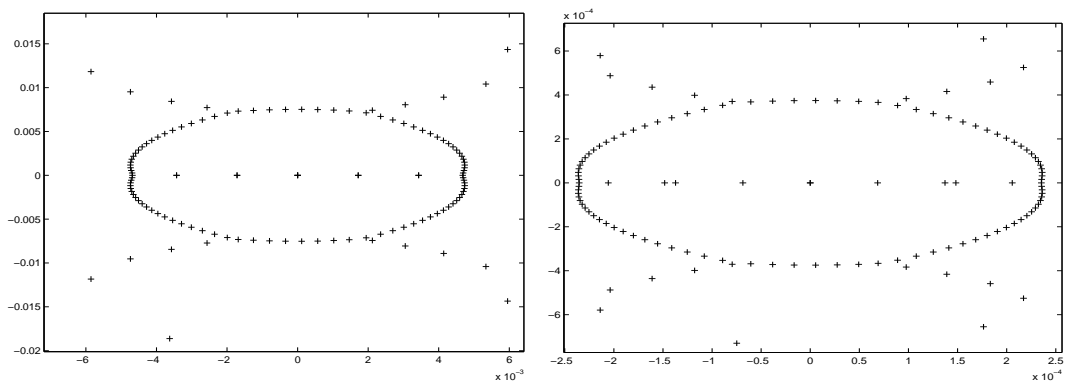


Figure 5.15: Adams-Bashforth (order 2): Left: $b_1 = \sin 2\pi t$
 Right: $b_3 = t(t - \frac{1}{2})(t - 1)$

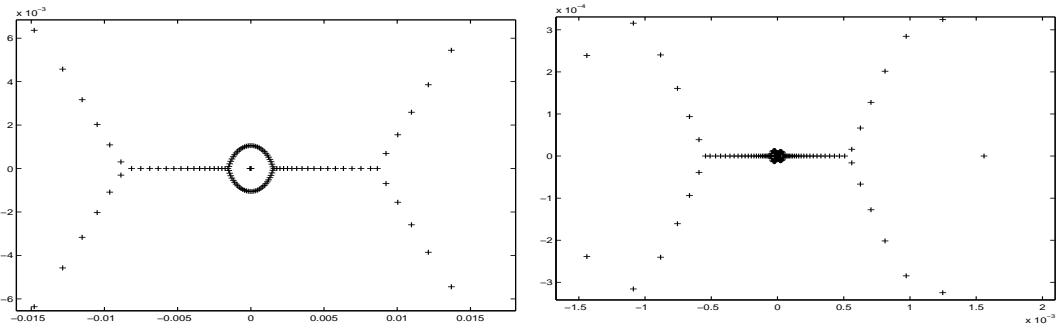


Figure 5.16: Adams-Moulton (order 3): Left: $b_1 = \sin 2\pi t$
 Right: $b_3 = t(t - \frac{1}{2})(t - 1)$

Eigenspectra when $b_i(t)$ changes sign on $[0, 1]$ and $\int_0^1 b_i(t) \neq 0$.
 The equation admits small solutions.

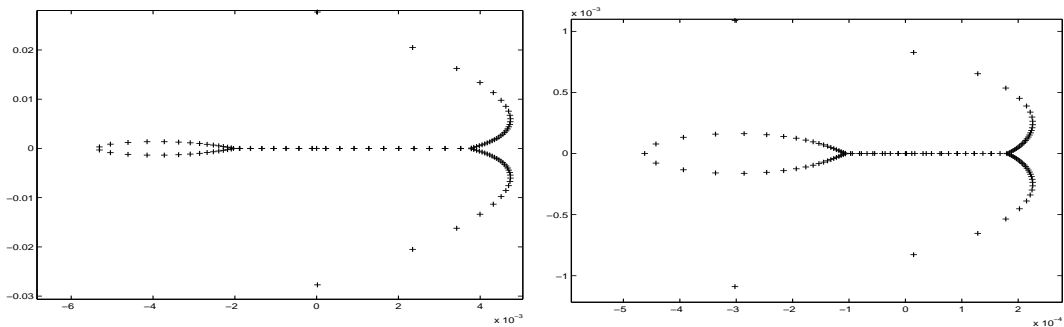


Figure 5.17: Forward Euler : Left: $b_1 = \sin 2\pi t + 0.5$
 Right: $b_3 = t(t - \frac{1}{2})(t - 1) + 1/64$

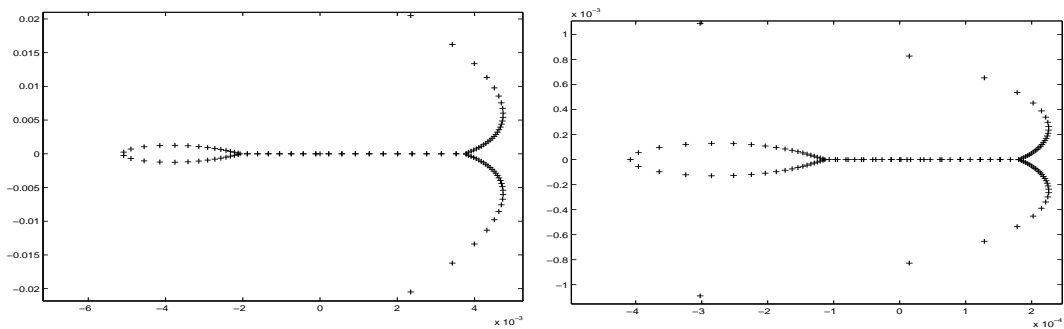


Figure 5.18: Backward Euler: Left: $b_1 = \sin 2\pi t + 0.5$
 Right: $b_3 = t(t - \frac{1}{2})(t - 1) + 1/64$

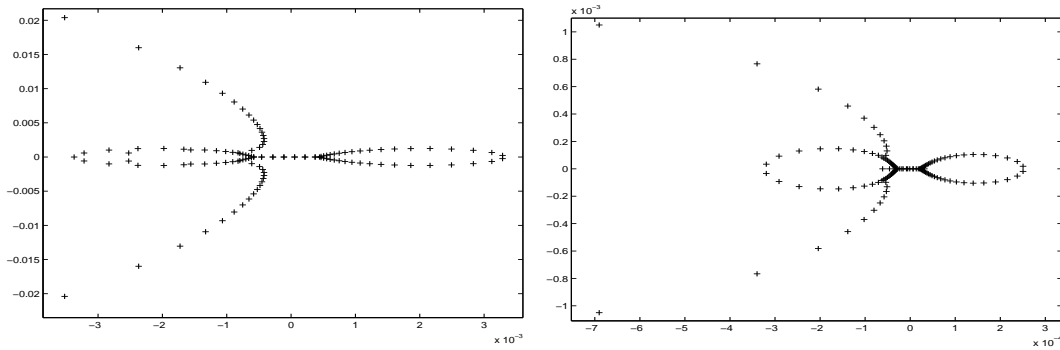


Figure 5.19: Trapezium rule: Left: $b_1 = \sin 2\pi t + 0.5$
 Right: $b_3 = t(t - \frac{1}{2})(t - 1) + 1/64$

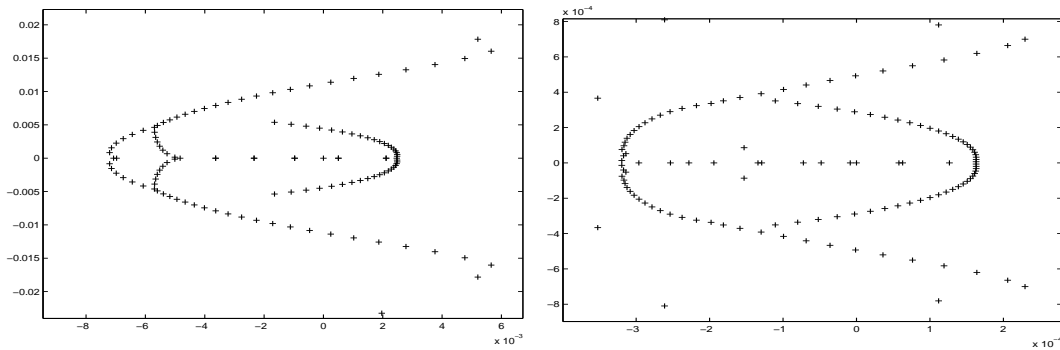


Figure 5.20: Adams-Bashforth (order 2): Left: $b_1 = \sin 2\pi t + 0.5$
 Right: $b_3 = t(t - \frac{1}{2})(t - 1) + 1/64$

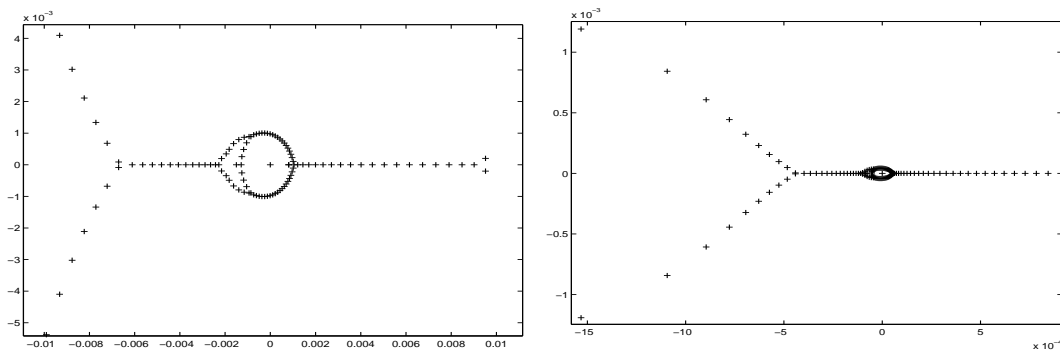


Figure 5.21: Adams-Moulton (order 3): Left: $b_1 = \sin 2\pi t + 0.5$
 Right: $b_3 = t(t - \frac{1}{2})(t - 1) + 1/64$

5.4 Further examples

We find that the similarities which we observe in the diagrams for the solutions to (4.1) with $b_i(t)$ as defined in equations (4.13), (4.14), (4.15) and (4.16) also occur in other functions satisfying the same constraints regarding periodicity, a change of sign on $[0,1]$ and the value of $\int_0^1 b_i(t)dt$.

5.4.1 Varying the function-type of $b(t)$

We have presented examples including $b(t)$ a trigonometric function, a linear function, and a cubic function. We now illustrate the application of our method using further variation in the function-type of $b(t)$ and include one example from each of the categories A, B and C (as defined in section 4.5.1).

Example 5.4.1 (Category A) We apply the trapezium rule with $b(t) = \ln(t + 1) + 0.5$. We note that $b(t)$ does not change sign on $[0, 1]$ and observe the similarity between the resulting eigenvalue trajectory, shown in the left-hand diagram of Figure 5.22, and those in Figures 4.11 and 4.12.

Example 5.4.2 (Category B) We apply the Adams-Moulton method of order 3 with $b(t) = e^{-0.1t} - 10(1 - e^{-0.1})$. We note that $b(t)$ changes sign on $[0, 1]$ and $\int_0^1 b(t)dt = 0$ and observe the similarity between the eigenvalue trajectory shown in the right-hand diagram of Figure 5.22 and those in Figures 5.3 and 5.4.

Example 5.4.3 (Category C) We apply the Forward Euler method with $b(t) = \frac{1}{(2t+1)} - \frac{1}{2}$. We note that $b(t)$ changes sign on $[0, 1]$ and $\int_0^1 b(t)dt \neq 0$. We observe the similarity between Figure 5.23 and the diagrams in Figure 5.17.

5.4.2 More complex forms of $b(t)$

Additive combinations of the original four functions $b_i(t)$ also produced similar diagrams when the same constraints on $b_i(t)$ were applied. We present the following illustrative examples.

Example 5.4.4 We apply the trapezium rule to solve $b(t) = \sin 2\pi t + t$, which can be considered as a combination of (4.13) and (4.14). We note that $b(t)$ changes sign on $[0, 1]$ and that $\int_0^1 b(t)dt \neq 0$. Compare the right-hand diagram in Figure 5.24 with Figures 4.15 and 4.16.

Example 5.4.5 We apply the Adams-Moulton method of order 3 to solve $b(t) = t(t - \frac{1}{2})(t - 1) + t - \frac{1}{2}$, which can be considered as a combination of (4.14) and (4.15). We note that $b(t)$ changes sign on $[0, 1]$ and that $\int_0^1 b(t)dt = 0$. Compare the left-hand diagram of Figure 5.24 with Figures 5.5 and 5.6.

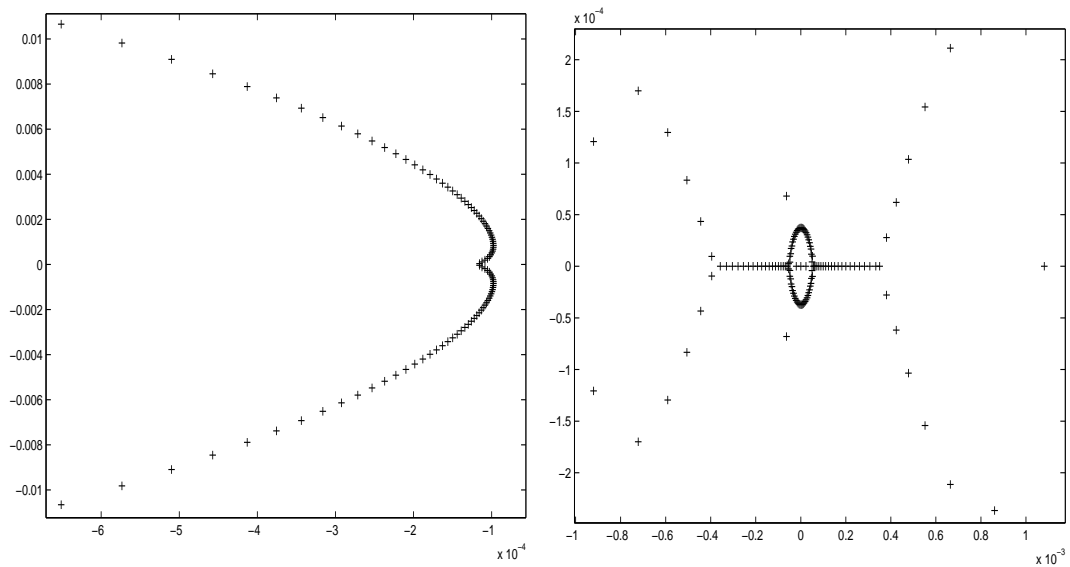


Figure 5.22: Left: $b(t) = \ln(t + 1) + 0.5$ using the trapezium rule.
 Right: $b(t) = e^{-0.1t} - 10(1 - e^{-0.1})$ using the Adams-Moulton method of order 3

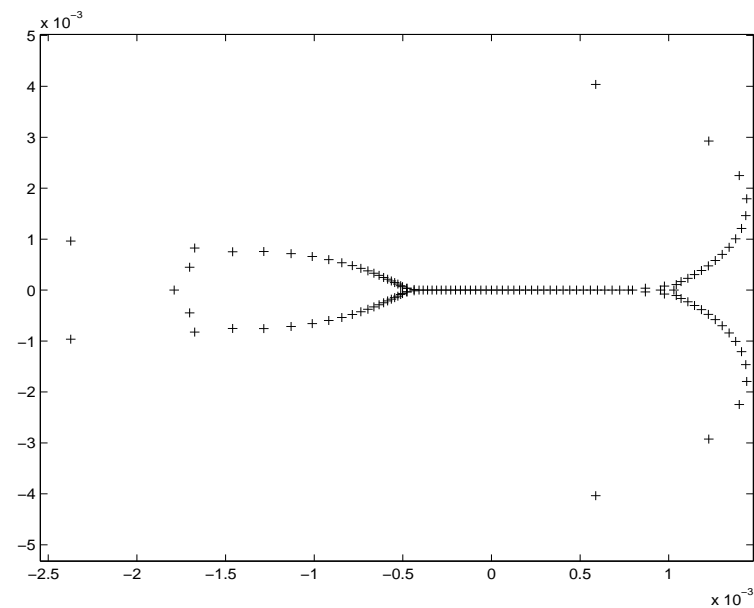


Figure 5.23: Using the Forward Euler method with $b(t) = \frac{1}{(2t+1)} - \frac{1}{2}$.

5.4.3 Values close to a critical value of c_i

We also considered whether any of the numerical methods produced consistent diagrams when the value of c_i was chosen to be very close to the value which is

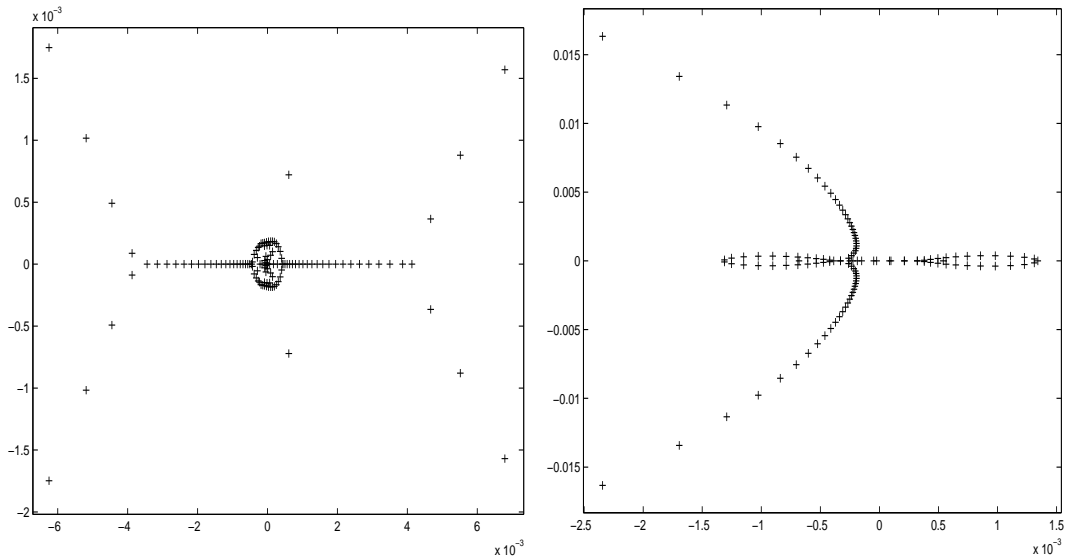


Figure 5.24: Left: Using the Adams-Moulton method of order 3 with $b(t) = t(t - \frac{1}{2})(t - 1) + t - \frac{1}{2}$. Right: Using the trapezium rule with $b(t) = \sin 2\pi t + t$

critical regarding a change in sign of $b_i(t)$ on $[0,1]$. In Figures 5.25 and 5.26 we give, as an example, the results of solving (4.13) using the trapezium rule with c_1 taking the values 0.99, 0.999, 1.001 and 1.01.

We considered other values of c_1 close to 1 and solved this and similar problems using other numerical methods. We observed that for several functions $b_i(t)$ the presence of small solutions to equation 4.1 was consistent with the eigenspectrum arising from applying a numerical scheme to the equation having eigenvalues lying on both sides of the origin. However, further work is needed on this problem before we can draw a reliable conclusion.

With regard to detecting the presence of small solutions the clarity of the diagrams near to the origin is less than ideal. The presence of small solutions is indicated by eigenvalues lying close to the real axis. These decrease in number as we approach the critical value but with the aid of the ‘zoom’ feature are visible. However, the decisions become more difficult and the dependence on an understanding of our methodology is increased.

5.5 Conclusions for the one-dimensional case

Our investigations into the one-dimensional case, reported in chapters 4 and 5, lead us to the following conclusions:

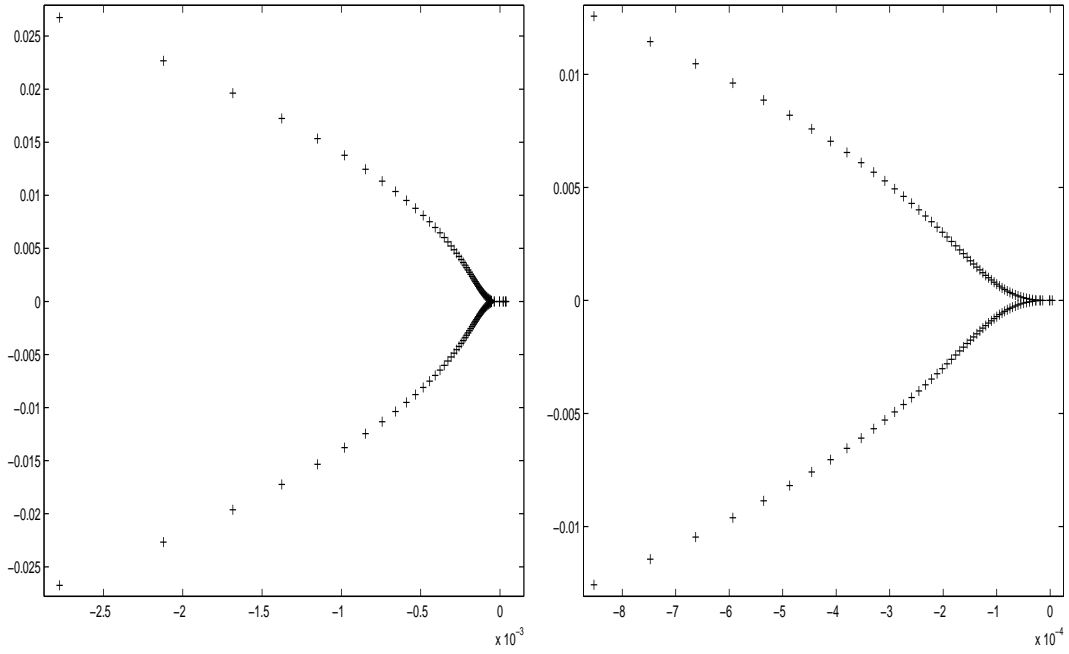


Figure 5.25: Left: $b_1 = \sin 2\pi t + 0.99$ Right: $b_1 = \sin 2\pi t + 0.999$

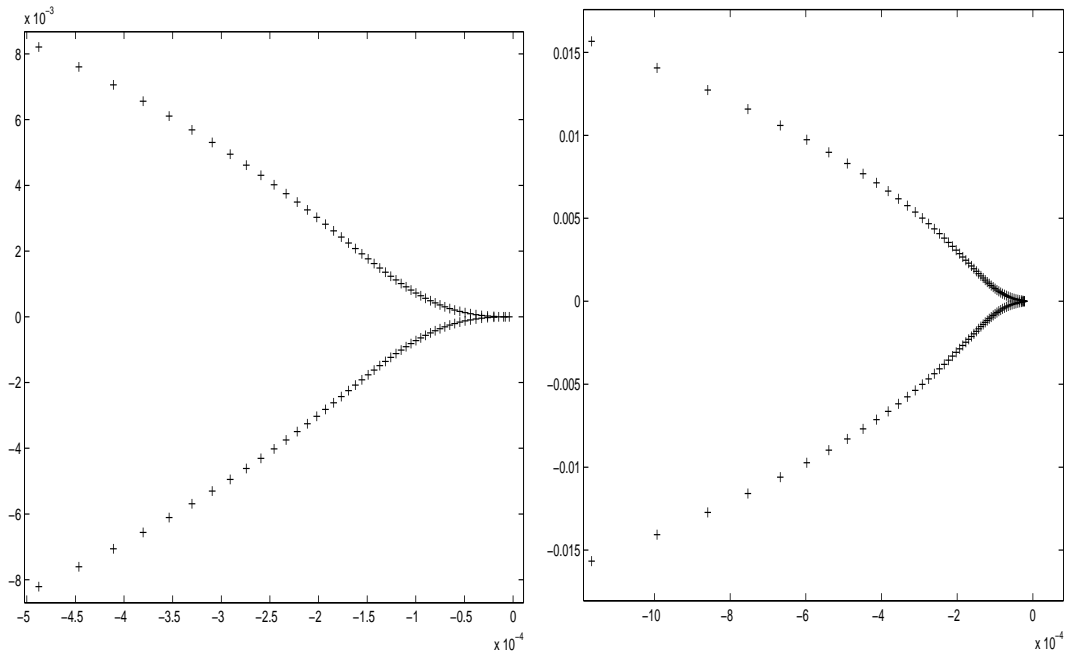


Figure 5.26: Left: $b_1 = \sin 2\pi t + 1.001$ Right: $b_1 = \sin 2\pi t + 1.01$

1. The eigenspectra display characteristic shapes that correspond to the property that the equation admits small solutions.
2. The characteristic shapes of the eigenspectra are dependent on whether or not $b(t)$ changes sign and whether or not $\int_0^1 b(s)ds$ is zero, rather than on the form of the function $b(t)$.
3. After consideration of the clarity and ease with which the presence of small solutions can be detected we decided that using a third-order method does not offer any obvious advantages over a second-order method.
4. To compare the three θ -methods we observe that the eigenvalue trajectories for functions in category C are most easily seen as a combination of those for functions in categories A and B when the trapezium rule is used.
5. Based on our experiments we find that the trapezium rule seems to be sufficiently accurate in cases when we are close to critical values of c_i , where the characteristics of the solution change.

In [38] Guglielmi discusses the optimal properties of the trapezium rule within the class of θ -methods. (See also comments in section 2.4.1.) In consequence, further experimental work when $b(t)$ is a real-valued function uses the trapezium rule as the numerical approximation scheme.

Chapter 6

Systems of delay differential equations

6.1 Introduction

In this chapter we move on from the scalar case considered in chapters 4 and 5 and consider systems of delay differential equations. One implication of the infinite dimensionality of the scalar delay equation is that a system of delay equations has essentially the same dimensionality as a scalar delay equation. However, systems of DDEs display some interesting and distinctive features, which we begin to develop.

In chapters 4 and 5 we considered the one-dimensional system represented by the equation

$$(6.1) \quad \begin{aligned} x'(t) &= b(t)x(t-1), \quad t \geq 0; \\ x(t) &= \phi(t), \quad t \in [-1, 0]. \end{aligned}$$

Systems of two delay equations exhibit all the important, relevant features of systems of DDEs (for example, the eigenvalues of the matrix $A(t)$ can be real for all t , complex for all t , or their nature may vary with t) and hence, for simplicity, we initially focus our attention on the two-dimensional case which can be represented by an equation of the form

$$(6.2) \quad \begin{aligned} y'(t) &= A(t)y(t-1) \quad \text{for } A \in \mathbb{R}^{2 \times 2} \text{ and } y \in \mathbb{R}^2, \\ y(t) &= \phi(t) \quad \text{for } -1 \leq t \leq 0. \end{aligned}$$

We consider the case when $A(t) = A(t-1)$ for all t . For the vector-valued case, a theorem stating the condition for small solutions to exist, proved by Verduyn Lunel [79], corresponding to the change in sign of $b(t)$ in the scalar case, was given in [29] as:

Theorem 6.1.1 (Theorem 1.1 in [29]) Consider the equation

$$(6.3) \quad y'(t) = A(t)y(t-1), \text{ where } A(t) = A(t-1),$$

and where $y \in \mathbb{R}^n$. The equation has small solutions if and only if at least one of the eigenvalues λ_i satisfies, for some \hat{t} ,

$$(6.4) \quad \Re\lambda_i(\hat{t}-) \times \Re\lambda_i(\hat{t}+) = 0.$$

Following communication with Verduyn Lunel [79], in [30] we clarified this condition with the statement of Theorem 6.1.2, developed from Theorem 6.1.1 and proved by Verduyn Lunel [79], but to date unpublished.

Theorem 6.1.2 Consider the equation

$$(6.5) \quad y'(t) = A(t)y(t-1), \text{ where } A(t) = A(t-1)$$

and where $y \in \mathbb{R}^n$. Let $\Lambda(t)$ denote the set of eigenvalues of $A(t)$. The equation has small solutions if and only if one of the following conditions is satisfied:

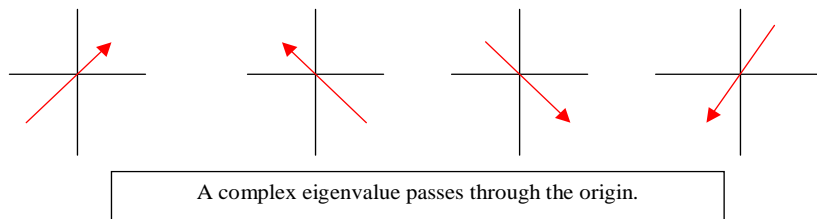
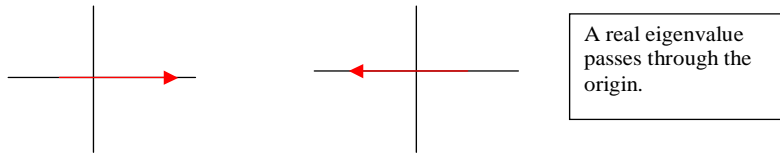
- (i) Given $\epsilon > 0$, there exists $\delta > 0$ such that for each $t \in [\hat{t} - \delta, \hat{t})$ there exists $\lambda \in \Lambda(t)$ such that $-\epsilon < R(\lambda) < 0$, for each $t \in (\hat{t}, \hat{t} + \delta]$ there exists $\lambda \in \Lambda(t)$ such that $0 < R(\lambda) < \epsilon$, and if $t = \hat{t}$ then there exists $\lambda \in \Lambda(t)$ such that $\lambda = 0$.
- (ii) Given $\epsilon > 0$, there exists $\delta > 0$ such that for each $t \in [\hat{t} - \delta, \hat{t})$ there exists $\lambda \in \Lambda(t)$ such that $0 < R(\lambda) < \epsilon$, for each $t \in (\hat{t}, \hat{t} + \delta]$ there exists $\lambda \in \Lambda(t)$ such that $-\epsilon < R(\lambda) < 0$, and if $t = \hat{t}$ then there exists $\lambda \in \Lambda(t)$ such that $\lambda = 0$.

This property was described in [29] using the words *an eigenvalue passes through the origin*. We note that, even for real matrices $A(t)$, the eigenvalues may be complex and that a pair of complex conjugate eigenvalues could cross the y -axis at a point $0 \pm iy, y \neq 0$. In the latter case the equation possesses small solutions only if some other eigenvalue crosses the y -axis at the origin. In Figure 6.1 we illustrate the different possibilities when an eigenvalue *approaches* the origin and visually clarify the reason for the replacement of Theorem 6.1.1 by Theorem 6.1.2.

In sections 6.2, 6.3 and 6.4 we will consider the three different cases of equation (6.2):

1. The matrix $A(t)$ is diagonal with $\beta(t) \equiv 0, \gamma(t) \equiv 0$ for all t .
2. The matrix is upper triangular with $\gamma(t) \equiv 0$ for all t .
3. The matrix $A(t)$ is neither diagonal or triangular.

We need to **include** the cases:



We need to **exclude** the cases:

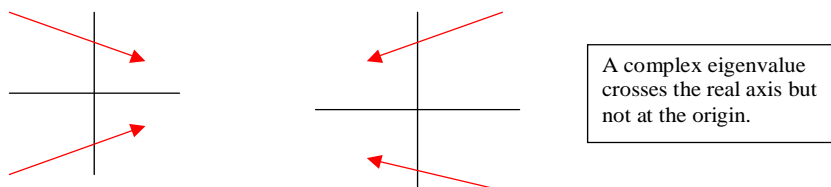
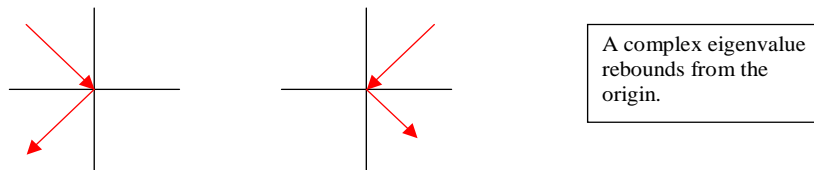


Figure 6.1: Visual clarification of ‘an eigenvalue passes through the origin’

We can deal with the first two cases quite quickly since real diagonal and triangular matrices have only real eigenvalues and these lie on the leading diagonal. We do not need to concern ourselves with possible complex eigenvalues whose real parts change sign away from the origin.

In the one-dimensional case when the non-autonomous equation $x'(t) = b(t)x(t-1)$ does not admit small solutions then it is equivalent to the autonomous equation $x'(t) = \hat{b}x(t-1)$ where $\hat{b} = \int_0^1 b(t)dt$ (see section 4.2). Hence, in our numerical investigations we compared the eigenspectra resulting from the numerical solution of $x'(t) = b(t)x(t-1)$ with that from $x'(t) = \hat{b}x(t-1)$. On this basis, in the two dimensional case we compare the eigenspectra resulting from the numerical solution of the non-autonomous problem represented by equation (6.2) with $A = \begin{pmatrix} \alpha(t) & \beta(t) \\ \gamma(t) & \delta(t) \end{pmatrix}$ with that from the autonomous problem in which $A = \begin{pmatrix} \int_0^1 \alpha(t)dt & \int_0^1 \beta(t)dt \\ \int_0^1 \gamma(t)dt & \int_0^1 \delta(t)dt \end{pmatrix}$.

It is known that if $A(t)$ can be uniformly diagonalised then a transformation to an autonomous problem can be made [79]. In this case equation (6.2), with A as above, will give the equivalent autonomous problem. When both eigenspectra are displayed on the same diagram we adopt the same convention as before and use the symbol $+$ to indicate that of the non-autonomous problem and the symbol $*$ to indicate that of the autonomous problem. When the two eigenspectra are very similar we conjecture that there exists an autonomous problem which is equivalent to the non-autonomous problem in the same sense as in section 4.2. When significant differences are observed the presence of more than one asymptotic trajectory is indicative of the existence of small solutions (see section 3.1). In this case, based on evidence from our numerical investigations, our research suggests that transformation to an autonomous problem is not possible.

6.1.1 The finite-dimensional solution map

We introduce

$$(6.6) \quad y(t) = \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix}, \quad A(t) = \begin{pmatrix} \alpha(t) & \beta(t) \\ \gamma(t) & \delta(t) \end{pmatrix}, \quad \phi(t) = \begin{pmatrix} \phi_1(t) \\ \phi_2(t) \end{pmatrix}$$

and rewrite (6.2) as

$$(6.7) \quad \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix}' = \begin{pmatrix} \alpha(t) & \beta(t) \\ \gamma(t) & \delta(t) \end{pmatrix} \begin{pmatrix} x_1(t-1) \\ x_2(t-1) \end{pmatrix}.$$

This is equivalent to

$$(6.8) \quad x_1'(t) = \alpha(t)x_1(t-1) + \beta(t)x_2(t-1)$$

$$(6.9) \quad x_2'(t) = \gamma(t)x_1(t-1) + \delta(t)x_2(t-1).$$

We again apply the trapezium rule with step length $h = \frac{1}{N}$. We introduce the approximations $x_{1,j} \approx x_1(jh), x_{2,j} \approx x_2(jh), j > 0; x_{1,j} = \phi_1(jh), x_{2,j} = \phi_2(jh), -N \leq j \leq 0$.

We are thus able to write

$$(6.10) \quad y_n = \left(x_{1,n} \quad x_{1,n-1} \quad \dots \quad \dots \quad x_{1,n-N} \quad x_{2,n} \quad x_{2,n-1} \quad \dots \quad \dots \quad x_{2,n-N} \right)^T.$$

As in the one-dimensional case (see section 4.4), $y_{n+1} = A(n)y_n$.

The matrix $A(n)$ now takes the form

$$A(n) = \begin{pmatrix} 1 & 0 & \dots & 0 & \frac{h}{2}\alpha_{n+1} & \frac{h}{2}\alpha_n & 0 & \dots & \dots & 0 & \frac{h}{2}\beta_{n+1} & \frac{h}{2}\beta_n \\ 1 & 0 & \dots & \dots & \dots & 0 & 0 & \dots & \dots & \dots & \dots & 0 \\ 0 & 1 & \ddots & & & \vdots & \vdots & & & & & \vdots \\ \vdots & \ddots & \ddots & \ddots & & \vdots & \vdots & & & & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots & \vdots & & & & & \vdots \\ 0 & \dots & \dots & 0 & 1 & 0 & 0 & \dots & \dots & \dots & \dots & 0 \\ 0 & \dots & \dots & 0 & \frac{h}{2}\gamma_{n+1} & \frac{h}{2}\gamma_n & 1 & 0 & \dots & 0 & \frac{h}{2}\delta_{n+1} & \frac{h}{2}\delta_n \\ 0 & \dots & \dots & \dots & \dots & 0 & 1 & 0 & \dots & \dots & \dots & 0 \\ \vdots & & & & & \vdots & 0 & 1 & \ddots & & & \vdots \\ \vdots & & & & & \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & & & \vdots & \vdots & \vdots & & \ddots & \ddots & \vdots \\ 0 & \dots & \dots & \dots & \dots & 0 & 0 & \dots & \dots & 0 & 1 & 0 \end{pmatrix}.$$

Using an argument similar to that used in section 4.4 we find that when the functions $\alpha(t), \beta(t), \gamma(t)$ and $\delta(t)$ are all periodic, with period 1, we can use the periodicity of the functions to write

$$(6.11) \quad y_{n+N} = Sy_n \text{ for } n = 1, 2, \dots, N,$$

where we find that the matrix S takes the form

$$(6.12) \quad S = \begin{pmatrix} 1 + \frac{h}{2}\alpha_{n+1} & h\alpha_n & h\alpha_{n-1} & \dots & h\alpha_2 & \frac{h}{2}\alpha_1 & \frac{h}{2}\beta_{n+1} & h\beta_n & h\beta_{n-1} & \dots & h\beta_2 & \frac{h}{2}\beta_1 \\ 1 & \frac{h}{2}\alpha_n & h\alpha_{n-1} & \dots & h\alpha_2 & \vdots & 0 & \frac{h}{2}\beta_n & h\beta_{n-1} & \dots & h\beta_2 & \vdots \\ 1 & 0 & \frac{h}{2}\alpha_{n-1} & \ddots & \vdots & \vdots & 0 & 0 & \frac{h}{2}\beta_{n-1} & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & h\alpha_2 & \vdots & \vdots & \vdots & \ddots & \ddots & h\beta_2 & \vdots \\ \vdots & \vdots & \ddots & \ddots & \frac{h}{2}\alpha_2 & \frac{h}{2}\alpha_1 & \vdots & \vdots & \ddots & \ddots & \frac{h}{2}\beta_2 & \frac{h}{2}\beta_1 \\ 1 & 0 & \dots & \dots & 0 & 0 & 0 & 0 & \dots & \dots & 0 & 0 \\ \frac{h}{2}\gamma_{n+1} & h\gamma_n & h\gamma_{n-1} & \dots & h\gamma_2 & \frac{h}{2}\gamma_1 & 1 + \frac{h}{2}\delta_{n+1} & h\delta_n & h\delta_{n-1} & \dots & h\delta_2 & \frac{h}{2}\delta_1 \\ 0 & \frac{h}{2}\gamma_n & h\gamma_{n-1} & \dots & h\gamma_2 & \vdots & 1 & \frac{h}{2}\delta_n & h\delta_{n-1} & \dots & h\delta_2 & \vdots \\ 0 & 0 & \frac{h}{2}\gamma_{n-1} & \ddots & \vdots & \vdots & 1 & 0 & \frac{h}{2}\delta_{n-1} & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & h\gamma_2 & \vdots & \vdots & \vdots & \ddots & \ddots & h\delta_2 & \vdots \\ \vdots & \vdots & \ddots & \ddots & \frac{h}{2}\gamma_2 & \frac{h}{2}\gamma_1 & \vdots & \vdots & \ddots & \ddots & \frac{h}{2}\delta_2 & \frac{h}{2}\delta_1 \\ 0 & 0 & \dots & \dots & 0 & 0 & 1 & 0 & \dots & \dots & 0 & 0 \end{pmatrix}.$$

Both $A(n)$ and S are considerably larger than the 2×2 matrix $A(t)$ in the original problem. However, the original block structure of four blocks in a 2×2 formation is retained in both matrices. This is key to extending our discussions to larger systems. The eight blocks present in $A(n)$ and S can be considered to belong to one of the four different matrix forms, defined in section 2.1.3, where relevant results pertaining to these matrix forms were established.

Using the definitions of P, Q, F and G in section 2.1.3 we can consider the $(2n + 2) \times (2n + 2)$ matrices $A(n)$ and S to be partitioned as follows:

$$(6.13) \quad A(n) = \begin{pmatrix} P(\alpha_n) & Q(\beta_n) \\ Q(\gamma_n) & P(\delta_n) \end{pmatrix}.$$

$$(6.14) \quad S = \begin{pmatrix} G(\alpha_n) & F(\beta_n) \\ F(\gamma_n) & G(\delta_n) \end{pmatrix}.$$

We note that the content of each block is completely determined by our numerical method (the trapezium rule) and the corresponding part of $A(t)$ (the values of the corresponding function, respectively α, β, γ and δ). We can see that $S = C_n$, as defined in proposition 2.1.2, and hence there is no *pollution* of the blocks in S from the neighbouring functions (see proposition 2.1.2).

6.2 Matrix $A(t)$ is diagonal with $\beta(t) \equiv 0, \gamma(t) \equiv 0$

6.2.1 The two-dimensional case

We begin our analysis by considering the subset of $\mathbb{R}^{2 \times 2}$ in which $\beta(t) \equiv 0$ for all t and $\gamma(t) \equiv 0$ for all t . In this case the system decouples into the two equations

$$(6.15) \quad x_1'(t) = \alpha(t)x_1(t-1)$$

$$(6.16) \quad x_2'(t) = \delta(t)x_2(t-1).$$

The eigenvalues of $A(t)$ are given by the roots of

$$(6.17) \quad \lambda^2 - \lambda[\alpha(t) + \delta(t)] + \alpha(t).\delta(t) = 0.$$

The roots of (6.17) are real if $[\alpha(t) + \delta(t)]^2 - 4\alpha(t).\delta(t) \geq 0$, which is equivalent to saying $[\alpha(t) - \delta(t)]^2 \geq 0$. Since this is true for all real-valued functions $\alpha(t)$ and $\delta(t)$, the eigenvalues of $A(t)$ are real for all real -valued functions $\alpha(t)$ and $\delta(t)$.

Some analytical results

We now establish some results relating to the two-dimensional case. We choose to refer to the matrix C , as defined in (4.12), as ‘the matrix associated with’ the one-dimensional equation and the matrix S , as defined in (6.12), as ‘the matrix associated with’ the two-dimensional equation.

Lemma 6.2.1 *If x_{1_e} is an eigenvector of the matrix C for equation (6.15), with C as defined in the one-dimensional case in equation (4.12), then $\begin{pmatrix} x_{1_e} \\ 0 \end{pmatrix}$ is an eigenvector of the matrix S , as defined in 6.12, for equation (6.7).*

Proof. $C = G(\alpha_n)$. $S = C_n$ (as defined in proposition 2.1.2).

If x_{1_e} is an eigenvector of C then $G(\alpha_n)x_{1_e} = \lambda_{1_e}x_{1_e}$ where λ_{1_e} is the associated eigenvalue. Using block matrix operations,

$$S \begin{pmatrix} x_{1_e} \\ 0 \end{pmatrix} = \begin{pmatrix} G(\alpha_n) & 0 \\ 0 & G(\delta_n) \end{pmatrix} \begin{pmatrix} x_{1_e} \\ 0 \end{pmatrix} = \begin{pmatrix} G(\alpha_n)x_{1_e} \\ 0 \end{pmatrix} = \begin{pmatrix} \lambda_{1_e}x_{1_e} \\ 0 \end{pmatrix} = \lambda_{1_e} \begin{pmatrix} x_{1_e} \\ 0 \end{pmatrix}. \text{ Hence } \begin{pmatrix} x_{1_e} \\ 0 \end{pmatrix} \text{ is an eigenvector of } S \text{ with associated eigenvalue } \lambda_{1_e}. \quad \square$$

Using a similar argument we can show that if x_{2_e} is an eigenvector of the $C = G(\delta_n)$ associated with (6.16) then $\begin{pmatrix} 0 \\ x_{2_e} \end{pmatrix}$ is an eigenvector of the S associated with (6.7).

Lemma 6.2.2 (Lemma 7.1.2 from [36])

If $T \in \mathbb{C}_{n \times n}$ is partitioned such that $T = \begin{pmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{pmatrix}$, then $\lambda(T) = \lambda(T_{11}) \cup \lambda(T_{22})$.

Corollary 6.2.3 The $(2N+2)$ eigenvalues of S consist of the $(N+1)$ eigenvalues of C_1 and the $(N+1)$ eigenvalues of C_2 .

Proof. Consider the matrix S to be of the form $S = \begin{pmatrix} C_1 & 0 \\ 0 & C_2 \end{pmatrix}$, where C_1 and C_2 are of the form given by equation (4.12). Applying lemma 6.2.2 gives $\lambda(S) = \lambda(C_1) \cup \lambda(C_2)$. In the case which we are considering $C_1 = G(\alpha_n)$ and $C_2 = G(\delta_n)$.

Hence $\lambda(S) = \lambda(G(\alpha_n)) \cup \lambda(G(\delta_n))$. \square

We observe that in our numerical approximation of the eigenvalues of the matrix S associated with (6.2), then the $2(N+1)$ eigenvalues calculated by the numerical method do indeed consist of the union of the $(N+1)$ eigenvalues of the matrix C associated with (6.15) and the $(N+1)$ eigenvalues of the matrix C associated with (6.16) when each of these is solved numerically as a one-dimensional equation. (See examples later in this section.)

Theorem 6.2.4 x_{1_s} is a small solution of (6.15) if and only if $\begin{pmatrix} x_{1_s} \\ 0 \end{pmatrix}$ is a small solution of (6.7).

Proof. If x_{1_s} is a small solution of (6.15) then $e^{kt}x_{1_s} \rightarrow 0$ as $t \rightarrow \infty$ for all $k \in \mathbb{R}$. Since $e^{kt} \begin{pmatrix} x_{1_s} \\ 0 \end{pmatrix} = \begin{pmatrix} e^{kt}x_{1_s} \\ 0 \end{pmatrix}$ which $\rightarrow \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ as $t \rightarrow \infty$ then $\begin{pmatrix} x_{1_s} \\ 0 \end{pmatrix}$ is a small solution of (6.7).

If $\begin{pmatrix} x_{1_s} \\ 0 \end{pmatrix}$ is a small solution of (6.7) then $e^{kt} \begin{pmatrix} x_{1_s} \\ 0 \end{pmatrix} \rightarrow 0$ as $t \rightarrow \infty$. From this we see that $e^{kt}x_{1_s} \rightarrow 0$ as $t \rightarrow \infty$ and hence x_{1_s} is a small solution of (6.15). \square

Similarly, we can show that x_{2_s} is a small solution of (6.16) if and only if $\begin{pmatrix} 0 \\ x_{2_s} \end{pmatrix}$ is a small solution of (6.7).

Corollary 6.2.5 Equation (6.7) possesses small solutions (see section 4.2) if either (6.15) or (6.16) possesses small solutions.

Proof. If $\alpha(t)$ changes sign on $[0, 1]$ then (6.15) possesses small solutions and hence, by Theorem 6.2.4, equation (6.7) admits small solutions. Similarly, if $\delta(t)$ changes sign on $[0, 1]$ then (6.16) possesses small solutions and hence, by Theorem 6.2.4, equation (6.7) admits small solutions. Hence, if either $\alpha(t)$ or $\delta(t)$ change sign on $[0, 1]$ then equation (6.16) admits small solutions. \square

Numerical results

We illustrate with the following examples. In each case we compare the trajectories with the expected trajectories (see chapter 4). We expect to see the superposition of the eigenspectra from the two block matrices on the diagonal of the associated matrix S .

Example 6.2.1 We solve (6.2) with $\alpha(t) = \sin 2\pi t + 1.4$ and $\delta(t) = \sin 2\pi t + 0.5$. In this case only (6.16) admits small solutions. The two distinct trajectories are

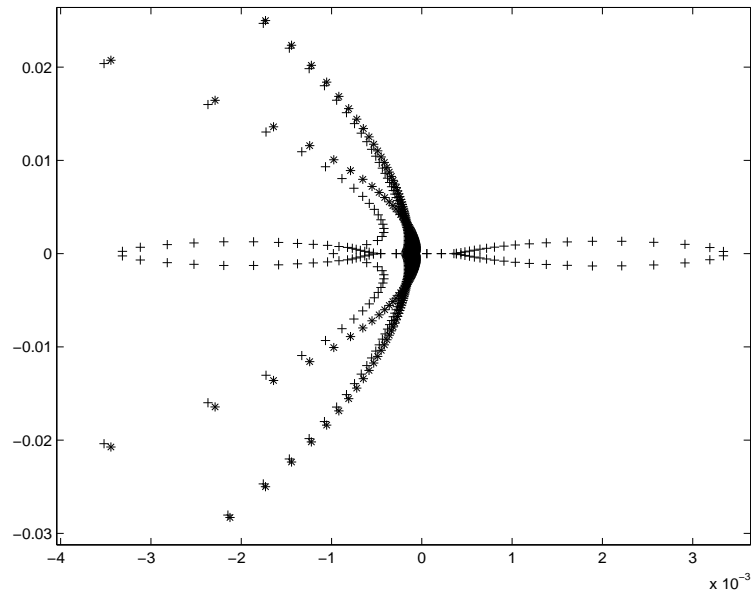


Figure 6.2: Solution of (6.2) with $\alpha(t) = \sin 2\pi t + 1.4$ and $\delta(t) = \sin 2\pi t + 0.5$. One additional trajectory: $\delta(t)$ changes sign but $\alpha(t)$ does not change sign.

easily identified in Figure 6.2. We observe that the trajectory arising from the non-autonomous problem (+++) consists of a trajectory similar to that in the left-hand trajectory in Figure 4.11 superimposed on the left-hand trajectory in Figure 4.15.

Example 6.2.2 We solve (6.2) with

$$\alpha(t) = \sin 2\pi t \text{ and } \delta(t) = \begin{cases} -0.3 & \text{for } t \in (0, \frac{1}{2}] \\ 0.7 & \text{for } t \in (\frac{1}{2}, 1] \end{cases}.$$

In this case both $\alpha(t)$ and $\beta(t)$ change sign and $\int_0^1 \alpha(t) dt = 0$. We observe that the trajectory arising from the non-autonomous problem ((++)) in Figure 6.3) consists of the trajectory in the left-hand diagram in Figure 4.13 superimposed on the right-hand trajectory in Figure 4.16. We expect small solutions and Figure

6.3 provides confirmation. There is clear evidence of two additional trajectories.

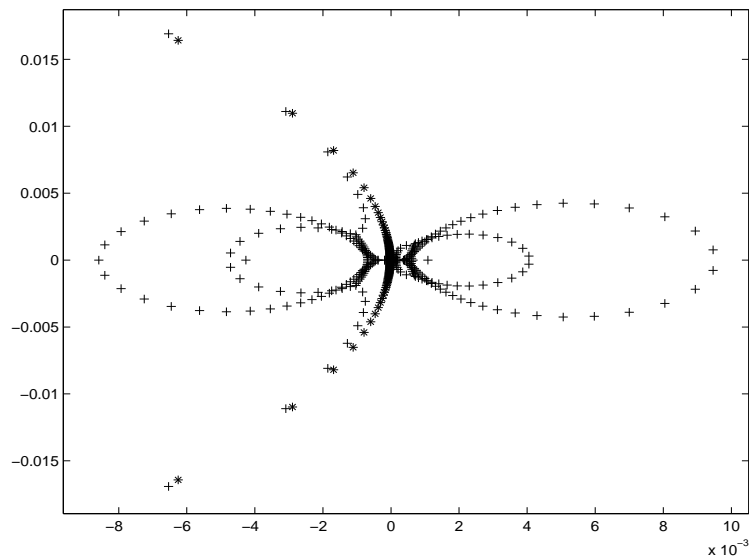


Figure 6.3: Solution of (6.2) with $\alpha(t)$ and $\delta(t)$ as in example 6.2.2. Two additional trajectories: Both $\alpha(t)$ and $\delta(t)$ change sign.

Example 6.2.3 We solve (6.2) with $\alpha(t) = t(t - \frac{1}{2})(t - 1) + \frac{1}{64}$ and $\delta(t) = t - \frac{1}{4}$. In this case both (6.15) and (6.16) admit small solutions and neither $\int_0^1 \alpha(t)dt$ or $\int_0^1 \delta(t)dt$ is equal to 0. We compare the trajectories illustrated in Figure 6.4 with the left-hand trajectory in Figure 4.16 and the right-hand trajectory in Figure 4.15. The presence of additional trajectories is clear.

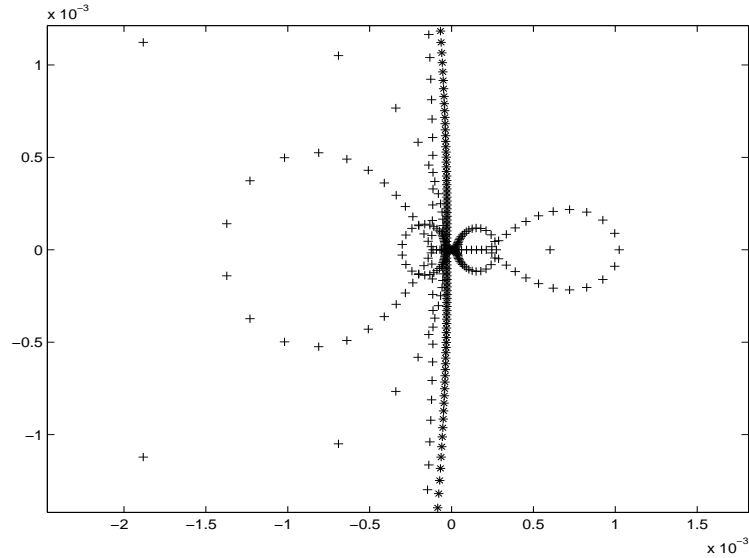


Figure 6.4: Solution of (6.2) with $\alpha(t) = t(t - \frac{1}{2})(t - 1) + \frac{1}{64}$ and $\delta(t) = t - \frac{1}{4}$. Two additional trajectories: Both $\alpha(t)$ and $\delta(t)$ change sign.

6.2.2 Extension to higher dimensions

We can extend results from section 6.2.1 to equations of the form

$$(6.18)$$

$$y'(t) = A(t)y(t - 1), A(t - 1) = A(t), \text{ where } A(t) \text{ is diagonal and } A(t) \in \mathbb{R}^{n \times n}.$$

We consider $A(t) = \{a_{ij}\}$ where $\begin{cases} a_{ij} = a_{ij}(t) & \text{for } i = j \\ a_{ij} = 0 & \text{for } i \neq j \end{cases}$.

In our examples the functions $a_{ij}(t)$ have been constructed such that $a_{ij}(t) = f(t) + c_{ij}$ and $\int_0^1 a_{ij}(t)dt = c_{ij}$. In our experimental work in sections 6.2.2 and 6.3.2 we choose to compare the eigenspectra resulting from the discretisation of the non-autonomous problem with that arising from the autonomous problem in which $A = \{c_{ij}\}$.

We introduce equations for $y(t)$ and y_n in equation (6.19)

$$(6.19) \quad y(t) = \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ \vdots \\ \vdots \\ \vdots \\ x_n(t) \end{pmatrix} \quad \text{and} \quad y_n = \begin{pmatrix} x_{1,n} \\ x_{1,n-1} \\ \vdots \\ x_{1,n-N} \\ x_{2,n} \\ x_{2,n-1} \\ \vdots \\ x_{2,n-N} \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ x_{n,n} \\ x_{n,n-1} \\ \vdots \\ x_{n,n-N} \end{pmatrix}.$$

In this case equation (6.2) decouples into a system of n equations of the form

$$(6.20) \quad x'_i(t) = a_{ii}(t)x_i(t-1) \quad \text{for } i = 1, 2, \dots, n.$$

For example, if $n = 3$,

$$A(t) = \begin{pmatrix} a_{11}(t) & 0 & 0 \\ 0 & a_{22}(t) & 0 \\ 0 & 0 & a_{33}(t) \end{pmatrix}.$$

In this case equation (6.18) becomes

$$\begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{pmatrix}' = \begin{pmatrix} a_{11}(t) & 0 & 0 \\ 0 & a_{22}(t) & 0 \\ 0 & 0 & a_{33}(t) \end{pmatrix} \begin{pmatrix} x_1(t-1) \\ x_2(t-1) \\ x_3(t-1) \end{pmatrix}$$

which decouples into

$$\begin{aligned} x'_1(t) &= a_{11}(t)x_1(t-1) \\ x'_2(t) &= a_{22}(t)x_2(t-1) \\ x'_3(t) &= a_{33}(t)x_3(t-1). \end{aligned}$$

Lemma 6.2.6 $x_{k_s}(t)$ is a small solution of (6.20) for some $i = k$ if and only if $(0, \dots, 0, x_{k_s}, 0, \dots, 0)^T$ is a small solution of (6.18).

Proof. If x_{k_s} is a small solution of (6.20) then $e^{kt}x_{k_s} \rightarrow 0$ as $t \rightarrow \infty$ for all $k \in \mathbb{R}$. In this case $e^{kt}(0, \dots, 0, x_{k_s}, 0, \dots, 0)^T = (0, \dots, 0, e^{kt}x_{k_s}, 0, \dots, 0)^T$ which $\rightarrow (0, \dots, 0, 0, 0, \dots, 0)^T$ as $t \rightarrow \infty$. Hence $(0, \dots, 0, x_{k_s}, 0, \dots, 0)^T$ is a small solution of (6.18). If $(0, \dots, 0, x_{k_s}, 0, \dots, 0)^T$ is a small solution of (6.18) then $e^{kt}(0, \dots, 0, x_{k_s}, 0, \dots, 0)^T \rightarrow (0, \dots, 0, 0, 0, \dots, 0)$ as $t \rightarrow \infty$. Hence $e^{kt}x_{k_s} \rightarrow 0$ as $t \rightarrow \infty$ and $x_{k_s}(t)$ is a small solution of (6.20) for $i = k$. \square

Proposition 6.2.1 If $A(t)$ is a diagonal matrix of the form

$$A(t) = \text{diag}(a_{11}(t), a_{22}(t), \dots, a_{nn}(t)),$$

where $a_{ii}(t)$ is continuous and $a_{ii}(t) = a_{ii}(t - 1)$ for $i = 1, 2, \dots, n$, then a sufficient condition for the equation $y'(t) = A(t)y(t - 1)$ to possess small solutions is that there exists at least one value of i such that $a_{ii}(t)$ changes sign on $[0, 1]$.

Proof. This follows from lemma 6.2.6. \square

We illustrate this in the following example. Again, we expect to find a superposition of eigenspectra arising from the block matrices on the leading diagonal.

Example 6.2.4 We solve equation (6.18) for $n = 4$ with $a_{11}(t) = t + 1.5$, $a_{22}(t) = \sin 2\pi t + c$, $a_{33}(t) = t(t - 0.5)(t - 1) + \frac{29}{64}$, $a_{44}(t) = \ln(t + 1) - 2 \ln 2 + 2.5$ and $a_{ii}(t) = a_{ii}(t - 1)$ and include the cases when $c = 1.5$ and 0.5 . The functions $a_{11}(t)$, $a_{44}(t)$ and $a_{33}(t)$ do not change sign on $[0, 1]$. The left-hand diagram in Figure 6.5 illustrates the eigenvalue trajectories when $c = 1.5$. In this case $a_{22}(t)$ does not change sign on $[0, 1]$ and no small solutions are predicted for equation (6.18). When $c = 0.5$ then $a_{22}(t)$ changes sign on $[0, 1]$ and in the right-hand diagram of Figure 6.5 we observe an additional trajectory, indicating the presence of small solutions, as expected. The four different eigenvalue trajectories are clearly distinguishable in each diagram.

6.3 Matrix $A(t)$ is triangular with $\gamma(t) \equiv 0$

6.3.1 The two-dimensional case

We now consider (6.2) when $\gamma(t) \equiv 0$. By a similar argument to that presented in section 6.2 we see that the resulting upper triangular matrix $A(t)$ has real

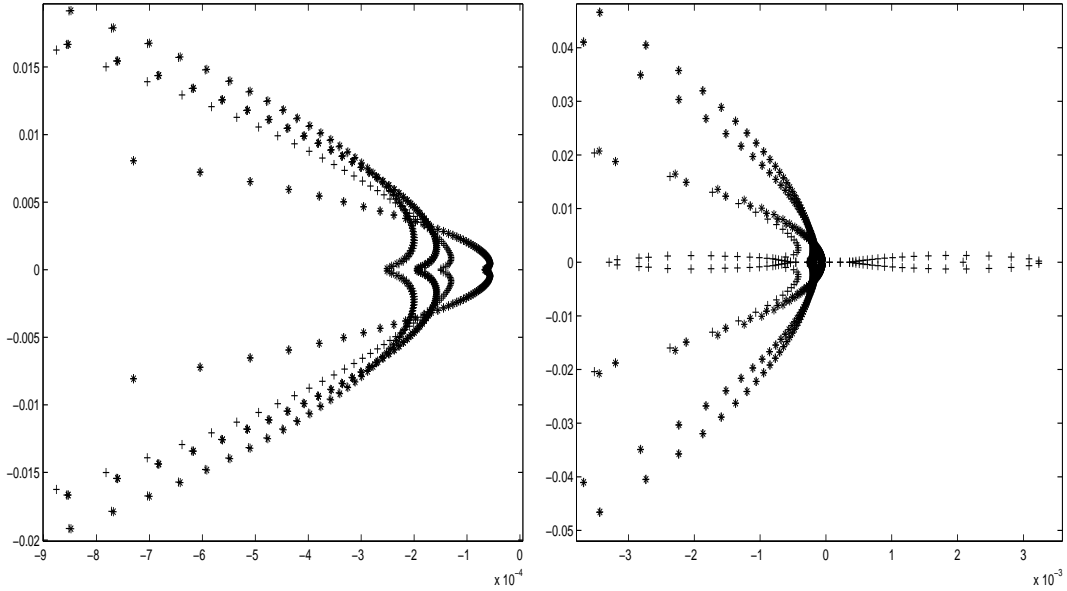


Figure 6.5: Left: No small solutions: $c = 1.5$
 Right: Equation admits small solutions: $c = 0.5$

eigenvalues. Equation (6.2) now takes the form

$$(6.21) \quad \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix}' = \begin{pmatrix} \alpha(t) & \beta(t) \\ 0 & \delta(t) \end{pmatrix} \begin{pmatrix} x_1(t-1) \\ x_2(t-1) \end{pmatrix}$$

which decouples into

$$(6.22) \quad x_1'(t) = \alpha(t)x_1(t-1) + \beta(t)x_2(t-1)$$

$$(6.23) \quad x_2'(t) = \delta(t)x_2(t-1).$$

Equation (6.23) possesses small solutions if $\delta(t)$ changes sign on $[0, 1]$ (see section 4.2). Applying Theorem 6.2.4 we see that if x_{2_s} is a small solution of (6.23) then $\begin{pmatrix} 0 \\ x_{2_s} \end{pmatrix}$ is a small solution of (6.21). Consequently, a sufficient condition for (6.21) to possess small solutions is that $\delta(t)$ changes sign on $[0, 1]$. This is supported by our numerical experiments and we illustrate with the following example:

Example 6.3.1 Figure 6.6 illustrates the eigenvalue trajectory when $\alpha(t) = \sin 2\pi t + 1.3$, $\beta(t) = \sin 2\pi t + 1.5$ and $\delta(t) = \sin 2\pi t + 0.4$. Only $\delta(t)$ changes sign on $[0, 1]$ and we observe the presence of small solutions

We now consider the case when (6.23) does not admit small solutions, that is when $\delta(t)$ does not change sign on $[0, 1]$. Theoretically, we note that the matrix

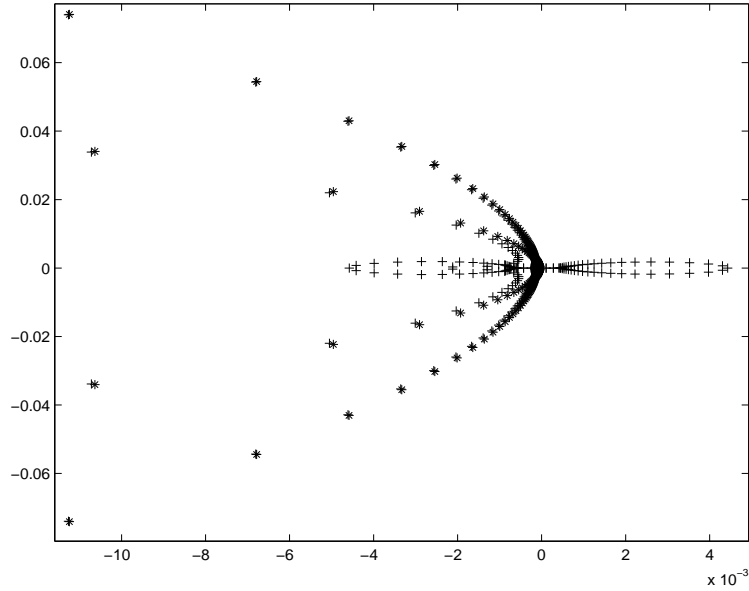


Figure 6.6: Eigenvalue trajectory when only $\delta(t)$ changes sign on $[0, 1]$

$A(t)$ in (6.21) is of the form T in Lemma 7.1.2 from [36]. Hence the eigenvalues of the S associated with $A(t)$ depend only on $\alpha(t)$ and $\delta(t)$ and not on $\beta(t)$. This is evidenced in our experimental work where we observed that allowing $\beta(t)$ to change sign on $[0, 1]$ does not induce small solutions to equation (6.2). Similar diagrams are obtained if $\delta(t)$ does not change sign, irrespective of the behaviour of $\beta(t)$. We illustrate this in the following example.

Example 6.3.2 We let $\alpha(t) = \sin 2\pi t + 1.3$, $\delta(t) = \sin 2\pi t + 1.7$, and illustrate the two cases $\beta(t) = \sin 2\pi t + 0.5$ and $\beta(t) = \sin 2\pi t + 1.5$. Neither $\alpha(t)$ or $\beta(t)$ change sign. In the first case $\beta(t)$ changes sign on $[0, 1]$ but in the second case there is no sign change. No additional trajectories are present (see Figure 6.7).

Irrespective of the behaviour of $\beta(t)$ the presence of small solutions was indicated in the eigenspectra arising from our numerical discretisations, when $\alpha(t)$ changed sign on $[0, 1]$.

Corollary 6.3.1 Equation 6.2 possess small solutions if either $\alpha(t)$ or $\delta(t)$ changes sign on $[0, 1]$.

Proof. By lemma 6.2.2 the set of eigenvalues of $\begin{pmatrix} \alpha(t) & \beta(t) \\ 0 & \delta(t) \end{pmatrix}$ is equal to the union of the sets of eigenvalues resulting from the relevant properties of $\alpha(t)$ and $\delta(t)$. If $\alpha(t)$ changes sign on $[0, 1]$ then there exists an eigenvalue of the matrix $C = G(\alpha_n)$, resulting from $\alpha(t)$ which passes through the origin. Hence

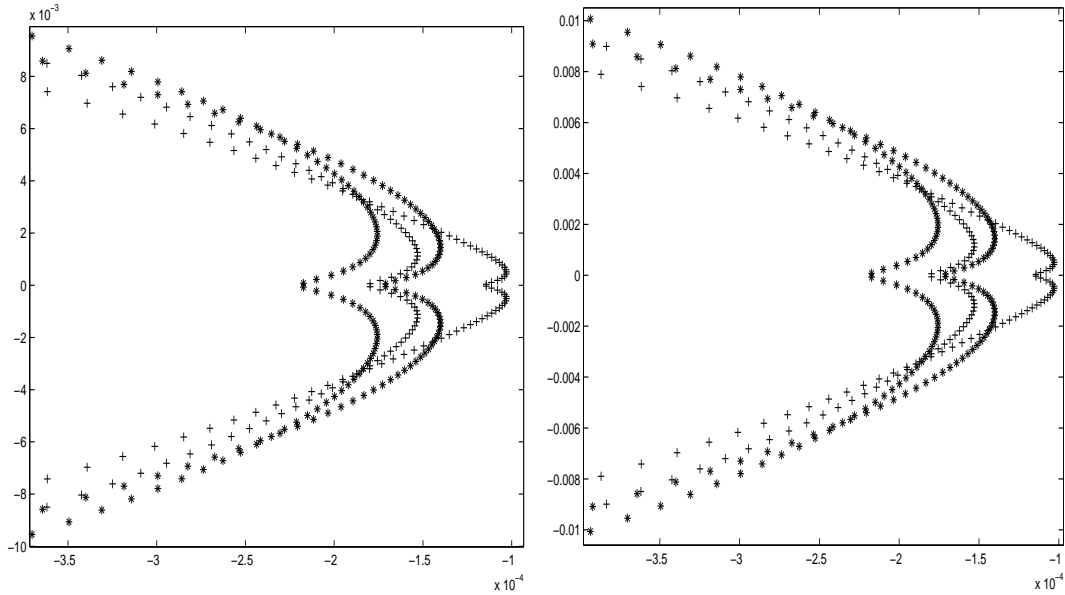


Figure 6.7: Left: $\beta(t)$ changes sign Right: $\beta(t)$ does not change sign

the equation admits small solutions.

Similarly, if $\delta(t)$ changes sign on $[0, 1]$ then there exists an eigenvalue of the matrix $C = G(\delta_n)$, resulting from $\delta(t)$, which *passes through the origin* and again this implies that the equation admits small solutions.

Hence, if either $\alpha(t)$ or $\delta(t)$ changes sign on $[0, 1]$ then equation 6.2 possesses small solutions. \square

6.3.2 Extension to higher dimensions

Results from section 6.3.1 can be extended to equations where $A(t)$ is an upper (or lower) triangular matrix, that is, equations of the form

$$(6.24) \quad \begin{aligned} y'(t) &= A(t)y(t-1), & A(t+1) &= A(t), & A(t) &\in \mathbb{R}^{n \times n}, \\ A(t) &= \{a_{ij}\} \text{ with } a_{ij} = 0 \text{ for } i > j. \end{aligned}$$

We adopt the same notation as in (6.19). In this case (6.24) decouples into a system of n equations. For example, if $n = 3$,

$$A(t) = \begin{pmatrix} a_{11}(t) & a_{12}(t) & a_{13}(t) \\ 0 & a_{22}(t) & a_{23}(t) \\ 0 & 0 & a_{33}(t) \end{pmatrix}.$$

In this case equation (6.24) becomes

$$\begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{pmatrix}' = \begin{pmatrix} a_{11}(t) & a_{12}(t) & a_{13}(t) \\ 0 & a_{22}(t) & a_{23}(t) \\ 0 & 0 & a_{33}(t) \end{pmatrix} \begin{pmatrix} x_1(t-1) \\ x_2(t-1) \\ x_3(t-1) \end{pmatrix}$$

which decouples into

$$\begin{aligned} x_1'(t) &= a_{11}(t)x_1(t-1) + a_{12}(t)x_2(t-1) + a_{13}(t)x_3(t-1) \\ x_2'(t) &= a_{22}(t)x_2(t-1) + a_{23}(t)x_3(t-1) \\ x_3'(t) &= a_{33}(t)x_3(t-1). \end{aligned}$$

Consider a matrix T of the form

$$T = \begin{pmatrix} T_{11} & T_{12} & T_{13} & \cdots & \cdots & T_{1n} \\ 0 & T_{22} & T_{23} & \cdots & \cdots & T_{2n} \\ \vdots & \ddots & \ddots & & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & T_{n-1,n-1} & T_{n-1,n} \\ 0 & \cdots & \cdots & \cdots & 0 & T_{nn} \end{pmatrix}.$$

If we let

$$H_{n-1} = \begin{pmatrix} T_{11} & T_{12} & T_{13} & \cdots & \cdots & T_{1,n-1} \\ 0 & T_{22} & T_{23} & \cdots & \cdots & T_{2,n-1} \\ \vdots & \ddots & \ddots & & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & T_{n-2,n-2} & T_{n-2,n-1} \\ 0 & \cdots & \cdots & \cdots & 0 & T_{n-1,n-1} \end{pmatrix}$$

and

$$P_{n-1} = \begin{pmatrix} T_{1,n} \\ T_{2,n} \\ \vdots \\ \vdots \\ T_{n-1,n} \end{pmatrix}$$

then we can write T as $T = \begin{pmatrix} H_{n-1} & P_{n-1} \\ 0 & T_{n,n} \end{pmatrix}$.

Lemma 7.1.2 from [36] then gives us that $\lambda(T) = \lambda(H_{n-1}) \cup \lambda(T_{n,n})$. By a similar argument we can show that $\lambda(H_{n-1}) = \lambda(H_{n-2}) \cup \lambda(T_{n-1,n-1})$. Continuing this argument leads to the result that $\lambda(T) = \lambda(T_{11}) \cup \lambda(T_{22}) \cup \dots \cup \lambda(T_{nn})$. A similar argument can be presented for the case when $A(t)$ is lower triangular.

We can hence extend corollary 6.3.1 to all upper triangular matrices in which all non-zero entries are periodic functions with period equal to one and say that a sufficient condition for the equation to possess small solutions is that at least one of the functions on the leading diagonal of $A(t)$ must change sign on $[0, 1]$.

Proposition 6.3.1 Let $A(t) \in \mathbb{R}^{n \times n}$ and $y \in \mathbb{R}^n$. Let $A(t) = \{a_{ij}(t)\}$, where $a_{ij}(t)$ is 1-periodic and continuous, and in which the a_{ij} are identically 0 for $i > j$. The equation $y'(t) = A(t)y(t-1)$ admits small solutions if there exists at least one value of i such that $a_{ii}(t)$ changes sign on $[0, 1]$.

Proof.

Let $\lambda(a_{kk}(t))$ be the set of eigenvalues of the matrix C associated with $a_{kk}(t)$. Using lemma 7.1.2 from [36] gives $\lambda(A(t)) = \lambda(a_{11}(t)) \cup \lambda(a_{22}(t)) \cup \dots \cup \lambda(a_{nn}(t))$. If $a_{kk}(t)$ changes sign on $[0, 1]$ then an eigenvalue of the associated matrix C passes through the origin, and hence the equation has small solutions. \square

We provide the following examples as illustration. Again we expect to find a superposition of the eigenspectra arising from block matrices on the leading diagonal.

Example 6.3.3 We solve equation (6.24) for $n = 4$ with $a_{11}(t) = t+1.5$, $a_{12}(t) = \sin 2\pi t + 1.7$, $a_{13}(t) = \sin 2\pi t + 1.2$, $a_{14}(t) = \sin 2\pi t + 1.8$, $a_{22}(t) = \sin 2\pi t + c$, $a_{23}(t) = \sin 2\pi t + 1.3$, $a_{24}(t) = \sin 2\pi t + 1.6$, $a_{33}(t) = t(t-0.5)(t-1) + \frac{29}{64}$, $a_{34}(t) = \sin 2\pi t + 1.4$, $a_{44}(t) = \ln(t+1) - 2\ln 2 + 2.5$ and $a_{ii}(t) = a_{ii}(t-1)$ and include the cases when $c = 1.5$ and 0.5 . The functions $a_{11}(t)$, $a_{33}(t)$ and $a_{44}(t)$ do not change sign on $[0, 1]$. In the left-hand diagram in Figure 6.8, when $c = 1.5$, $a_{22}(t)$ does not change sign on $[0, 1]$ and small solutions are not indicated. In the right-hand diagram in Figure 6.8, when $c = 0.5$, $a_{22}(t)$ does change sign on $[0, 1]$ and we observe an additional trajectory indicating that the equation admits small solutions. In this example none of the elements which do not lie on the leading diagonal change sign on $[0, 1]$.

Since the eigenvalues of S depend only on the nature of its diagonal elements the theory predicts that changing the non-zero elements which do not lie on the leading diagonal of $A(t)$ to functions which do change sign on $[0, 1]$ does not induce small solutions. Figure 6.9 shows the resulting eigenspectra when each non-zero element not lying on the leading diagonal of $A(t)$ is reduced by one. All elements affected by this reduction now change sign on $[0, 1]$. We illustrate using $c = 1.1$ and $c = 0.1$, and find that, as predicted, the eigenspectra in the left-hand diagram of Figure 6.9 indicate that the equation does not admit small solutions.

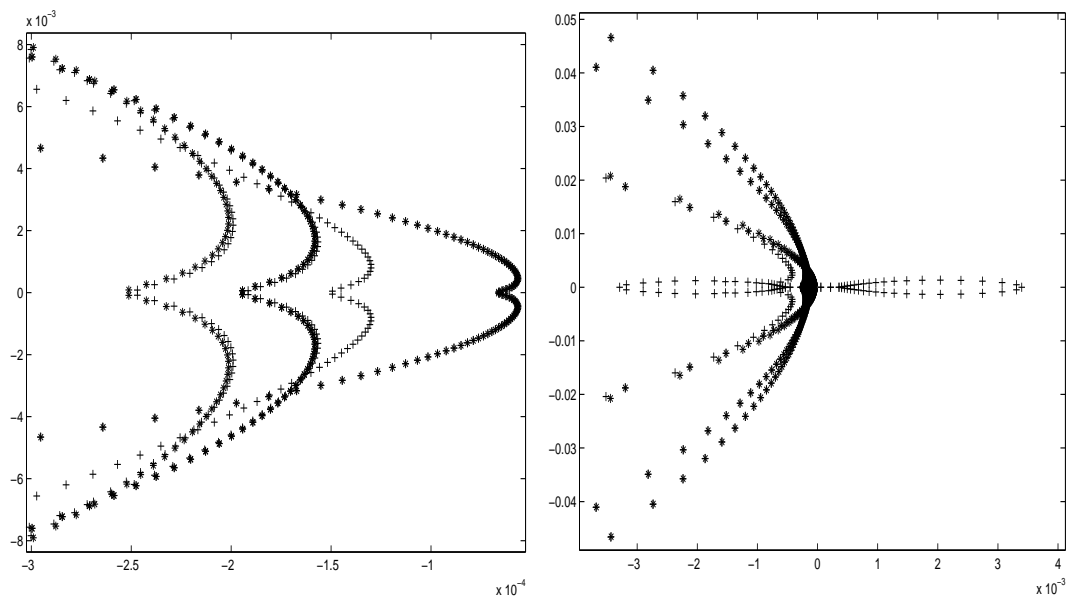


Figure 6.8: Left: None of the diagonal elements change sign on $[0, 1]$
 Right: One of the diagonal elements changes sign on $[0, 1]$

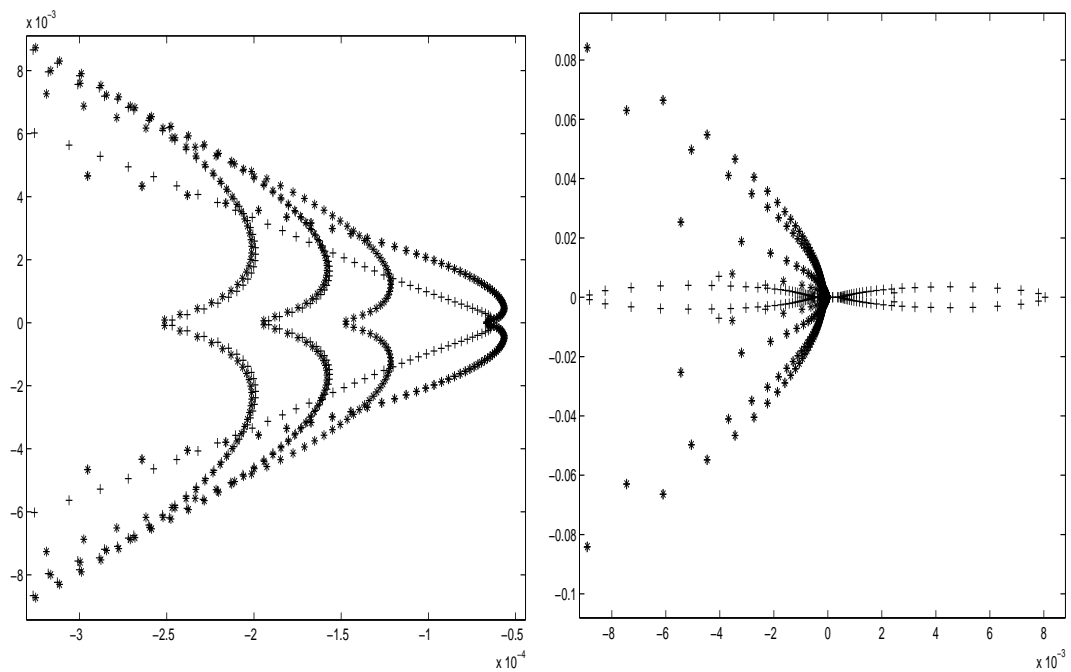


Figure 6.9: Left: $c = 1.1$ Right: $c = 0.1$

Remark 6.3.1 If $A \in \mathbb{R}^{n \times n}$ and k elements on the leading diagonal of $A(t)$ change sign then we expect to observe k , ($k = 1, \dots, n$), additional trajectories in our eigenspectra.

6.4 The general real two-dimensional case

We now move on to consider the case of equation (6.18) when none of $\alpha(t)$, $\beta(t)$, $\gamma(t)$ or $\delta(t)$ are identically zero. In this situation we can see that for each value of t the eigenvalues of $A(t)$ can be real and distinct, real and equal, complex with real part equal to zero or complex with real part not equal to zero. Eigenvalues of $A(t)$ may cross the y -axis away from the origin.

If equation (6.18) possesses small solutions then for some j , $\lambda_j = 0$, and hence it follows that $\det(A) = 0$ is a necessary condition for small solutions.

Lemma 6.4.1 *Let $y \in \mathbb{R}^n$ and $A(t) \in \mathbb{R}^{n \times n}$. $\det(A) = 0$ is a necessary condition for the equation $y'(t) = A(t)y(t-1)$ to admit small solutions.*

Proof.

$\det(A) = |A| = \prod_{j=1}^n \lambda_j$ [36]. We make use of theorem 6.1.2 and observe that if the equation admits small solutions then $\lambda_i(\hat{t}) = 0$ for at least one eigenvalue λ_i and for some \hat{t} . If $\lambda_i(\hat{t}) = 0$ then $\prod_{j=1}^n \lambda_j = 0$ and hence $\det(A) = 0$. \square

However, the condition $\det(A(t)) = 0$ cannot be used to characterise equations where small solutions arise. For example, in the case of an eigenvalue which ‘rebounds from’ rather than ‘passes through’ the origin $\det(A(t))$ attains the value of zero.

6.4.1 The eigenvalues of $A(t)$ are always real

We first consider the case when the eigenvalues of $A(t)$ are always real as t varies. In this case $[Tr(A(t))]^2 - 4|A(t)| \geq 0$. When one eigenvalue of $A(t)$ passes through the origin, then, using a continuity argument we see that for equation (6.18) to admit small solutions we need $\det(A)$ to change sign on $[0, 1]$. This remark supports observations made previously from our numerical discretisation of the equation in which we observed the presence of small solutions when $\det(A(t))$ changed sign on $[0, 1]$.

Remark 6.4.1 If two or more eigenvalues pass through the origin simultaneously then the determinant may or may not change sign. However, numerical computation involves rounding errors. As early as publication of the classical text by Wilkinson [81] it has been appreciated that, due to the occurrence of rounding errors, repeated eigenvalues are not a phenomenon that normally occurs in practice. In consequence, it is unlikely that the situation in which two

eigenvalues pass through the origin simultaneously will occur, and therefore we would expect to see the determinant change sign whenever one eigenvalue passes through the origin. (In the event of the situation arising we would expect to observe more than one asymptotic trajectory in the eigenspectrum.)

We summarise the criteria for a real matrix $A(t)$ with real eigenvalues as follows:

1. If $\det(A)$ changes sign then the equation admits small solutions.
2. If $\det(A)$ does not change sign but does attain the value zero then the equation is unlikely to admit small solutions but the reader is referred to remark 6.4.1.

6.4.2 $A(t)$ has complex eigenvalues

However, if the eigenvalues of $A(t)$ form a complex conjugate pair which cross the imaginary axis at the origin then $\det(A)$ will instantaneously take the value zero but will otherwise remain positive. In this situation we need to distinguish between the two possibilities that the eigenvalue may either pass through, or rebound from, the origin. We use the fact that the trace of $A(t)$ is equal to the sum of the eigenvalues to distinguish between these two cases.

We summarise the criteria for a real matrix $A(t)$ with complex eigenvalues as follows:

1. If $\det(A)$ changes sign then the equation admits small solutions.
2. If $\det(A)$ becomes instantaneously zero, (and is otherwise positive), and $\text{trace}(A)$ simultaneously changes sign then the equation admits small solutions.
3. If $\det(A)$ becomes zero instantaneously, (and is otherwise positive), but $\text{trace}(A)$ does not simultaneously change sign, then the equation does not admit small solutions.

6.4.3 How does this relate to the scalar case?

Previous statements, both for the one-dimensional and higher-dimensional equations, concerning sufficient conditions for particular forms of (6.18) to admit small solutions are easily related to this more general result. In the one-dimensional case $A(t)$ is the 1×1 matrix $(b(t))$ with determinant $b(t)$. The well-established result that small solutions are present if $b(t)$ changes sign on $[0, 1]$ is consistent with the more general result where we require the determinant to change sign

on $[0, 1]$. In the case of the general triangular matrix the determinant of A is equal to the product of the diagonal elements, $a_{11}, a_{22}, \dots, a_{nn}$. Consequently the determinant will change sign if one of the diagonal elements changes sign which was a sufficient condition for the 2-D equation to possess small solutions (see Corollary 6.3.1). We note that if two (or more) of the diagonal elements simultaneously change sign then the determinant may or may not change sign but will attain the value of 0. In our numerical experiments we would expect to observe two (or more) different sets of trajectories indicating the presence of small solutions.

Remark 6.4.2 If two diagonal elements of $A(t)$ are equal then we are unable to distinguish between the two associated eigenvalue spectra.

Based on extensive numerical investigation we make the following conjecture:

Conjecture 6.4.2 *The equation $y'(t) = A(t)y(t - 1)$ where $A \in \mathbb{R}^{n \times n}$, $A(t) = \{a_{ij}\}$ and each non-zero element a_{ij} of $A(t)$ has period equal to one, admits small solutions if either the determinant of $A(t)$ changes sign on $[0, 1]$ or the determinant takes the value zero at the same instant at which the trace of $A(t)$ changes sign.*

6.4.4 Numerical results

Does $\det A(t)$ change sign?

In examples 6.4.1 and 6.4.2 we illustrate the case in which a change of sign in the determinant characterises the presence of small solutions.

Example 6.4.1 We first consider the case when the matrix A takes the form

$$A(t) = \begin{pmatrix} \sin 2\pi t + a & \sin 2\pi t + b \\ \sin 2\pi t + c & \sin 2\pi t + d \end{pmatrix}.$$

One can see that $|A(t)| = (a + d - b - c) \sin 2\pi t + (ad - bc)$. Our condition for small solutions to exist requires that at least one solution of $\sin 2\pi t = \frac{-(ad-bc)}{(a+d-b-c)}$ can be found on $[0, 1]$ to ensure that the determinant changes sign on $[0, 1]$. Careful choice for the constants a, b, c, d allows different types of behaviour to be produced.

We will illustrate with the following four cases:

Case 1: $a = 1.5, b = 0.7, c = 0.5, d = 0.5$. Determinant changes sign.

Case 2: $a = -2, b = 0.8, c = 1.8, d = 0.7$. Determinant changes sign.

Case 3: $a = 1.6, b = 0.8, c = 1.8, d = 0.7$. The determinant never becomes zero.

Case 4: $a = -0.4, b = 1.5, c = -1.2, d = 1.2$. The determinant never becomes zero.

In the first two cases when the determinant changes sign on $[0,1]$ we detect the presence of small solutions in the eigenspectra shown in Figure 6.10. In the last two cases, when the determinant does not change sign on $[0, 1]$ the eigenspectra in Figure 6.11 indicate that no small solutions are present, as expected. We observe that, in case 4, the eigenvalues of the matrix in the autonomous problem are complex and that the characteristic shape of the eigenspectrum differs from that arising when the eigenvalues are real.

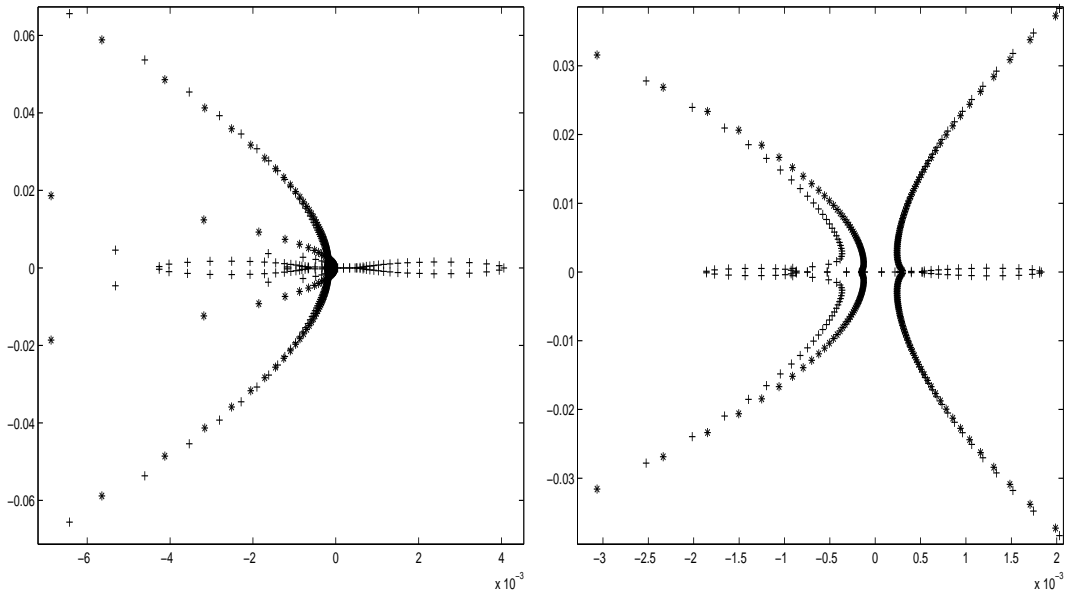


Figure 6.10: Left: Case 1 Right: Case 2

Example 6.4.2 Next, we consider the case when the matrix A takes the form

$$A(t) = \begin{pmatrix} \sin 2\pi t + a & -(\sin 2\pi t + b) \\ \sin 2\pi t + c & \sin 2\pi t + d \end{pmatrix}.$$

We find that $|A(t)| = 2 \left[\sin 2\pi t + \frac{(a+b+c+d)}{4} \right]^2 - \frac{(a+b+c+d)^2}{8} + (ad + bc)$.

Hence, if $|A(t)|$ is to change sign on $[0, 1]$ we need either (i) $\frac{(a+b+c+d)^2}{8} - (ad + bc) > 0$ and $2 \left[1 + \frac{(a+b+c+d)}{4} \right]^2 > \frac{(a+b+c+d)^2}{8} - (ad + bc)$ or (ii) $\frac{(a+b+c+d)^2}{8} - (ad + bc) = 0$ and $|a + b + c + d| < 4$.

To illustrate we include the eigenspectra for the following two cases.

Case 1: $a = 1.5, b = 1.5, c = 1.5, d = 1.5$.

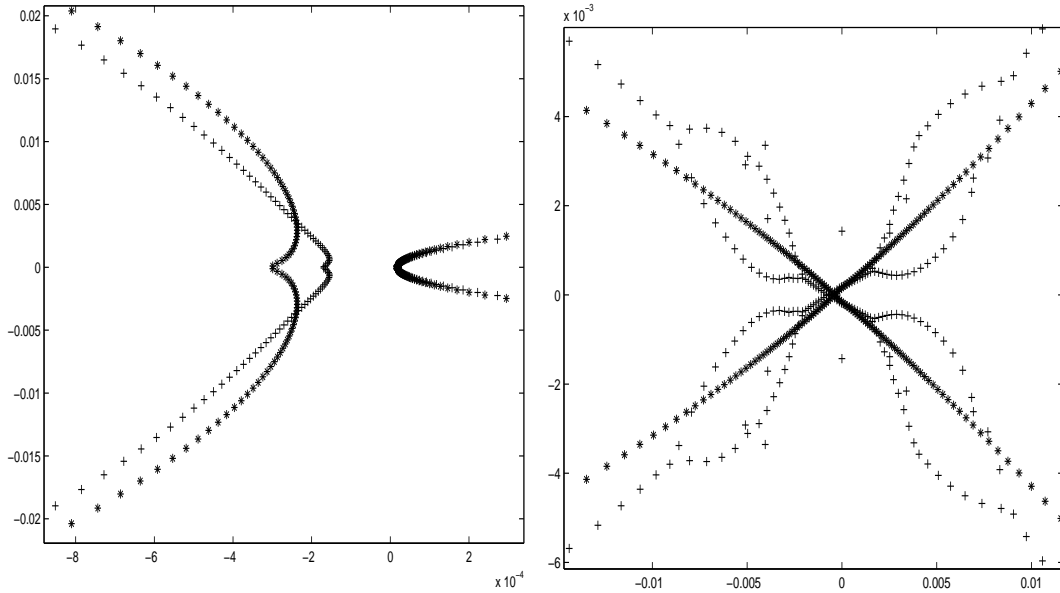


Figure 6.11: Left: Case 3 Right: Case 4

Case 2: $a = 0.4, b = 0.5, c = 0.5, d = 1.5$.

Case 1 illustrates the situation when $\frac{(a+b+c+d)^2}{8} - (ad+bc) = 0$ and $(a+b+c+d) > 4$ and in case 2 $\frac{(a+b+c+d)^2}{8} - (ad+bc) > 0$. We detect, in Figure 6.12, the presence of small solutions in case 2 but not in case 1.

Remark 6.4.3 The eigenspectra in the right-hand diagram of Figure 6.11 and the left-hand diagram of Figure 6.12 arise from problems where the eigenvalues of $A(t)$ are always complex. The eigenspectrum in the right-hand diagram of Figure 6.12 arises from an equation where the nature of the eigenvalues of $A(t)$ changes as t varies. Matrices with complex eigenvalues arose ‘naturally’ during our investigations into systems of DDEs and we choose to include them here whilst fully acknowledging that further work is needed in this interesting area. Analytical theory for this case is less well established and less readily available in the literature than for the case when the eigenvalues are real. A complete classification of the eigenspectra when $A(t)$ can have complex eigenvalues is not easy. Progress has been made but we hope to gain further insight into this case through our research reported in chapter 11.

We include two three-dimensional examples to illustrate the potential for our methodology to extend to higher dimensions.

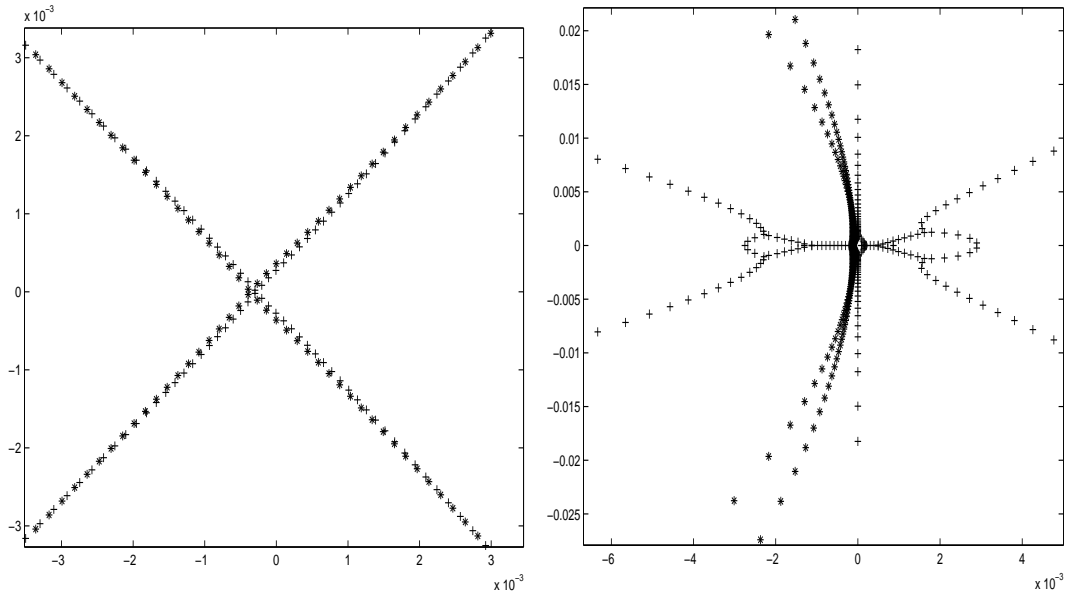


Figure 6.12: Left: Case 1 Right: Case 2

Example 6.4.3 We can show that for equation (6.18) with $A(t)$ given by

$$A(t) = \begin{pmatrix} \sin(2\pi t) + a & \sin(2\pi t) + 2 & \sin(2\pi t) + 5 \\ \sin(2\pi t) + 4 & \sin(2\pi t) + 3 & \sin(2\pi t) + 6 \\ \sin(2\pi t) + 7 & \sin(2\pi t) + 8 & \sin(2\pi t) + 9 \end{pmatrix}$$

then $\det A(t)$ changes sign if $\frac{73}{23} < a < \frac{61}{19}$. Hence, if we take $a = 3.2$ then we expect the equation to admit small solutions. This is confirmed by the left-hand diagram of Figure 6.13.

Example 6.4.4 We can show that for equation (6.18) with $A(t)$ given by

$$A(t) = \begin{pmatrix} \sin(2\pi t) - 0.5 & \sin(2\pi t) - 0.3 & \sin(2\pi t) - 0.5 \\ \sin(2\pi t) + 0.6 & \sin(2\pi t) - 0.5 & \sin(2\pi t) + 0.6 \\ \sin(2\pi t) + 0.7 & \sin(2\pi t) - 0.4 & \sin(2\pi t) - 0.5 \end{pmatrix}$$

then $\det A(t)$ changes sign. Hence, we expect the equation to admit small solutions. This is confirmed by the right-hand diagram of Figure 6.13.

$\det A$ does not change sign. Does the equation admit small solutions?

We begin our consideration of the case when $\det(A)$ does not change sign but does attain the value zero instantaneously with some examples.

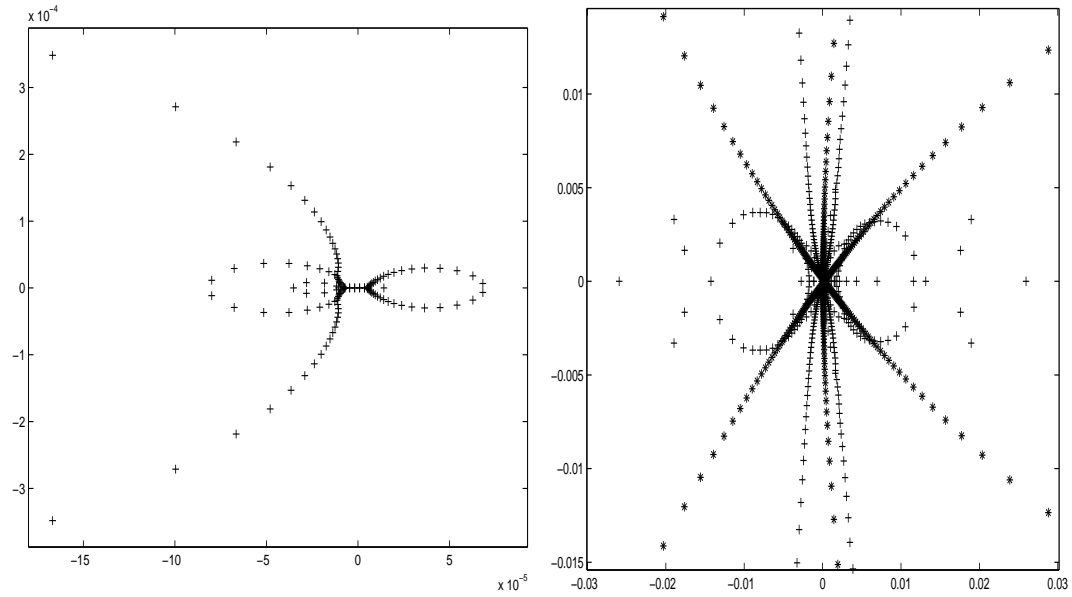


Figure 6.13: Left: $a=3.2$ Right: Eigenspectra for example 6.4.4

Example 6.4.5 We consider (6.5) with

$$A(t) = \begin{pmatrix} \sin 2\pi t & -(\sin 2\pi t + b) \\ \sin 2\pi t + b & \sin 2\pi t \end{pmatrix}.$$

If $b = 0$ then $\det(A)$ becomes instantaneously zero, $\text{trace}(A(t))$ changes sign and the otherwise complex eigenvalues are real simultaneously. Hence the complex eigenvalues of $A(t)$ cross the y -axis at the origin. We note that $\det(A) > 0$ for all other values of b and the eigenvalues cross the y -axis away from the origin. Figure 6.14 illustrates the two cases $b = 0$ and $b = 0.05$. We give zoomed in versions in Figure 6.15, We show only the eigenvalues from the non-autonomous equation.

Example 6.4.6 Now we consider the case when the matrix A takes the form

$$A(t) = \begin{pmatrix} t & -t + b \\ -t - b & t \end{pmatrix}$$

for $t \in [-0.5, 0.5)$ with $A(t) = A(t - 1)$ for $t \geq 0.5$. A has complex eigenvalues that cross the y -axis at $y = b$ when $t = 0$. In Figure 6.16 we plot the eigenspectra for the cases: (i) $b = 0$ so that the eigenvalues of A cross the y -axis at the origin, (ii) $b = 0.01$ so that the eigenvalues of A cross the y -axis away from the origin. We give zoomed-in versions in Figure 6.17.

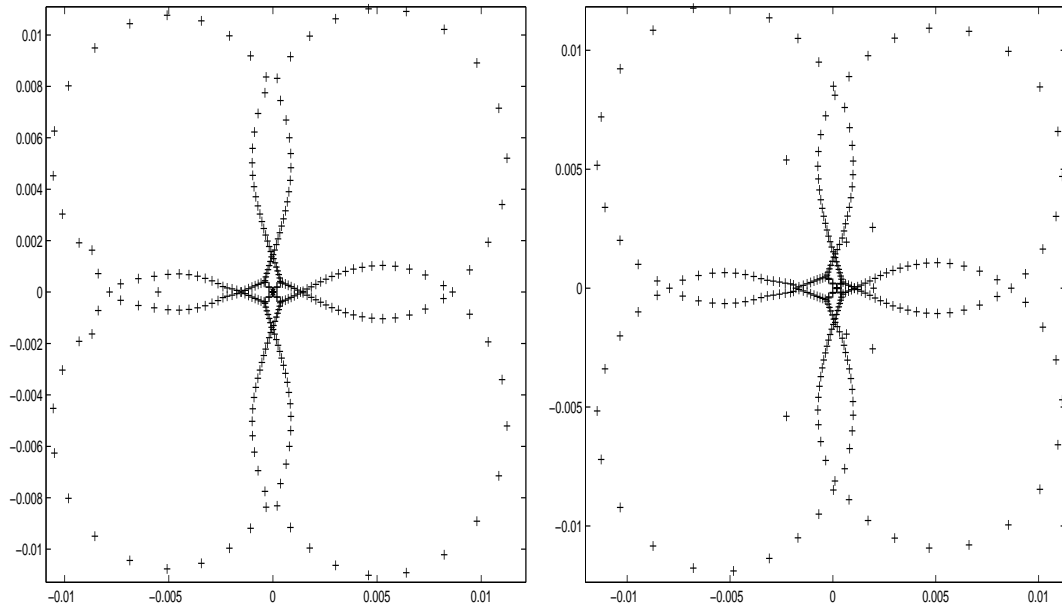


Figure 6.14: Left: Complex eigenvalues cross the y-axis at the origin
 Right: Complex eigenvalues cross the y-axis away from the origin

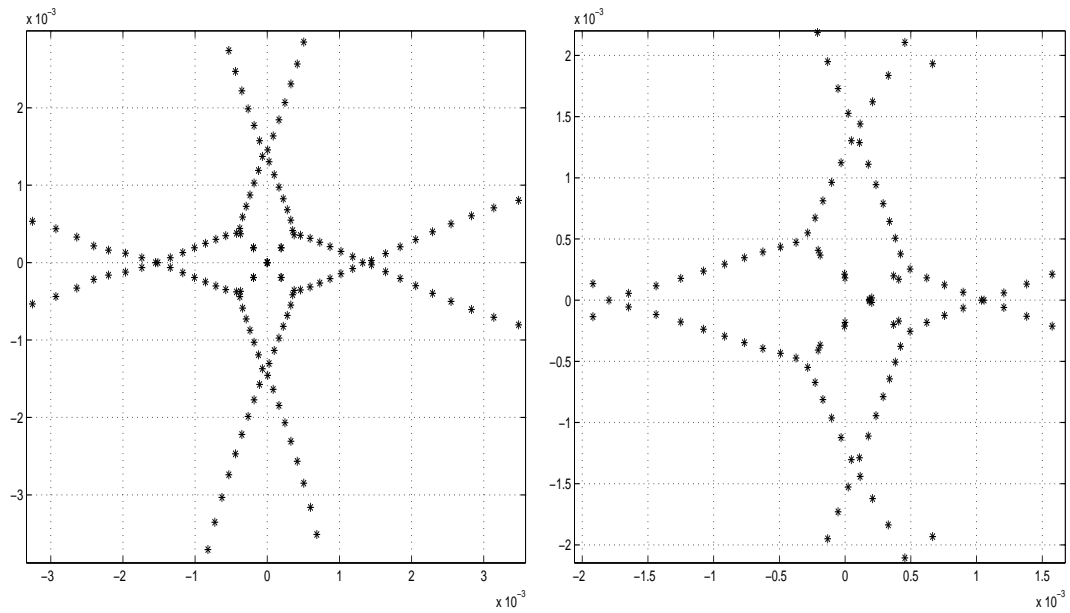


Figure 6.15: Left: Complex eigenvalues cross the y-axis at the origin
 Right: Complex eigenvalues cross the y-axis away from the origin

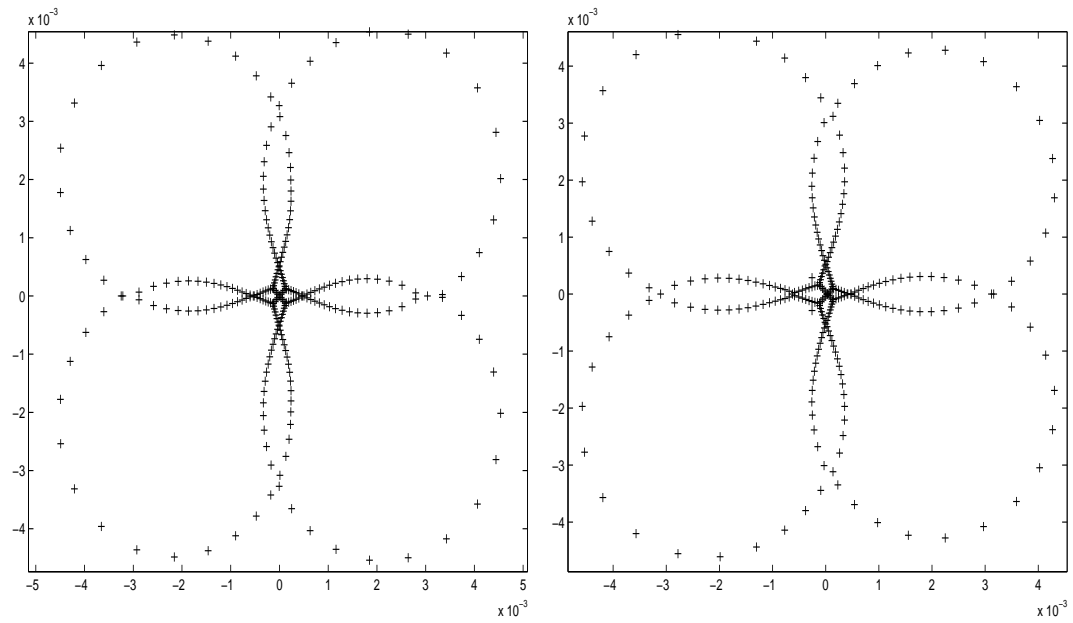


Figure 6.16: Left: Complex eigenvalues cross the y-axis at the origin
 Right: Complex eigenvalues cross the y-axis away from the origin

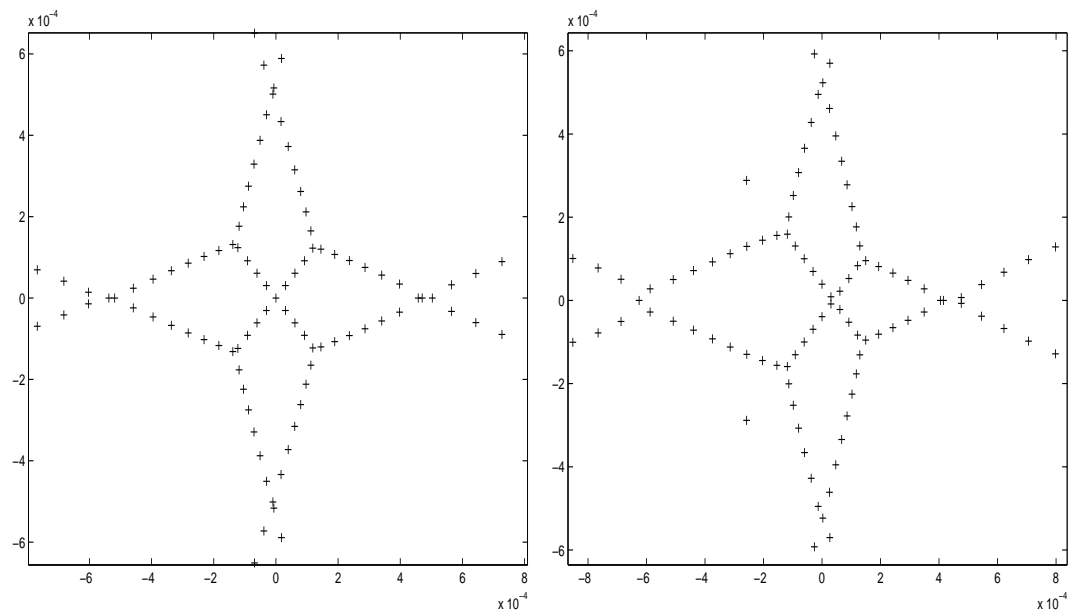


Figure 6.17: Left: Complex eigenvalues cross the y-axis at the origin
 Right: Complex eigenvalues cross the y-axis away from the origin

Example 6.4.7 We now consider (6.5) with

$$A(t) = \begin{pmatrix} \sin 2\pi t + c & -(\sin 2\pi t + c) \\ \sin 2\pi t + c & \sin 2\pi t + c \end{pmatrix}.$$

When $A(t)$ takes this form then

$$\det(A(t)) = 2(\sin 2\pi t + c)^2,$$

$$\text{Tr}(A(t)) = 2(\sin 2\pi t + c) \text{ and}$$

$$[\text{Tr}(A(t))]^2 - 4|A(t)| = -4(\sin 2\pi t + c)^2.$$

If $|c| < 1$ then values of t exist such that simultaneously $\det(A(t))$ is instantaneously zero (and otherwise positive), the eigenvalues are instantaneously real (and otherwise complex) and $\text{Tr}(A(t))$ changes sign. In this case (6.5) admits small solutions. We note that the characteristic shapes of the eigenvalue trajectories resulting from the numerical discretisation of the problem differ from those encountered in our previous work. Further investigation is called for. We compare the eigenvalue trajectory with that resulting from the autonomous problem in which $A(t) = \begin{pmatrix} c & -c \\ c & c \end{pmatrix}$ and conjecture that the presence of small solutions is indicated by an additional trajectory which passes through the origin. We illustrate using the cases $c = -0.3$, $c = 0.95$ and $c = 1.5$ in Figure 6.18.

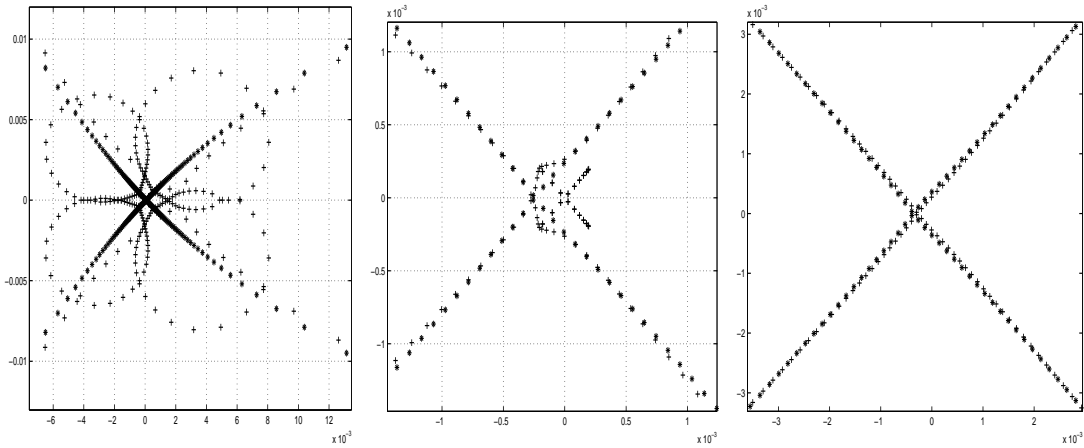


Figure 6.18: Left: $c = -0.3$ Centre: $c = 0.95$ Right: $c = 1.5$

Example 6.4.8 We consider a more general matrix of the form

$$A(t) = \begin{pmatrix} p(\sin 2\pi(t) + c) & q(\sin 2\pi(t) + c) \\ r(\sin 2\pi(t) + c) & s(\sin 2\pi(t) + c) \end{pmatrix}.$$

We choose values of p , q , r and s such that two complex eigenvalues pass through the origin. In Figure 6.19 we observe the presence of an additional trajectory passing through the origin, as expected.

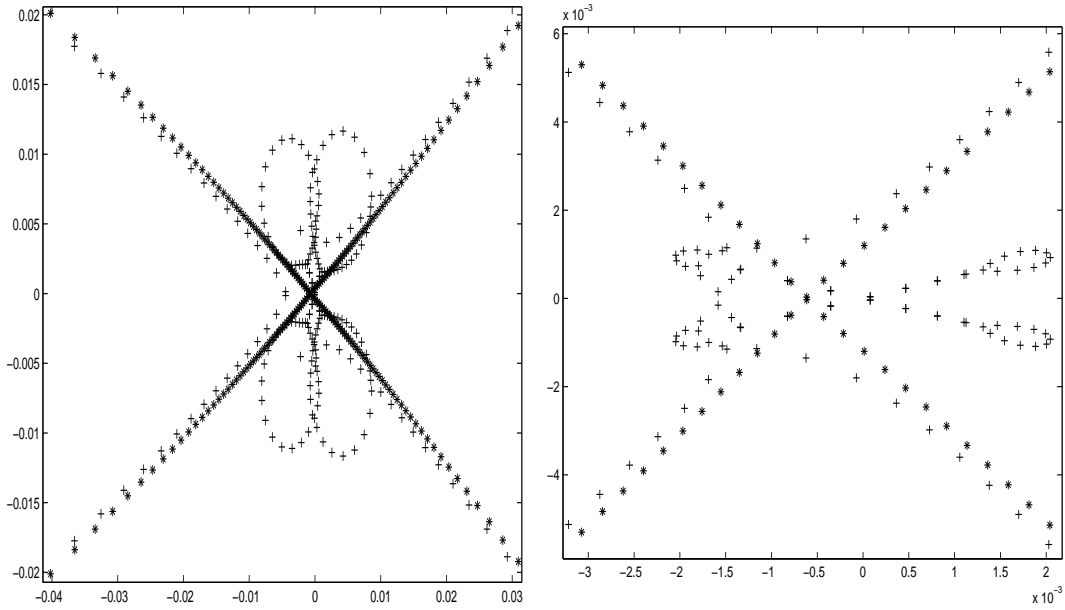


Figure 6.19: Left: $c = 0.6, p = 2, q = 5, r = -4, s = 3$ Right: $c = 0.9, p = 3, q = 5, r = -2, s = 7$

Example 6.4.9 This example is illustrative of the case in which $\det(A(t)) \geq 0$ and $\text{Tr}(A(t)) \leq 0$, and in which both are instantaneously zero simultaneously but in which $\text{Tr}(At)$ does not change sign. In this case the complex eigenvalue *rebounds from the origin* and (6.2) does not admit small solutions. We use $A(t) = \begin{pmatrix} \sin 2\pi t - 0.3 & \sin 2\pi t + 0.4 \\ \sin 2\pi t - 1.35 & \sin 2\pi t - 1.7 \end{pmatrix}$ and present the eigenspectra (and the zoomed in version) in Figure 6.20.

6.4.5 Conclusions

We have demonstrated that we can easily extend our method of detecting small solutions from the one-dimensional case to the two-dimensional case when the eigenvalues of $A(t)$ are always real. When the determinant changes sign the non-autonomous problem admits small solutions and we expect to observe additional eigenvalue trajectories to that resulting from the numerical discretisation of the potentially equivalent autonomous problem. We conjecture that the condition for small solutions to exist, regarding the change in sign of the determinant, can be extended to higher dimensions. Based on the evidence from our numerical investigations, we conjecture that it is possible to use a numerical method to distinguish between higher dimensional problems which admit small solutions and those for which an equivalent autonomous problem exists by considering the

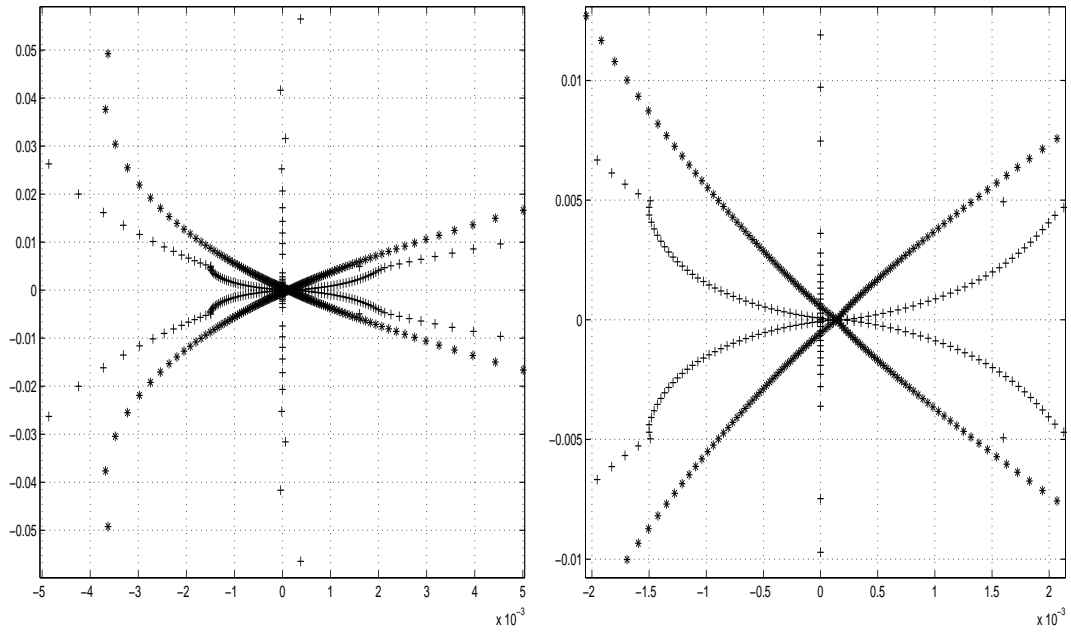


Figure 6.20: Right: Zoomed in version of eigenspectra

visual representation of the associated eigenspectra.

In the case when the eigenvalues can be complex then our experimental work to date suggests that the presence of small solutions is characterised by eigenspectra plots that pass through the origin. Further investigation is needed.

Chapter 7

Equations with multiple delays

7.1 Introduction and theoretical results

In this chapter we consider scalar linear periodic delay differential equations of the form

$$(7.1) \quad \dot{x}(t) = \sum_{j=0}^m b_j(t)x(t - jw), \quad x_s = \phi, \quad t \geq s,$$

where $b_j, j = 0, \dots, m$ are continuous periodic functions with period w . In chapter 4 (see also [28]) we considered non-autonomous periodic single delay differential equations (the case $m = 1, w = 1, b_0 = 0$) of the form

$$(7.2) \quad x'(t) = b_1(t)x(t - 1), \quad t \geq 0$$

with $b_1(t + 1) = b_1(t)$, and established that it is possible, using a numerical discretisation of the equation, to identify whether or not (7.2) admits small solutions. We now extend our investigations to cases where $m > 1$. First we show that our existing numerical method can be adapted to produce results about small solutions that are consistent with known theory. We then develop a more sophisticated approach which leads to a simpler and more efficient method of detecting small solutions for equations with multiple delays.

7.2 Known analytical results

The analysis of delay differential equations of the form (7.1) is quite well developed (see [41] for example). Even in the case $m = 1$ equation (7.1) represents an infinite dimensional system. We note that the initial function must be defined over an interval of length mw if a unique solution is required. Although the presence of multiple delays does not generally present additional difficulties beyond

those encountered in the one-delay case, we note that the presence of more than one delay may lead to a more chaotic proliferation of the discontinuity points (see page 27 in [11] or page 327 in [82]). The difference with respect to the single delay case when applying a numerical method is, in general, technical rather than conceptual [82].

Remark 7.2.1 Bélair in [10], referring to a DDE with two delays, comments on the availability of theory concerning the stability of the null solution, and states that ‘the introduction of multiple delays can have devastating effects on the simplicity of the stability analysis’ and suggests that a more thorough investigation of DDEs with more than one delay is needed.

We will assume that the zeros of b_m , in (7.1), are isolated. In this case, we know that equation (7.1) has small solutions if and only if b_m has a sign change (see Theorem 5.4 in [69] or page 250 in [41]). We are interested to see whether, by adapting the numerical method used in chapter 4 and in [28], we are able to detect the presence of small solutions to (7.1).

In section 7.3 we show how we can use our existing work directly. In section 7.4 we show how Floquet solutions can be used to simplify the numerical solution of the problem by reducing both its complexity and the computational time needed.

7.3 Using our existing ideas directly

7.3.1 The case when $m = 1$ and $w = 1$

When $m = 1$ and $w = 1$ equation (7.1) takes the form $\dot{x}(t) = b_0(t)x(t) + b_1(t)x(t-1)$, where $b_0(t)$ and $b_1(t)$ are both 1-periodic. From Theorem 4.1 in [75] we know that, providing the zeros of $b_1(t)$ are isolated, the equation has small solutions if and only if $b_1(t)$ has a sign change. Following the ideas described in section 2 of [33], we use the transformations $y(t) = x(t)\exp^{-\int_{-1}^t b_0(\sigma)d\sigma}$ and $\hat{b}_1(t) = b_1(t)\exp^{-\int_{t-1}^t b_0(\sigma)d\sigma}$, with initial data $y(\theta) = \phi(\theta)\exp^{-\int_{-1}^{\theta} b_0(\sigma)d\sigma}$, $-1 \leq \theta \leq 0$, to write (7.1) in the form

$$(7.3) \quad y'(t) = \hat{b}_1(t)y(t-1).$$

Due to the periodicity of $b_0(t)$, $\exp^{-\int_{t-1}^t b_0(\sigma)d\sigma}$ is constant and hence $\hat{b}_1(t)$ changes sign on $[0,1]$ if and only if $b_1(t)$ changes sign on $[0,1]$. We illustrate this transformation in the following example.

Example 7.3.1 Consider

$$(7.4) \quad \dot{x}(t) = (\sin 2\pi t + c_0)x(t) + b(t)x(t-1)$$

Introducing

$$\hat{b}_1(t) = b_1(t) \exp^{-\int_{t-1}^t b_0(\sigma) d\sigma} = e^{-c_0} b_1(t)$$

and

$$y(t) = x(t) e^{\frac{1}{2\pi}(\cos 2\pi t - 1)} e^{-c_0(t+1)},$$

we can rewrite (7.4) as

$$(7.5) \quad y'(t) = e^{-c_0} b_1(t) y(t-1).$$

If $e^{-c_0} b_1(t)$ changes sign on $[0, 1]$ then $b_1(t)$ must change sign on $[0, 1]$.

Equation (7.5) is of the form (7.1) with $m = 1$, $b_0(t) = 0$ and $b_1(t) = e^{-c_0} b_1(t)$.

The discussion above shows that equation (7.5) admits small solutions if $e^{-c_0} b_1(t)$ changes sign on $[0, 1]$, and hence if $b_1(t)$ changes sign on $[0, 1]$.

If $y_s(t)$ is a small solution of (7.5) then $\lim_{t \rightarrow \infty} e^{kt} y_s(t) \rightarrow 0$ for all $k \in \mathbb{R}$.

$$x(t) = e^{-\frac{1}{2\pi}(\cos 2\pi t - 1)} e^{c_0(t+1)} y(t).$$

$$e^{k_1 t} x(t) = e^{k_1 t} e^{-\frac{1}{2\pi}(\cos 2\pi t - 1)} e^{c_0(t+1)} y(t) \leq e^{\frac{1}{\pi}} e^{c_0} e^{(k_1 + c_0)t} y(t).$$

Let $y_s(t)$ be a small solution of (7.5). In this case $e^{(k_1 + c_0)t} y_s(t) \rightarrow 0$ as $t \rightarrow \infty$ and hence $x(t)$ is a small solution of (7.4).

Having already established in [28] that we can use a numerical discretisation to detect the presence of small solutions to equation (7.2) our discussion in [28], concerning the identification of the presence of small solutions, is thus immediately extended to equation (7.1) following the transformation, with $m = 1$ and $w = 1$.

7.3.2 The more general case

A similar transformation is possible in the case of equations with multiple delays of the form

$$(7.6) \quad x'(t) = \sum_{j=0}^m b_j(t) x(t-j).$$

Setting

$$(7.7) \quad y(t) = f(t) x(t)$$

where

$$(7.8) \quad f'(t) = -b_0(t) f(t),$$

$$(7.9) \quad f(t) = k_i f(t-i) \text{ for some constant } k_i$$

and

$$(7.10) \quad \hat{b}_i(t) = k_i b_i(t)$$

we obtain an equation (with no instantaneous term on the right hand side) of the form

$$(7.11) \quad y'(t) = \sum_{i=1}^m \hat{b}_i(t) y(t-i).$$

We now consider the discrete forms of (7.6) and (7.11) when solved using the trapezium rule with fixed step length $h = \frac{1}{N}$. We obtain, respectively, the equations

$$(7.12) \quad x_{n+1} = x_n + \frac{h}{2} \sum_{j=0}^m \{b_{j,n} x_{n-jN} + b_{j,n+1} x_{n+1-jN}\}$$

and

$$(7.13) \quad y_{n+1} = y_n + \frac{h}{2} \sum_{i=1}^m \{\hat{b}_{i,n} y_{n-iN} + \hat{b}_{i,n+1} y_{n+1-iN}\}.$$

We derive the approximate transformation that relates these two equations from the discrete forms (using the trapezium rule) of the transformations that applied exactly in the continuous case:

$$(7.14) \quad y_n = f_n x_n$$

$$(7.15) \quad f_{n+1} = f_n - \frac{h}{2} \{b_{0,n} f_n + b_{0,n+1} f_{n+1}\}$$

$$(7.16) \quad f_n = k_i f_{n-iN}$$

$$(7.17) \quad \hat{b}_{i,n} = k_i b_{i,n}$$

where each k_i is a constant.

We proceed to investigate how good an approximation to the solutions of (7.12) is provided by solutions to (7.13) using the transformation (7.14). We rewrite (7.12) as

$$(7.18) \quad x_{n+1} = x_n \left\{ \frac{1 + \frac{h}{2} b_{0,n}}{1 - \frac{h}{2} b_{0,n+1}} \right\} + \frac{h}{2(1 - \frac{h}{2} b_{0,n+1})} \sum_{j=1}^m \{b_{j,n} x_{n-jN} + b_{j,n+1} x_{n+1-jN}\}.$$

Using (7.14) and (7.17) we can write (7.13) as

$$(7.19) \quad f_{n+1}x_{n+1} = f_nx_n + \frac{h}{2} \sum_{i=1}^m \{k_i b_{i,n} f_{n-iN} x_{n-iN} + k_i b_{i,n+1} f_{n+1-iN} x_{n+1-iN}\}.$$

which, using (7.16), can be written

$$(7.20) \quad f_{n+1}x_{n+1} = f_nx_n + \frac{h}{2} \sum_{i=1}^m \{f_n b_{i,n} x_{n-iN} + f_{n+1} b_{i,n+1} x_{n+1-iN}\}.$$

Rewriting (7.15) in the form

$$(7.21) \quad \frac{f_n}{f_{n+1}} = \left\{ \frac{1 + \frac{h}{2} b_{0,n+1}}{1 - \frac{h}{2} b_{0,n}} \right\}.$$

enables (7.20) to be written in the form

$$(7.22) \quad x_{n+1} = \left\{ \frac{1 + \frac{h}{2} b_{0,n+1}}{1 - \frac{h}{2} b_{0,n}} \right\} x_n + \frac{h}{2} \sum_{i=1}^m \left\{ \left[\frac{1 + \frac{h}{2} b_{0,n+1}}{1 - \frac{h}{2} b_{0,n}} \right] b_{i,n} x_{n-iN} + b_{i,n+1} x_{n+1-iN} \right\}.$$

We note that each of the expressions

$\left(\frac{1 + \frac{h}{2} b_{0,n}}{1 - \frac{h}{2} b_{0,n+1}} - \frac{1 + \frac{h}{2} b_{0,n+1}}{1 - \frac{h}{2} b_{0,n}} \right)$, $\left(\frac{h}{2(1 - \frac{h}{2} b_{0,n+1})} - \frac{1 + \frac{h}{2} b_{0,n+1}}{1 - \frac{h}{2} b_{0,n}} \right)$ and $\left(\frac{h}{2(1 - \frac{h}{2} b_{0,n+1})} - \frac{h}{2} \right)$ is of the order of h^2 . Hence, by comparing the coefficients of x_{n+1} , x_n , $b_n x_{n-N}$ and $b_{n+1} x_{n+1-N}$ in equations (7.18) and (7.22), we are able to conclude that the errors in the sequence $\{x_n\}$ resulting from approximating (7.12) by (7.13) under the transformation described are (at worst) of the order of h^2 .

We observe that the difference equations (7.12) to (7.17) are discretisations of the differential equations (7.6), (7.11) and (7.7) to (7.10) using the trapezium rule with the notation $x_n = x(nh)$. We observe that the error term is $O(h^2)$. The local error of the trapezium rule is $O(h^3)$. Hence, the consequence of discretizing (7.8) and using the result in the discretisation of (7.11) is a reduction in the accuracy achieved. We note that using the forward Euler or backward Euler method to discretise the equations we expect to find that the error is $O(h^2)$. No reduction in the expected accuracy results from the transformation of the equations in this case and we achieve accuracy to $O(h)$.

Remark 7.3.1 To justify our discretisation of the multi-term continuous delay differential equation to give a multi-term discrete equation we observe that equation (7.6) can be transformed to equation (7.11) using the transformations $y(t) = \exp^{-\int_{-1}^t b_0(\sigma) d\sigma} x(t)$ and $\hat{b}_j(t) = \exp^{-\int_{t-j}^t b_0(\sigma) d\sigma} b_j(t)$ (see [33]). If $f(t) = \exp^{-\int_{-1}^t b_0(\sigma) d\sigma}$ then $f'(t) = -b_0(t)f(t)$ and $f(t) = \exp^{-\int_{t-1}^t b_0(\sigma) d\sigma} f(t-1)$.

1).

We note that $\exp^{-\int_{t-j}^t b_j(\sigma) d\sigma}$ is constant due to the periodicity of $b_j(t)$.

Introducing $k_j = \exp^{-\int_{t-j}^t b_0(\sigma) d\sigma}$ gives $\hat{b}_j = k_j b_j(t)$ and $f(t) = k_j f(t-j)$.

7.3.3 Applying a numerical method

We begin with some notation that we shall need. We let $x_n = x(nh)$ and $b_{i,j} = b_i(jh)$. We continue to use a numerical method with constant step size $h = \frac{1}{N} = \frac{m\omega}{N^*}$.

We introduce $D_1 \in \mathbb{R}^{1 \times (N+1)}$, $D_j \in \mathbb{R}^{1 \times N}$ for $j = 2, 3, \dots, m-1$, $D_m \in \mathbb{R}^{1 \times (N-1)}$, $D(n) \in \mathbb{R}^{1 \times mN}$ and $A(n) \in \mathbb{R}^{(mN+1) \times (mN+1)}$ where

$$1. D_1 = \left(\begin{array}{cccc} \frac{(2+hb_{0,n})}{(2-hb_{0,n+1})} & 0 & \dots & 0 \\ \frac{h}{(2-hb_{0,n+1})} b_{1,n+1} & \frac{h}{(2-hb_{0,n+1})} b_{1,n} & & \end{array} \right)$$

$$2. D_j = \left(\begin{array}{cccc} 0 & \dots & \dots & 0 \\ \frac{h}{(2-hb_{0,n+1})} b_{j,n+1} & \frac{h}{(2-hb_{0,n+1})} b_{j,n} & & \end{array} \right)$$

for $j = 2, 3, \dots, m-1$.

$$3. D_m = \left(\begin{array}{cccc} 0 & \dots & \dots & 0 \\ \frac{h}{(2-hb_{0,n+1})} b_{m,n+1} & & & \end{array} \right)$$

$$4. D(n) = (D_1 \ D_2 \ D_3 \ \dots \ \dots \ D_m)$$

$$5. A(n) = \left(\begin{array}{cc} D(n) & \frac{h}{(2-hb_{0,n+1})} b_{m,n} \\ I & 0 \end{array} \right)$$

$$6. y_{n+1} = \left(\begin{array}{c} x_{n+1} \\ x_n \\ \vdots \\ x_{n+1-N} \\ x_{n-N} \\ \vdots \\ x_{n+1-2N} \\ x_{n-2N} \\ \vdots \\ x_{n+1-mN} \end{array} \right)$$

Discretisation of (7.1) using the trapezium rule gives

$$(7.23) \quad x_{n+1} = x_n + \frac{h}{2} \sum_{j=0}^m (b_{j,n} x_{n-jN} + b_{j,n+1} x_{n+1-jN}).$$

from which we obtain

$$(7.24) \quad y_{n+1} = A(n)y_n.$$

It follows that $y(t + m\omega) \approx y_{n+N^*} = Cy_n$ where $C = \prod_{i=0}^{N^*-1} A(n+i)$.

In [28] we considered the autonomous problem arising from the replacement of $b_1(t)$, in the non-autonomous problem, by $\int_0^1 b_1(t)dt$. We then compared the eigenspectra arising from the autonomous problem with that from the non-autonomous problem. Here we consider the autonomous problem in which we replace each $b_i(t)$ with $\frac{1}{\omega} \int_0^\omega b_i(t)dt$ and use this to create the constant matrix A .

Remark 7.3.2 Our motivation for this approach arises from the fact that the characteristic equation for the Floquet exponents is $\det \left(e^{\mu\omega} - e^{w \sum_{j=0}^m \hat{b}_j e^{-j\mu\omega}} \right) = 0$ where $\hat{b}_j = \frac{1}{\omega} \int_0^\omega b_j(s)ds$, for $j = 0, 1, \dots, m$. The characteristic matrix for the exponents may be taken to be $\mu = \sum_{j=0}^m \hat{b}_j e^{-j\mu\omega}$, which is the characteristic matrix for the autonomous equation $x'(t) = \sum_{j=0}^m \hat{b}_j x(t - j\omega)$ (see page 249 of [41]).

We are then able to compare the eigenvalues of C with the eigenvalues of A^{N^*} . Our interest lies in the proximity of the two eigenvalue trajectories to each other. When the two trajectories are close to each other then the dynamics of the two problems are approximately the same. Obviously we can use the periodicity of the $b_i(t)$ to improve the efficiency of calculations of the eigenspectrum of C , since if $C_1 = \prod_{i=0}^{\frac{N^*}{m}-1} A(n+i)$ then $C = C_1^m$.

7.3.4 Numerical examples

We present some examples illustrating the results of this approach.

Example 7.3.2 In our first example we consider four cases of equation (7.1) with $b_0(t) \equiv 0$, $w = 1$, $m = 2$. In this case the established theory informs us that if $b_2(t)$ changes sign on $[0, 1]$ then small solutions are admitted. In Figure 7.1 $b_2(t)$ does not change sign and we observe the proximity of the two trajectories. In Figure 7.2 $b_2(t)$ does change sign and we observe the presence of two additional trajectories, which, (cf. [28]), we take to indicate the presence of small solutions. In both Figures the left-hand eigenspectra is illustrative of the case when $b_1(t)$ does change sign and the right-hand one of the case when $b_1(t)$ does not change sign, showing that, for small solutions to be admitted, it is necessary for $b_m(t)$ to change sign.

Our numerical experiments included cases when $b_2(t) = \sin 2\pi t + c$ and $|c|$ was close to 1. We found that it was still possible to detect the presence of small solutions when $|c| \leq 1$, that is, when $b_2(t)$ changes sign.

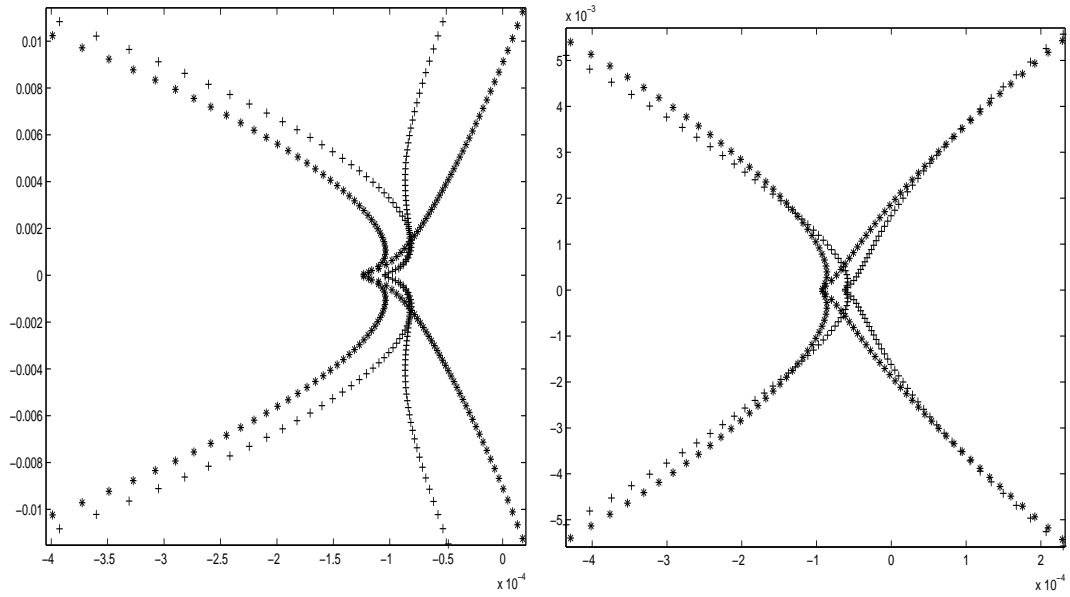


Figure 7.1: Left: $b_1(t) = \sin 2\pi t + 0.5$
 $b_2(t) = \sin 2\pi t + 1.8$

Right: $b_1(t) = \sin 2\pi t + 1.5$
 $b_2(t) = \sin 2\pi t + 1.3$

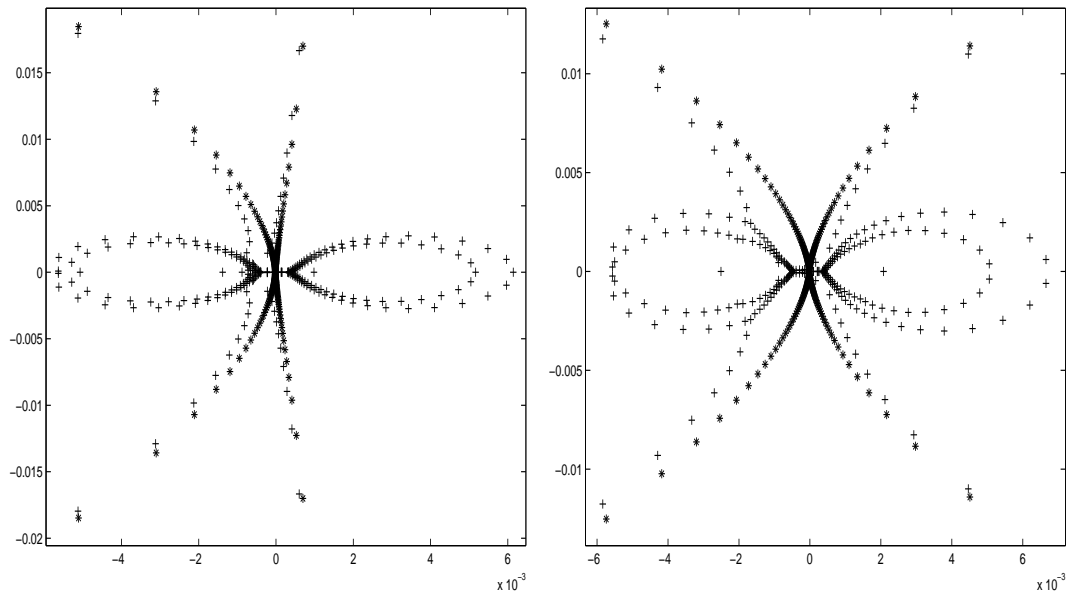


Figure 7.2: Left: $b_1(t) = \sin 2\pi t + 0.5$
 $b_2(t) = \sin 2\pi t + 0.3$

Right: $b_1(t) = \sin 2\pi t + 1.5$
 $b_2(t) = \sin 2\pi t + 0.3$

Example 7.3.3 We now include two eigenspectra resulting from equation (7.1) with $w = 1, m = 4$ and $b_0(t) \neq 0$.

- (a) $b_0(t) = \sin 2\pi t + 0.6, b_1(t) = \sin 2\pi t + 0.3, b_2(t) = \sin 2\pi t + 0.2,$
 $b_3(t) = \sin 2\pi t + 0.7, b_4(t) = \sin 2\pi t + 1.4.$
 (b) $b_0(t) = \sin 2\pi t + 1.8, b_1(t) = \sin 2\pi t + 1.3, b_2(t) = \sin 2\pi t + 1.2,$
 $b_3(t) = \sin 2\pi t + 1.7, b_4(t) = \sin 2\pi t + 0.4.$

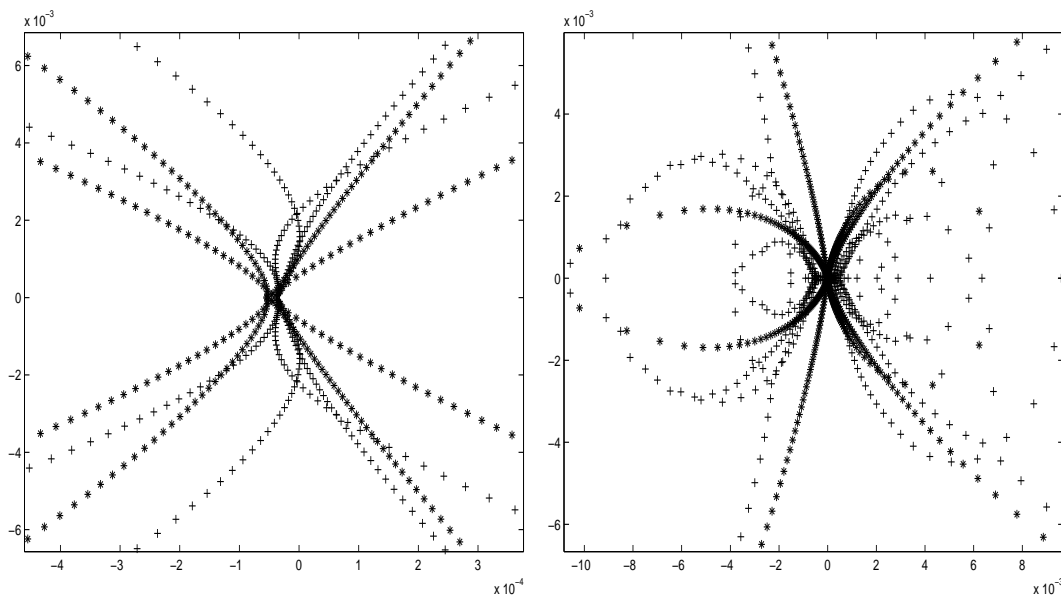


Figure 7.3: Left: $b_4(t)$ does not change sign Right: $b_4(t)$ changes sign

In Figure 7.3 we observe the presence of additional trajectories in the right-hand eigenspectra, that is when $b_4(t)$ changes sign, in accordance with the theory.

In chapter 4 (see also [28]) we successfully used a numerical method to identify whether or not equations of the form (7.11) with $m = 1$ admit small solutions. Section 7.3.2 justifies the adaptation of our numerical method to determine whether or not an equation of the form (7.6) admit small solutions. In [28] our decision to use the trapezium rule was partially based on the ease and clarity with which we could interpret our eigenspectra in terms of the presence, or otherwise, of small solutions. Based on our conclusions in section 7.3.2 we would not expect using the trapezium rule to achieve greater clarity than the forward Euler method or the backward Euler method in this case.

7.3.5 Some observations

1. For some equations we have a choice of values for j and w . For example, the equation $\dot{x} = (\sin 2\pi t + 1.3)x(t - 2) + (\sin 2\pi t + 0.6)x(t - 4)$ can be considered as a system with $w = 2, m = 2$ or with $w = 1, m = 4$ and the

equation $\dot{x} = (\sin 2\pi t + 1.6)x(t-3) + (\sin 2\pi t + 0.4)x(t-6)$ can be considered as a system with $w = 3, m = 2$ or with $w = 1, m = 6$. Clearly our decision regarding the presence, or otherwise, of small solutions must ideally be independent of our choice of m and w . It would also be interesting to consider the relative efficiency of possible choices in terms of the ‘cost’ of implementing our numerical scheme.

2. We have already observed in section 7.3.3 that using the periodicity of the $b_i(t)$ and evaluating $C = C_1^m$ can be effective in improving the efficiency of our numerical scheme.
3. We have considered a time interval equal to the maximum delay in our numerical discretisations by considering y_{n+N^*} . However we have experimented with eigenspectra arising from the consideration of y_{n+N} . The conclusions were consistent with those from the numerical scheme outlined in this chapter, in that it is still possible to detect the presence of small solutions when they exist. This would seem to be computationally more efficient and further investigation is needed. (See appendix D for examples of eigenspectra resulting from the consideration of y_{n+N} .)

7.4 A more sophisticated approach using Floquet solutions

When we adapt our numerical method as indicated in section 7.3.2 our algorithm involves the computation of the $(mN + 1)$ eigenvalues of matrix C . This can be very costly in terms of computational time. For example:

1. Equation $x'(t) = a(t)x(t-1) + b(t)x(t-1.1)$ can be considered to be of the form (7.1) with $w = 0.1, m = 11$ and would involve the calculation of $11N + 1$ eigenvalues.
2. Equation $x'(t) = a(t)x(t-1) + b(t)x(t-1.01)$ can be considered to be of the form (7.1) with $w = 0.01, m = 101$ and would involve the calculation of $101N + 1$ eigenvalues.

In section 7.3.2 our discretisation of a multi-term continuous problem led to a multi-term discrete problem. In this section we use an approach involving Floquet solutions. We obtain a single term continuous problem from our multi-term continuous problem which we are then able to discretise and produce a single term discrete problem. The reduction in the computational time needed is potentially very significant.

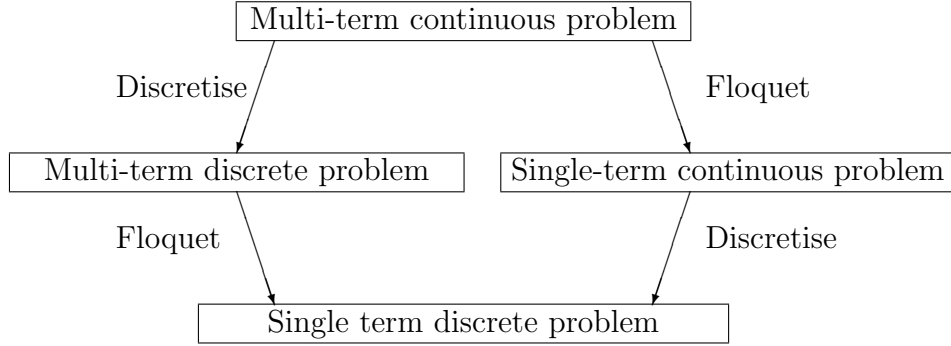


Figure 7.4: Possible approaches to the problem

7.4.1 Developing the rationale

Analytical results state that ‘the system of Floquet solutions is complete if and only if there are no small solutions’ [41, 69]. We are reminded that Floquet solutions are non-zero solutions such that $x(t+w) \equiv \lambda x(t)$, $-\infty < t < \infty$ (see [54]). These solutions can be represented in the form $x(t) = e^{\mu t} p(t)$ where $p(t+w) = p(t)$ and $\lambda = e^{\mu w}$. The λ are known as the characteristic multipliers. In chapter 8 of [41] theory relating to delay differential equations, analagous to the Floquet Theory for ODEs (see [46, 80]), is developed.

We consider the continuous equation (7.6) which, following discretisation by the trapezium rule and rearrangement, becomes equation (7.12). If $X(t)$ is a Floquet type solution of equation (7.1) then it satisfies

$$(7.25) \quad X'(t) = \sum_{j=0}^m b_j(t) X(t - jw)$$

with

$$(7.26) \quad X(t) = e^{\mu t} p(t) \text{ where } p(t+w) = p(t).$$

This expression for X satisfies

$$(7.27) \quad X(t - jw) = \lambda^{m-j} X(t - mw) \text{ with } \lambda = e^{\mu w}.$$

and

$$(7.28) \quad X'(t) = \sum_{j=0}^m \lambda^{m-j} b_j(t) X(t - mw).$$

The discrete scheme corresponding to (7.25) is

$$(7.29) \quad X_{n+1} = X_n + \frac{h}{2} \sum_{j=0}^m (b_{j,n} X_{n-jN} + b_{j,n+1} X_{n+1-jN}).$$

For Floquet solutions, we set

$$(7.30) \quad X_n = e^{\mu n h} p_n = \Lambda^n p_n \text{ where } \Lambda = e^{\mu h} \text{ and } \Lambda^N = \lambda.$$

so that

$$(7.31) \quad X_n = \lambda X_{n-N} = \lambda^m X_{n-mN}.$$

We set

$$(7.32) \quad p_n = p_{n-N}.$$

We can use (7.31) to write (7.29) as

$$(7.33) \quad X_{n+1} = X_n + \frac{h}{2} \sum_{j=0}^m \lambda^{m-j} (b_{j,n} X_{n-mN} + b_{j,n+1} X_{n+1-mN})$$

which is the discretisation of (7.28) using the trapezium rule.

From a Floquet viewpoint, for a solution to be a small solution we require λ to be very small. As a consequence, instead of considering whether (7.1) admits small solutions, we are able to consider whether the equation $x'(t) = b_m(t)x(t - m\omega)$ admits small solutions. Computationally this is clearly more efficient.

Since a function which is ω -periodic is also $m\omega$ -periodic this is equivalent to considering whether the equation $y'(t) = b_m(t)y(t - \tau)$ with $b_m(t + \tau) = b_m(t)$ admits small solutions. We make the following observations:

$x'(t) = b_m(t)x(t - m\omega)$ admits small solutions if and only if $b_m(t)$ changes sign on $[0, m\omega]$.

$b_m(t)$ changes sign on $[0, m\omega]$ if and only if $b_m(t)$ changes sign on $[0, \omega]$.

$b_m(t)$ changes sign on $[0, \omega]$ if and only if $x'(t) = b_m(t)x(t - \omega)$ admits small solutions.

We are thus able to consider a much simpler problem, since only $b_m(t)$ is involved in the discretisation.

7.4.2 Numerical results

We illustrate our approach using Floquet solutions of some of the examples from section 7.3.4. We display the eigenspectra arising from the discretisation of equation $x'(t) = b_m(t)x(t - m\omega)$ using the trapezium rule.

Example 7.4.1 We consider two of the cases of equation (7.1) with $b_0(t) \equiv 0, w = 1, m = 2$ which were presented in example 7.3.2. In this case the theory states that if $b_2(t)$ changes sign on $[0, 1]$ then small solutions are admitted. The

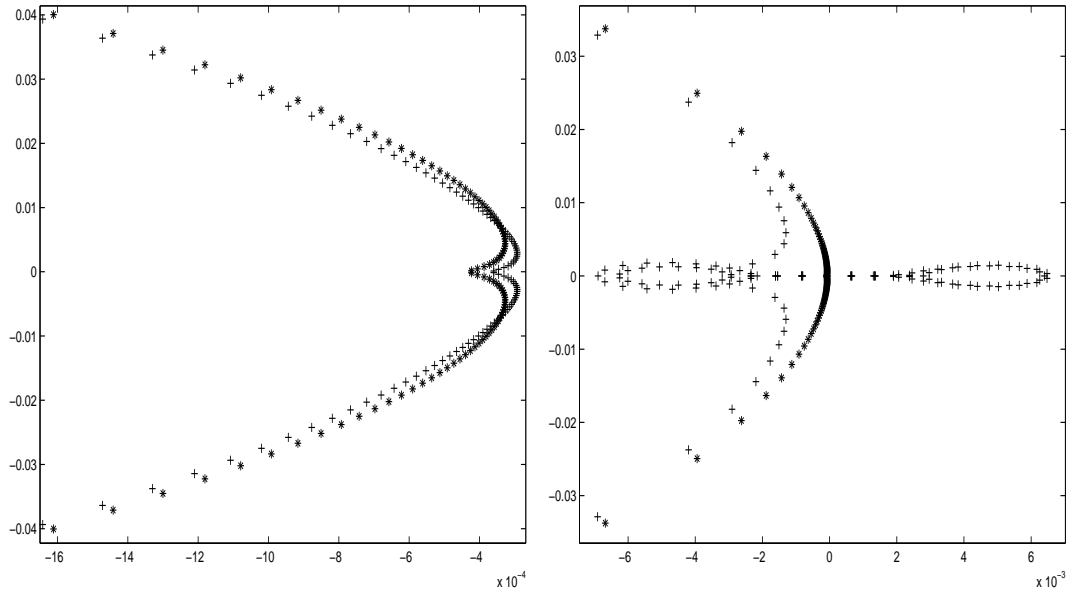


Figure 7.5: Left: $b_2(t)$ does not change sign Right: $b_2(t)$ changes sign

left-hand eigenspectra of Figure 7.5 arises from (7.1) with $b_1(t) = \sin 2\pi t + c$, $b_2(t) = \sin 2\pi t + 1.8$ and the right-hand eigenspectra arises from (7.1) with $b_1(t) = \sin 2\pi t + c$, $b_2(t) = \sin 2\pi t + 0.3$. As expected we observe additional eigenspectra in the case when $b_2(t)$ changes sign.

Example 7.4.2 In Figure 7.6 we present the eigenspectra resulting from equation (7.1) with $w = 1$, $m = 4$ and $b_i(t)$ as in example 7.3.3 for $i = 0, \dots, 4$. As expected we observe additional eigenspectra in the case when $b_4(t)$ changes sign.

If we compare the eigenspectra in examples 7.4.1 and 7.4.2 with the corresponding eigenspectra in examples 7.3.2 and 7.3.3 we observe a decrease in the complexity of the eigenspectra without losing the ease and clarity with which the presence of small solutions can be detected. We also observed a decrease in the computational time needed.

7.4.3 Some observations

1. We note that the diagrams in section 7.3.4 involve only small values of m . We would expect to see an even more significant decrease in complexity for larger values of m , particularly with small values of ω .
2. An alternative approach to using Floquet solutions of the multi-term continuous problem followed by discretisation would be to reduce the multi-term continuous problem to a multi-term discrete problem, (as in section

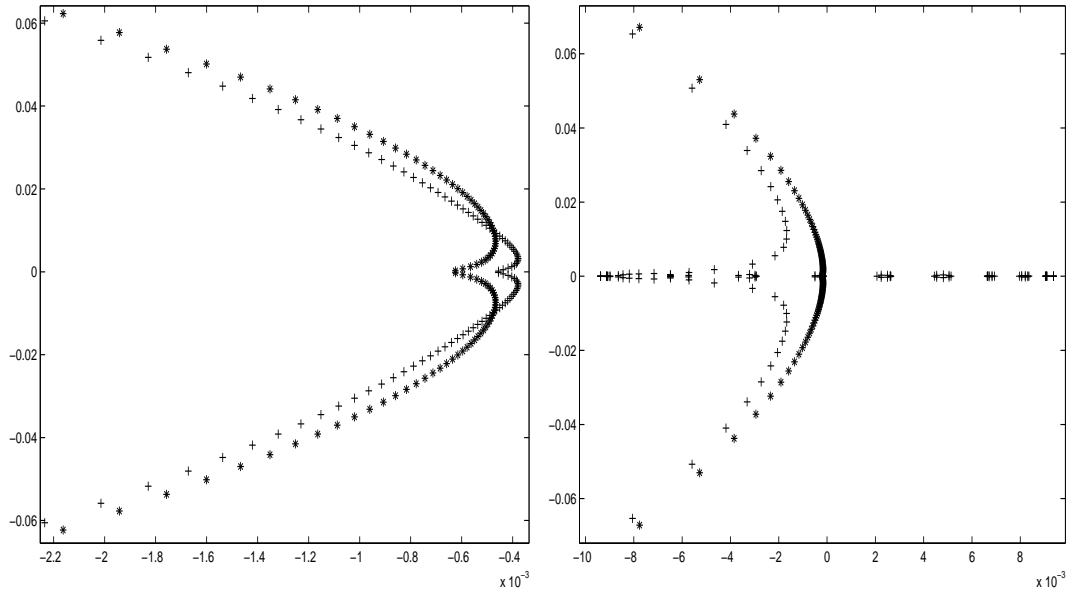


Figure 7.6: Left: $b_4(t)$ does not change sign Right: $b_4(t)$ changes sign

7.3), followed by consideration of the Floquet solutions to reduce it to a single term discrete problem. This would not seem as efficient but may provide additional insight for non-autonomous difference equations.

7.5 Conclusion

The decisions made, based on the eigenspectra resulting from our numerical scheme, about the presence, or otherwise, of small solutions to equations of the form (7.1) are consistent with the known theory. We again take the presence of additional trajectories in the eigenspectra arising from the non-autonomous problem, when compared to that arising from the equivalent autonomous problem, to indicate the presence of small solutions and we conclude that we are indeed able to adapt our numerical method to predict the presence of small solutions for equations of the form (7.1). We have seen that there may be significant advantages in considering Floquet type solutions in terms of the complexity of the eigenspectra obtained. Indeed, by using a Floquet solution approach, we have reduced the problem to a type considered in chapter 4 (see also [28]), that is, to a scalar DDE with a single delay where the delay and the period are equal. Having already established a reliable method for detecting small solutions to single delay DDEs (see chapter 4) we conclude that the Floquet approach leads to a reliable method for successfully detecting the presence, or otherwise, of small

solutions to multi-delay differential equations of the form (7.1).

Chapter 8

Single delays revisited

8.1 The one-dimensional case

In this chapter we return to the scalar case with a single delay and consider delay differential equations with periodic coefficients such that the delay, d , and the period, p , are not equal but are commensurate. We consider equations of the form

$$(8.1) \quad x'(t) = a(t)x(t) + b(t)x(t-d), \quad t \geq 0.$$

where $a(t)$ and $b(t)$ are bounded, real, continuous functions with period, p , so that $a(t+p) = a(t)$ and $b(t+p) = b(t)$. In this case we need to specify a continuous function on the interval $[-d, 0]$ in order for (8.1) to possess a unique solution, $x(t)$. We let $d = \frac{d_1}{d_2}$ and $p = \frac{p_1}{p_2}$ and assume that d_1, d_2, p_1 and p_2 are positive integers such that the delay, d , and the period, p , are expressed in their lowest terms.

8.1.1 Using a transformation to remove the instantaneous term

Adopting the method used in section 2 of [33] we can use the transformations

$$y(t) = x(t) \exp^{-\int_{-\frac{d_1}{d_2}}^t a(\sigma) d\sigma} \quad \text{and} \quad \hat{b}(t) = b(t) \exp^{-\int_{t-\frac{d_1}{d_2}}^t a(\sigma) d\sigma},$$

with initial data

$$y(\theta) = \phi(\theta) \exp^{-\int_{-\frac{d_1}{d_2}}^{\theta} a(\sigma) d\sigma}, \quad -\frac{d_1}{d_2} \leq \theta \leq 0,$$

to write (8.1) in the form

$$(8.2) \quad y'(t) = \hat{b}(t)y(t - \frac{d_1}{d_2}).$$

We observe that $\hat{b}(t)$ is a p -periodic function which changes sign if and only if $b(t)$ changes sign. We are thus able to consider equation (8.1) in reduced form, with $a(t) = 0, b(t + \frac{p_1}{p_2}) = b(t)$ as

$$(8.3) \quad x'(t) = b(t)x(t - \frac{d_1}{d_2}), \quad t \geq 0.$$

We illustrate the above transformation with example 8.1.1.

Example 8.1.1 Consider

$$(8.4) \quad x'(t) = (\sin \frac{3\pi t}{2} + c)x(t) + (\sin 2\pi t + 0.3)x(t - \frac{1}{2}).$$

In this case $d = 0.5, p = 4, b(t) = \sin(2\pi t + 0.3), a(t) = \sin(\frac{3\pi t}{2}) + c$. We see that

$$\begin{aligned} \int_{-d}^t a(\sigma)d\sigma &= \int_{-0.5}^t \left(\sin \left(\frac{3\pi(\sigma)}{2} \right) + c \right) d\sigma \\ &= -\frac{2}{3\pi} \left(\cos \left(\frac{3\pi t}{2} \right) + \frac{1}{\sqrt{2}} \right) + c(t + 0.5), \end{aligned}$$

and

$$\int_{t-0.5}^t a(\sigma)d\sigma = \frac{2}{3\pi} \left\{ 2 \sin \left(\frac{3\pi t}{2} - \frac{1}{8} \right) \sin \frac{3\pi}{8} \right\} + \frac{c}{2}.$$

This leads to

$$y(t) = e^{\frac{2}{3\pi} \left(\cos \frac{3\pi t}{2} + \frac{1}{\sqrt{2}} \right)} e^{-c(t+0.5)} x(t)$$

and

$$\hat{b}(t) = e^{-\frac{2}{3\pi} \left\{ 2 \sin \left(\frac{3\pi t}{2} - \frac{1}{8} \right) \sin \frac{3\pi}{8} \right\}} e^{-\frac{c}{2}} (\sin 2\pi t + 0.3).$$

We observe that the period of \hat{b} is 4. The two problems can be shown to be equivalent.

8.2 Analytical results

Analytical theory concerning equations of the form (8.1) is generally less well developed than that concerning the particular case in which $p = 1, d = 1$.

If b is a continuous function of rational period with isolated zeros and $b(t) > 0$ then equation (8.1) with $d = 1$ has no small solutions [75]. From Theorem 3.4 in [41] we know that, if the zeros of $b_m(t)$ are isolated, then

$$(8.5) \quad \dot{x}(t) = \sum_{j=0}^m b_j(t)x(t - jw), t \geq s, x_s = \phi$$

with $b_j(t), j = 0, \dots, m$ continuous w -periodic functions, has small solutions if and only if b_m changes sign. If the delay is an integer multiple of the period, say $d = mp$, then we can regard equation (8.3) to be of the form (8.5) with $b_m(t) = b(t), w = p$ and $b_j(t) \equiv 0$ for $j = 0, 1, \dots, m - 1$. Hence we know that if $b(t)$ changes sign then (8.3) has small solutions. Alternatively, from [73], we know that if $d = mp$ where $m \in \mathbb{N}$ then the system of eigenvectors and generalised eigenvectors is complete for $\dot{x}(t) = b_0(t) + b_1(t)x(t - d)$ if $b_1(t)$ does not change sign. Much less is known if the ratio between the delay, d , and the period, p , is non-integer.

Remark 8.2.1 It is not possible to write equation (8.3) in the form of equation (8.5) if the delay is not an integer multiple of the period, as can be seen by the following argument: Assume that it is possible to write

$$x'(t) = b(t)x(t - d), \quad b(t + p) = b(t)$$

in the form

$$x'(t) = \sum_{j=0}^m b_j(t)x(t - jw), \quad b_j(t + w) = b_j(t).$$

In this case there exists $j \in \mathbb{N}$ and $k \in \mathbb{N}$ such that

$$jw = d \text{ and } w = kp.$$

It follows that $j(kp) = d$ or $d = (jk)p$. Since $jk \in \mathbb{N}$ this equation is only satisfied if the delay is an integer multiple of the period.

For example, if $p = \frac{2}{3}$, and $d = \frac{1}{2}$ then we would require $j, k \in \mathbb{N}$ such that $\frac{2}{3}k = w$ and $jw = \frac{1}{2}$. This leads to $jk = \frac{3}{4}$ which cannot be satisfied with $j, k \in \mathbb{N}$.

The following results concerning equations of the form (8.1) can be found in [50].

- The time dependence of $a(t)$ can be eliminated by considering the Floquet decomposition of the non-delayed part
- W. Just states that ‘the competition between the two timescales, the delay and the external period cause intricate structures’.
- Equation (8.1) can be reduced to a system of ODEs if the ratio of the period and the delay is rational, but a full analysis of the resulting system is not easy. A variation in the period or the delay changes the dimension of the system.

Remark 8.2.2 The autonomous system is not clearly defined when p and d are not equal [79]. However, when the detection of small solutions is the major concern this is not of vital importance. The existence of more than one asymptotic curve in the eigenspectrum is evidence that small solutions are present.

Justification for our approach

We now provide justification for our approach. If we let

$$(8.6) \quad y(t) = f(t)x(t) \text{ with } f'(t) = -a(t)f(t) \text{ and hence } f(t) = e^{-\int_{-d}^t a(\sigma)d\sigma}$$

then we can transform

$$(8.7) \quad x'(t) = a(t)x(t) + b(t)x(t-d)$$

into an equation of the form

$$(8.8) \quad y'(t) = \hat{b}(t)y(t-d) \text{ where } \hat{b}(t) = e^{-\int_{t-d}^t a(\sigma)d\sigma}b(t).$$

If we consider

$$(8.9) \quad \hat{b}(t) = g(t)b(t)$$

we see that

$$(8.10) \quad f(t) = g(t)f(t-d).$$

As in section 7.3.2 we can now consider the discrete forms of (8.7) and (8.8) when solved using the trapezium rule with fixed step length $h = \frac{1}{N}$. We obtain respectively the equations

$$(8.11) \quad x_{n+1} = x_n + \frac{h}{2} \{a_n x_n + b_n x_{n-N}\} + \frac{h}{2} \{a_{n+1} x_{n+1} + b_{n+1} x_{n+1-N}\}$$

and

$$(8.12) \quad y_{n+1} = y_n + \frac{h}{2} \left\{ \hat{b}_n y_{n-N} + \hat{b}_{n+1} y_{n+1-N} \right\}.$$

We continue in a similar manner to that used in section 7.3.2 (see also section 2.2 in [30]). We derive the approximate transformation that relates these two equations from the discrete forms (using the trapezium rule) of the transformation that applied exactly in the continuous case.

$$(8.13) \quad f_{n+1} = f_n - \frac{h}{2} \{a_n f_n + a_{n+1} f_{n+1}\}$$

$$(8.14) \quad y_n = f_n x_n$$

$$(8.15) \quad \hat{b}_n = g_n b_n$$

$$(8.16) \quad f_n = g_n f_{n-N}$$

Equation (8.11) can be written as

$$(8.17) \quad x_{n+1} = x_n \left\{ \frac{1 + \frac{h}{2}a_n}{1 - \frac{h}{2}a_{n+1}} \right\} + \frac{h}{2} \left\{ \frac{1}{1 - \frac{h}{2}a_{n+1}} \right\} \{b_n x_{n-N} + b_{n+1} x_{n+1-N}\}.$$

Using (8.14) and (8.15) in (8.12) gives

$$(8.18) \quad f_{n+1} x_{n+1} = f_n x_n + \frac{h}{2} \{g_n b_n f_{n-N} x_{n-N} + g_{n+1} b_{n+1} f_{n+1-N} x_{n+1-N}\}$$

which leads to

$$(8.19) \quad f_{n+1} x_{n+1} = f_n x_n + \frac{h}{2} \{b_n f_n x_{n-N} + b_{n+1} f_{n+1} x_{n+1-N}\}$$

giving

$$(8.20) \quad x_{n+1} = \frac{f_n}{f_{n+1}} x_n + \frac{h}{2} \left\{ \frac{f_n}{f_{n+1}} b_n x_{n-N} + b_{n+1} x_{n+1-N} \right\}.$$

Equation (8.13) can be arranged to give

$$(8.21) \quad \frac{f_n}{f_{n+1}} = \frac{1 + \frac{h}{2}a_{n+1}}{1 - \frac{h}{2}a_n}.$$

Hence

$$(8.22) \quad x_{n+1} = \frac{1 + \frac{h}{2}a_{n+1}}{1 - \frac{h}{2}a_n} x_n + \frac{h}{2} \left\{ \frac{1 + \frac{h}{2}a_{n+1}}{1 - \frac{h}{2}a_n} b_n x_{n-N} + b_{n+1} x_{n+1-N} \right\}.$$

We can continue as in section 7.3.2 and show that the error term is $O(h^2)$.

We are thus able to focus our attention on equation (8.3) in reduced form, with $a(t) \equiv 0$ as

$$(8.23) \quad x'(t) = b(t)x(t-d), \quad t \geq 0.$$

with $b(t+p) = b(t)$ and p and d commensurate.

8.3 Introductory background theory

Proposition 8.3.3 states that equation (8.3) admits small solutions if $b(t)$ changes sign on $[0, p]$. We first prove some results which are necessary to underpin our proof of this proposition.

Proposition 8.3.1 Let $d = \frac{d_1}{d_2}, p = \frac{p_1}{p_2}$ where p_1, p_2, d_1, d_2 are positive integers such that d and p are expressed in their lowest terms. Let $b(t)$ be a periodic function with period p . If $p > d$ and $b(t)$ changes sign on $[0, p]$ but not on $[0, d]$ then the shortest interval of the form $[0, jd]$ on which we can guarantee that $b(t)$ changes sign is $[0, p_1d_2d]$.

Proof. Let $b(t)$ change sign on $[0, p]$. Since $p > d$, if $b(t)$ does not change sign on $[0, d]$ then it may change sign on $[0, 2d]$.

Similarly, if $b(t)$ does not change sign on $[0, kd]$ then it may change sign on $[0, (k+1)d]$.

Since $b(t)$ changes sign on $[0, p]$ then $b(t)$ is guaranteed to change sign on $[0, (k+1)d]$ if $(k+1)d \geq p$, that is if $(k+1)\frac{d_1}{d_2} \geq \frac{p_1}{p_2}$.

It is clear that $(k+1)\frac{d_1}{d_2} \geq \frac{p_1}{p_2}$ if and only if $(k+1)d_1p_2 \geq p_1d_2$. Here, $d_1p_2 \in \mathbb{N}$ and when d_1p_2 takes its minimum value of 1, then the inequality holds if and only if $(k+1) \geq p_1d_2$.

For larger values of d_1p_2 the minimum value of $(k+1)$ required to satisfy the inequality is reduced by a factor equal to d_1p_2 .

Hence the inequality is satisfied in all cases if and only if $(k+1)$ is at least p_1d_2 . Hence, if $b(t)$ changes sign on $[0, \frac{p_1}{p_2}]$ then it is guaranteed to change sign on $[0, p_1d_2d]$. \square

Proposition 8.3.2 If $b(t)$ changes sign on $[0, p]$ then $b(t - id)$ changes sign on $[0, d]$ for some $i = 1, 2, \dots, p_1d_2$.

Proof. If $b(t)$ changes sign on $[0, p]$ then, by proposition 8.3.1, $b(t)$ is guaranteed to change sign on $[0, p_1d_2d]$.

Since $b(t)$ changes sign on $[0, p_1d_2d]$ there exists an $\alpha \in [0, p_1d_2d]$ such that $b(\alpha) = 0$.

We can cover the interval $[0, p_1d_2d]$ by p_1d_2 intervals of the form $[kd, (k+1)d]$.

Let $\alpha \in [\gamma d, (\gamma+1)d]$, $\gamma \in \mathbb{N}$, $0 \leq \gamma \leq p_1d_2$.

The graph of $b(t)$ is transformed to that of $b(t-d)$ by a shift of d units to the right.

$b(t)$ can be regarded as being of period (d_2p_1d) .

Hence, if $b(t)$ changes sign on $[kd, (k+1)d]$ then, for $k \geq p_1d_2$, $b(t)$ also changes sign on $[(k-d_2p_1)d, (k+1-d_2p_1)d]$.

If $b(t)$ changes sign on $[\gamma d, (\gamma+1)d]$ then

$b(t-d)$ changes sign on $[(\gamma+1)d, (\gamma+2)d]$,

$b(t-2d)$ changes sign on $[(\gamma+2)d, (\gamma+3)d]$,

\vdots

\vdots

$b(t-id)$ changes sign on $[(\gamma+i)d, (\gamma+1+i)d]$.

If $(\gamma+i) = p_1d_2$ then $b(t-id)$ changes sign on $[p_1d_2d, (p_1d_2+1)d]$ and hence also

on $[0, d]$. If $\gamma + i = p_1 d_2$ then $i = p_1 d_2 - \gamma$.

Hence if $b(t)$ changes sign on $[\gamma d, (\gamma + 1)d]$ then $b(t - (p_1 d_2 - \gamma)d)$ changes sign on $[0, d]$. \square

Proposition 8.3.3 The equation

$$(8.24) \quad x'(t) = b(t)x(t - \frac{d_1}{d_2}), \quad t \geq 0, \quad \text{with } b(t + \frac{p_1}{p_2}) = b(t),$$

admits small solutions if $b(t)$ changes sign on $[0, \frac{p_1}{p_2}]$.

The result has been proven analytically [79]. However, to our knowledge, the proof cannot currently be found in the literature and hence we choose to include the following proof.

Proof. If $d = mp$ then theory states that the equation admits small solutions if $b(t)$ changes sign. If the p -periodic function $b(t)$ has not changed sign on $[0, \frac{p_1}{p_2}]$ then it will not change sign on $[0, k\frac{p_1}{p_2}]$ for any $k \geq 1$.

If the delay, d , and the period, p , are equal then

$$(8.25) \quad x'(t) = b(t)x(t - d)$$

admits small solutions if $b(t)$ changes sign on $[0, p]$, that is, on $[0, d]$.

In general if $b(t)$ changes sign on $[0, p]$ then by remark 8.3.1 $b(t)$ is guaranteed to change sign on $[0, p_1 d_2 d]$.

Using (8.25) we can write

$$\begin{aligned} x'(t) &= b(t)x(t - d) \\ x'(t - d) &= b(t - d)x(t - 2d) \\ x'(t - 2d) &= b(t - 2d)x(t - 3d) \\ &\vdots \\ &\vdots \\ x'(t - p_1 d_2 d) &= b(t - p_1 d_2 d)x(t - (p_1 d_2 - 1)d). \end{aligned}$$

We introduce $y(t) = (x(t), x(t - d), x(t - 2d), \dots, x(t - p_1 d_2 d))^T$ and write

$$(8.26) \quad \begin{pmatrix} x'(t) \\ x'(t-d) \\ x'(t-2d) \\ \vdots \\ \vdots \\ x'(t-p_1 d_2 d) \end{pmatrix} = \begin{pmatrix} b(t) & 0 & \dots & \dots & \dots & 0 \\ 0 & b(t-d) & 0 & & & 0 \\ 0 & 0 & b(t-2d) & 0 & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & \ddots & \vdots \\ 0 & \dots & & & 0 & b(t-p_1 d_2 d) \end{pmatrix} \begin{pmatrix} x(t-d) \\ x(t-2d) \\ x(t-3d) \\ \vdots \\ \vdots \\ x(t-(p_1 d_2 + 1)d) \end{pmatrix},$$

giving

$$(8.27) \quad y'(t) = B(t)y(t - d).$$

Here $B(t) \in \mathbb{R}^{(p_1 d_2 + 1) \times (p_1 d_2 + 1)}$ and $B(t)$ is $\text{diag}(b(t), b(t-d), \dots, b(t-p_1 d_2 d))$. We know, using proposition 6.2.1, that equation (8.27) admits small solutions if at least one of $b(t-id)$ changes sign on $[0, d]$ for $i = 0, 1, \dots, p_1 d_2 d$. This is guaranteed by proposition 8.3.2 if $b(t)$ changes sign on $[0, p]$. Hence, (8.25) with $b(t)$ p -periodic, where $p = \frac{p_1}{p_2}$, and delay $d = \frac{d_1}{d_2}$ admits small solutions if $b(t)$ changes sign on $[0, p]$. \square

8.4 Applying the trapezium rule

We consider equation (8.3) and assume that d_1, d_2, p_1 and p_2 are positive integers such that the delay, d , and the period, p , are expressed in their lowest terms. We note that a function which has period $\frac{p_1}{p_2}$ can also be considered to have period $k \frac{p_1}{p_2}$ for $k \in \mathbb{N}$. The periodicity of $b(t)$ implies that $b_n = b_{n - \frac{pN}{d}}$. We apply the trapezium rule with step size $h = \frac{d}{N}$. We observe that when $\frac{pN}{d}$ is an integer we can use the approach adopted in [28] (see also chapter 4). In this case we have

$$(8.28) \quad y_{n+1} = A(n)y_n,$$

$$(8.29) \quad y_{n+\frac{pN}{d}} = A\left(n + \frac{pN}{d} - 1\right) \cdot A\left(n + \frac{pN}{d} - 2\right) \cdots A(n)y_n,$$

which we can write as

$$(8.30) \quad y_{n+\frac{pN}{d}} = C y_n \text{ where } C = \prod_{i=0}^{\frac{pN}{d}-1} A(n-i).$$

We note that taking $N = mp_2 d_1$ is guaranteed to fulfil the requirement that $\frac{pN}{d}$ is an integer¹ and gives the step length $h = \frac{1}{mp_2 d_2}$. Hence, when we apply our method to a particular problem, the step length is governed by the value of $p_2 d_2$ and the arbitrary choice of m .

We again compare the eigenspectra arising from the non-autonomous problem and the potentially equivalent autonomous problem, in this case, $x'(t) = b(t)x(t-d)$, $b(t+p) = b(t)$ and $x'(t) = \hat{b}x(t-d)$, $\hat{b} = \frac{1}{p} \int_0^p b(t)dt$.

8.5 Numerical results

We consider equation

$$(8.31) \quad x'(t) = \left\{ \sin\left(\frac{2\pi t}{p}\right) + c \right\} x(t-d)$$

¹If f_i is equal to the highest common factor of p_i and d_i then $N = \frac{p_2 d_1}{f_1 f_2}$ is the smallest number which guarantees that $\frac{pN}{d} \in \mathbb{N}$.

with $d = \frac{d_1}{d_2}, p = \frac{p_1}{p_2}$, where $d_1, d_2, p_1, p_2 \in \mathbb{N}$. We apply the trapezium rule with $h = \frac{d}{N}$.

We first consider the case when $p = 1$ and the delay $d \in \mathbb{N}$ before moving on to a more general case.

In chapter 4 and in [28] we classified diagrams of the eigenspectra under three headings, according to whether or not the equation admitted small solutions and whether or not $\int_0^1 b(t)dt = 0$. In this chapter we choose to refer to these three characteristic shapes of the eigenspectrum, illustrated in Figure 8.1, as *basic pattern A* when no small solutions are admitted (Left), *basic pattern B* when almost all solutions are small (Centre) and *basic pattern C* when the equation admits small solutions (Right).

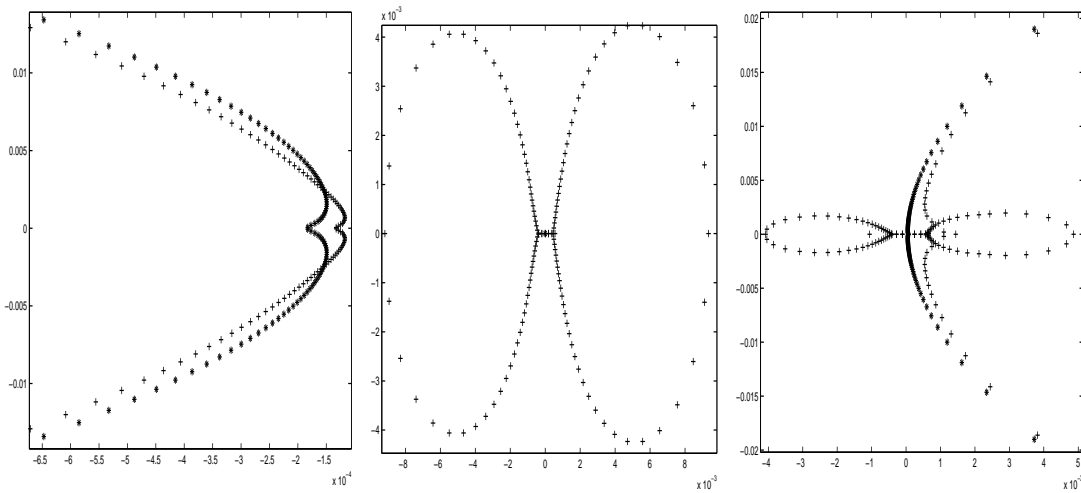


Figure 8.1: Left: *Basic pattern A*: No small solutions.
 Centre: *Basic pattern B*: Almost all solutions are small.
 Right: *Basic pattern C*: Equation admits small solutions.

8.5.1 $p = 1, d \in \mathbb{N}$

In this case, using $N = mp_2d_1$ implies that $h = \frac{1}{m}$ and $A(n) = A(n - m)$. We are again able to consider the results of our experiments in 3 categories, depending upon whether or not $b(t)$ changes sign on $[0, p]$ and whether or not $\frac{1}{p} \int_0^p b(t)dt = 0$. In Figures 8.2, 8.3 and 8.4 we illustrate the cases detailed in Table 8.1.

We compare our diagrams to those in Figure 8.1. We observe the presence of additional trajectories in the eigenspectra in cases 3, 4, 5 and 6, which, based on our previous work, we take to indicate the presence of small solutions. This is in accordance with the theory (presented in section 8.3) since $b(t)$ changes sign. We observe that in each case the characteristic shape of the trajectory resulting

Figure	Statement concerning small solutions	Case	c	p	d	m	Compare with basic pattern
8.2	Equation does not admit small solutions	1	1.3	1	3	128	A
		2	1.3	1	7	128	
8.3	Almost all solutions are small	3	0	1	2	128	B
		4	0	1	5	128	
8.4	Equation admits small solutions	5	0.5	1	3	128	C
		6	0.5	1	8	128	

Table 8.1: Examples used to illustrate the case when $p = 1$ and $d \in \mathbb{N}$

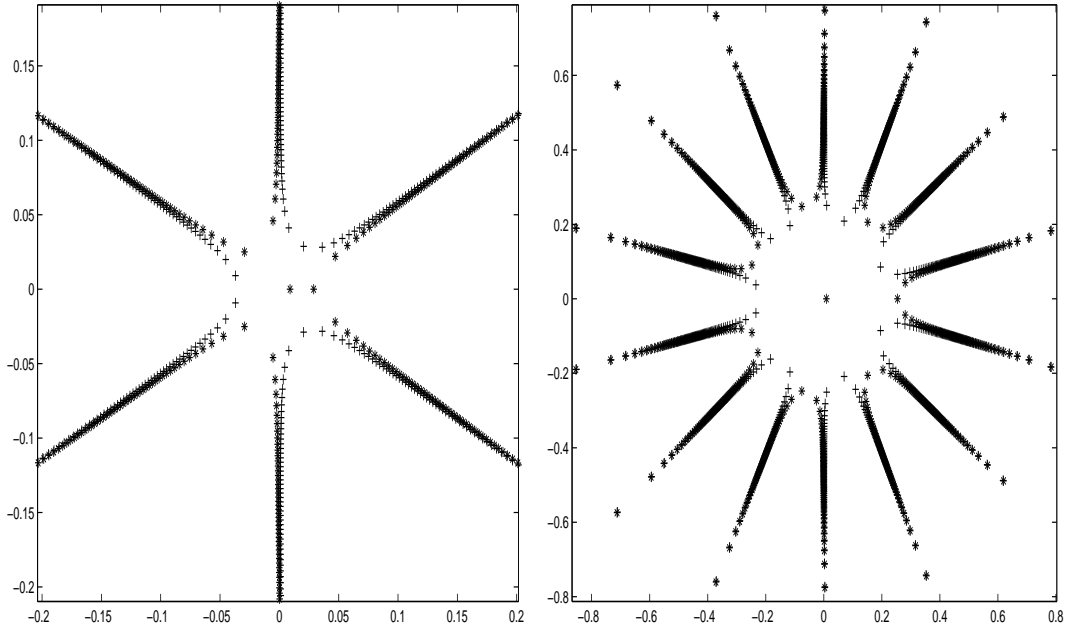


Figure 8.2: No small solutions. Left: Case 1 Right: Case 2

from the case when $p = 1$ and $d = 1$ is repeated d times. The proximity of the trajectories in cases 1 and 2 indicates the existence of an equivalent autonomous problem when $b(t)$ does not change sign on $[0, p]$, in accordance with known theory. Figure 8.3 illustrates the case when almost all solutions are small solutions, which occurs when $\frac{1}{p} \int_0^p b(t) dt = 0$.

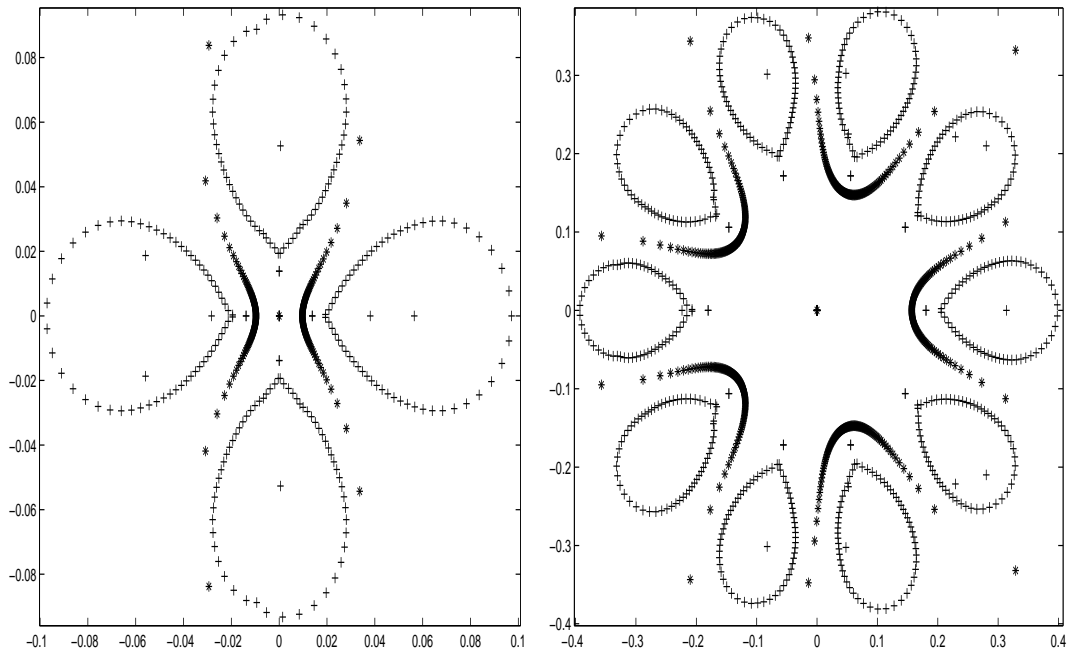


Figure 8.3: Almost all solutions are small. Left: Case 3 Right: Case 4

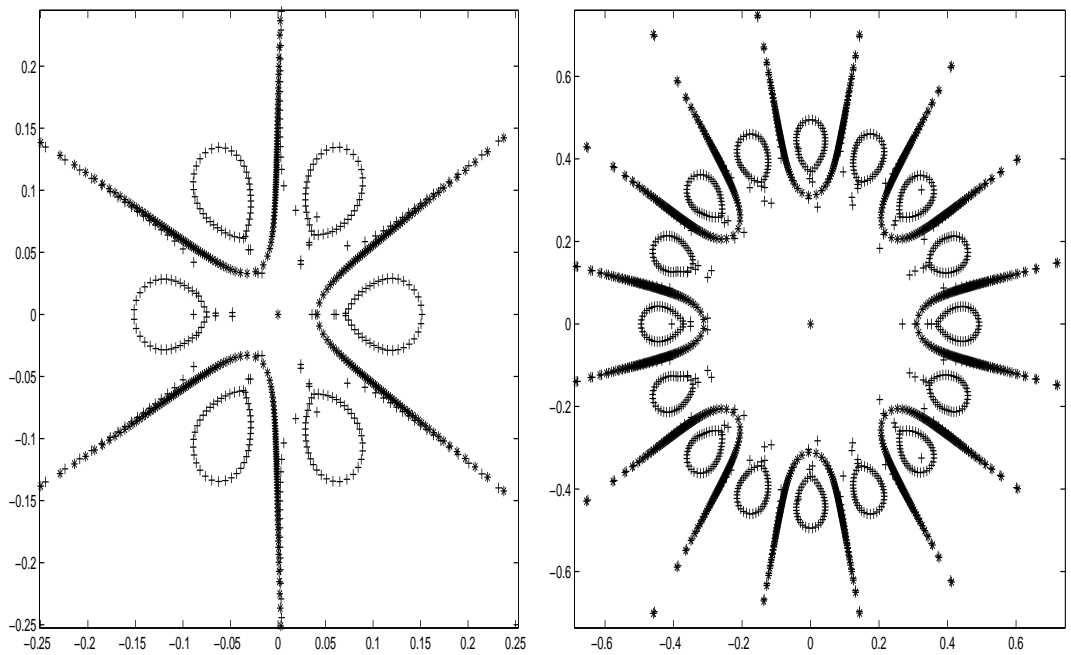


Figure 8.4: Equation admits small solutions. Left: Case 5 Right: Case 6

8.5.2 A more general case

We now consider the case when $d = \frac{d_1}{d_2}$ and $p = \frac{p_1}{p_2}$. We begin with equations for which $p < d$, and for which p_1 and d_1 , p_2 and d_2 , are relatively prime. In Figures 8.5, 8.6 and 8.7 we illustrate results of our experiments using the examples detailed in Table 8.2.

Fig.	Statement concerning small solutions	Eg.	c	p	d	m	Compare with basic pattern
8.5	Equation does not admit small solutions	1	1.3	1/7	1/3	36	A
		2	1.3	1/7	2/3	36	
8.6	Almost all solutions are small	3	0	1/8	1/3	36	B
		4	0	2/5	3/4	36	
8.7	Equation admits small solutions	5	0.5	2/5	1/2	50	C
		6	0.3	2/5	3/4	36	

Table 8.2: Examples used to illustrate the case when $p < d$, with p_i and d_i relatively prime for $i = 1, 2$

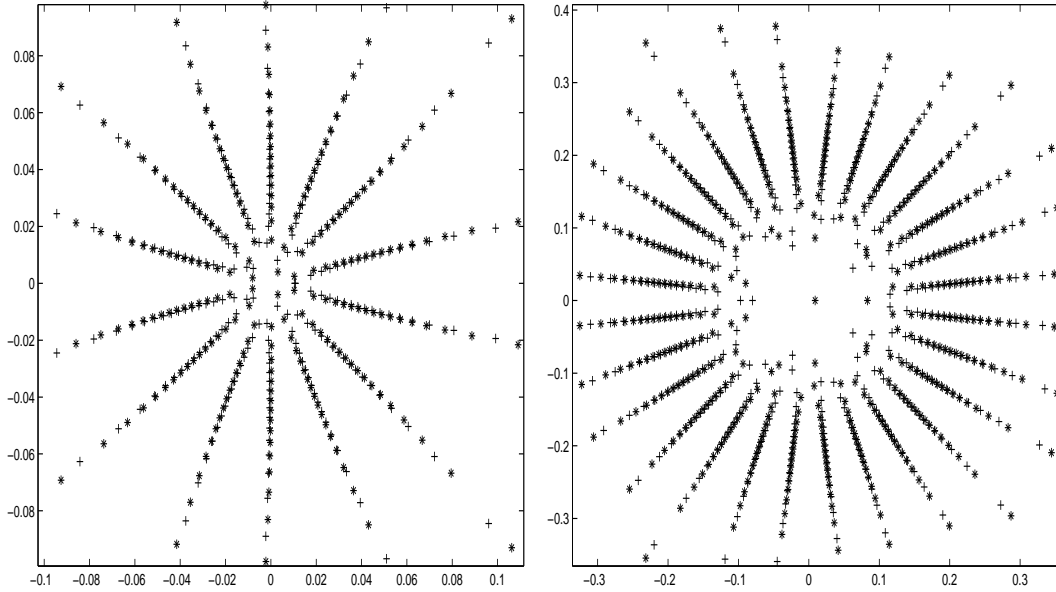


Figure 8.5: No small solutions. Left: Example 1 Right: Example 2

In accordance with the theory the trajectories shown in Figures 8.6 and 8.7 indicate clearly the presence of small solutions. We observe that the number of repetitions of the characteristic shape of the trajectory resulting from the case when $d = 1$ and $p = 1$ is equal to $p_2 d_1$.

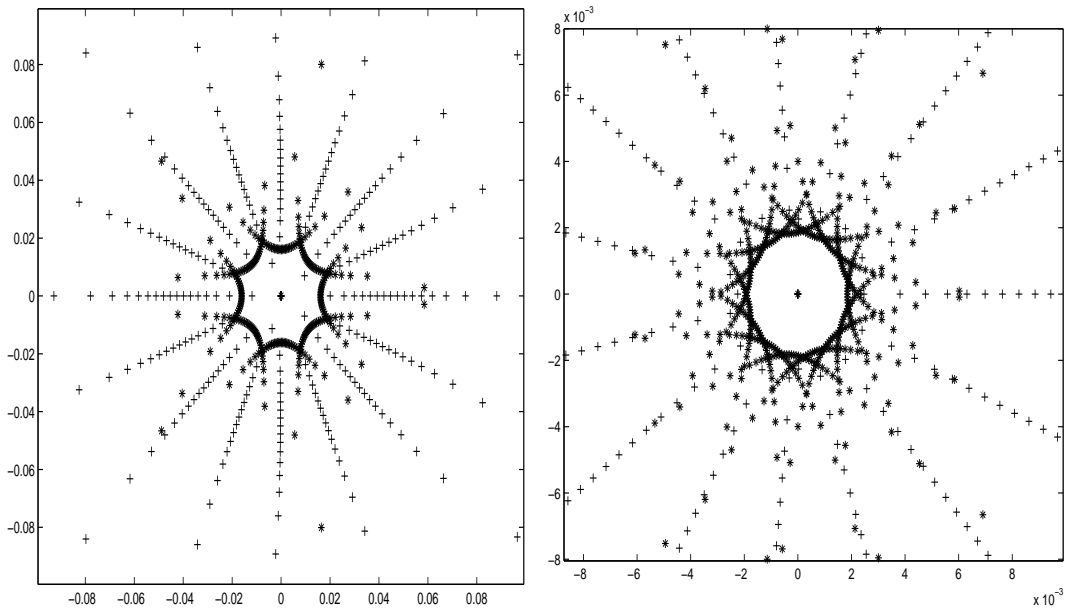


Figure 8.6: Almost all solutions are small.
 Left: Example 3 Right: Example 4

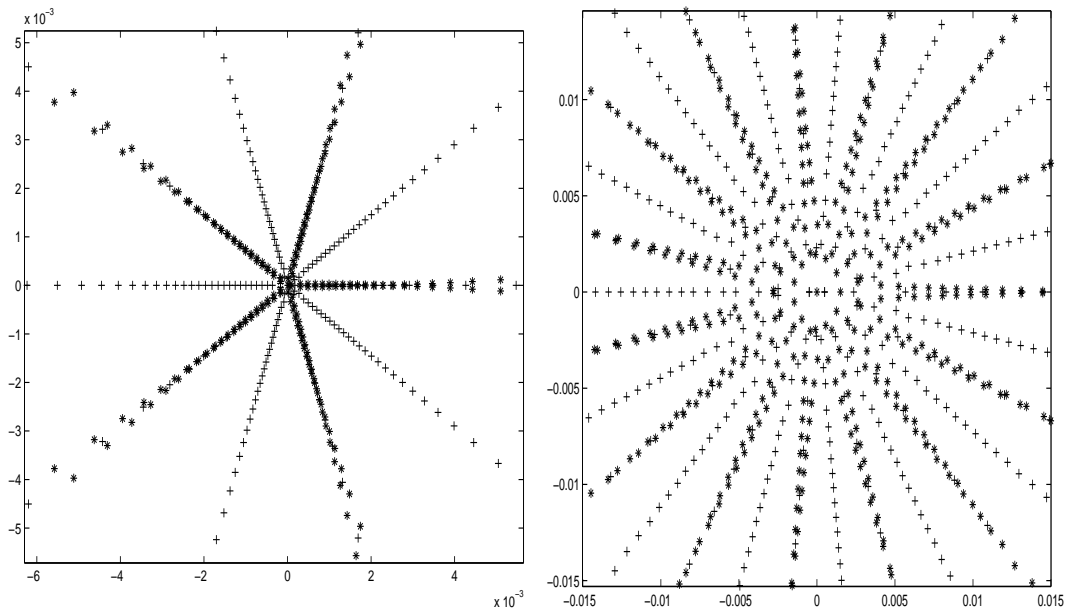


Figure 8.7: Equation admits small solutions.
 Left: Example 5 Right: Example 6

Proposition 8.5.1 Consider the equations $x'(t) = b(t)x(t - \frac{d_1}{d_2})$, $b(t + \frac{p_1}{p_2}) = b(t)$, $t \geq 0$,

and $x'(t) = \hat{b}x(t - \frac{d_1}{d_2})$ where $\hat{b} = \frac{1}{p} \int_0^p b(t)dt$ and $p = \frac{p_1}{p_2}$.

Let f_i be the highest common factor of p_i and d_i for $i = 1, 2$. The characteristic eigenspectra which result from the application of the trapezium rule to both equations in the case when $p_1 = 1, d_1 = 1, p_2 = 1, d_2 = 1$ is repeated $\frac{d_1 p_2}{f_1 f_2}$ times in the more general case.

Proof. In this proof we again refer to the pattern of the eigenvalue trajectories resulting when $p = 1$ and $d = 1$ as the *basic pattern* (see section 8.5). We can write $p_1 = f_1 p_u, d_1 = f_1 d_u, p_2 = f_2 p_\ell, d_2 = f_2 d_\ell$.

Due to the periodicity of $b(t)$ we obtain 1 *basic pattern* after $\frac{Np}{d}$ matrices, that is, 1 *basic pattern* after $\frac{Np_1 d_2}{d_1 p_2}$ matrices.

If $f_1 = 1$ and $f_2 = 1$ we obtain $d_1 p_2$ *basic patterns* after $Nd_2 p_1$ matrices.

More generally, we obtain 1 *basic pattern* after $\frac{Np_u d_\ell}{d_u p_\ell}$ matrices and hence $d_u p_\ell$ *basic patterns* after $Np_u d_\ell$ matrices, $N \in \mathbb{N}$.

Since $d_u p_\ell = \frac{d_1 p_2}{f_1 f_2}$ we obtain $\frac{d_1 p_2}{f_1 f_2}$ repetitions of the *basic pattern* after $Np_u d_\ell$, ($k \in \mathbb{N}$) matrices and the proposition is proved. \square

We provide illustration of this result in Figures 8.8 and 8.9 using the examples detailed in Table 8.3:

Fig. No.	Eg. No.	Small Solutions?	c	p	d	m	f_1	f_2	$\frac{p_2 d_1}{f_1 f_2}$	Number of repeats and basic pattern
8.8	7	Yes	0.5	2/3	4/5	36	2	1	6	6, C
8.8	8	Yes	0.4	1/4	3/8	36	1	4	3	3, C
8.9	9	Yes	0.8	3/14	6/7	10	3	7	4	4, C
8.9	10	No	1.4	4/9	2/3	20	2	3	3	3, A

Table 8.3: Examples used to illustrate the case when $p < d$, with p_i and d_i , for $i = 1, 2$, not relatively prime

We have chosen not to include diagrams for the case when $p > d$ but our experimental work confirmed the validity of proposition 8.5.1.

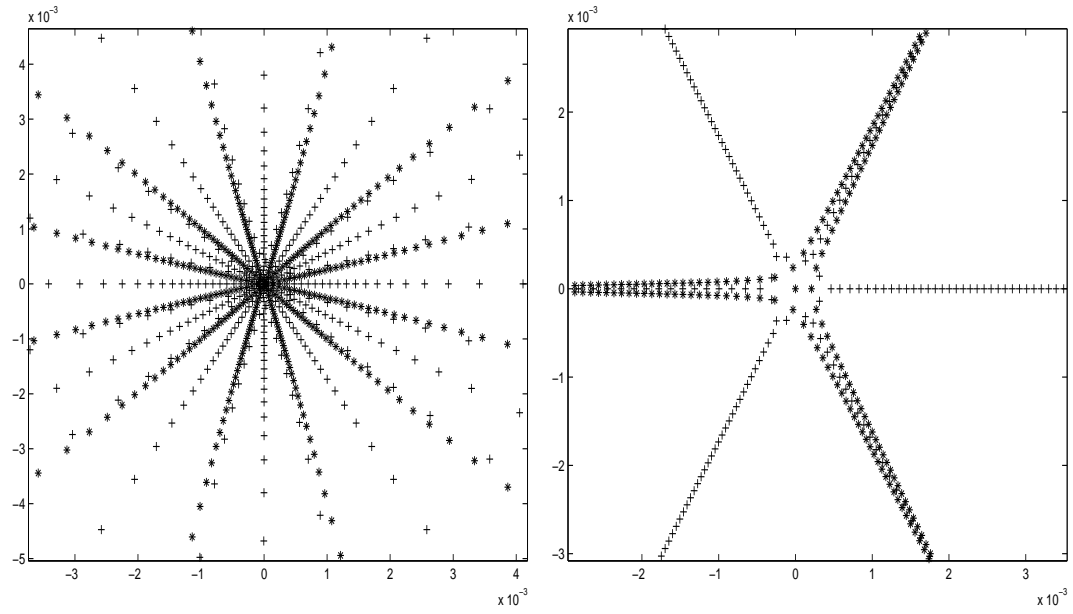


Figure 8.8: Additional trajectories are present. Small solutions are admitted.
 Left: Example 7 Right: Example 8

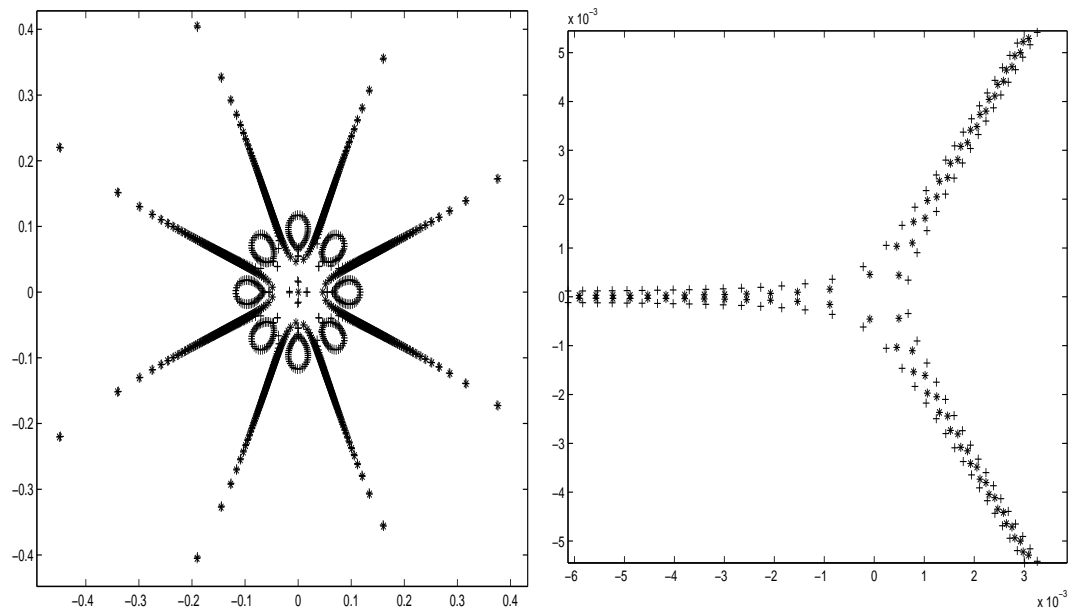


Figure 8.9: Left: Small solutions are admitted. Example 9
 Right: The equation does not admit small solutions. Example 10

8.6 Extension to higher dimensions

8.6.1 The two-dimensional case

Next we consider the two-dimensional case represented by the equation
(8.32)

$$y'(t) = A(t)y(t-d), \text{ where } A(t) = \begin{pmatrix} a_{11}(t) & a_{12}(t) \\ a_{21}(t) & a_{22}(t) \end{pmatrix} \text{ and } y(t) = \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix},$$

with $d = \frac{d_1}{d_2}$, $p = \frac{p_1}{p_2}$, and $a_{ij}(t+p) = a_{ij}(t)$.

Using $y(t) = (x_1(t), x_1(t-d), x_1(t-2d), \dots, x_1(t-p_1 d_2 d), x_2(t), x_2(t-d), x_2(t-2d), \dots, x_2(t-p_1 d_2 d))^T$.

We can write

$$y'(t) = \begin{pmatrix} x'_1(t) \\ x'_1(t-d) \\ x'_1(t-2d) \\ \dots \\ \dots \\ x'_1(t-p_1 d_2 d) \\ x'_2(t) \\ x'_2(t-d) \\ x'_2(t-2d) \\ \vdots \\ \vdots \\ x'_2(t-p_1 d_2 d) \end{pmatrix} = \begin{pmatrix} a_{11}(t)x_1(t-d) + a_{12}(t)x_2(t-d) \\ a_{11}(t-d)x_1(t-2d) + a_{12}(t-d)x_2(t-2d) \\ a_{11}(t-2d)x_1(t-3d) + a_{12}(t-2d)x_2(t-3d) \\ \vdots \\ \vdots \\ a_{11}(t-p_1 d_2 d)x_1(t-p_1 d_2 d - d) + a_{12}(t-p_1 d_2 d)x_2(t-p_1 d_2 d - d) \\ a_{21}(t)x_1(t-d) + a_{22}(t)x_2(t-d) \\ a_{21}(t-d)x_1(t-2d) + a_{22}(t-d)x_2(t-2d) \\ a_{21}(t-2d)x_1(t-3d) + a_{22}(t-2d)x_2(t-3d) \\ \vdots \\ \vdots \\ a_{21}(t-p_1 d_2 d)x_1(t-p_1 d_2 d - d) + a_{22}(t-p_1 d_2 d)x_2(t-p_1 d_2 d - d) \end{pmatrix}.$$

This can be written as $y'(t) = B(t)y(t-d)$ where $B(t) = \begin{pmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{pmatrix} y(t-d)$

with $D_{ij} = \text{diag}(a_{ij}(t), a_{ij}(t-d), a_{ij}(t-2d), \dots, a_{ij}(t-p_1 d_2 d))$. We will consider the case when $B(t)$ is triangular.

$B(t)$ is triangular

If $B(t)$ is upper triangular then small solutions exist if one of the diagonal elements changes sign on $[0, d]$ (see proposition 6.2.1 and proposition 6.3.1). Since $a_{11}(t)$ and $a_{22}(t)$ have period $p = \frac{p_1}{p_2}$, then if either (or both) changes sign on $[0, p]$ then at least one of the $a_{11}(t-id)$ or $a_{22}(t-id)$, ($i = 1, 2, \dots, p_1 d_2$), changes sign on $[0, d]$. Hence, if $a_{11}(t)$ or $a_{22}(t)$ changes sign on $[0, p]$ then the equation has small solutions. A similar statement can be made if $B(t)$ is lower triangular. We illustrate with the following examples.

Example 8.6.1 We consider equation $y'(t) = A(t)y(t-d)$, $A(t+p) = A(t)$ with p and d commensurate, $A(t) \in \mathbb{R}^{2 \times 2}$, $A(t) = \{a_{ij}(t)\}$ and $a_{ij} = \sin(\frac{2\pi t}{p}) + c_{ij}$. We consider the case when $a_{21} = 0$, that is when $A(t)$ is upper triangular and include the cases detailed in Table 8.4.

Fig.	Example	p	d	c_{11}	c_{12}	c_{22}	m	$\frac{p_2 d_1}{f_1 f_2}$
8.10	1	1/5	1	1.5	0.4	-1.6	40	5
8.10	2	2/5	3/10	1.8	1.5	1.1	16	3
8.11	3	1/4	1	0.2	0.4	1.6	64	4
8.11	4	2/3	4/9	0.4	-0.3	1.4	16	2
8.12	5	1/2	1	0.6	1.4	0.2	128	2
8.12	6	3/4	5/6	0.8	0.7	-0.3	20	10

Table 8.4: Examples used to illustrate the two-dimensional case

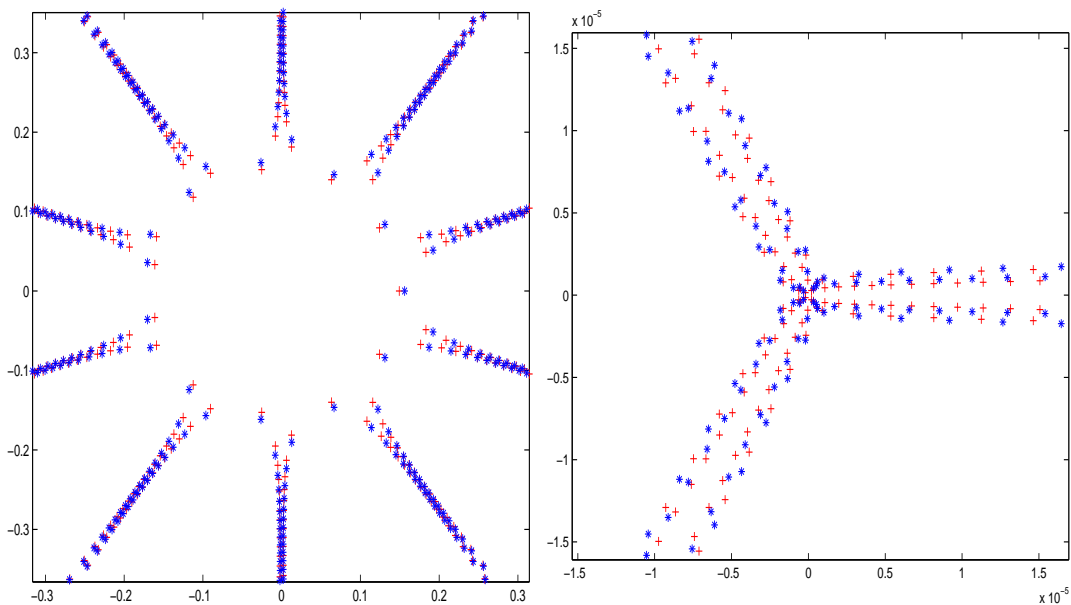


Figure 8.10: The two-dimensional case. Left: Example 1 Right: Example 2
No small solutions are present

We find that in the two-dimensional case we observe two sets of the multiples of the basic pattern that we observed in the one-dimensional case. This is particularly clear in the left-hand diagram of Figure 8.12 where we observe the presence of two sets of additional trajectories. In the left-hand diagram of Figure 8.11 we observe just one set of additional trajectories when only $a_{11}(t)$

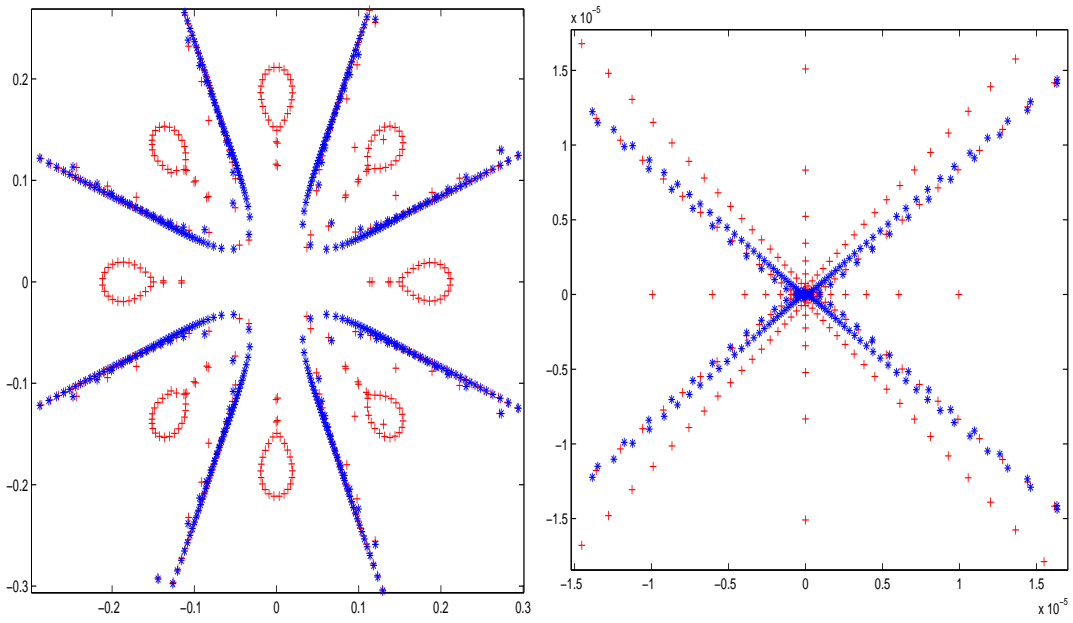


Figure 8.11: The two-dimensional case. Left: Example 3 Right: Example 4
 One function on the leading diagonal of $A(t)$ changes sign

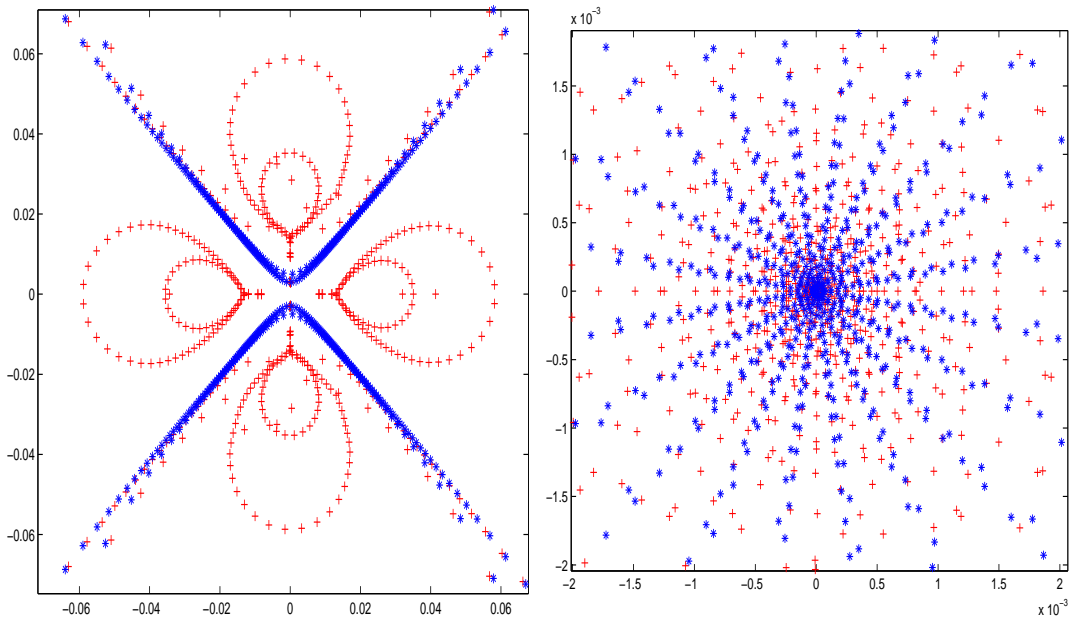


Figure 8.12: The two-dimensional case. Left: Example 5 Right: Example 6
 Both functions on the leading diagonal of $A(t)$ change sign

changes sign. We note that in the right-hand diagram of Figure 8.12 the shape of the eigenspectra is becoming less clear. Use of additional computational time to produce eigenspectra symmetrical about the real axis would increase the ease and clarity with which making a decision about the presence, or otherwise, of small solutions can be made. This will be discussed in chapter 10.

8.6.2 An example of the three-dimensional case

To illustrate application beyond two dimensions we include an example of the three-dimensional case. We consider equation $x'(t) = A(t)x(t - d)$ where $A(t)$ is such that $A(t + p) = A(t)$ and

$$A(t) = \begin{pmatrix} \sin(\frac{2\pi t}{p}) + 0.2 & \sin(\frac{2\pi t}{p}) + 1.3 & \sin(\frac{2\pi t}{p}) + 0.1 \\ 0 & \sin(\frac{2\pi t}{p}) - 0.6 & \sin(\frac{2\pi t}{p}) + -1.6 \\ 0 & 0 & \sin(\frac{2\pi t}{p}) + c \end{pmatrix}$$

with $p = \frac{1}{4}$ and $d = 1$.

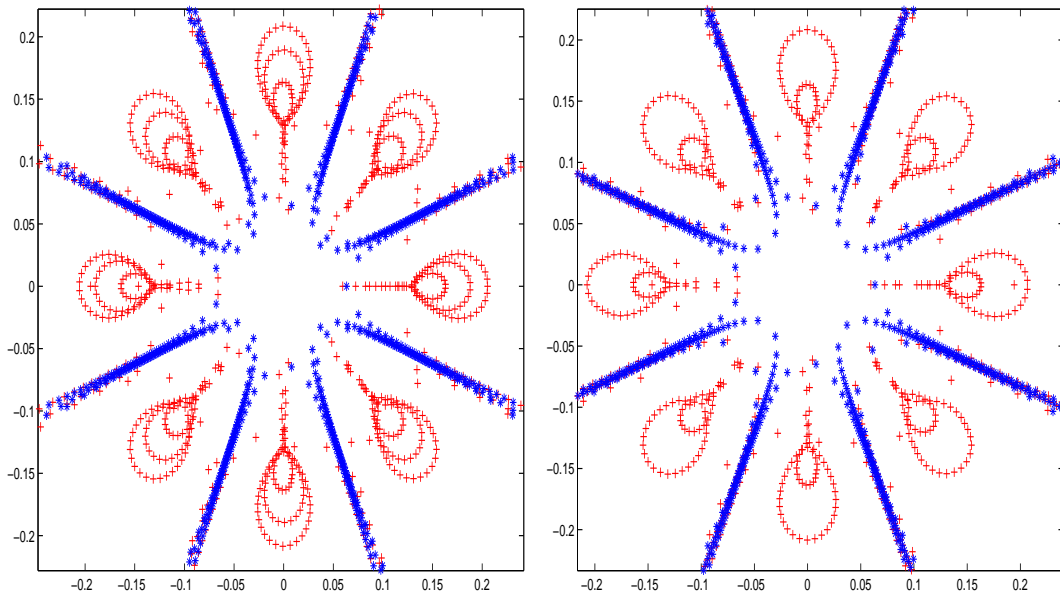


Figure 8.13: An example of the three-dimensional case

In Figure 8.13 we observe a similar phenomenon to that seen in the two-dimensional case in that three sets of multiples of the basic pattern are present. Here $p_2 d_1 = 4$. The product of $\frac{Np}{d}$ matrices has been used and we observe 4 axes of symmetry. In the left-hand diagram, with $c = 0.4$, $a_{11}(t)$, $a_{22}(t)$ and $a_{33}(t)$ all change sign and we can see three sets of additional trajectories indicating the

presence of small solutions. In the right-hand diagram, with $c = 1.7$, only $a_{11}(t)$ and $a_{22}(t)$ change sign, leading to only two sets additional trajectories. We note that the correct number of additional trajectories are not always as clearly visible as in these diagrams.

8.6.3 Conclusion

We have described an effective approach to detecting small solutions to equations of the form

$$(8.33) \quad x'(t) = b(t)x(t-d), b(t+p) = b(t) \text{ where } b \text{ and } p \text{ are commensurate.}$$

Remark 8.6.1 In section 10.7.1 we explain how we can adapt our method and produce eigenspectra with only one axis of symmetry (the real axis) for single delay equations with delay and period commensurate.

By restricting ourselves to a particular class of equation, we have demonstrated that our approach may be extendable to higher dimensional delay differential equations.

Remark 8.6.2 If $B(t)$ is not upper (or lower) triangular then, based on our investigations presented in chapter 6 (see also [29]), we conjecture that a change in sign of $\det(B(t))$ on $[0, p_1 d_1]$ is a sufficient condition for the equation to admit small solutions. Further work is needed in this area.

Chapter 9

Can statistics help?

Chapters 9 and 10 focus on the development of a computer program that will automate the detection of small solutions to a particular class of DDE.

In this chapter we provide evidence of a statistical analysis, motivated partly by the eigenspectra in Figure 4.6. Eigenspectra arising from a numerical discretisation can be used to provide information about the exact eigenspectra [27]. As the step length decreases the ‘distance’ between the trajectories arising from the autonomous equation and the non-autonomous equation decreases when the equation does not admit small solutions. Given that we can calculate actual numerical values of the eigenvalues we are interested to see whether features of equations that admit small solutions are identifiable through the application of statistical techniques and calculations. In section 9.2 we perform a statistical analysis using the cartesian form of the eigenvalues of C , as defined in section 4.4. We explore several possible approaches and then justify our belief that the detection of small solutions using the cartesian form of the eigenvalues arising from our approach is unlikely to be successful, particularly near ‘critical’ values. We follow this in section 9.3 by an analysis involving the eigenvalues of C in polar form.

The basic delay equation

$$(9.1) \quad \dot{x}(t) = b(t)x(t-1) \text{ with } b(t+1) = b(t),$$

(a non-autonomous problem).

$$(9.2) \quad \dot{x}(t) = \hat{b}x(t-1) \text{ where } \hat{b} = \int_0^1 b(t)dt,$$

(an autonomous problem).

We first restrict ourselves to equations of the form (9.1), which we considered in chapters 4 and 5 (see also [28]). We recall that if $b(t)$ does not change sign

then equation (9.1) does not admit small solutions [41, 69]. In this case (9.1) and (9.2) are equivalent.

9.1 Which statistics? A reasoned choice.

We established in section 4.4 that applying a numerical method with step-length $h = \frac{1}{N}$ to (9.1) yields an equation for y_{n+1} of the form $y_{n+1} = A(n)y_n$, where $A(n)$, with $A(n) = A(n - N)$ for all $n > N$, is a companion matrix, dependent upon the numerical method applied (see [28]). From this it followed that $y_{n+N} = Cy_n$, for $n = 1, 2, \dots$ where $C = \prod_{i=1}^N A(N - i)$. In the autonomous problem (9.2) $A(n) = A$ is a constant matrix. This leads to a comparison of the eigenvalues of C with those of A^N .

We introduce

$\Lambda_1 = \{\text{eigenvalues of } C\} = \{z_{1,j}, j=1, 2, \dots, N+1 : z_{1,j} \text{ is an eigenvalue of } C \text{ with } |z_{1,j}| \geq |z_{1,j+1}| \text{ and if } |z_{1,j}| = |z_{1,j+1}| \text{ then } \arg(z_{1,j}) < \arg(z_{1,j+1})\}$.

$\Lambda_2 = \{\text{eigenvalues of } A^N\} = \{z_{2,j}, j=1, 2, \dots, N+1 : z_{2,j} \text{ is an eigenvalue of } A^N \text{ with } |z_{2,j}| \geq |z_{2,j+1}| \text{ and if } |z_{2,j}| = |z_{2,j+1}| \text{ then } \arg(z_{2,j}) < \arg(z_{2,j+1})\}$.

We examine whether the two sets of eigenvalues, Λ_1 and Λ_2 , arise from equivalent problems. When the two problems are equivalent, that is equation (9.1) does not admit small solutions, then the eigenspectra lie close to each other. Each eigenvalue arising from discretisation of (9.1) will approximate an eigenvalue arising from discretisation of (9.2). The approximation should improve as we increase the dimensionality of the problem, that is, as the step size decreases.

We let $z_{1,j} = x_{1,j} + iy_{1,j}$, $z_{2,j} = x_{2,j} + iy_{2,j}$.

We define a one-one mapping between these two ordered sets of eigenvalues (after choosing the ordering as above) and for $j = 1, \dots, N + 1$ we evaluate the distance d_j where $d_j = \sqrt{(x_{2,j} - x_{1,j})^2 + (y_{2,j} - y_{1,j})^2}$. In the absence of small solutions a decrease in step length leads to a better approximation and the values of d_j will tend to zero. The improvement in the approximation (as the step size decreases) should be reflected in measures of location and dispersion of the distribution of the d_j . However, when small solutions are present the ordering will *match up the wrong pairs* and in this case $d_j \not\rightarrow 0$ for some j . This is illustrated in example 9.2.1.

We now apply some basic statistical techniques in our analysis of the distribution of the d_j , including calculation of the mean, the standard deviation, skewness and kurtosis. These provide useful descriptive information about the shape of a distribution. Skewness reflects the degree to which a distribution is asymmetrical. Kurtosis reflects the degree to which a distribution is ‘peaked’,

providing information regarding the *height* of a distribution relative to the value of its standard deviation. We explore whether differences (in the shape of the distributions of the d_j) arising as the result of the problem admitting or not admitting small solutions are identifiable through our statistical analysis. We ask the question ‘Is it possible to impose a threshold, possibly dependent upon N , which would lead to the automatic detection of small solutions using this approach?’

9.2 Our initial approach: Using the cartesian form of the eigenvalues

9.2.1 Examples

Example 9.2.1 We consider first the distributions of the distances d_j for equation (9.1) with $b(t) = \sin(2\pi t) + c$ for different values of c . In this case small solutions are known to arise if $b(t)$ changes sign on $[0, 1]$, that is, if $|c| < 1$ (see [41]). In Figure 9.1 the box plots illustrate the cases $c = 0.5$ and $c = 1.5$. In both cases we observe a decrease in the range of values of d_j and in the median value as the step size decreases. The interquartile range is seen to decrease steadily as the step length decreases when small solutions are not admitted, but the situation is less clear when $c = 0.5$ and the equation admits small solutions.

We make the following observations:

1. The existence of outliers is evidenced in the box plots for which $c = 0.5$, illustrating the greater variation in d_j when the problem admits small solutions.
2. Practical considerations of displaying distributions of d_j on the same axes prevent a representation of the results from a wider range of values of N , particularly in the case when small solutions are admitted.

As the step length decreases we expect the mean and standard deviation of the distribution of the d_j to decrease. This is evidenced in Figure 9.2 which shows the ninety-five per cent confidence intervals for the mean value of the distance between *corresponding* eigenvalues in Λ_1 and Λ_2 for $b(t) = \sin(2\pi t) + c$ and $c = 0.5, 1.5$. When the equation admits small solutions both the mean and the standard deviation are much larger than in problems without small solutions. We observe the much wider confidence intervals when small solutions are admitted.

Example 9.2.2 Figures 9.3 and 9.4 illustrate differences in the distributions of the d_j for different values of c , dependent upon whether or not $|c| < 1$. Again, a much greater variation in the values of d_j is observed for values of c for which

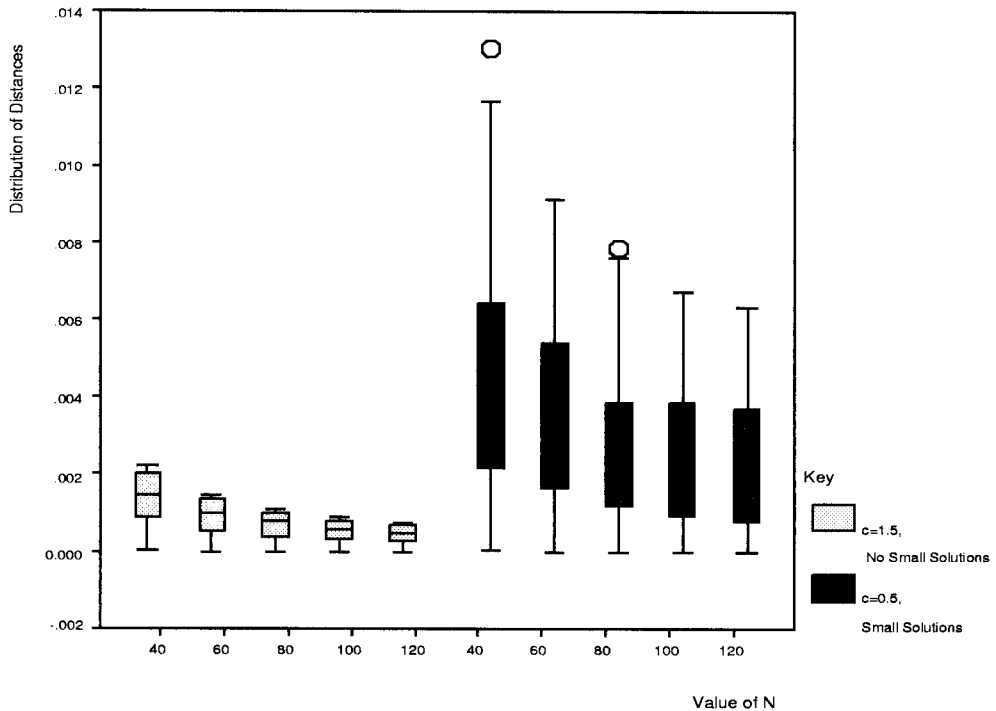


Figure 9.1: A comparison of the distributions of the d_j for varying values of N , with $c = 0.5$ and $c = 1.5$

the equation admits small solutions. When $c = 0$ almost all solutions are small. In Figure 9.3 we observe an increase in both the presence of outliers and in the mean distance as c approaches 0.

Example 9.2.3 In Tables 9.1 and 9.2 we present the values of the kurtosis and skewness of the distribution of the d_j for varying values of c . Values of c equal to -1.5, 1.1, 1.5 and 3 correspond to problems which do not admit small solutions and in this case we observe similar values for different values of the step-length. Values of c of -0.5, 0.1 and 0.5 correspond to problems which admit small solutions. The situation here is very different. Considerable variation is observed for different step lengths. Support for a one-one correspondence between the ordered sets of eigenvalues, Λ_1 and Λ_2 is weak. Differences in the distributions, as indicated by the values of the skewness and kurtosis, are clearly visualised in the histograms in Figure 9.5, again using the distributions of the d_j for $c = 1.5$ and $c = 0.5$. The histograms in the left-hand diagrams are clearly negatively skewed and similar to each other in *shape*. The histograms on the right are positively skewed and show greater variation in *shape*.

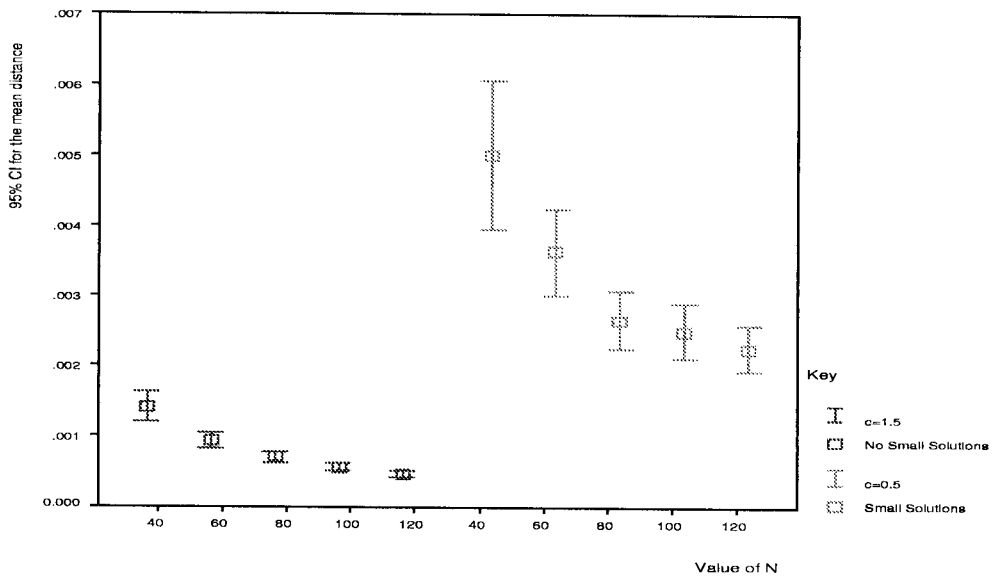


Figure 9.2: The 95% confidence interval for the mean distance between corresponding eigenvalues for varying values of N , with $c = 0.5$ and $c = 1.5$.

N	Value of the constant c						
	-1.5	-0.5	0.1	0.5	1.1	1.5	3
20	-0.9805	-1.3494	-0.8987	-1.0654	-0.7645	-1.0068	-1.1627
40	-1.1103	-0.8628	-1.1396	-0.3964	-1.0624	-1.1287	-1.1593
60	-1.1212	-0.1906	-1.2311	-0.7312	-1.0962	-1.1307	-1.1397
80	-1.1170	-0.4978	-0.5819	0.1171	-1.0981	-1.1227	-1.1256
100	-1.1110	0.1868	-0.5294	-0.8360	-1.0940	-1.1145	-1.1157
120	-1.1053	-0.8658	0.9663	-0.8150	-1.0890	-1.1080	-1.1080

Table 9.1: Values of the kurtosis of the distribution of d_j for different values of c and N

Remark 9.2.1 When N is even we have chosen to disregard the one real eigenvalue of each of the matrices C and A^N since it is an outlier for the distribution and has the potential to affect any conclusions which we may be able to draw.

Limitations of using the distribution of the d_j as the basis for decision-making about the existence, or otherwise, of small solutions to an equation are demonstrated in Figures 9.3 and 9.4. Close to the critical values of c , ($c = \pm 1$) the distributions of the d_j on either side of the boundary values are very similar. Reliable conclusions based on either a threshold for the mean or one for the standard deviation of the d_j seem unlikely.

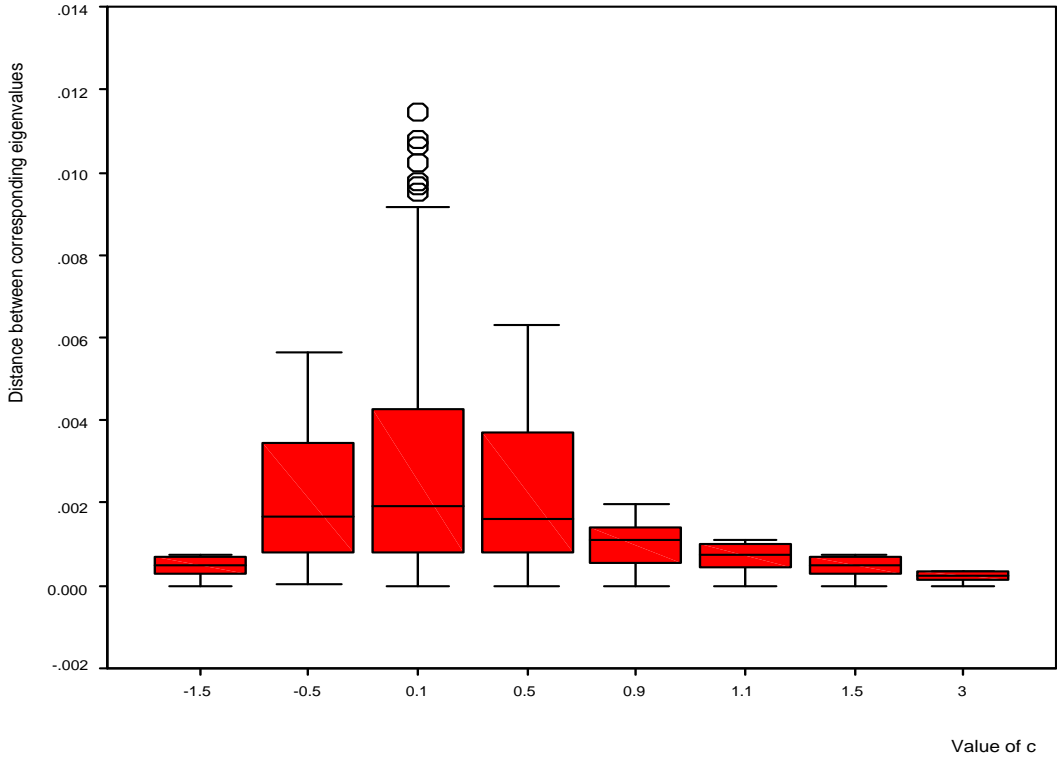


Figure 9.3: Distribution of d_j for different values of c

	Value of the constant c						
N	-1.5	-0.5	0.1	0.5	1.1	1.5	3
20	-0.4977	-0.1484	-0.4673	0.1665	-0.5702	-0.4754	-0.4248
40	-0.4647	0.4142	-0.3549	0.5712	-0.4866	-0.4565	-0.4487
60	-0.4682	0.6600	0.1157	0.4281	-0.4840	-0.4642	-0.4626
80	-0.4736	0.7745	0.6337	0.7630	-0.4883	-0.4713	-0.4710
100	-0.4780	0.8656	0.8861	0.6758	-0.4926	-0.4766	-0.4764
120	-0.4814	0.6019	1.3091	0.6755	-0.4963	-0.4804	-0.4801

Table 9.2: Values of the skewness of the distribution of d_j for different values of c and N

We conclude this section by considering Spearman's rank correlation coefficient, r_s , the value of which determines the degree to which a monotonic relationship (increasing or decreasing) exists between two variables (see [67] for example). A visual comparison of the eigenspectra suggested that, for the equations that we are considering, the relationship between $|z_{1,j}|$ and $|y_{1,j}|$ would be

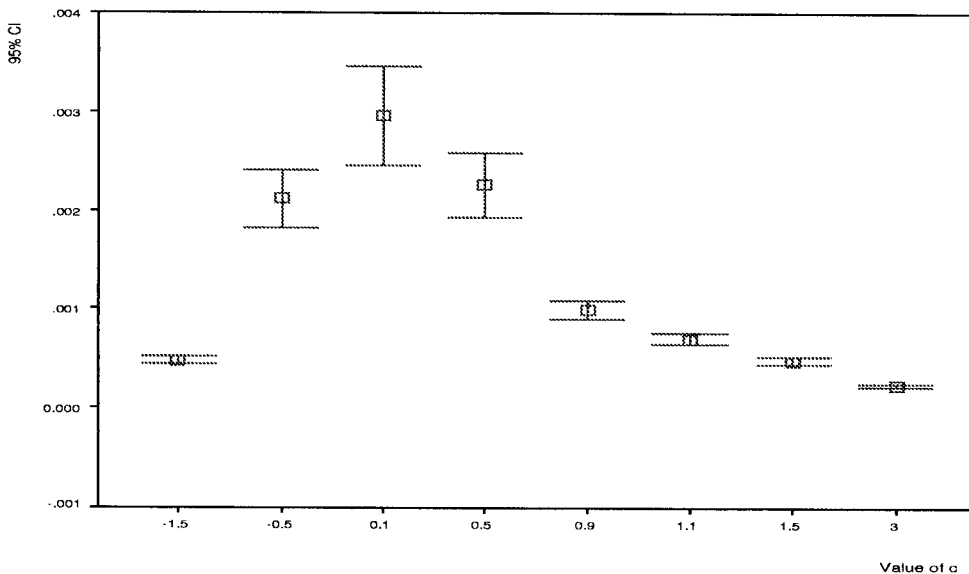


Figure 9.4: 95% Confidence intervals for the mean distance between corresponding eigenvalues for different values of c

expected to be monotonic when the equation does not admit small solutions, but not otherwise. We explore whether the automatic calculation of an appropriate Spearman's rank correlation coefficient can reliably answer the question 'Does an equation admit small solutions?'

Remark 9.2.2 (Remark 2.2 from [31]) This approach, if it was successful, would be extremely attractive. Notice that the calculation of the statistic involves calculation only in terms of features of the eigenspectrum of the original delay differential equation under discretisation, and no longer relies on the computation of eigenspectra for an equivalent autonomous problem. For more general equations, we might be unable to write down an equivalent autonomous problem (for many equations, the formula is unknown analytically) yet this type of method would remain applicable.

Remark 9.2.3 Chapter XI in [41] states that for the characteristic equation $z - \alpha - \beta e^{-z} = 0$ (for the delay equation $\dot{x}(t) = \alpha x(t) + \beta x(t - 1)$) the ordering of the imaginary parts of the roots coincides with the ordering of the real parts. This would support our use of Spearman's rank correlation coefficient. However, the authors state that 'this is a special feature which does not carry over to a wide class of problems'. Hence, if successful the usage of it is likely to be limited.

We illustrate its potential usage in the following example.

Example 9.2.4 In Table 9.3 we present values of Spearman’s rank correlation coefficient between the magnitude of the eigenvalue and the magnitude of its imaginary part for the non-autonomous equation (9.1), with $b(t) = t - 0.5 + c$, $b(t + 1) = b(t)$, and for the autonomous equation (9.2) with $\hat{b} = c$. For this example small solutions are admitted if $|c| < 0.5$. We observe that the relationship is monotonic when small solutions are not admitted. A similar pattern emerged for other $b(t)$, including $b(t) = \sin 2\pi t + c$, $b(t) = t(t - 0.5)(t - 1) + c$ and $b(t) = \sin 2\pi t + t(t - 0.5)(t - 1)$.

	(non- autonomous)	(autonomous)		(non- autonomous)	(autonomous)
c	r_s	r_s	c	r_s	r_s
-1	1	1	0.1	0.871913	1
-0.9	1	1	0.2	0.893179	1
-0.8	1	1	0.3	0.935066	1
-0.7	1	1	0.4	0.967120	1
-0.6	1	1	0.5	1	1
-0.5	1	1	0.6	1	1
-0.4	0.954099	1	0.7	1	1
-0.3	0.851343	0.961307	0.8	1	1
-0.2	0.845831	0.963283	0.9	1	1
-0.1	0.836725	0.962309	1.0	1	1
0	0.829479	1			

Table 9.3: Values of Spearman’s rank-order correlation coefficient between the magnitudes of the eigenvalues and their imaginary part using the eigenvalues of (9.1) with $b(t) = t - 0.5 + c$ and c varying.

When c is not close to a critical value, where small solutions begin to arise, the calculations do provide some indication of the presence of small solutions. However, it is clear that close to the boundary r_s is not sufficiently sensitive to enable decisions to be made. Further work is needed using Spearman’s rank correlation coefficient in this context before a statement about its usage can be made with confidence.

In summary, in this section we have reviewed some elementary statistical measures which could be calculated to determine whether or not small solutions arise for a particular problem. We also considered the use of non-parametric statistical tests such as the Wilcoxon Rank Sum test, to test for differences between the medians of two populations, and the Kruskal-Wallis test, to test for differences between three or more populations (see [67]). (Some preliminary results of using these tests with the data used in Figure 9.1 were of interest). However,

although we have gained some useful insight, sensitivity near to critical values is poor and we have yet to establish a process using the cartesian form of the eigenvalues which satisfies the aim of our investigations. In the next section we explore a quite different approach.

9.3 Insight from visualisation: Consideration of the eigenvalues in polar form

Based on our experimental results (see [28, 29, 30, 31]) we believe that results arising from the use of the polar form of the eigenvalues might be more easily extendable to other class of equation, in particular to equations of the form $\dot{x}(t) = \sum_{j=0}^m b_j(t)x(t - jw)$ and to those higher dimensional systems when the eigenvalues of $A(t)$ in equation $y'(t) = A(t)y(t - 1)$ are always real. Hence, although further insight was achieved using the cartesian form of the eigenvalues of the two matrices concerned, the potential for wider application of our results encouraged us to turn our attention to the consideration of the eigenvalues in polar form.

When the analytical theory informs us that small solutions exist then we observe consistently some of the eigenvalues arising from discretisation of the non-autonomous problem lying close to the real axis and others lying on the negative real axis [34]. In our work, (see [28, 29]), we used ‘the presence of closed loops that cross the x-axis to be characteristic of the cases where small solutions arise’. We observe that the sizes of the arguments of the eigenvalues whose representation forms the ‘additional’ trajectory lie closer to 0 or 2π than those represented in the trajectory lying close to that arising from the autonomous problem. We use this idea as a basis for developing our method.

We use $z_{1,j}$ and $z_{2,j}$ as defined in section 9.1 and introduce

$$M_1 = \{\alpha_{1,j} : \alpha_{1,j} = \arg(z_{1,j}), j = 1, 2, \dots, N + 1\}.$$

$$M_2 = \{\alpha_{2,j} : \alpha_{2,j} = \arg(z_{2,j}), j = 1, 2, \dots, N + 1\}.$$

$$L_1 = \{\alpha : 0 \leq \alpha < 0.5, \alpha = |\alpha_{1,j}|, \alpha_{1,j} \in M_1\}.$$

$$L_2 = \{\alpha : 3 < \alpha \leq \pi, \alpha = |\alpha_{1,j}|, \alpha_{1,j} \in M_1\}.$$

We focus our interest on the distribution of $\alpha = \{|\alpha_{1,j}| : \alpha_{1,j} \in M_1\}$ for α lying in the intervals $[0, 0.5]$, $(0.5, 1.0]$, $(1.0, 1.5]$, $(1.5, 2.5]$, $(2.5, 3.0]$, $(3.0, \pi]$.

Decreasing the step length from $\frac{1}{N_1}$ to $\frac{1}{N_2}$ increases the dimensions of the matrices C and A^N and leads to the calculation of a further $(N_2 - N_1)$ eigenvalues. We consider the question ‘Where do the larger set of eigenvalues lie in relation to the previous set of eigenvalues?’. We investigated a range of step-lengths, observing where the additional eigenvalues fitted into the distribution and whether this depended upon the presence, or otherwise, of small solutions.

9.3.1 Numerical results

In the case when (9.1), with $b(t) = \sin 2\pi t + c$, does not admit small solutions then, for $h \geq \frac{1}{300}$, all the additional eigenvalues have arguments whose magnitudes lie in the range 0.5 to 2.5. This is not the case when (9.1) admits small solutions and we illustrate this difference in Tables 9.4 and 9.5. We note also that in Table 9.4 where the problem does not admit small solutions we observe no values of $\alpha > 2.5$, but in Table 9.5, when small solutions are admitted, we observe values of $\alpha > 2.5$ for all values of N .

N	Range of values for α					
	[0, 0.5)	[0.5, 1.0)	[1.0, 1.5)	[1.5, 2.5)	[2.5, 3.0)	[3, π]
30	1	2	26	2	0	0
60	1	2	56	2	0	0
90	1	4	52	34	0	0
120	1	4	48	68	0	0
150	1	4	48	98	0	0
300	1	4	48	248	0	0
500	3	2	50	446	0	0
1000	3	4	54	940	0	0

Table 9.4: Distribution of the magnitudes of the arguments of the eigenvalues for $c = -1.4$. No small solutions are admitted.

N	Range of values for α					
	[0, 0.5)	[0.5, 1.0)	[1.0, 1.5)	[1.5, 2.5)	[2.5, 3.0)	[3, π]
30	15	0	0	0	2	14
60	30	0	0	2	18	11
90	25	18	0	18	20	10
120	20	38	0	40	14	9
150	19	40	12	54	18	8
300	18	26	98	136	12	11
500	16	24	196	240	16	9
1000	18	20	432	498	24	9

Table 9.5: Distribution of the magnitudes of the arguments of the eigenvalues for $c=0.1$. Small solutions are admitted.

We now consider equation (9.1) with $b(t) = \sin 2\pi t + c$ for a range of values of c . In this case the *critical functions* are when $c = \pm 1$. In Table 9.6 we present the number of eigenvalues of C for which the magnitude of the argument lies in each

specified range and, in brackets, the corresponding figure for A^N . The divisions in the table effectively discriminate between the middle section where $|c| < 1$ and the non-autonomous equation admits small solutions and the other cases where small solutions are not present. It is clear that for equations of the form (9.1) which admit small solutions then the two sets of figures are very dissimilar. We observe that (using $h = \frac{1}{128}$):

1. $n(L_2) = 0$ and $n(L_1) = 1$ except near the critical functions when $c = \pm 1$.
2. Near the critical functions when $c = \pm 1$ at least one of the statements $n(L_2) = 0, n(L_1) = 1$ is true.

The results from our experiments lead us to present the following tool as the basis on which our program decides whether or not an equation admits small solutions.

Decision tool 9.3.1 Let M_1 be the set of eigenvalues arising from discretisation of $x'(t) = b(t)x(t-1), b(t+1) = b(t)$ using the trapezium rule (as in chapter 4) and define

$$L_1 = \{\alpha : \alpha \in M_1, 0 \leq |\alpha| < 0.5\},$$

$$L_2 = \{\alpha : \alpha \in M_1, 3 < |\alpha| \leq \pi\}.$$

When the equation $x'(t) = b(t)x(t-1), b(t+1) = b(t)$ does not admit small solutions then at least one of the following statements is true.

- (i) $L_2 = \phi$ (or $n(L_2) = 0$).
- (ii) $n(L_1) = 1$.

We note that we have also considered the distribution of the magnitudes of the arguments of the eigenvalues after discretisation using the Backward Euler and Forward Euler methods. The shape of the distributions differed from that obtained using the trapezium rule, but distinguishing between problems which admitted small solutions and those for which an equivalent autonomous problem exists can be achieved using a similar and equally effective approach to that described here.

N	Range of values for α					
	[0, 0.5)	[0.5, 1.0)	[1.0, 1.5)	[1.5, 2.5)	[2.5, 3.0)	[3, π]
-1.5	1 (1)	4 (4)	46 (40)	78 (84)	0 (0)	0 (0)
-1.4	1 (1)	4 (4)	48 (42)	76 (82)	0 (0)	0 (0)
-1.3	1 (1)	4 (4)	52 (44)	72 (80)	0 (0)	0 (0)
-1.2	1 (1)	4 (4)	60 (44)	64 (80)	0 (0)	0 (0)
-1.1	1 (1)	4 (4)	68 (48)	56 (76)	0 (0)	0 (0)
-1.0	4 (1)	6 (4)	74 (48)	45 (76)	0 (0)	0 (0)
-0.9	16 (1)	4 (4)	60 (48)	30 (76)	0 (0)	19 (0)
-0.8	24 (1)	4 (4)	62 (50)	12 (74)	14 (0)	13 (0)
-0.7	30 (1)	4 (4)	62 (52)	0 (72)	22 (0)	11 (0)
-0.6	28 (1)	14 (6)	50 (52)	0 (70)	28 (0)	9 (0)
-0.5	26 (1)	20 (6)	40 (54)	12 (68)	22 (0)	9 (0)
-0.4	26 (3)	26 (4)	30 (56)	18 (66)	20 (0)	9 (0)
-0.3	24 (3)	32 (4)	22 (60)	26 (62)	16 (0)	9 (0)
-0.2	20 (3)	38 (4)	16 (68)	28 (54)	20 (0)	7 (0)
-0.1	20 (5)	44 (2)	6 (78)	34 (44)	18 (0)	7 (0)
0	18 (1)	46 (0)	0 (0)	40 (128)	20 (0)	5 (0)
0.1	18 (1)	42 (0)	0 (0)	42 (126)	18 (2)	9 (0)
0.2	18 (1)	38 (0)	0 (0)	44 (126)	18 (2)	11 (0)
0.3	20 (1)	32 (0)	0 (0)	50 (126)	16 (2)	11 (0)
0.4	20 (1)	28 (0)	0 (0)	48 (126)	22 (2)	11 (0)
0.5	22 (1)	16 (0)	0 (0)	52 (126)	26 (2)	13 (0)
0.6	22 (1)	16 (0)	0 (0)	52 (126)	26 (2)	13 (0)
0.7	30 (1)	4 (0)	0 (0)	64 (126)	20 (2)	11 (0)
0.8	28 (1)	0 (0)	0 (0)	76 (126)	14 (2)	11 (0)
0.9	20 (1)	0 (0)	0 (0)	92 (126)	4 (2)	13 (0)
1.0	1(1)	0 (0)	0 (0)	123 (126)	2 (2)	3 (0)
1.1	1(1)	0 (0)	0 (0)	126 (126)	2 (2)	0 (0)
1.2	1(1)	0 (0)	0 (0)	126 (126)	2 (2)	0 (0)
1.3	1(1)	0 (0)	0 (0)	126 (126)	2 (2)	0 (0)
1.4	1(1)	0 (0)	0 (0)	126 (126)	2 (2)	0 (0)
1.5	1(1)	0 (0)	0 (0)	126 (126)	2 (2)	0 (0)

Table 9.6: The distribution of the magnitudes of the arguments of the eigenvalues, α , arising from discretisation of (9.1) and (9.2) with $b(t) = \sin 2\pi t + c$ for different values of c

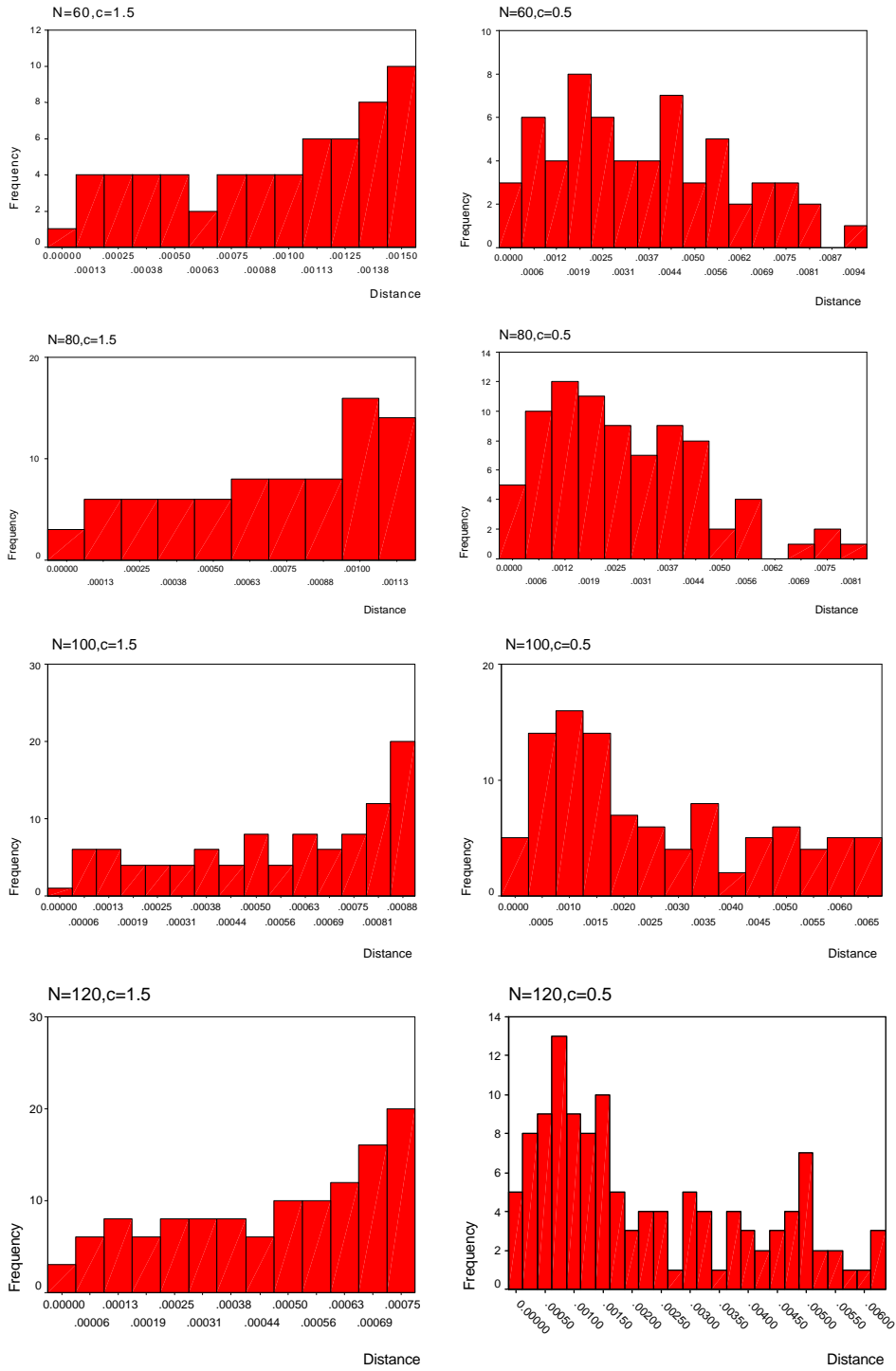


Figure 9.5: Histograms for the distributions of the d_j for:
 Left: $c = 1.5$. No small solutions are admitted.
 Right: $c = 0.5$. The equation admits small solutions.

Chapter 10

Automating the process

In chapters 4 to 8 we have established that numerical methods can be used effectively to detect small solutions. However, the idea of a reliable algorithm that can be used without an understanding of the methodology underlying the decision-making process is attractive. Ideally we would like to develop a ‘black box’ approach. The aim of the (heuristic) algorithm presented in this chapter, and developed using Matlab, is to automate the detection of small solutions to a particular class of DDE. It is desirable that we might be able to adapt the algorithm to handle other classes of DDE and we demonstrate some progress in this direction in section 10.7. The Matlab code for the algorithm is found in appendix A.

10.1 Introducing ‘smallsolutiondetector1’

‘Smallsolutiondetector1’ is a Matlab program written to answer the question ‘Does an equation of the form

$$(10.1) \quad x'(t) = b(t)x(t - p), b(t + p) = b(t)$$

admit small solutions?’ The program allows the user to detect small solutions to equations of the form (10.1) but transforms this equation to an equation of the form

$$(10.2) \quad y'(t) = b_1(t)y(t - 1), b_1(t + 1) = b_1(t)$$

using the transformation $b_1(t) = pb(pt)$. This transformation is internal to the program and transparent to the user.

10.2 The Rationale behind the algorithm

Our aim is to produce a program where the user does not need to understand the methodology underlying the process by which the decision is made.

For some non-autonomous problems of the form (10.1) there exists an equivalent autonomous problem (in the sense that the solution is the same whenever the initial vector is the same [34]), the existence or otherwise of which is an important question to a mathematical modeller. Our previous work in chapters 4 to 8 (see also [28, 29]) involved a visual representation of the eigenspectra arising from numerical discretisations of a non-autonomous problem and the potentially equivalent autonomous problem. We identified characteristics of the eigenspectra which correctly indicated the presence, or otherwise, of small solutions, and hence determined whether or not an equivalent autonomous problem existed. The insight gained from this visualisation motivated a statistical analysis of the two sets of eigenvalues, as detailed in chapter 9, and the subsequent development of the algorithm presented in this chapter.

When we consider eigenspectra our interest in small solutions focusses our attention on the eigenvalues near the origin. Envisioning the data generated by the Matlab program effectively, so that the part of the eigenspectra close to the origin is displayed clearly whilst maintaining a clear overview of the characteristic shape of the whole eigenspectra, can be challenging. Although our diagrams are conclusive, except when we are close to a *critical function* (where the nature of the problem changes from not admitting small solutions to admitting small solutions and vice-versa), we would like to automate the process, hence removing the need to be able to interpret our eigenspectra. In addition, the difficulty in correctly detecting the presence of small solutions, (by consideration of the two eigenspectra), near a critical function may lead to an incorrect or unreliable decision. We experimented with smaller step lengths but found no improvement in their detection. We are interested to see whether using our algorithm can reduce the likelihood of this event.

10.2.1 The underlying methodology

The methodology underlying the algorithm is based on decision tool 9.3.1. We use the term *critical function* to refer to a function at the bifurcation point when the behaviour of the equation changes from admitting small solutions to not admitting small solutions and vice-versa. The program consists of the following stages:

1. The user is asked to state the period/delay and to input their function $b(t)$.
2. The eigenvalues of the matrix C , with C as defined in section 4.4, are calculated.
3. The numbers of these eigenvalues with magnitudes lying in the intervals $[0, 0.5]$ and $(3, \pi]$ are calculated. The algorithm refers to these numbers as

$n1$ and $n6$ respectively.

4. If $n6 = 0$ we conclude that the equation does not admit small solutions.
5. If $n6 > 0$ we also consider the value of $n1$.
 - (a) If $n6 > 0$ and $n1 = 1$ we conclude that the equation does not admit small solutions but the user is warned that their function is near to a critical function.
 - (b) If $n6 > 0$ and $n1 > 1$ we conclude that the equation admits small solutions.
 - (c) We note that, to date, we have not experienced the situation when $n6 > 0$ and $n1 = 0$. If this case does arise then the user is informed that a decision cannot be made using the algorithm.

In view of the potential for an incorrect decision we developed a modification of our algorithm to provide the user with a check on the decision made, but at a cost in terms of the additional time in reaching a decision. It was felt that this would be particularly useful when the function is near to a critical function, providing the user with an indication of the unreliability of the decision when appropriate. The modified algorithm repeats the decision-making process outlined above, but this time with $b(t)$ and each of the two neighbouring functions $b(t) \pm \epsilon$. For each of the three functions the program decides whether the equation admits small solutions. Three decisions are possible for each of the three functions. We will refer to these decisions as:

Yes: The equation admits small solutions.

No: The equation does not admit small solutions.

No/Near: It is unlikely that the equation admits small solutions but you are near to a critical function.

The algorithm considers all 27 possibilities and a decision is made for the function $b(t)$ dependent on the decisions using the nearby functions $b(t) \pm \epsilon$. The user can choose their own value of ϵ , referred to in the program as the tolerance, or use the pre-selected value of ϵ . The decisions made by the algorithm are reflected in Table 10.1.

If the user chooses to run the modified algorithm the program then compares the two answers produced. A re-run of the modified algorithm with a reduced tolerance (pre-selected or of the user's own choice) is advised when appropriate. The user can elect whether or not to accept the advice.

$b(t) - \epsilon$	$b(t)$	$b(t) + \epsilon$	Decision: Does the equation admit small solutions?	Re-run algorithm with a reduced tolerance?
Yes	Yes	Yes	Yes	
No/Near	Yes	Yes	Very Likely	
No	Yes	Yes	Likely	
Yes	Yes	No/Near	Very Likely	
Yes	Yes	No	Very Likely	
No/Near	Yes	No/Near	Likely	
No	Yes	No/Near	Likely	Yes
No/Near	Yes	No	Likely	Yes
No	Yes	No	Likely	Yes
Yes	No/Near	No	Unlikely	Possibly
No/Near	No/Near	No	Very Unlikely	
No/Near	No/Near	No/Near	Unlikely	
No/Near	No/Near	Yes	Unlikely	
Yes	No/Near	Yes	Very Unlikely	Yes
Yes	No/Near	No/Near	Very Unlikely	
No	No/Near	Yes	Very Unlikely	
No	No/Near	No/Near	Very Unlikely	
No	No/Near	No	Unlikely	Yes
No	No	No	No	No
No/Near	No	No	Very Unlikely	Yes
No	No	No/Near	Unlikely	Yes
Yes	No	No	Very Unlikely	Yes
No	No	Yes	Unlikely	Yes
Yes	No	No/Near	Unlikely	Yes
No/Near	No	No/Near	Very Unlikely	No
No/Near	No	Yes	Unlikely	Yes
Yes	No	Yes	Unlikely	Yes

Table 10.1: Decisions made using the modified algorithm

10.3 A theoretical basis for the algorithm

(Section 3.1 in [31]) We provide a simple mathematical justification for our approach. It is straightforward to show that only one characteristic value (the real root itself) of the autonomous problem lies close to the real axis (see, for example, [22] p. 305-316). We adopt the approach in [27] to show that for the numerical scheme, as $h \rightarrow 0$, there will be only a single characteristic root close to the real axis. Therefore, for an equation without small solutions, all but one of the characteristic roots should lie away from the real axis. Hence, when we

detect more than one characteristic root in a neighbourhood of the real axis, this is sufficient to indicate the presence of small solutions.

10.4 Consideration of the reliability of the algorithm

We have considered the reliability of our algorithm with particular reference to the decisions made near a critical function. An algorithm was written, using Matlab, to determine the value of c at which the decision made by the algorithm changed from ‘Yes’ to ‘No’ with regard to the existence of small solutions. The Matlab code can be found in appendix B. In Table 10.2 we show, for three different $b(t)$, the value of c at which the algorithm’s decision changes and the absolute difference between that value and the *theoretically correct value* to eight decimal places.

We make the following observations for the step lengths that we have considered:

1. For $b(t) = \sin(2\pi t) + c$ the error is zero to 8 decimal places.
2. For $b(t) = t - 0.5 + c$ the reduction in the error as the step length decreases is of order h .
3. For $b(t) = t(t - 0.5)(t - 1) + c$ the error is at most of the order of 10^{-5} .

	$b(t)=\sin(2\pi t)+c$		$b(t)=t-0.5+c$		$b(t)=t(t-0.5)(t-1)+c$	
CV	$c = 1$		$c = \frac{1}{2}$		$c = \frac{\sqrt{3}}{36}$	
N	Actual	Error	Actual	Error	Actual	Error
32	1	0	0.46875000	0.03125000	0.04806519	0.00004733
64	1	0	0.48437500	0.01562500	0.04806519	0.00004733
96	1	0	0.48958333	0.01041667	0.04810475	0.00000777
128	1	0	0.49218750	0.00781250	0.04811239	0.00000013
160	1	0	0.49375000	0.00625000	0.04811133	0.00000119

Table 10.2: Values of c at which the decision changes
NB. CV = the value of c which gives the critical function.

Remark 10.4.1 The negative value of c at which the decision changes is correct to 8 decimal places for each $b(t)$.

Remark 10.4.2 The first generation of our algorithm was based purely on the number of eigenvalues with magnitude lying in $(3, \pi]$, a result of 0 implying

that the equation does not admit small solutions and a value > 0 implying that the equation admits small solutions. The magnitudes of the errors was considered in a similar way (see appendix E). Including the number of eigenvalues with magnitudes less than 0.5 in the decision-making process led to a significant increase in the reliability of our algorithm in detecting the presence of small solutions.

10.5 Illustrative examples

Example 10.5.1 Input: period = 1, $b(t) = \frac{t(t-0.5)(t-1)}{1000}$

The algorithm decides that the equation admits small solutions. Running the modified algorithm with the specified tolerance results in the advice to re-run the modified algorithm with a reduced tolerance. Re-running the modified algorithm with the tolerance reduced by a factor of 10 results in confirmation of the first decision. We can see from figure 10.1 that adjusting the function by a constant amount of 0.0001 will result in a function which does not change sign, hence the advice to reduce the tolerance.

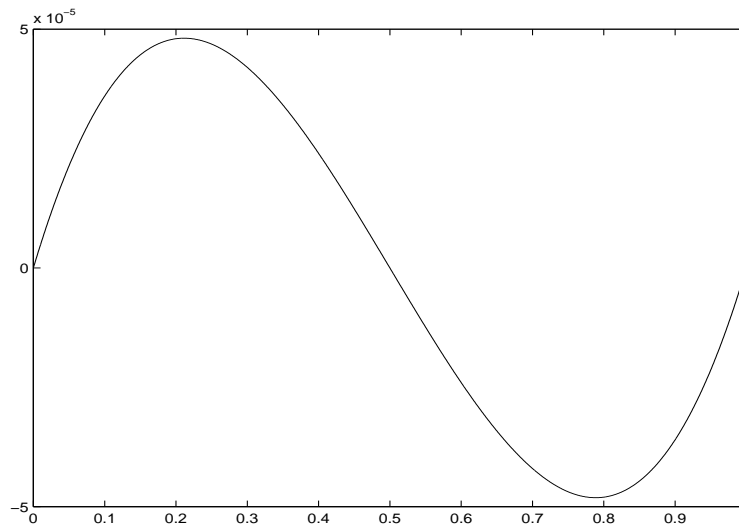


Figure 10.1: Graph of $b(t) = \frac{t(t-0.5)(t-1)}{1000}$ on $[0, 1]$

Example 10.5.2 Input: period = 4, $b(t) = t - 3.5$; decision:- the equation admits small solutions.

Input: period = 3, $b(t) = t - 3.5$; decision:- the equation does not admit small solutions.

In this case $b(t)$ changes sign when $t=3.5$, hence with a period of 3 there is no change of sign.

Example 10.5.3 In examples 10.5.1 and 10.5.2 the decision was easily predictable. If $b(t) = \sin(\pi t) - e^{0.4t} + \log(2.6t + 0.1) - \frac{t}{2+4t}$ the decision is less obvious. The algorithm returns a decision that the equation admits small solutions. This result is confirmed by the graph of $b(t)$ in figure 10.2 (the function changes sign on $[0, 1]$).

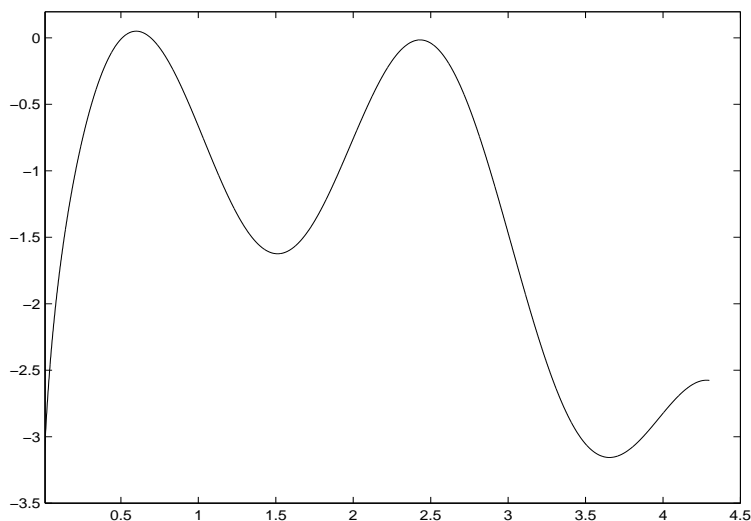


Figure 10.2: Graph of $b(t) = \sin(\pi t) - e^{0.4t} + \log(2.6t + 0.1) - \frac{t}{2+4t}$ on $[0, 4.3]$

10.6 Algorithm: Summary

We have developed and tested an algorithm which automates the decision concerning the existence, or otherwise, of small solutions to the equation $x'(t) = b(t)x(t - p)$, with $b(t + p) = b(t)$. Consideration has been given to its reliability and any reservations about the decision are communicated to the user.

- Remark 10.6.1**
1. We have adapted the algorithm to answer the same question of the multi-delay equation $x'(t) = \sum_{j=0}^m b_0(t)x(t - jw)$.
 2. We anticipate that we will be able to modify the algorithm to answer the same question of the system $y'(t) = A(t)y(t - 1)$ where the eigenvalues of $A(t)$ as t varies are always real. (See also section 13.1)

10.7 Algorithm: Possible future developments

The question that we begin to address in this section is: Can our algorithm be modified or extended to answer the same question for other classes of DDE?

10.7.1 DDEs with delay and period commensurate

In chapter 8 we considered the equation

$$(10.3) \quad \dot{x}(t) = b(t)x(t-d) \text{ with } b(t+p) = b(t)$$

with p and d commensurate. We are reminded that we use $p = \frac{p_1}{p_2}, d = \frac{d_1}{d_2}$ or $p = \frac{f_1 p_u}{f_2 p_\ell}, d = \frac{f_1 d_u}{f_2 d_\ell}$ where f_i is the highest common factor of p_i and d_i for $i = 1, 2$. In anticipation of being able to develop an automated approach to detecting the existence of small solutions to (10.3), we indicate how, using in general more computational time, we can produce diagrams similar to those encountered in previous work and which underpin the development of the algorithm.

Progress towards an automated approach

'Smallsolutiondetector1' automates the answer to the question 'Does the equation $x'(t) = b(t)x(t-d), b(t+d) = b(t)$ admit small solutions?' The algorithm involves calculating the magnitudes of the arguments of the eigenvalues of the matrix C , with C defined as in section 4.4. The real axis is, as expected, an axis of symmetry of the eigenspectrum arising from C . In our diagrams small solutions are indicated by the presence of eigenvalues lying close to the real axis and the decision regarding the existence, or otherwise, of small solutions is based on the number of eigenvalues lying very close to the real axis, that is the number of eigenvalues that have arguments that are very close to 0 or π in magnitude.

We observe that using the approach adopted in chapters 4 and 5 (see also [32]) we obtained diagrams with $p_\ell d_u = \frac{p_2 d_1}{f_1 f_2}$ axes of symmetry in chapter 8 for equation (10.3).

The equation of the axes of symmetry can be written as

$$(10.4) \quad \theta = \begin{cases} \frac{\pi k}{p_\ell d_u} \\ -\left(\pi - \frac{k\pi}{p_\ell d_u}\right) \end{cases} \text{ for } k = 0, 1, \dots, (p_\ell d_u).$$

If a matrix C has an eigenvalue with argument α then the matrix C^N has an eigenvalue with magnitude $N\alpha$ which, in this section, we choose to refer to as the *associated* arguments of α . If we consider an eigenvalue of C with argument α , $\alpha \in [0, \frac{\pi}{p_\ell d_u}]$, then, due to the symmetry of the eigenspectrum, eigenvalues of C exist with arguments given by $(2k \pm 1) \left(\frac{\pi}{p_\ell d_u}\right) \mp \alpha, k = 0, 1, \dots, p_\ell d_u$.

The arguments of the associated eigenvalues of $C^{p_\ell d_u}$ will thus be given by $p_\ell d_u \left\{ \frac{(2k \pm 1)\pi}{p_\ell d_u} \mp \alpha \right\}$.

In consequence, since $p_\ell d_u \left\{ \frac{(2k \pm 1)\pi}{p_\ell d_u} \mp \alpha \right\} = (2k \pm 1)\pi \mp p_\ell d_u \alpha$, the real axis will be the axis of symmetry for eigenvalues of $C^{p_\ell d_u}$.

We include the following example as an illustration.

Example 10.7.1 We consider equation $x'(t) = \left\{ \sin\left(\frac{2\pi t}{p}\right) + c \right\} x(t - d)$ with $p = \frac{1}{6}, d = 1, c = 0.4$. In this case $p\ell d_u = 6$. If an eigenvalue exists with argument α then eigenvalues also exist with arguments given by $(2k \pm 1)\frac{\pi}{6} \mp \alpha$ for $k = 1, 2, \dots, 6$. The eigenspectra will have six axes of symmetry. Arguments of *associated* eigenvalues of C^2 are $2(2k \pm 1)\frac{\pi}{6} \mp 2\alpha$ and hence the eigenspectrum will have three axes of symmetry. In a similar way we can show that we expect eigenspectra arising from C^3, C^4, C^5 to display 2, 3 and 6 axes of symmetry. Arguments of *associated* eigenvalues of C^6 are $6(2k \pm 1)\frac{\pi}{6} \mp 6\alpha = (2k \pm 1)\pi \mp \alpha$ and hence the eigenspectrum will have just one axis of symmetry, the real axis. We illustrate this in Figures 10.3 and 10.4.

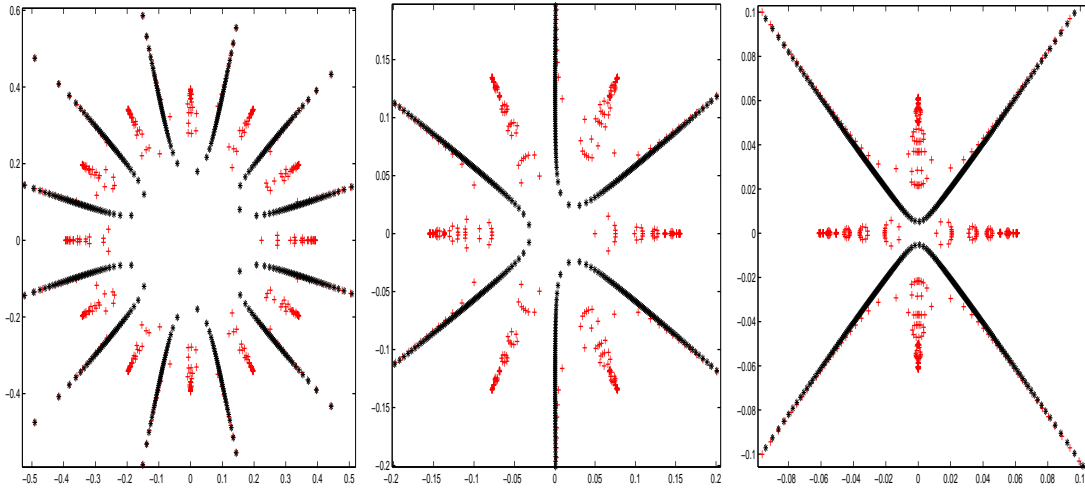


Figure 10.3: Example 10.7.1. Left: $k=1$; 6 axes of symmetry
Centre: $k=2$; 3 axes of symmetry. Right: $k=3$; 2 axes of symmetry

Since the real axis is the axis of symmetry in the right-hand diagram of Figure 10.4 and small solutions are clearly indicated by an additional trajectory lying close to the real axis we anticipate that it will be possible to modify our algorithm to detect whether or not equations of the form (10.3) admit small solutions. We now present further examples to illustrate this approach.

Examples of producing eigenspectra with just one axis of symmetry

We choose to use examples which were presented in chapter 8 using the product of only $\frac{Np}{d}$ matrices. In Figures 10.5 and 10.6 we observe that the real axis is indeed the only axis of symmetry. In the right-hand diagram in Figure 10.5 we emphasise the very small scale used on the imaginary axis. In this example nearly all solutions are small solutions and we note that all displayed eigenvalues

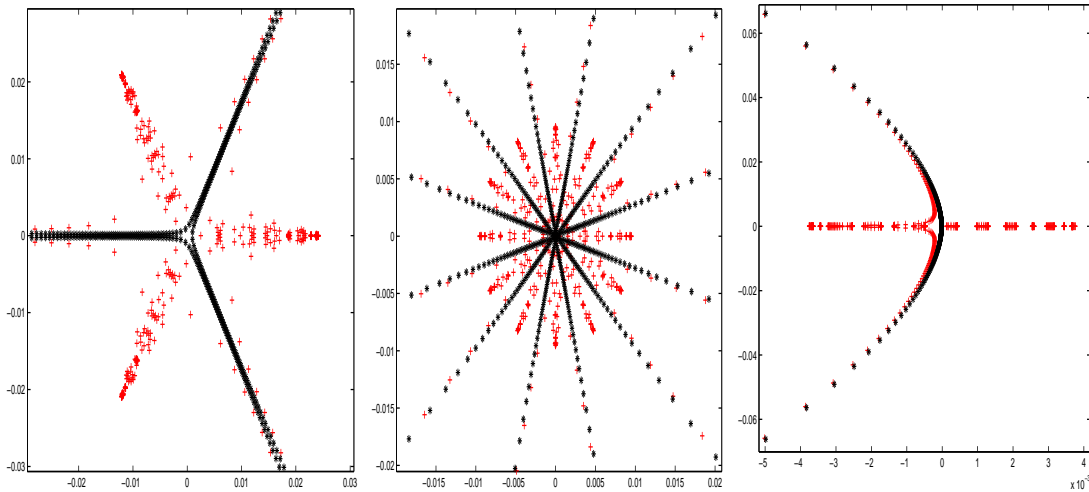


Figure 10.4: Example 10.7.1. Left: $k=4$; 3 axes of symmetry
 Centre: $k=5$; 6 axes of symmetry Right: $k=6$; 1 axis of symmetry

will have arguments whose magnitude is very close to 0 or π . The eigenspectra in the left-hand diagram of Figure 10.5 arise from an equation which does not admit small solutions and we note the lack of an additional trajectory lying close to the real axis.

In Figure 10.6, when the eigenspectra arise from equations which admit small solutions we note the presence of an additional trajectory composed of eigenvalues lying close to the real axis. The characteristic shapes of these diagrams are very similar to those encountered in our earlier work. Our experimental work supports our view that our algorithm can be modified to detect whether or not equations of the form (10.3) admit small solutions.

Additional computational time is likely to be needed since we are using $y_{n+p_2d_1(mp_1d_2)} = C^{p_2d_1}y_n$ instead of $y_{n+mp_1d_2} = Cy_n$. In support of this conjecture we present, in Table 10.3, the number of flops executed by Matlab in producing the eigenspectra using the matrix C_1 where $C_1 = \prod_{i=0}^{\frac{kNp}{d}-1} A(n-i)$ for $k = 1, 2, \dots, 6$ and for three different step lengths defined by the values of m . There is clear evidence of an additional cost for larger values of k .

The periodicity of $b(t)$, and hence of $A(t)$, implies that $\prod_{i=\frac{Npk}{d}}^{(k+1)\frac{Np}{d}-1} A(n-i) = C$ for $k = 0, 1, \dots, 5$. Hence we can reduce the computational time needed by evaluating $C^{p_2d_1}$, where $C = \prod_{i=0}^{\frac{Np}{d}-1} A(n-i)$, instead of evaluating C^* with C^* defined by $C^* = \prod_{i=0}^{(p_2d_1)(\frac{Np}{d})-1} A(n-i)$. In Table 10.4, for example 10.7.1, we state the number of flops executed by Matlab in producing the eigenspectra, using C , C^6 and C^* . We observe a reduction when C^6 is used instead of C^* and a further

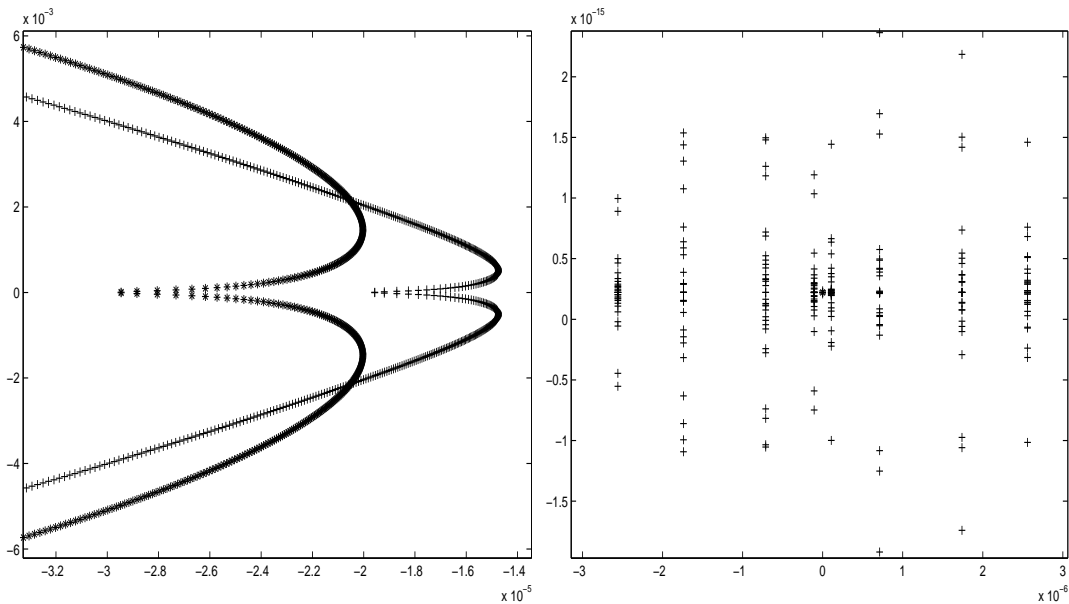


Figure 10.5: Left: Eigenvalues displayed in Figure 8.2, Case 2, raised to the power of 7

Right: Eigenvalues displayed in Figure 8.6, Example 3, raised to the power of 8

k	$m = 20$	$m = 40$	$m = 80$
1	1.8658×10^8	2.5806×10^9	3.8058×10^{10}
2	3.2880×10^8	4.7997×10^9	7.3628×10^{10}
3	4.7029×10^8	7.0349×10^9	1.0921×10^{11}
4	6.0754×10^8	9.2480×10^9	1.4462×10^{11}
5	7.4698×10^8	1.1483×10^{10}	1.8026×10^{11}
6	8.8468×10^8	1.3685×10^{10}	2.1552×10^{11}

Table 10.3: The number of flops executed by Matlab in producing the eigenspectra when $k \times \frac{Np}{d}$ matrix multiplications are performed prior to the calculation of the eigenvalues

reduction when C is used instead of C^* . (However, the eigenspectra arising from the use of C is not symmetrical about the real axis).

Remark 10.7.1 Our motivation in producing the diagrams is to detect the presence of small solutions. We observe that they are clearly detectable using the product of only $\frac{Np}{d}$ matrices, thus saving the additional computational time needed to produce diagrams similar to those produced in our previous work. However, the development of our algorithm was motivated by diagrams which are symmetrical about the real axis only. The detection of the presence of small

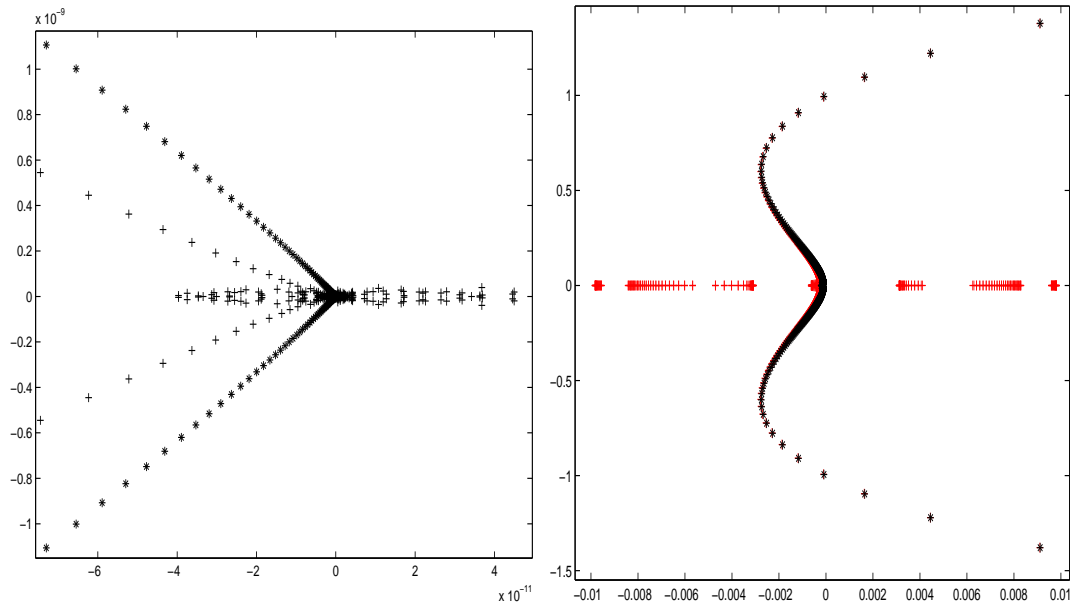


Figure 10.6: Left: Eigenvalues displayed in Figure 8.8, Example 7, raised to the power of 6
 Right: Eigenvalues displayed in Figure 8.9, Example 9 raised to the power of 4

m	Using C	Using C^6	Using C^*
20	1.8658×10^8	2.0440×10^8	8.8468×10^8
40	2.5806×10^9	2.7113×10^9	1.3685×10^{10}
80	3.8058×10^{10}	3.9247×10^{10}	2.1552×10^{11}

Table 10.4: Comparing the number of flops executed by Matlab when the matrix used is C , C^6 or C^* .

solutions (when they are present) is through additional trajectories lying close to the real axis. Additional computational time is needed to produce eigenspectra with only one axis of symmetry. Hence, if we wish to modify our algorithm to automate the process of detecting small solutions to equations of the form (10.3), we anticipate that we will need to accept the cost of the additional computational time.

An example of the systems case

We now demonstrate the potential for extending our approach to higher dimensions by including an example of the three dimensional case.

We consider equation $x'(t) = A(t)x(t-d)$ where $A(t)$ is such that $A(t+p) = A(t)$

and

$$A(t) = \begin{pmatrix} \sin\left(\frac{2\pi t}{p}\right) + 0.6 & \sin\left(\frac{2\pi t}{p}\right) + 1.3 & \sin\left(\frac{2\pi t}{p}\right) + 1.7 \\ 0 & \sin\left(\frac{2\pi t}{p}\right) + 0.5 & \sin\left(\frac{2\pi t}{p}\right) + 1.4 \\ 0 & 0 & \sin\left(\frac{2\pi t}{p}\right) + 0.2 \end{pmatrix} \text{ with } p = \frac{1}{3} \text{ and } d = 1.$$

Here $p_2d_1 = 3$. We note that $a_{11}(t)$, $a_{22}(t)$ and $a_{33}(t)$ all change sign and in Figure 10.7 we can see three sets of additional trajectories indicating the presence of small solutions. For the left-hand diagram of Figure 10.7 the product of $\frac{Np}{d}$ matrices has been used and we observe three axes of symmetry. In the right-hand diagram we show the eigenvalues of $C^{p_2d_1}$. We observe that the real axis is the only axis of symmetry and in Table 10.5 we present further evidence of the increase in computational time needed by displaying the number of flops executed in the production of the eigenspectra in Figure 10.7.

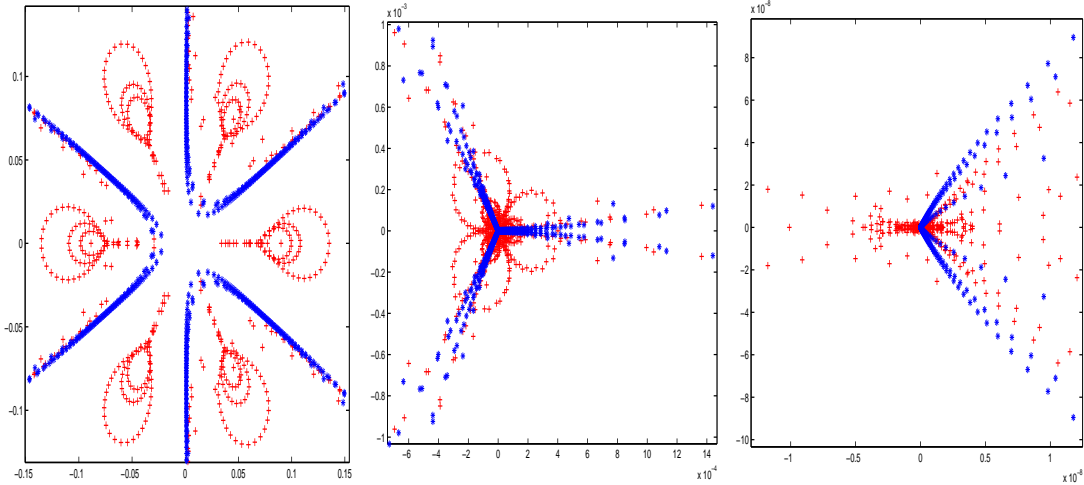


Figure 10.7: Left: Eigenvalues of C Centre: Eigenvalues of C^2
Right: Eigenvalues of $C^3 = C^{p_2d_1}$

Eigenspectra	Matrix Product used	Number of flops
Left	C	1.2665×10^{11}
Centre	C^2	2.4855×10^{11}
Right	$C^3 = C^{p_2d_1}$	3.7038×10^{11}

Table 10.5: A comparison of the number of flops used in the production of the eigenspectra displayed in Figure 10.7

The idea of an algorithm that automates the process of determining whether or not equations of the form (8.33) admit small solutions is attractive. Our algorithm, 'smallsolutiondetector1', has been developed from eigenspectra that are

symmetrical about the real axis. We have demonstrated that such eigenspectra can be produced using additional computational time. However, further work is needed before we can implement an effective algorithm.

Chapter 11

Complex-valued functions

11.1 Introduction

In chapters 4 and 5 we considered the equation

$$(11.1) \quad x'(t) = b(t)x(t-1)$$

with $b(t)$ a real-valued, 1-periodic function. In this chapter we revisit this equation for the case when $b(t)$ is a scalar-valued complex periodic function of period 1.

Guglielmi's heading in [38] "Instability of the trapezoidal rule" is effective in alerting the reader to the fact that the trapezium rule is not τ -stable, a definition of stability concerning (11.1) when $b(t)$ is a complex-valued function (see section 2.4.1 for the definition of τ -stability). However, the backward Euler method is τ -stable (see [11, 38]). This questions the use of the trapezium rule for this case and indicates that the backward Euler method is appropriate. This adds a layer of complexity not encountered previously in our work. The delay in our equation is fixed, hence delay dependent stability conditions are appropriate. Eigenspectra using the backward Euler were judged to be less efficient in the real case in chapter 5. Early experimentation for the complex case involved the use of the trapezium rule. The initial results from using the backward Euler method seemed promising (see section 11.3). However, some later results provoked further interest and motivated our decision that it would be of interest to compare results of using the two methods. (We note here that the authors of [38] say "numerous numerical experiments have shown that the numerical and the true stability regions may be remarkably dissimilar (especially for small values of m)).

In this chapter we:

- investigate whether our approach to detecting small solutions is likely to be successful when $b(t)$ is complex-valued,

- compare the eigenspectra arising from discretisation using the trapezium rule to that arising from use of the backward Euler method, that is, we compare the use of a method that is unstable for the problem to one that is stable for the problem,
- explain how we can interpret our eigenspectra in relation to the known analytical theory in this case.

We begin by stating known analytical results for equation (11.1). We then show that we can employ the methods used in chapters 4, 5 and 6, resulting in eigenspectra based on which we make our decision concerning the existence, or otherwise, of small solutions to equation (11.1). Examples of eigenspectra arising from equations which are known not to admit small solutions are then presented. In this way we can begin to characterise the eigenspectra for this case. We then consider the case when a sufficient condition for small solutions is satisfied. This provides examples of eigenspectra, the interpretation of which must be that the equation admits small solutions to be consistent with known theory. These provide further insight and begin our characterisation of eigenspectra that indicate the presence of small solutions.

We have found that the question of invertibility of a function $F : \mathbb{R} \rightarrow \mathbb{R} \times \mathbb{R}$ does not appear to be readily addressed in the literature. This is presenting problems in finding suitable examples on which to test our approach. We report on the progress made to date.

We present examples of two types of function $b(t)$ and solve the problems using each of the numerical methods. We discuss the effect of using a method that is not τ -stable on the ease and accuracy with which we can detect small solutions.

11.2 Known analytical results

Theorem 11.2.1 (*Theorem 4.7 in [77]*) *If $b(t)$ is a nonzero complex scalar-valued periodic function with period 1 such that the real and imaginary part of $b(t)$ have constant sign, then the monodromy operator associated with $\dot{x}(t) = b(t)x(t-1), t \geq 0$, has a complete span of eigenvectors.*

Remark 11.2.1 (see page 504 in [77])

Completeness is determined by the behaviour of the curve $\zeta : [0, 1] \rightarrow \mathbb{C}$ by $\zeta(t) = \int_{-1}^t b(s)ds$. A sufficient condition for the presence of small solutions, that is for completeness to fail, is that there exist θ_1, θ_2 with $-1 \leq \theta_1 < \theta_2 \leq 0$ such that $\int_{\theta_1}^{\theta_2} b(s)ds = 0$. This is equivalent to requiring the curve $\zeta(t)$ to have a self intersection.

Remark 11.2.2 In [77] Verduyn Lunel gives the necessary and sufficient condition for the operator to have a complete set of eigenvectors as ‘ $\zeta(t)$ is an invertible function.’

11.3 Justification for our approach

We adopt the approach used in section 4 of [33], (used there with the backward Euler scheme), with $b(t)$ a real valued function. We illustrate how known theory about the characteristic values of the solution map, here with $b(t)$ a complex-valued function, is evidenced through our visualisation following numerical discretisation using both the trapezium rule and the backward Euler method.

We again compare the eigenspectra arising from the discretisation of (11.1) with that arising from $x'(t) = \hat{b}x(t-1)$ where $\hat{b} = \int_0^1 b(t)dt$. In the examples in this section $b(t) = \sin(2\pi t + d_1 + d_2i) + c_1 + c_2i$ leading to $\hat{b} = \int_0^1 \{\sin(2\pi t + d_1 + d_2i) + c_1 + c_2i\}dt = c_1 + c_2i$. We adopt our usual notation for the scalar case.

key to Figures 11.1, 11.2, 11.3 and 11.4

- **The solid line** shows the locus of the true characteristic values, $|\lambda| = |\hat{b}e^{-\lambda}|$ for $\hat{b} = 1.2 + 0.4i$.
- ******* shows $\frac{1}{h} \times$ the complex logarithms of the eigenvalues of the matrix A where $y_{n+1} = Ay_n$, arising from the autonomous equation, $x'(t) = \hat{b}x(t-1)$. (By Theorem 3.2 in [27] we are guaranteed that the characteristic values of the discrete solution should approximate the true characteristic values).
- **+++** shows $\frac{1}{h} \times$ the complex logarithms of the eigenvalues of the matrix $A(1)$ where $A(1)$ is such that $y_{n+1} = A(1)y_n$, arising from the equation $x'(t) = b(t)x(t-1)$, with $b(t) = \sin(2\pi t + 0.3 + 0.2i) + 1.2 + 0.4i$. This equation does not admit small solutions since both the real and imaginary parts have constant sign (see Theorem 11.2.1). Consequently the eigenspectra arising from this equation should closely mimic that from the autonomous problem in use.
- We illustrate the known theory [33] that there is one eigenvalue in each horizontal band of width 2π .

In Figures 11.1 and 11.2 the numerical method used is the trapezium rule. In Figures 11.3 and 11.4 the backward Euler method has been used with the same equation and step lengths.

Comparing the results from the two numerical schemes we observe that for the trapezium rule the eigenspectrum arising from the autonomous equation when $h = \frac{1}{1000}$ is closer to the true characteristic curve than that when $h =$

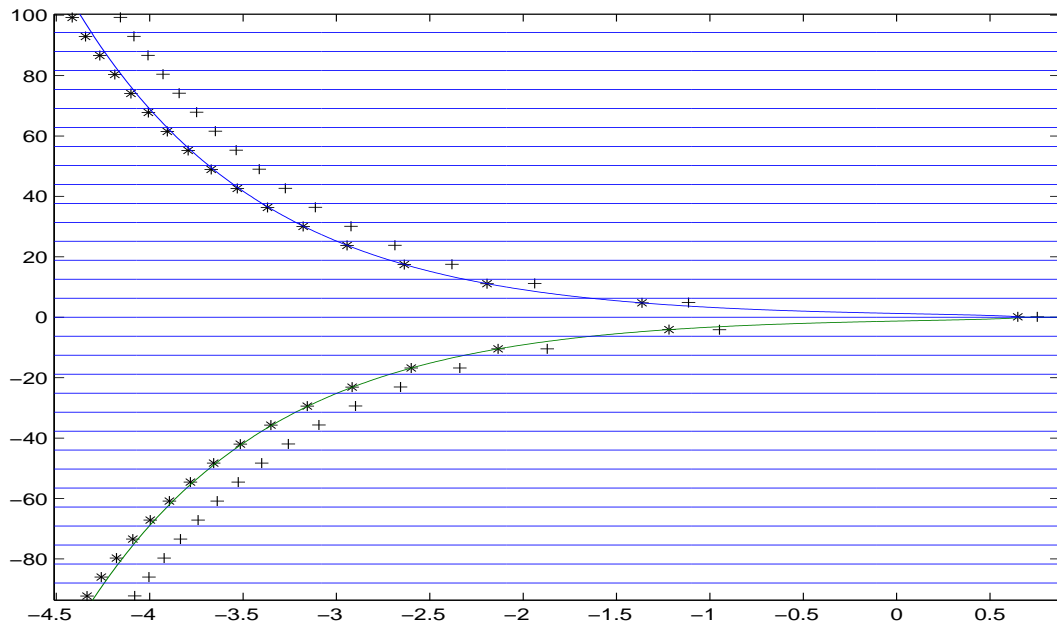


Figure 11.1: Trapezium rule: $b(t) = \sin(2\pi t + 0.3 + 0.2i) + 1.2 + 0.4i$, step length = $\frac{1}{128}$

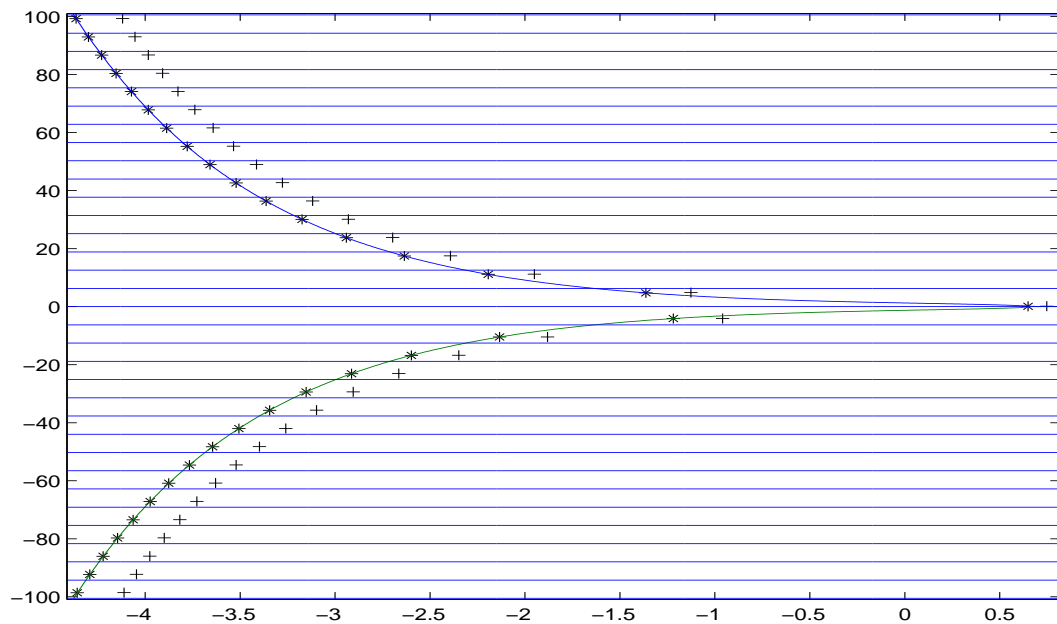


Figure 11.2: Trapezium rule: $b(t) = \sin(2\pi t + 0.3 + 0.2i) + 1.2 + 0.4i$, step length = $\frac{1}{1000}$

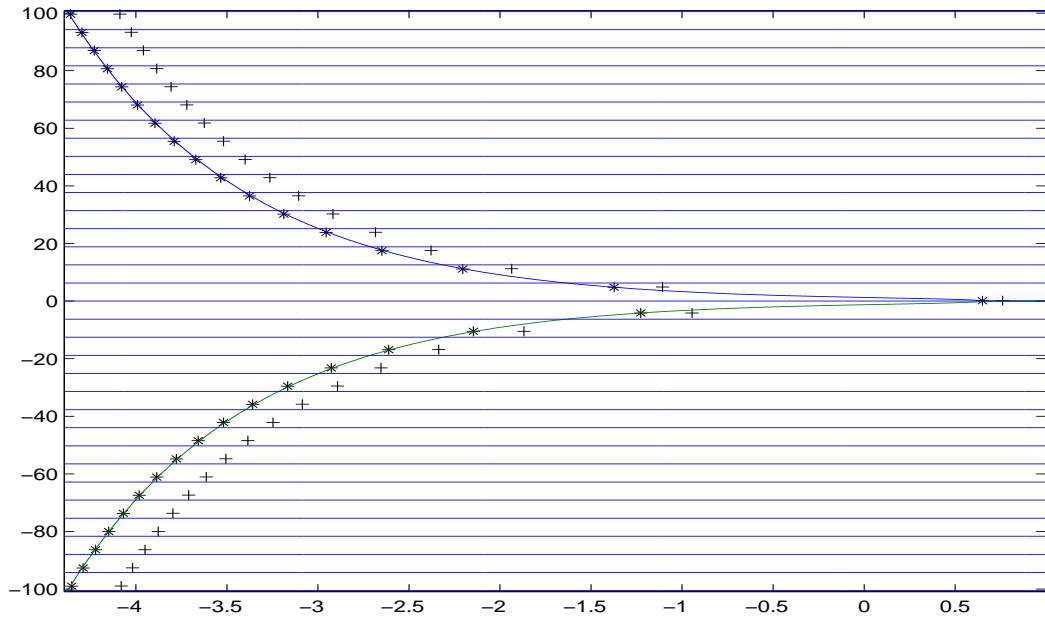


Figure 11.3: Backward Euler: $b(t) = \sin(2\pi t + 0.3 + 0.2i) + 1.2 + 0.4i$, step length = $\frac{1}{128}$

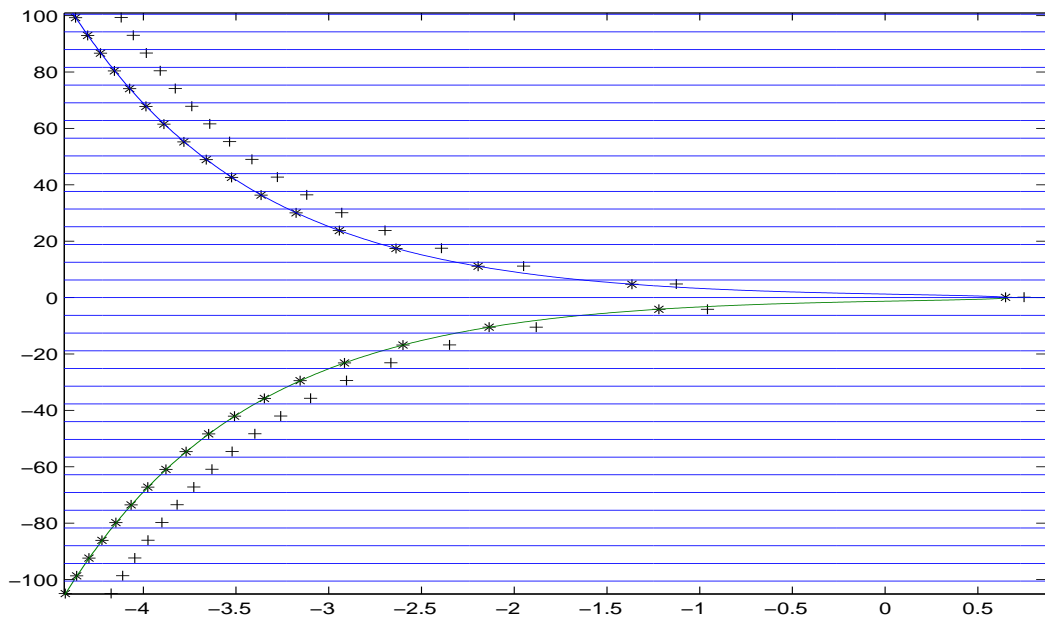


Figure 11.4: Backward Euler: $b(t) = \sin(2\pi t + 0.3 + 0.2i) + 1.2 + 0.4i$, step length = $\frac{1}{1000}$

$\frac{1}{128}$, that is, a decrease in step length has *visibly* improved the approximation. The improvement is not visible (on the chosen scale) when the backward Euler method is used. The scales have been chosen to enable easy comparison between the two numerical methods. However, although we note the improvement in the approximation, we feel that a step size of $h = \frac{1}{128}$ is again an appropriate compromise between accuracy and speed.

Similar diagrams have been produced for equation (11.1) with $b(t) = \sin(2\pi t + 0.5 + 0.4i) + 0.1 + 0.3i$ and $\sin(2\pi t + 1.9 + 1.3i) + 0.6 - 0.4i$. For these cases we are able to find t_1, t_2 such that $\int_{t_1}^{t_2} b(s)ds = 0, t_1 \neq t_2$. Hence, the equation admits small solutions (see remark 11.2.1). In Figures 11.5 and 11.7 the numerical method used is the trapezium rule and in Figures 11.6 and 11.8 the backward Euler method has been used. In this case, when the non-autonomous problem admits small solutions, the trajectories arising from the non-autonomous and autonomous problems are not close to each other.

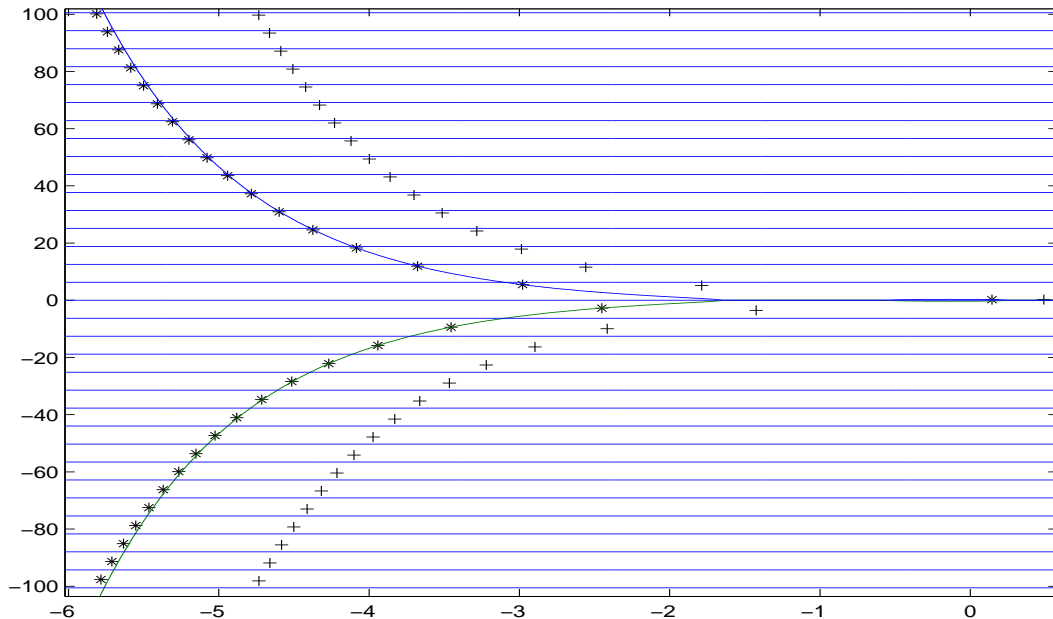


Figure 11.5: Trapezium rule: $b(t) = \sin(2\pi t + 0.5 + 0.4i) + 0.1 + 0.3i$, step length = $\frac{1}{128}$

The eigenspectra in this section support our view that our approach will be effective in detecting small solutions for this class of DDE. The eigenspectra display characteristics that can be interpreted in a way that is consistent with known theory.

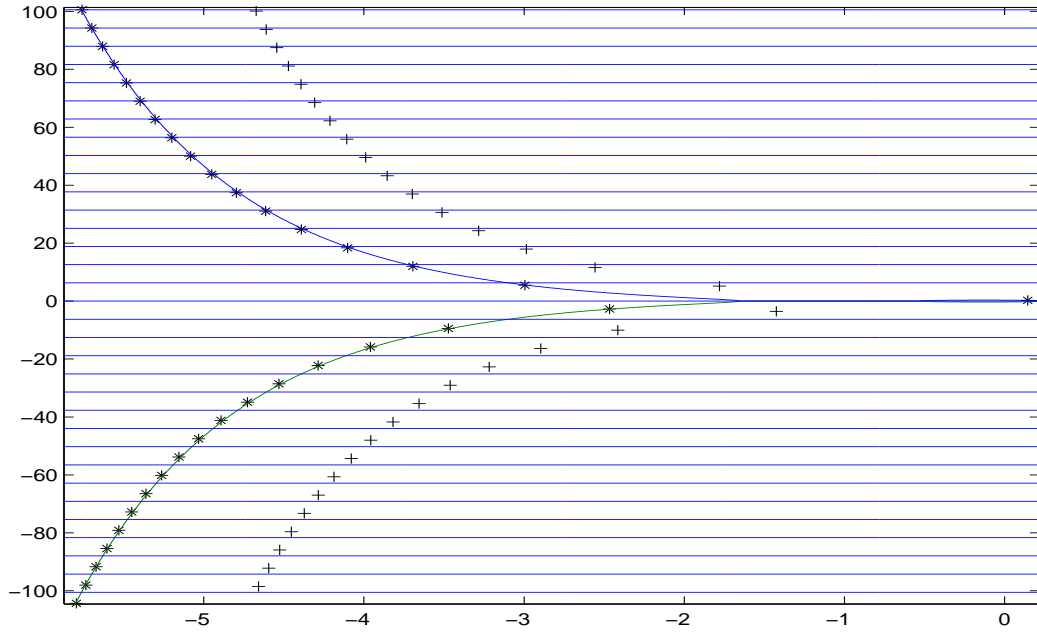


Figure 11.6: Backward Euler: $b(t) = \sin(2\pi t + 0.5 + 0.4i) + 0.1 + 0.3i$, step length = $\frac{1}{128}$

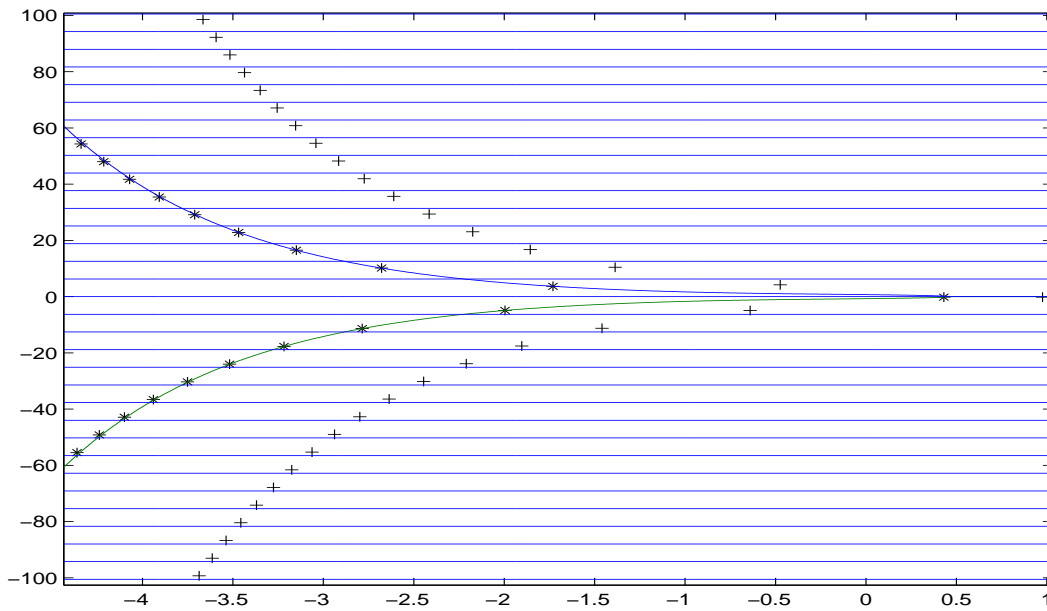


Figure 11.7: Trapezium rule: $b(t) = \sin(2\pi t + 1.9 + 1.3i) + 0.6 - 0.4i$, step length = $\frac{1}{128}$

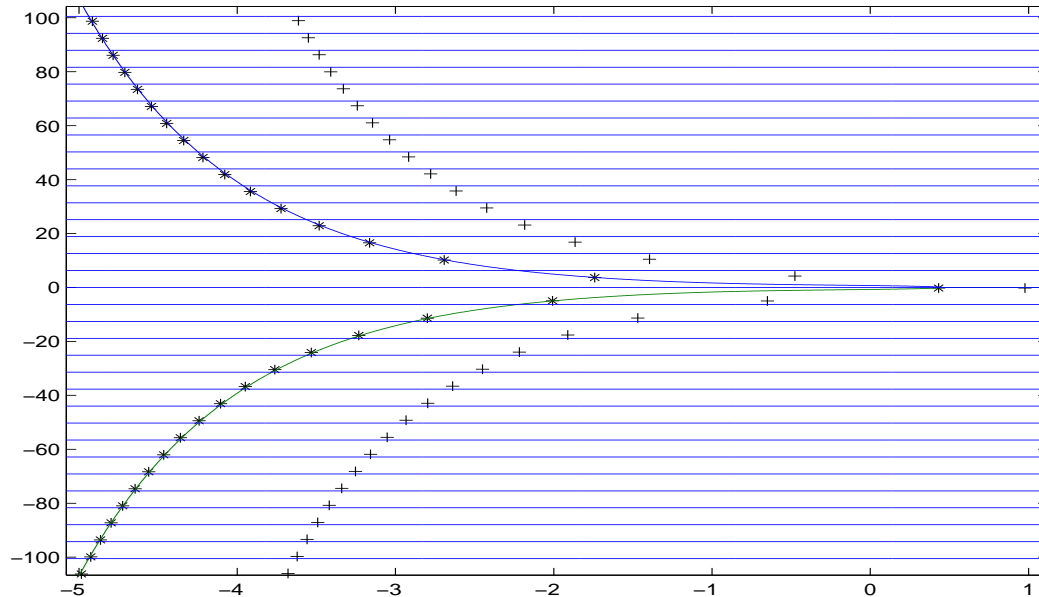


Figure 11.8: Backward Euler: $b(t) = \sin(2\pi t + 1.9 + 1.3i) + 0.6 - 0.4i$, step length = $\frac{1}{128}$

11.4 Numerical results

We have justified our methodology in the previous section. For each problem considered we will present eigenspectra arising from the use of each of the two numerical methods, the trapezium rule and the backward Euler method. Our focus is on the detection of small solutions but we are interested to see whether we have evidence of how the stability of a numerical method might affect our method. We present illustrative examples in which $b(t)$ is a trigonometric function and a linear function. Based on earlier work in chapters 4 and 5 we would anticipate that any similarities arising from these two functions may also arise for a wider function-type of $b(t)$. We can define the solution map as in section 4.4. We again compare the eigenspectrum from the non-autonomous problem (11.1) with that arising from the autonomous problem $x'(t) = \hat{b}x(t-1)$ where $\hat{b} = \int_0^1 b(t)dt$.

11.4.1 The equation does not admit small solutions

In this section we present eigenspectra arising from problems that are known not to have small solutions. This will enable us to begin our characterisation of the eigenspectra arising from the complex case. By Theorem 11.2.1 we know that if both the real and imaginary components of $b(t)$ are of constant sign then equation (11.1) does not admit small solutions. We first give examples of the

case when $b(t)$ is a trigonometric function and follow this with examples when $b(t)$ is a linear function.

Examples where $b(t)$ is a trigonometrical function

Consider $b(t) = \sin(2\pi t + d_1 + d_2i) + c_1 + c_2i$ where $c_1, c_2, d_1, d_2 \in \mathbb{R}$. We note that $\hat{b} = c_1 + c_2i$.

We can rewrite $b(t)$ as

$$(11.2) \quad b(t) = \{\sin(2\pi t + d_1) \cosh(d_2) + c_1\} + i\{\cos(2\pi t + d_1) \sinh(d_2) + c_2\}.$$

If $|c_1| > \cosh(d_2)$ and $|c_2| > |\sinh(d_2)|$ then both the real and imaginary parts of $b(t)$ are of constant sign. Hence by Theorem 11.2.1 we know that the equation does not admit small solutions and we expect the eigenspectra arising from the non-autonomous and autonomous problems to be very similar. Details of the illustrative examples for this case are given in Table 11.1. In each figure the left-hand diagram is the eigenspectra arising from the trapezium rule and that in the right-hand diagram that from the backward Euler method. (Values of $\cosh(x)$ and $\sinh(x)$ are given to 3 decimal places.)

Example	Figure	c_1	c_2	d_1	d_2	$\cosh(d_2)$	$(\sinh(d_2))$
1	11.9	2	1	0.3	0.6	1.185	0.637
2	11.10	5	2.5	0.1	1.5	2.129	2.129
3	11.11	-1.5	0.2	1.6	-0.1	1.005	-0.100
4	11.12	1.05	0.31	0.5	0.3	1.0453	0.305

Table 11.1: Details of examples where $b(t)$ is a trigonometric function that does not change sign

Examples where $b(t)$ is a linear function of t

We consider $b(t) = (d_1 + d_2i)t + c_1 + c_2i$ with $b(t+1) = b(t)$. In the autonomous problem $\hat{b} = 0.5(d_1 + d_2i) + c_1 + c_2i$.

We illustrate with the following examples:

- (i) $b(t) = t - 2 + 0.2i$ (see Figure 11.13).
- (ii) $b(t) = (2 - i)t + 0.1 - 0.5i$ (see Figure 11.14).
- (iii) $b(t) = (0.1 + 2i)t + 0.3 + 0.2i$ (see Figure 11.15).

Observations

Unlike earlier eigenspectra we observe, as expected, that the trajectories are not symmetrical about the real axis. To be consistent with known theory eigen-

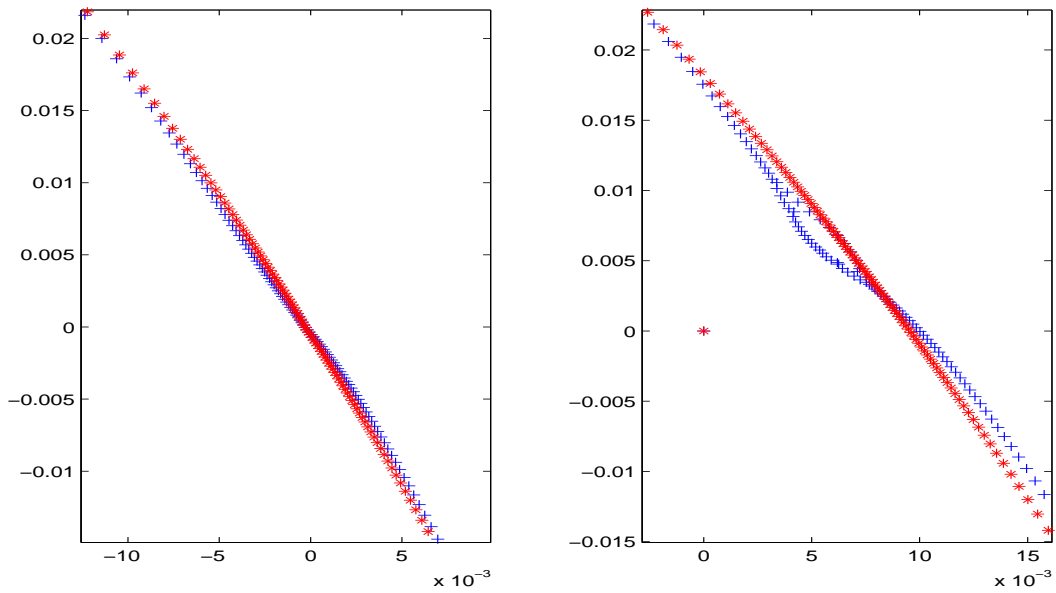


Figure 11.9: Example 1 (Table 11.1). The equation does not admit small solutions.

Left: Trapezium rule. Right: Backward Euler.

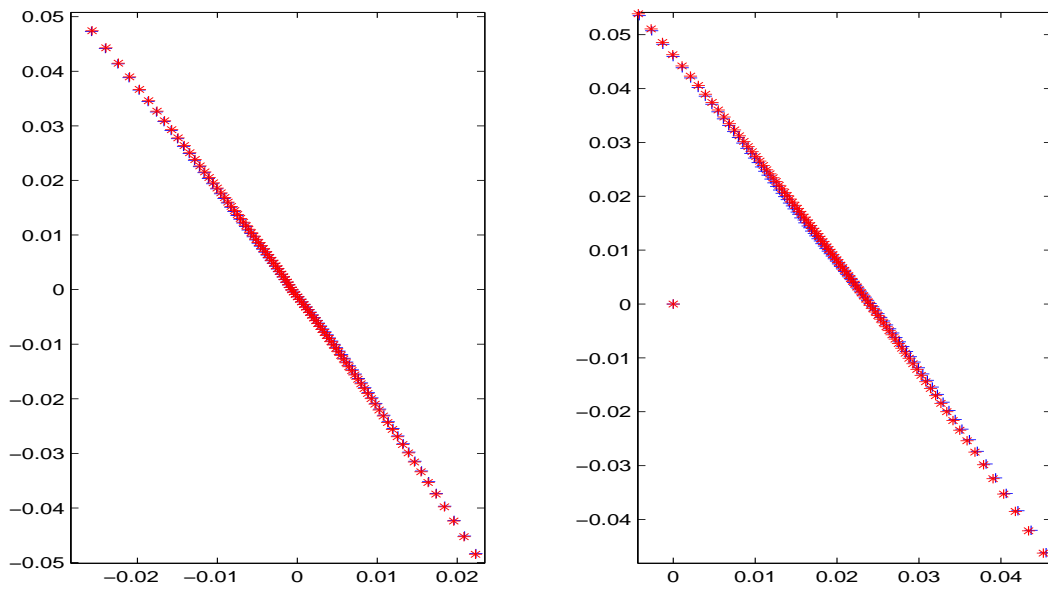


Figure 11.10: Example 2 (Table 11.1). The equation does not admit small solutions.

Left: Trapezium rule. Right: Backward Euler.

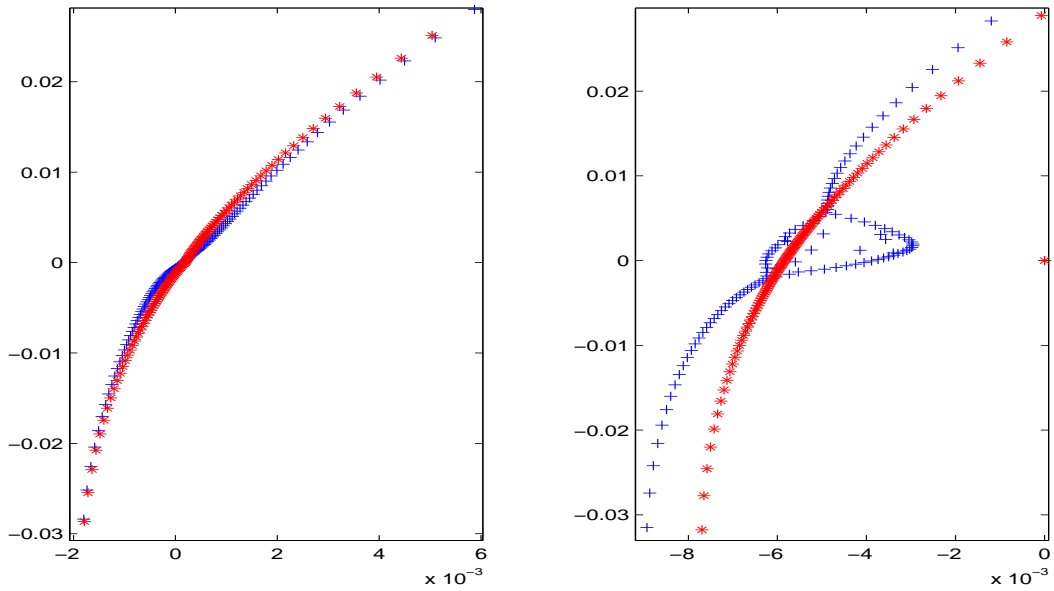


Figure 11.11: Example 3 (Table 11.1). The equation does not admit small solutions.

Left: Trapezium rule. Right: Backward Euler

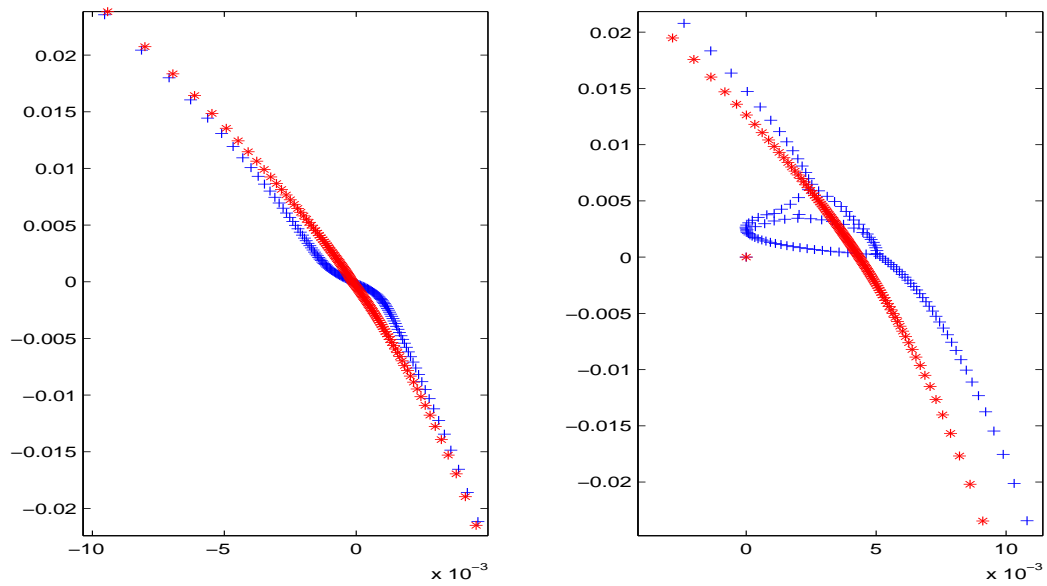


Figure 11.12: Example 4 (Table 11.1). The equation does not admit small solutions.

Left: Trapezium rule. Right: Backward Euler.

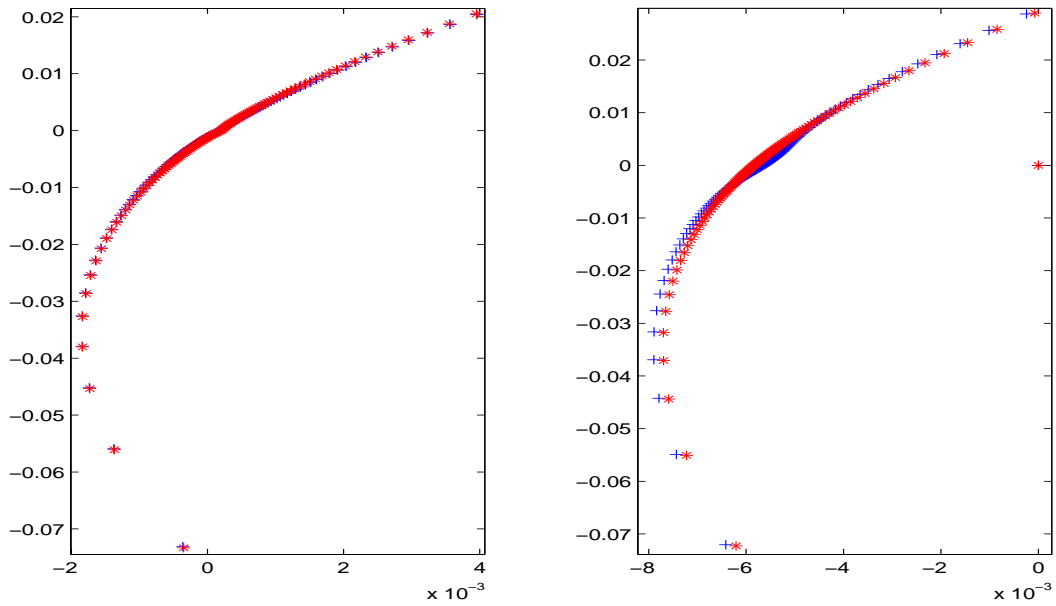


Figure 11.13: $b(t) = t - 2 + 0.2i$. The equation does not admit small solutions. Left: Trapezium rule. Right: Backward Euler.

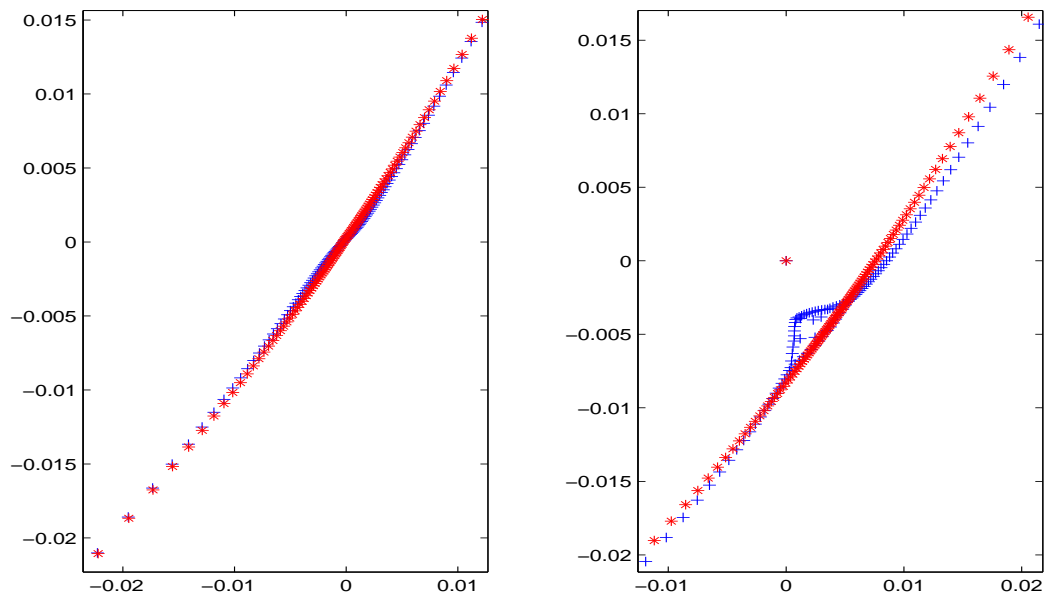


Figure 11.14: $b(t) = (2 - i)t + 0.1 - 0.5i$. The equation does not admit small solutions. Left: Trapezium rule. Right: Backward Euler.

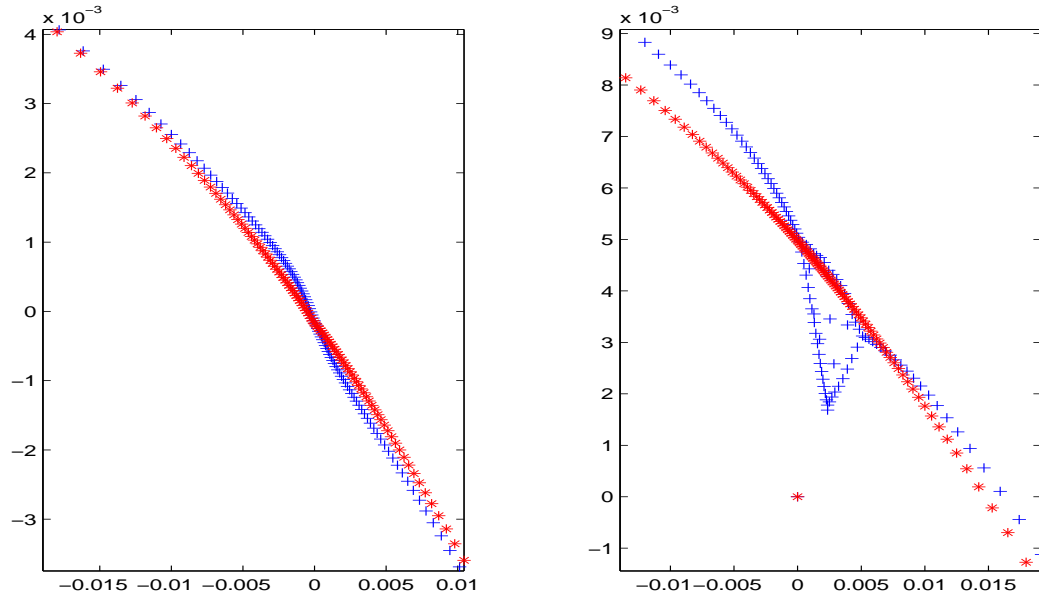


Figure 11.15: $b(t) = (0.1 + 2i)t + 0.3 + 0.2i$. The equation does not admit small solutions.

Left: Trapezium rule. Right: Backward Euler.

spectra with the characteristic shapes in this section must be interpreted as indicating that no small solutions are present. The eigenvalues arising from use of the trapezium rule clearly lie on one asymptotic curve. We observe similarities between the eigenspectra in the right-hand diagram of Figures 11.11, 11.12, 11.14 and 11.15 and the left-hand diagram of Figure 5.8 (when the backward Euler method was used for an equation that does not admit small solutions). As a consequence the observed deviation from the eigenspectrum of the autonomous problem does not give cause for concern. The scales used on the diagrams are similar. The trapezium rule appears to lead to eigenspectra that are closer together and more easily interpreted. From the results of this section we conclude that we are able to detect the *absence* of small solutions using our approach.

11.4.2 A sufficient condition for small solutions is satisfied

In our desire to detect small solutions using a numerical discretisation, followed by a visual representation of the resulting eigenvalues, we seek a positive response to the question ‘Is it clear from our eigenspectra whether or not the equation admits small solutions?’ For the examples presented in this section it is known that the equation admits small solutions. This begins our characterisation of

eigenspectra which, to be consistent with known theory, need to clearly indicate the presence of small solutions to the equation.

Examples where $b(t)$ is a trigonometrical function

We again consider $b(t) = \sin(2\pi t + d_1 + d_2i) + c_1 + c_2i$ where $c_1, c_2, d_1, d_2 \in \mathbb{R}$. From remark 11.2.1 we can see that, since $b(t)$ is a 1-periodic function, equation (11.1) will admit small solutions if we can find t_1, t_2 with $0 \leq t_1 < t_2 \leq 1$ such that

$$(11.3) \quad \int_{t_1}^{t_2} \{\sin(2\pi t + d_1 + d_2i) + c_1 + c_2i\} = 0.$$

This requires

$$(11.4) \quad \int_{t_1}^{t_2} \{\sin(2\pi t + d_1) \cosh(d_2) + c_1\} = 0$$

and

$$(11.5) \quad \int_{t_1}^{t_2} \{\cos(2\pi t + d_1) \sinh(d_2) + c_2\} = 0,$$

which leads to

$$(11.6) \quad \frac{1}{\pi} \sin[\pi(t_1 + t_2) + d_1] \sin[\pi(t_2 - t_1)] \cosh(d_2) + c_1(t_2 - t_1) = 0$$

and

$$(11.7) \quad \frac{1}{\pi} \cos[\pi(t_1 + t_2) + d_1] \sin[\pi(t_2 - t_1)] \sinh(d_2) + c_2(t_2 - t_1) = 0.$$

Our interest lies in finding a solution in which $t_1 \neq t_2$. In this case we can use (11.6) and (11.7) to obtain

$$(11.8) \quad \frac{c_2}{c_1} = \frac{\tanh(d_2)}{\tan[\pi(t_1 + t_2) + d_1]}, \quad c_1 \neq 0, \quad \pi(t_1 + t_2) + d_1 \neq n\pi, n \in \mathbb{Z},$$

and

$$(11.9) \quad \frac{\pi^2(t_2 - t_1)^2}{\sin^2[\pi(t_2 - t_1)]} \left\{ \frac{c_1^2}{\cosh^2(d_2)} + \frac{c_2^2}{\sinh^2(d_2)} \right\} = 1.$$

From equation (11.8) we see that

$$(11.10) \quad \pi(t_1 + t_2) + d_1 = n\pi + \tan^{-1} \left[\frac{c_1 \tanh(d_2)}{c_2} \right].$$

Equation (11.9) is of the form

$$(11.11) \quad \frac{\pi^2 x^2}{\sin^2(\pi x)} \{k\} = 1, x \neq 0,$$

where $x = t_2 - t_1, k = \frac{c_1^2}{\cosh^2(d_2)} + \frac{c_2^2}{\sinh^2(d_2)}$. Our analytical search for equations that admit small solutions reduces in this case to the following question. For a given problem can we find values of t_1 and t_2 such that both (11.8) and (11.9) are satisfied?

Remark 11.4.1 A visual inspection of the intersection of the curves

$$f_1(x) = k\pi^2 x^2 \text{ and } f_2(x) = \sin^2(\pi x),$$

combined with a search for the zeros of $f_1(x) = f_2(x)$ (using the Newton-Raphson method), enabled us to determine whether or not non-zero values of $(t_2 - t_1)$ satisfying (11.9) existed. Non-zero values of $(t_2 - t_1)$ exist if $0 < k < 1$. An infinite number of values of t_1 and t_2 are possible. We choose the value to give t_1 and t_2 in the required range. (Values are given to 4 decimal places when appropriate.)

In Table 11.2 we give details of the equation being used for Figures 11.16 to 11.19. In Figure 11.16 an additional trajectory is observed for the non-autonomous problem. In Figure 11.17 the two trajectories are very different. The right-hand diagram of Figure 11.18 compares favourably with those produced using the backward Euler for the case when $b(t)$ is real and the equation admits small solutions (see chapter 5). The eigenspectra in Figure 11.19 resembles more closely those found in the real case (see chapter 5).

Example	c_1	c_2	d_1	d_2	k	$(t_2 - t_1)$	$(t_1 + t_2)$	Figure
1	0.1	0.3	0.5	0.4	0.5420	0.4182	0.8809	11.16
2	0.3	0.4	0.1	2.5	0.0068	0.7062	0.8681	11.17
3	0.8	1.1	0.6	1.1	0.9082	0.1703	0.9678	11.18
4	0.6	0.01	0.2	0.1	0.3664	0.5243	1.3836	11.19

Table 11.2: Examples of equations that satisfy the sufficient condition for small solutions to exist.

Remark 11.4.2 If $d_2 = 0$ then $\sinh(d_2) = 0$. This implies that $c_2(t_2 - t_1) = 0$. For non-zero $(t_2 - t_1)$ this gives $c_2 = 0$ and we return to the case when $b(t)$ is a real function.

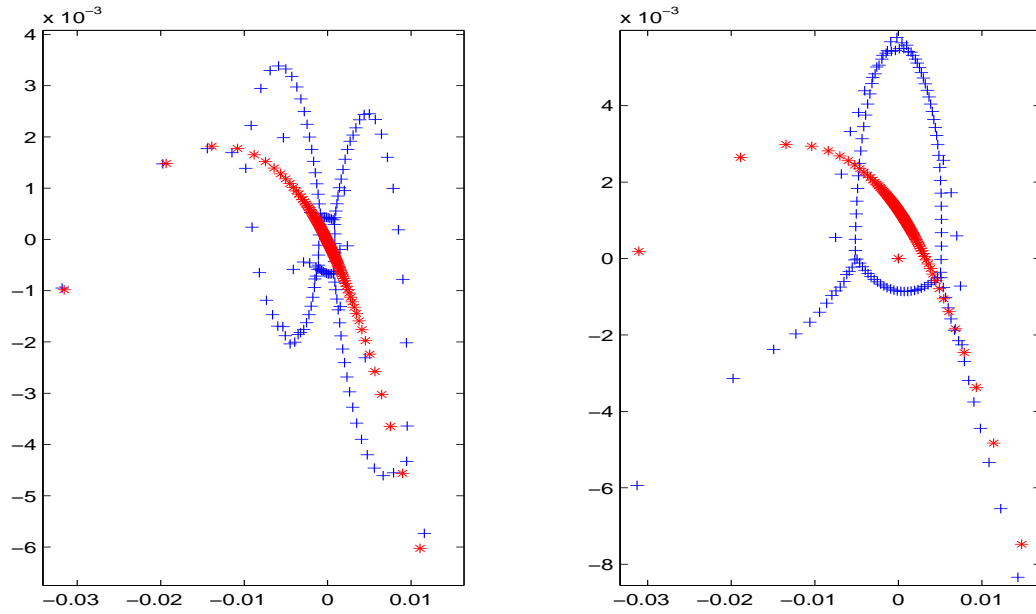


Figure 11.16: Example 1 (Table 11.2). The equation admits small solutions. Left: Trapezium rule. Right: Backward Euler.

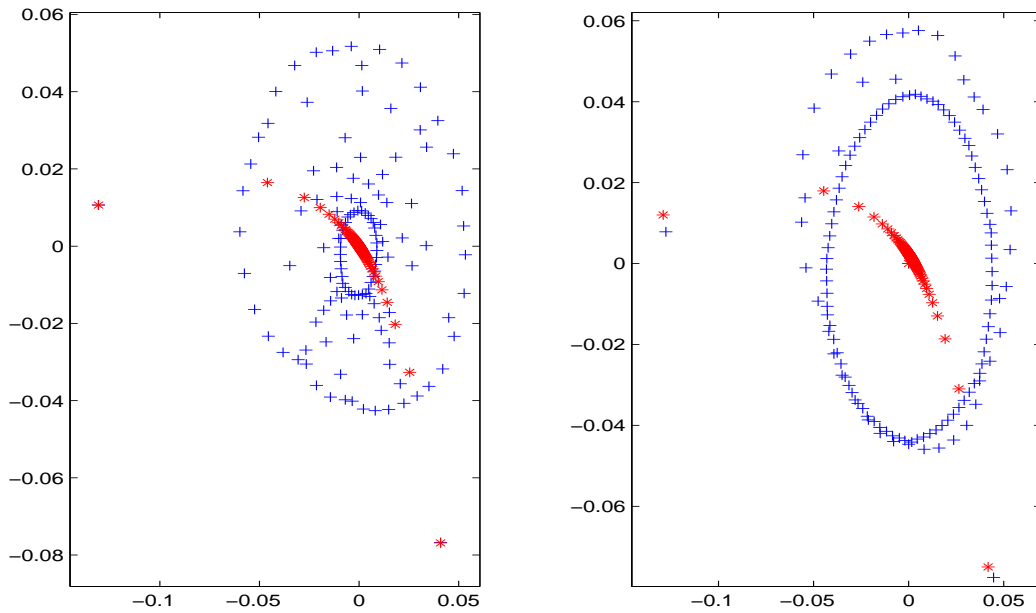


Figure 11.17: Example 2 (Table 11.2). The equation admits small solutions. Left: Trapezium rule. Right: Backward Euler.

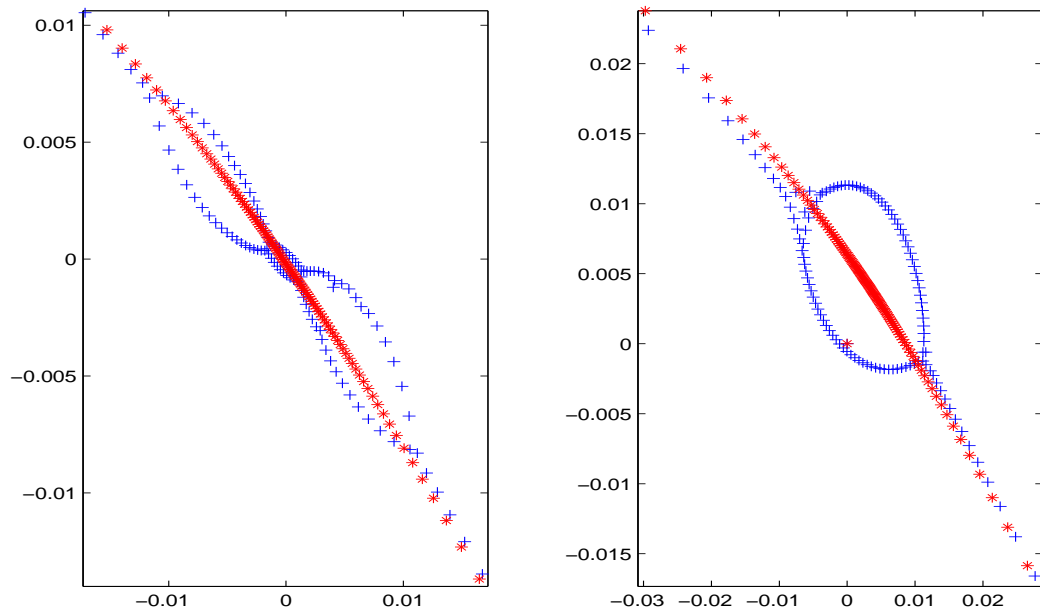


Figure 11.18: Example 3 (Table 11.2). The equation admits small solutions. Left: Trapezium rule. Right: Backward Euler.

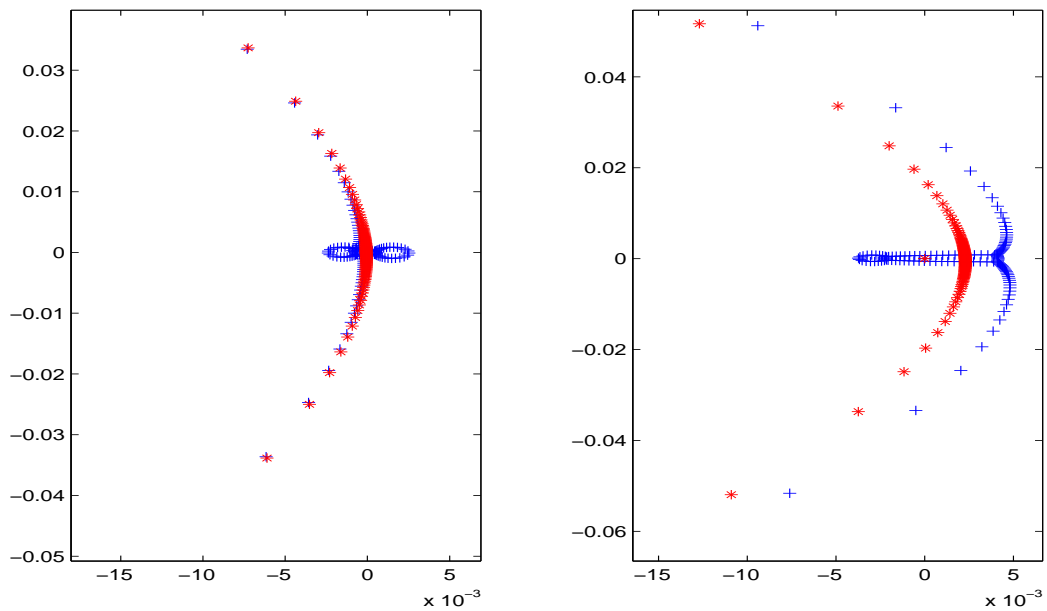


Figure 11.19: Example 4 (Table 11.2). The equation admits small solutions. Left: Trapezium rule. Right: Backward Euler.

Remark 11.4.3 If $c_1 = 0$ we can show that, if $t_1 \neq t_2$, we need to find non-zero solutions to the equation

$$\pm \frac{\sinh(d_2)}{\pi} \sin(\pi x) + c_2 x = 0, \text{ where } x = t_2 - t_1,$$

to satisfy the sufficient condition for small solutions (Remark 11.2.1). A similar condition applies if $c_2 = 0$.

Remark 11.4.4 We observe that, since $\lim_{x \rightarrow 0} \left\{ \frac{\pi^2 x^2}{\sin^2(\pi x)} \right\} = 1$, a value of $k = 1$ may lead to eigenspectra from which the decision about the existence, or otherwise, of small solutions may be unclear. We do not include examples of this case in this section since the sufficient condition for the presence of small solutions is not satisfied (remark 11.2.1). See section 11.4.3 for illustrative examples.

Examples where $b(t)$ is a linear function of t

We again consider

$$b(t) = (d_1 + d_2 i)t + c_1 + c_2 i$$

with $b(t+1) = b(t)$. Satisfying the sufficient condition for the equation to admit small solutions (Remark 11.2.1) leads to the requirement that

$$(t_2 - t_1) \left\{ \frac{1}{2}(d_1 + d_2 i)(t_2 + t_1) + c_1 + c_2 i \right\} = 0.$$

If $t_1 \neq t_2$ then we find that

$$t_2 + t_1 = \frac{-2(c_1 d_1 + c_2 d_2) - 2(c_2 d_1 - c_1 d_2)i}{d_1^2 + d_2^2}.$$

Since $(t_1 + t_2) \in \mathbb{R}$ this is satisfied only if $c_2 d_1 = c_1 d_2$, which is equivalent to the requirement that $\frac{c_1}{c_2} = \frac{d_1}{d_2}$. If $d_1 \neq 0$ then $c_2 = \frac{c_1 d_2}{d_1}$, which, with the requirement on the values of t_1 and t_2 leads to further conditions, such as $d_1 > 0$ implies that we need $c_1 < 0$. We illustrate with the following examples:

- (i) $b(t) = (3 - 6i)t - 1 + 2i$ (see Figure 11.20).
- (ii) $b(t) = (-0.3 - 0.6i)t + 0.2 + 0.4i$ (see Figure 11.21).

The eigenspectra in both of Figures 11.20 and 11.21 resemble those found in the case when $b(t)$ is real but we note that a rotation of the eigenspectra seems to have occurred.

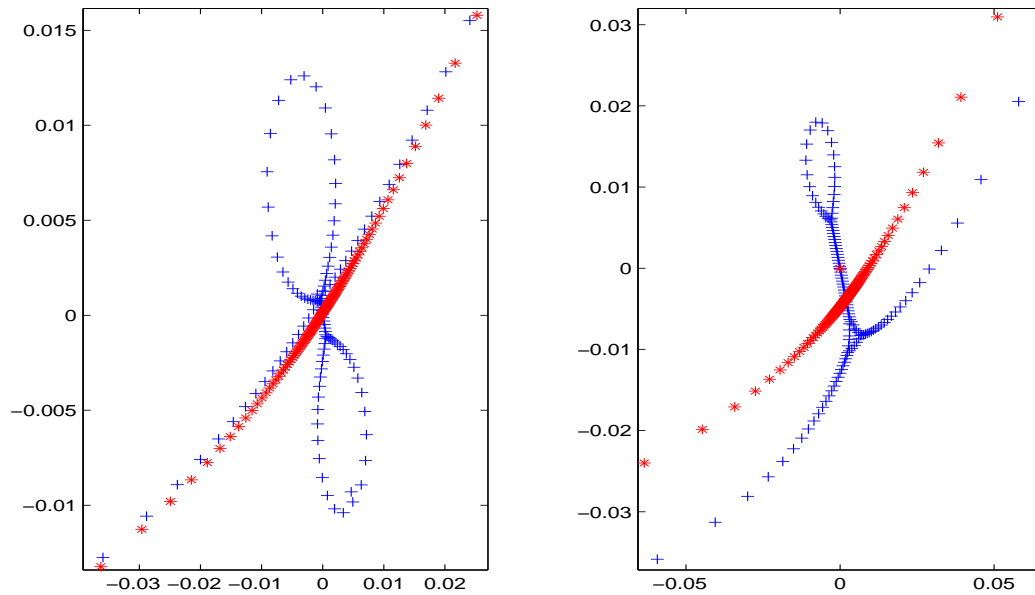


Figure 11.20: $b(t) = (3 - 6i)t - 1 + 2i$. The equation admits small solutions. Left: Trapezium rule. Right: Backward Euler.

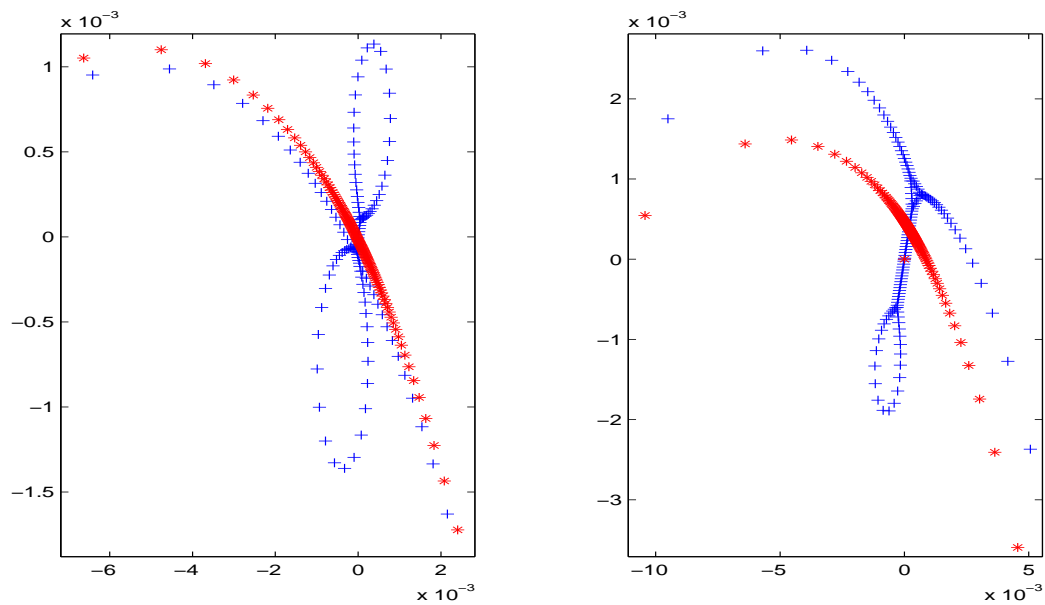


Figure 11.21: $b(t) = (-0.3 - 0.6i)t + 0.2 + 0.4i$. The equation admits small solutions. Left: Trapezium rule. Right: Backward Euler.

Observations

Based on the visual evidence presented and seen in our experimental work there are several characteristic shapes of eigenspectra that we need to be able to interpret as indicating the presence of small solutions to the equation. We illustrate those *discovered* to date in Figures 11.16 to 11.19.

The eigenspectra arising from the trapezium rule are more clearly different to those when the equation does not admit small solutions. However, we might possibly view those produced using backward Euler as more similar to each other.

In the case when $b(t)$ is real, and the trapezium rule is used, the presence of an additional trajectory consisting of two ‘circles’ is an indication that small solutions are present. Based on the diagrams in this section it seems unlikely that a single characteristic feature can be identified (using the same approach) in the case when $b(t)$ is a complex-valued function.

From remark 11.2.2 we know that the functions $b(t)$ used in the examples in this section are not invertible.

11.4.3 The question of invertibility

We have commented earlier on the problem of deciding whether or not $f : \mathbb{R} \rightarrow \mathbb{R} \times \mathbb{R}$ is an invertible function. In this section we present illustrative examples of eigenspectra that arise from equations where $b(t)$ does not satisfy either the condition for no small solutions or the sufficient condition for small solutions to exist.

For all the examples in this section $b(t)$ is a trigonometrical function of the form $b(t) = \sin(2\pi t + d_1 + d_2i) + c_1 + c_2i$ where $c_1, c_2, d_1, d_2 \in \mathbb{R}$.

In Figures 11.22, 11.23 and 11.24 the eigenspectra arise from problems where non-zero $(t_2 - t_1)$ cannot be found to satisfy equation (11.9). Comparing the eigenspectra with Figures 11.9 to 11.14 we might conjecture that they arise from equations that do not admit small solutions. The eigenspectra in Figures 11.25 and 11.26 arise from functions $b(t)$ where values of t_1 and t_2 can be found, but not within the range given in remark 11.2.1. An inspection of the eigenspectra suggests that they arise from equations that do admit small solutions, that is, from equations where $b(t)$ is not invertible. Research into this case continues.

The case when $k = 1$

Here we present examples of the case when $k = 1$, referred to earlier in remark 11.4.4. We include the examples detailed in Table 11.4. The three eigenspectra are very different.

Question: Do any of these eigenspectra indicate that the equation admits small solutions?

Example	c_1	c_2	d_1	d_2	k	$\cosh(d_2)$	$\sinh(d_2)$	Figure
1	0.5	-0.4	1.3	0.2	4.1874	1.0201	0.2013	11.22
2	0.5	-1	1.3	0.2	22.5043	1.0201	0.2013	11.23
3	1	0.6	2	0.7	1.2603	1.2552	0.7586	11.24
4	0.3	0.4	0	2	0.018522	3.7622	3.62686	11.25
5	-7	0.5	-6	4.1	0.0541	30.1784	30.1619	11.26

Table 11.3: Are small solutions indicated? Is $b(t)$ invertible?

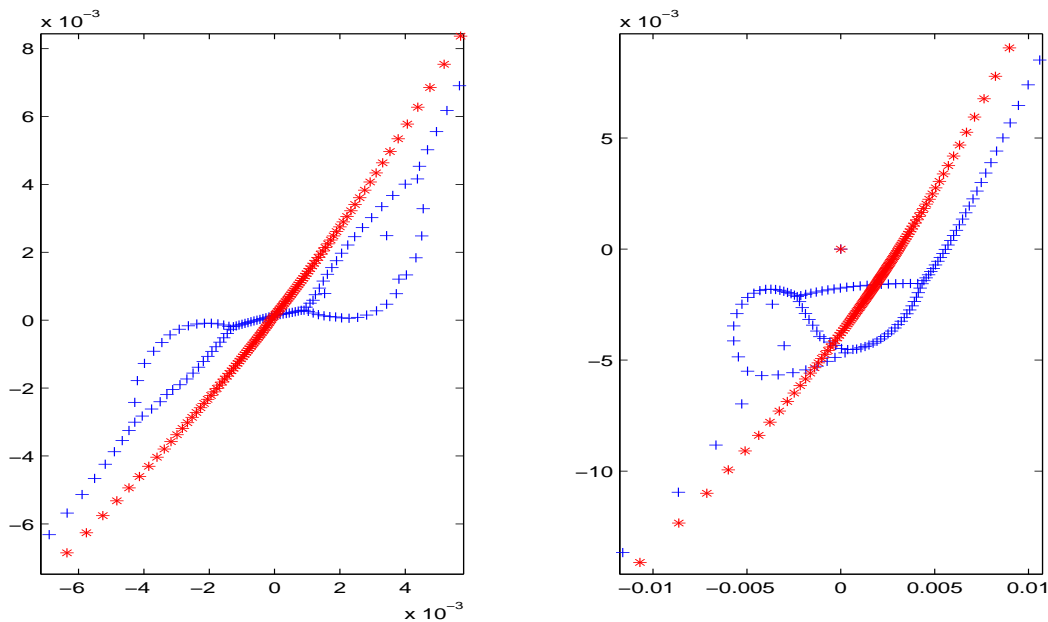


Figure 11.22: Example 1 (Table 11.3).
Left: Trapezium rule. Right: Backward Euler.

We conjecture that, based on earlier eigenspectra, Figures 11.27 and 11.29 indicate that the equation admits small solutions.

Example	c_1	d_1	d_2	k	$\cosh(d_2)$	$\sinh(d_2)$	Figure
1	2	1.3	-1.5	1	2.3524	-2.1293	11.27
2	1.4	1.3	100	1	1.3441×10^43	1.3×10^43	11.28
3	0.4	1.3	0.1	1	1.0050	0.1002	11.29

Table 11.4: The case when $k = 1$.
Are small solutions indicated? Is $b(t)$ invertible?

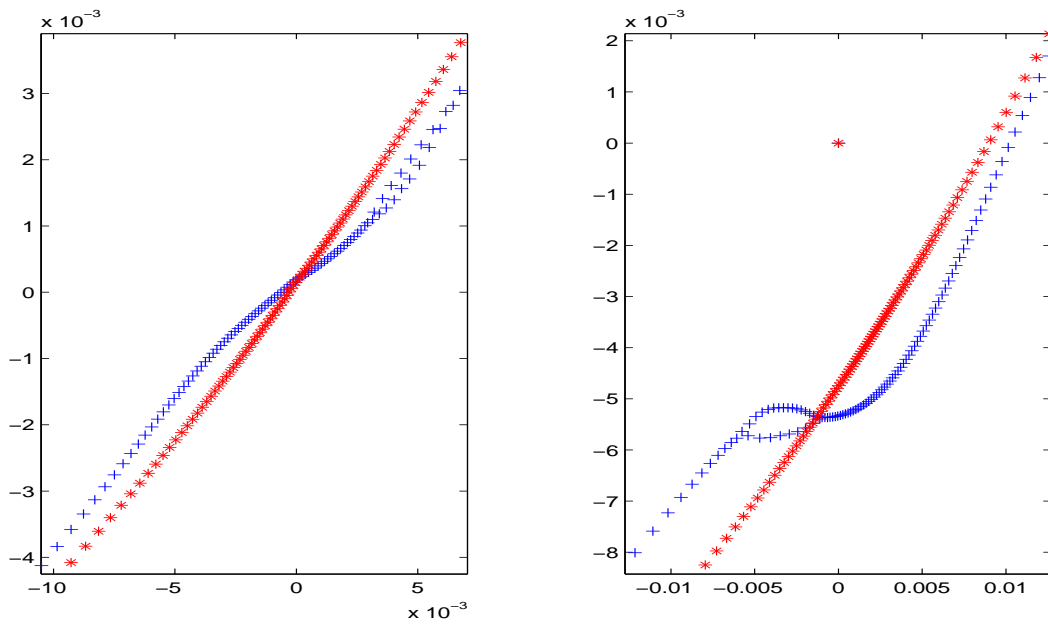


Figure 11.23: Example 2 (Table 11.3).
 Left: Trapezium rule. Right: Backward Euler.

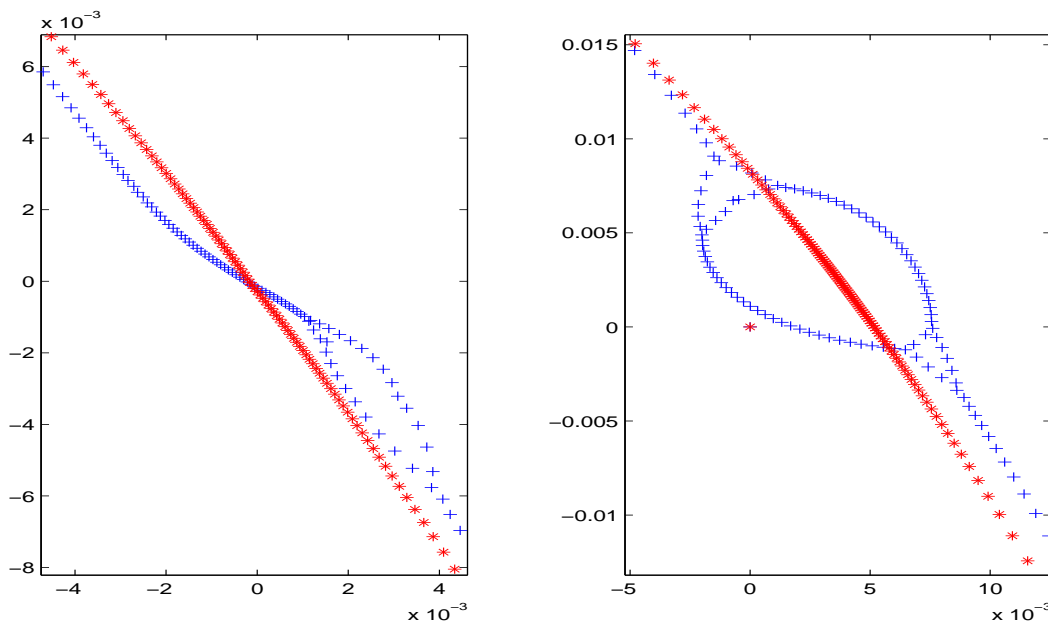


Figure 11.24: Example 3 (Table 11.3).
 Left: Trapezium rule. Right: Backward Euler.

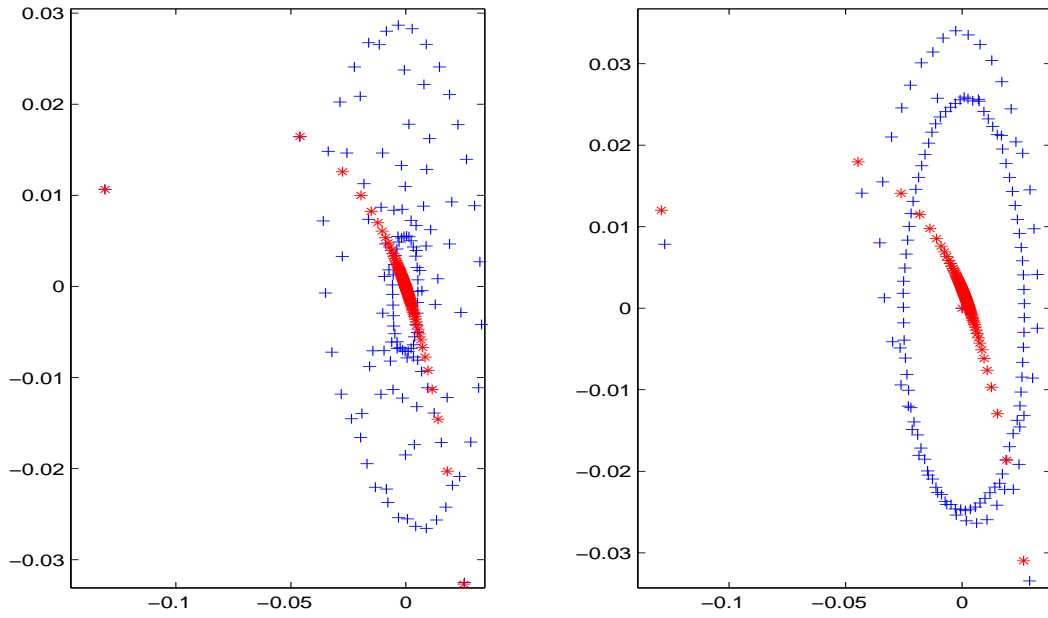


Figure 11.25: Example 4 (Table 11.3).
 Left: Trapezium rule. Right: Backward Euler.

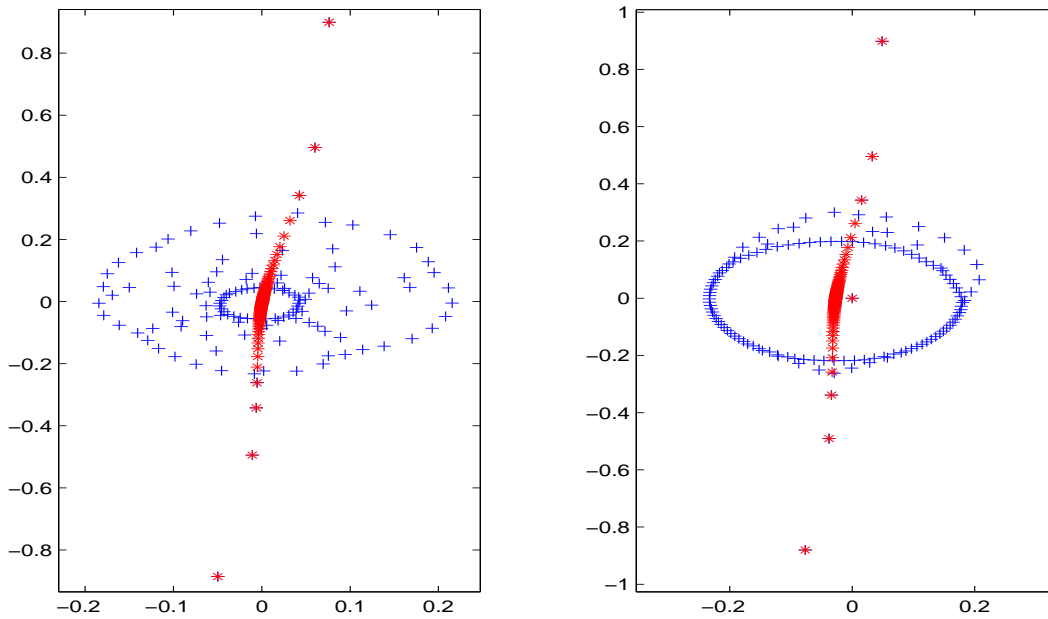


Figure 11.26: Example 5 (Table 11.3).
 Left: Trapezium rule. Right: Backward Euler.

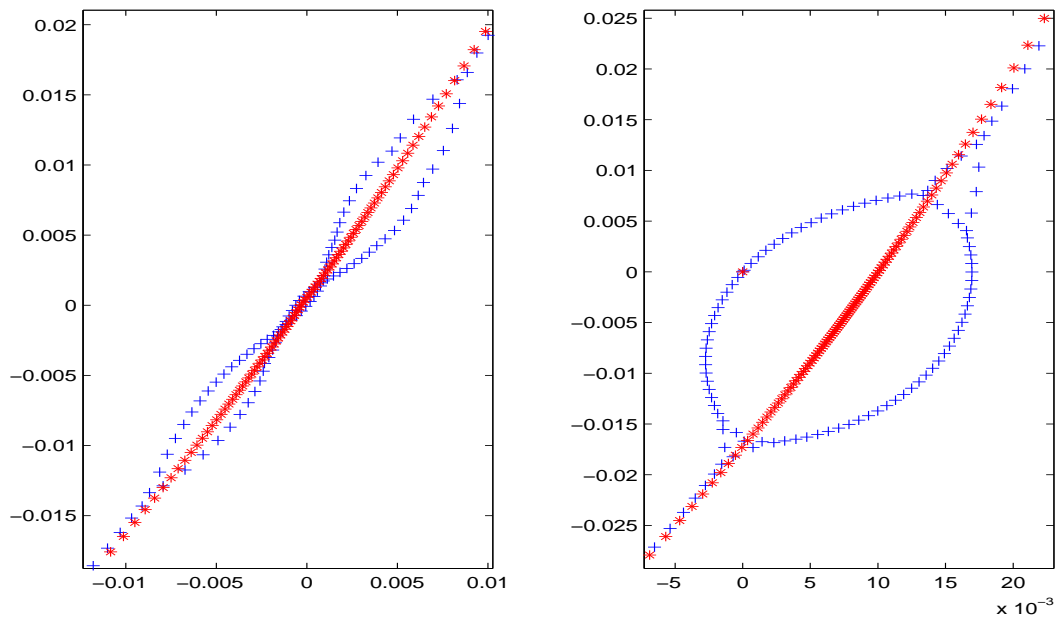


Figure 11.27: Example 1 (Table 11.4).
 Left: Trapezium rule. Right: Backward Euler.

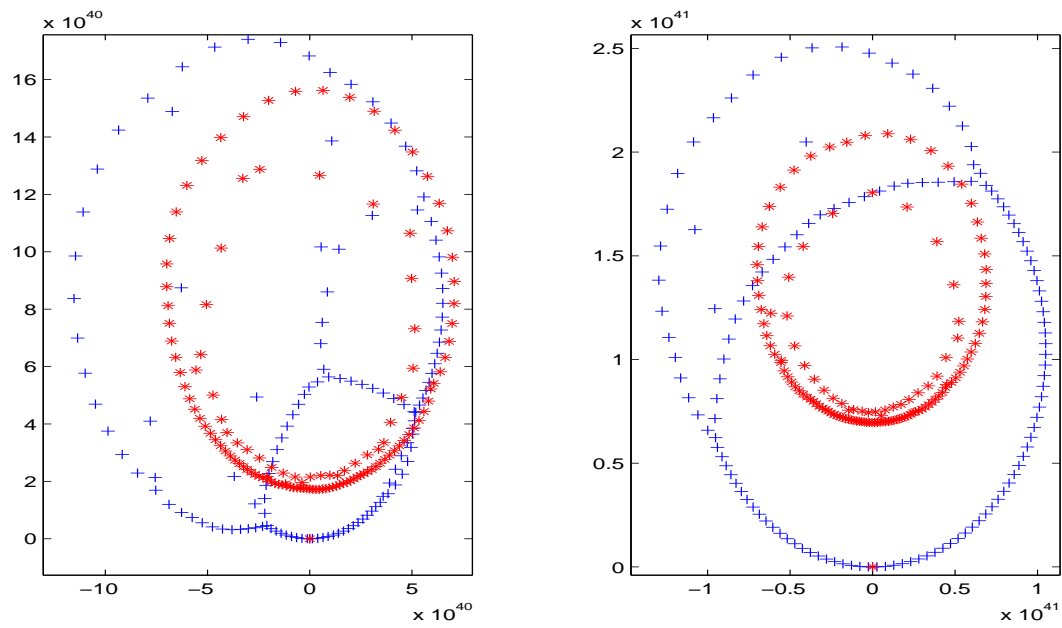


Figure 11.28: Example 2 (Table 11.4).
 Left: Trapezium rule. Right: Backward Euler.

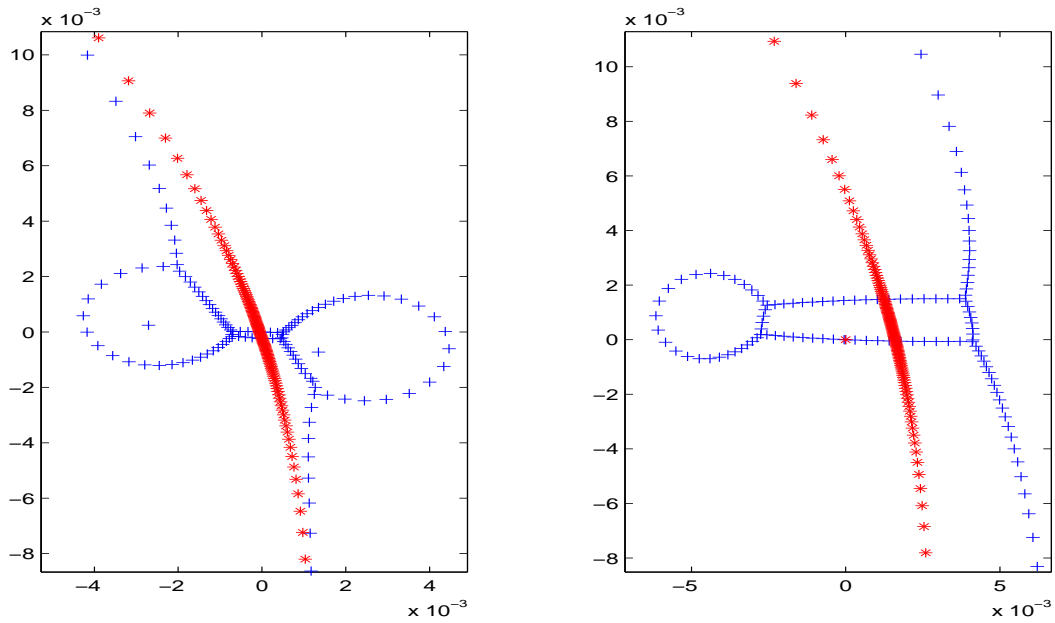


Figure 11.29: Example 3 (Table 11.4).
 Left: Trapezium rule. Right: Backward Euler.

Remark 11.4.5 We note that Theorem 11.2.1 does not imply that equation (11.1) admits small solutions if both the real and the imaginary components of $b(t)$ change sign.

Question: Does the eigenspectra in Figure 11.24 indicate that the equation admits small solutions?

11.4.4 Other observations and investigations

We have also considered whether the step size of $h = \frac{1}{128}$ is the most appropriate to use. Could we improve the clarity of our diagrams by decreasing the step size? Figures 11.30 and 11.31 illustrate the eigenspectra for step sizes $h = \frac{1}{N}$ with $N = 32, 63, 96, 120$. These confirm that we are using an appropriate step size.

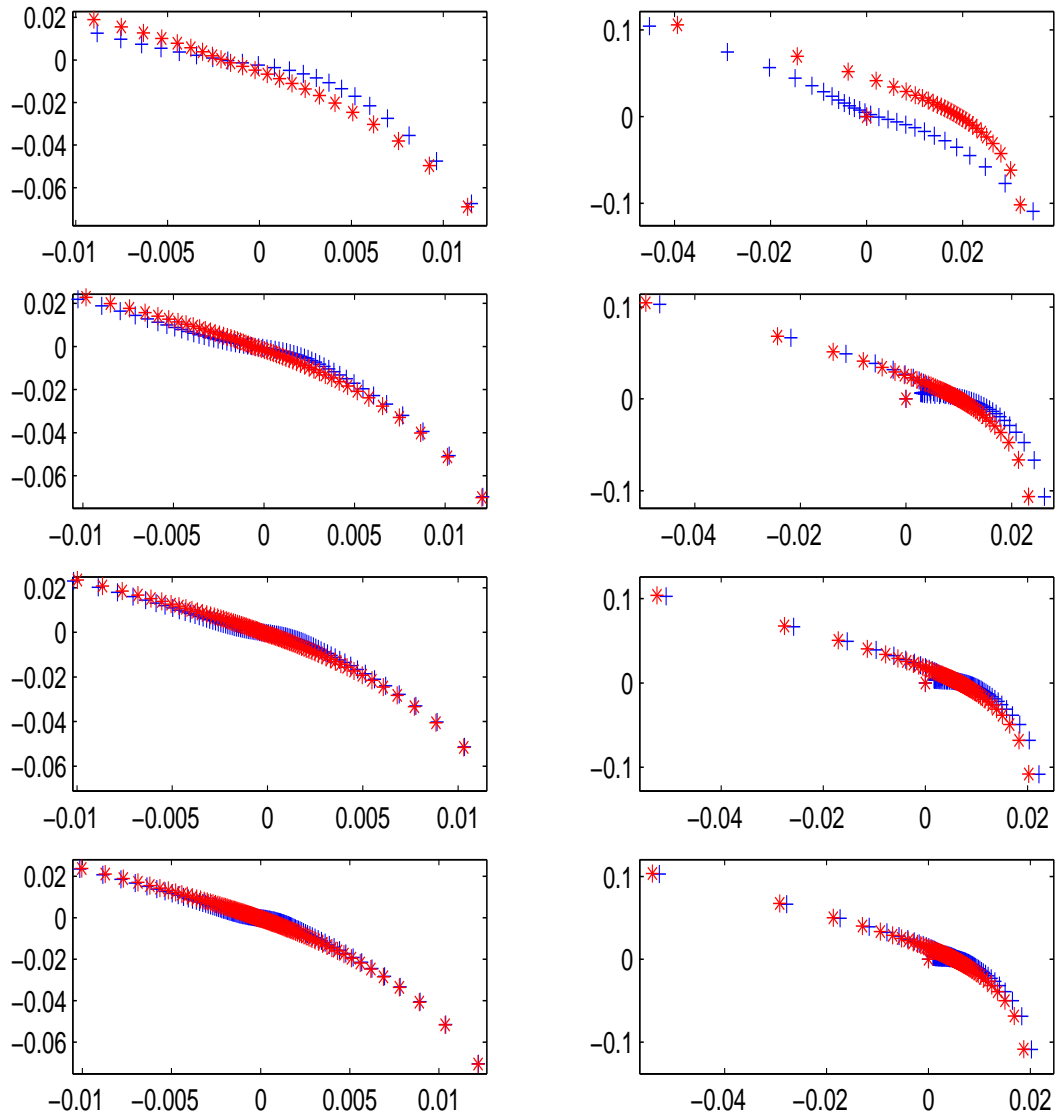


Figure 11.30: Using different step lengths for an equation that does not admit small solutions.

Left: Trapezium rule. Right: Backward Euler.

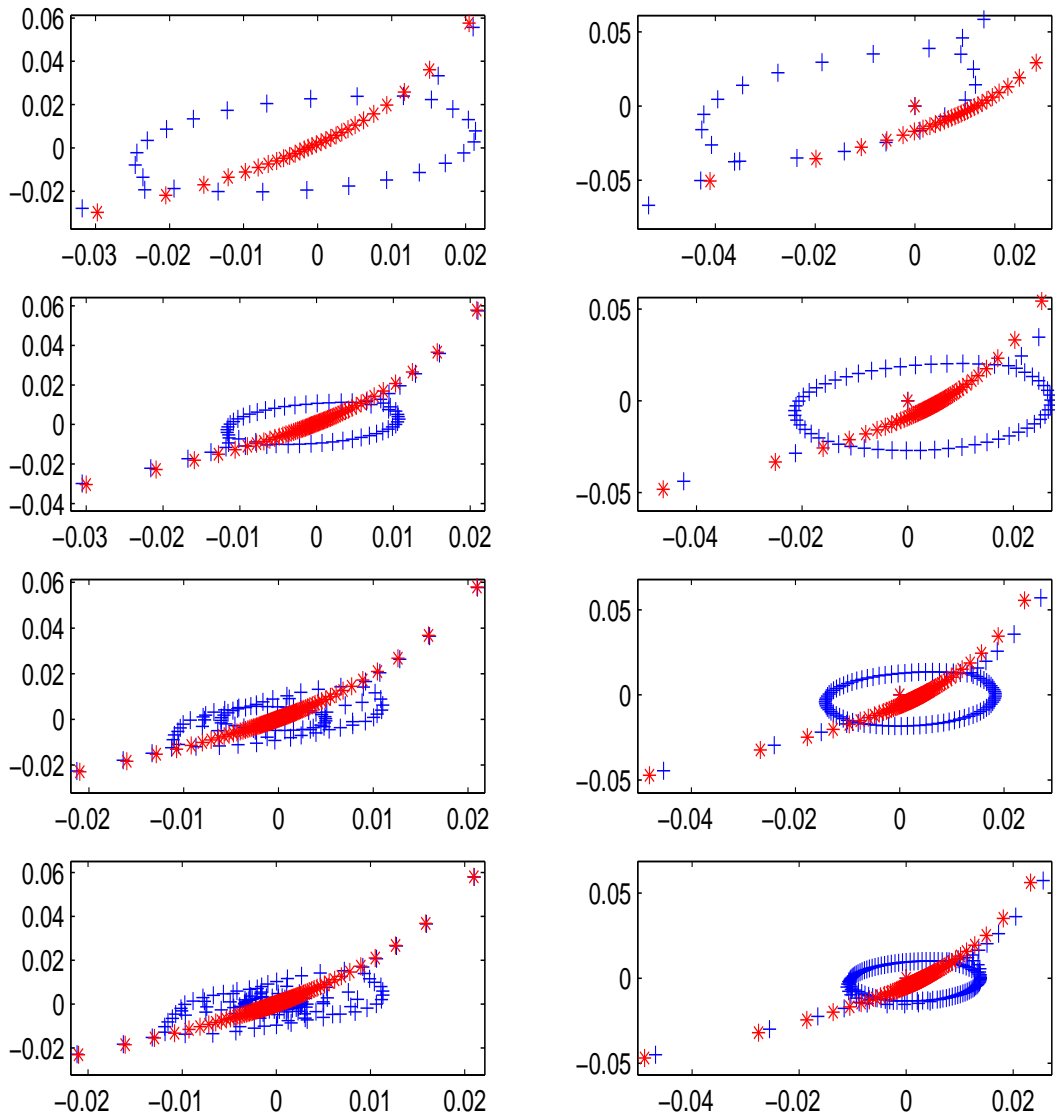


Figure 11.31: Using different step lengths for an equation that admits small solutions.

Left: Trapezium rule. Right: Backward Euler.

Should we plot something different?

Question: Can we find an alternative plot that will ease the detection of small solutions for this case?

We consider plotting the natural logarithm of the eigenvalues arising from our discretisation of the equation. If $z = x + iy$ then a plot of $\ln(z)$ will result in a plot $(\ln|r|, \theta)$ where r and θ are the magnitude and the principal argument of z

respectively, ($r^2 = x^2 + y^2$ and $\theta = \tan^{-1}(\frac{y}{x})$).

We present the resulting log-plots for the cases detailed in Table 11.5 in Figures 11.17 to 11.25, where we also indicate where earlier use of the equation can be found.

Details of the equation	Small solutions?	Original Eigenspectra	The log-plot
Table 11.2	Yes	Figure 11.17	Figure 11.32
Table 11.2	Yes	Figure 11.18	Figure 11.33
Section 11.4.1	No	Figure 11.9	Figure 11.34
Section 11.4.1	No	Figure 11.10	Figure 11.35
Table 11.3	??	Figure 11.22	Figure 11.36
Table 11.3	??	Figure 11.25	Figure 11.37

Table 11.5: Details for the examples of the log-plots

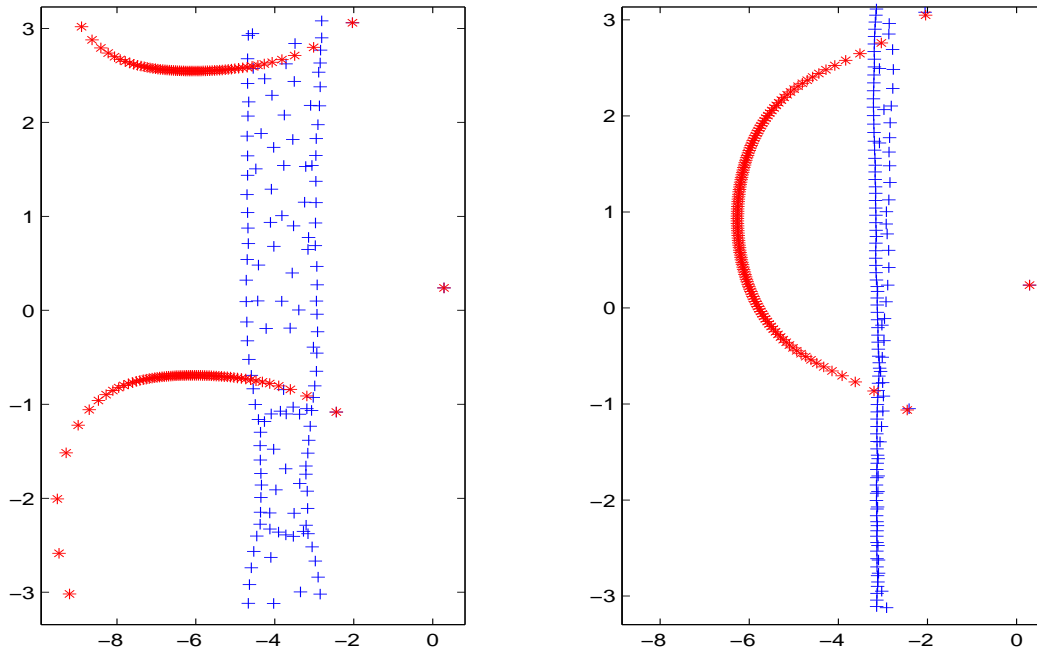


Figure 11.32: Left: Trapezium rule.

Right: Backward Euler.

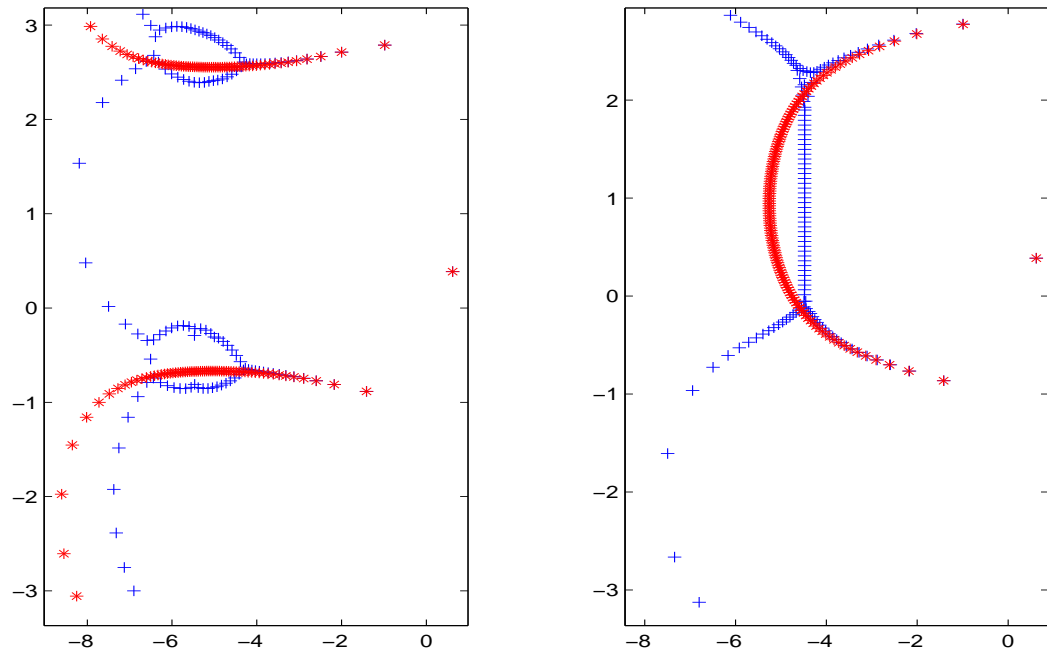


Figure 11.33: Left: Trapezium rule. Right: Backward Euler.

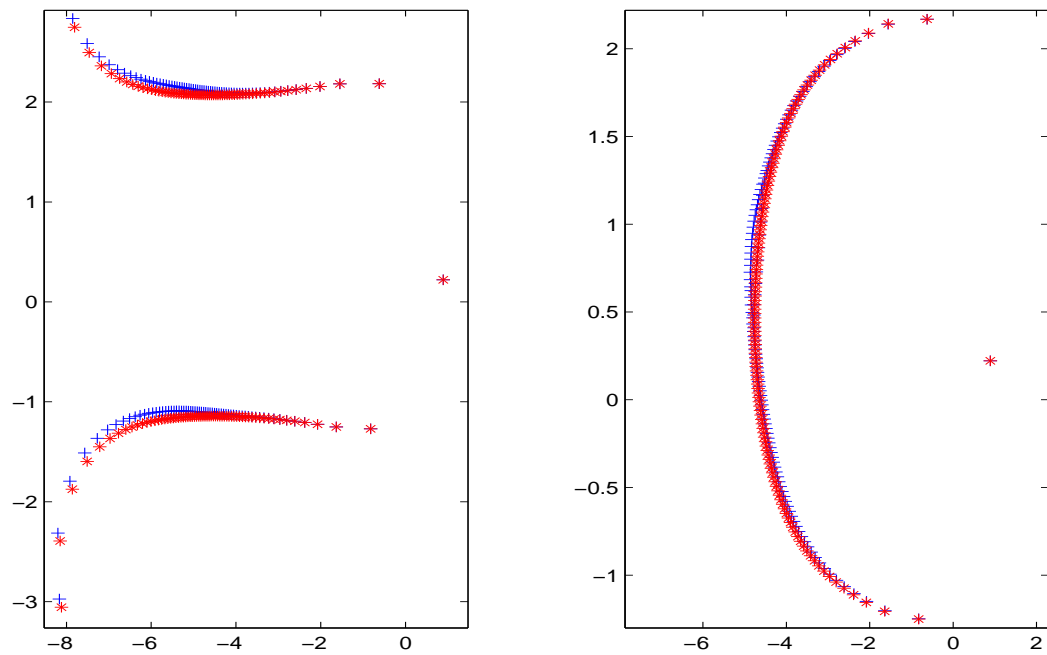


Figure 11.34: Left: Trapezium rule. Right: Backward Euler.

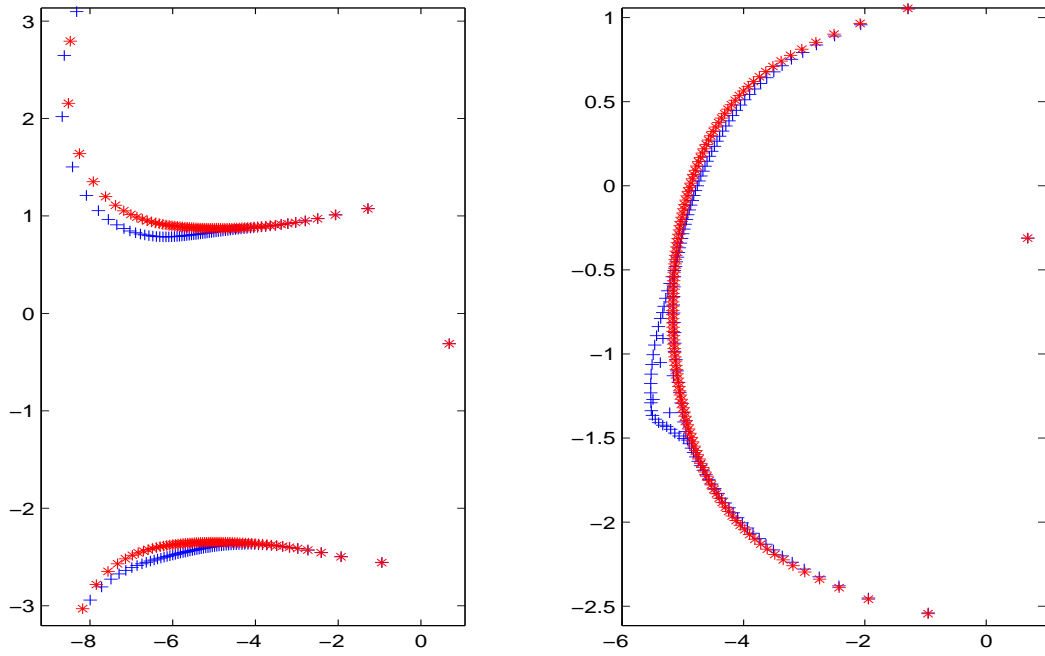


Figure 11.35: Left: Trapezium rule. Right: Backward Euler.

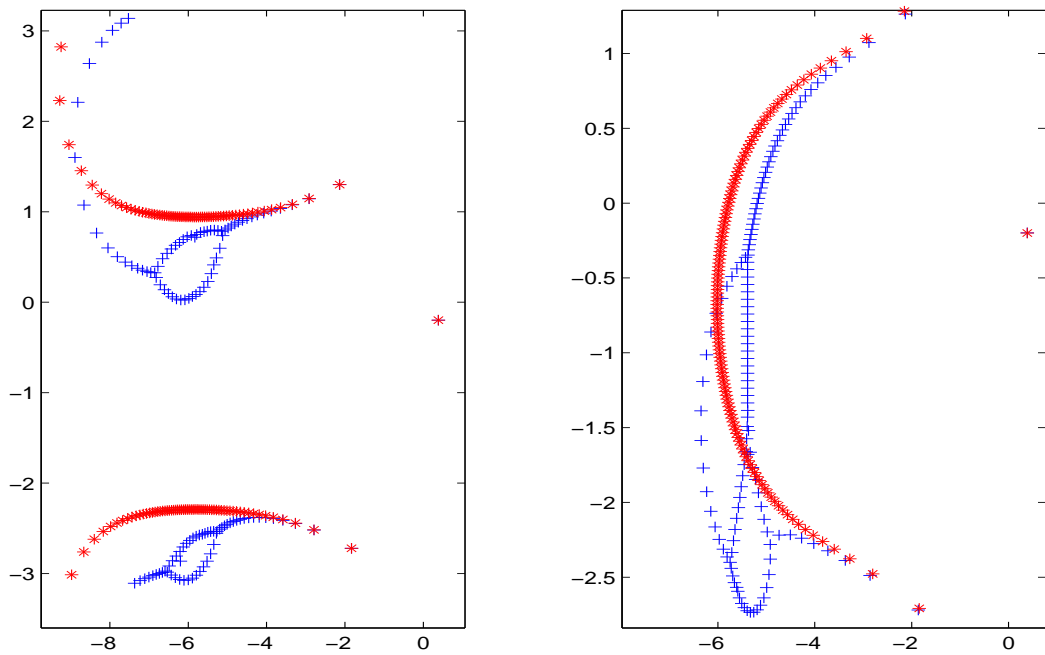


Figure 11.36: Left: Trapezium rule. Right: Backward Euler.

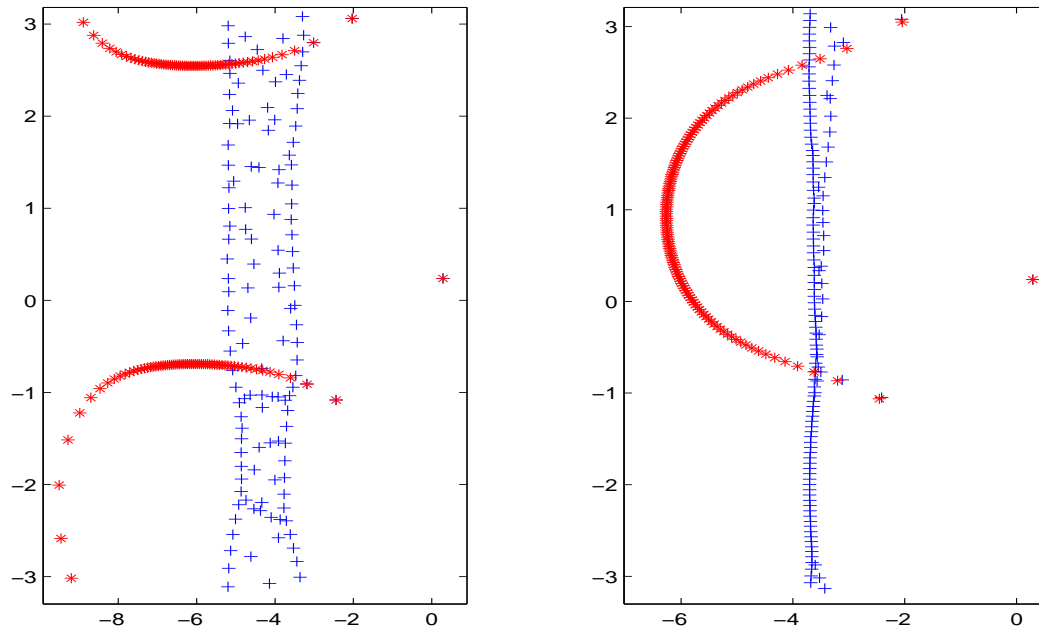


Figure 11.37: Left: Trapezium rule. Right: Backward Euler.

Observations

1. When the equation is known to admit small solutions the values of the arguments for the non-autonomous problem cover the whole range $(-\pi, \pi]$.
2. When the equation is known not to admit small solutions the range of arguments does not cover the whole range $(-\pi, \pi]$ and the ranges are very similar for both the non-autonomous and the autonomous problem.

In the examples considered to date conjectures made based on the evidence from our *usual* approach are supported by the observations above.

Remark 11.4.6 We found that plotting the exponentials of the eigenvalues did not enhance our ability to detect small solutions in this case.

Chapter 12

Summary and conclusions

“Does this delay differential equation admit small solutions?”

The difficulty in detecting small solutions by analytical methods encourages investigation using a numerical approach. In this thesis, following on from earlier work by Ford and Verduyn Lunel [33, 34]:

1. We have adopted a standard approach of the numerical analyst. The success of our method is established using test equations for cases where the analytical theory is known.
2. We have developed analytical results that either support, or are suggested by, our experimental work.
3. We have used the insight gained from computation where analytical theory is known, to inform our experimental work for cases where analytical theory is either less well developed or less readily available. This has enabled interpretation of our results, leading to the formulation of conjectures which, we anticipate, will be both useful to, and inform, the analysts.
4. We have developed a ‘black box’ approach in the form of an algorithm that automates the process of detecting small solutions to a particular class of DDE. The visualisation step, requiring human intervention/interpretation, is removed. An understanding of our methodology is not required by the user.

12.1 Further commentary

We have extended the range of function-type of $b(t)$, from that used in earlier work, for the scalar delay differential equation $x'(t) = b(t)x(t-1)$, $b(t+1) = b(t)$. We have identified characteristics of the resulting eigenspectrum (dependent on properties of $b(t)$), that are indicative of the existence, or otherwise, of small solutions to this class of equation and which lead to an interpretation that is consistent with known analytical theory.

Having achieved success in the numerical detection of small solutions to the scalar DDE with delay and period equal, we then addressed the question of whether the use of an alternative numerical method could enhance the clarity and ease with which decisions (about the existence of small solutions) could be made. Further investigation led to the same, but more informed, choice of numerical method. We then considered other classes of equation where the relevant analytical theory was established. This enabled us to test the success of our approach with a view to using it when the analytical theory is less well developed or less well known.

The numerical detection of small solutions to the classes of scalar DDE considered in chapters 7 and 8 had not been considered previously. We have successfully adapted our method to these classes and have established a connection between the eigenspectra resulting from these equations and those in our earlier work, an important factor with regard to the possible automation of the process of detecting small solutions.

For systems of DDEs, when the eigenvalues of the matrix $A(t)$ in $y'(t) = A(t)y(t-1)$ are always real, we have established that when $A(t)$ is triangular we are able to view the eigenspectra produced as a superposition of eigenspectra seen in the scalar case. However, the situation is more complicated in the case that the eigenvalues of $A(t)$ can be complex. Published analytical theory concerning the existence of small solutions in this case is less readily available. However, our approach has provided further insight and the results of our experimental work has led to the formulation of conjectures. (See, for example, the conclusion of chapter 6).

For scalar DDEs with complex coefficients Guglielmi's heading in [38], regarding the instability of the trapezium rule for this case, necessitated careful consideration and prompted new questions. This motivated the decision to carry out the investigations by applying two numerical methods, one of which is known to be stable and one unstable, to the same problems. In chapter 11 we have used known analytical theory to make progress with the characterisation of the eigenspectra (regarding the existence, or otherwise, of small solutions) for this case. We are currently of the view that the detection of small solutions is not

hindered by the instability of the trapezium rule.

Statistics is a tool not commonly used in this area of research. It was interesting to investigate the possibility of using statistical techniques to determine whether or not an equation admitted small solutions. However, although our conclusion (that we did not find those considered to be more useful in the long term) was a little disappointing the investigation did provide further confirmation of earlier results (such as the difficulty in making a correct decision near to a critical function). In addition, since the motivation for the research which ultimately led to the development of ‘smallsolutiondetector1’ was a ‘by-product’ of the statistical analysis we are pleased to report that, not only was it an interesting viewpoint to consider (that is, that statistics might help), but, that it indirectly influenced the initial development of the ‘black-box’ approach.

Successful detection of small solutions has been achieved through our visualisation of eigenspectra. However, without an understanding of our methodology, appreciation and interpretation of the diagrams produced by our approach is not possible. Automation of the process is both attractive and desirable. The results of our research would then be accessible and usable by a wider mathematical/scientific community. Automation has been achieved for the scalar DDE with delay and period equal, with the development of ‘smallsolutiondetector1’, an algorithm that detects the presence of small solutions to equations of this particular class of DDE. We have extended the algorithm to a class of multi-delay equations and have justified our belief that, with further extensive experimental work, automation of the detection of small solutions is achievable for two further classes of DDE.

The Floquet approach (see chapter 7) has enabled automation of the process to be extended to a class of scalar multi-delay differential equations. An algorithm with wider application, or a collection of algorithms each dealing with particular classes of DDEs, would be both more attractive and useful to users of DDEs. Some of our thoughts concerning possible modifications of, or extensions to, our algorithm to enable automatic detection of small solutions to other classes of DDE have been ‘put into action’. Justification has been provided (see, for example, sections 13.1 and 10.7.1), along with reasons why additional work is needed before further automation can be considered.

In Figure 12.1 we summarise the classes of DDE that we have considered in our research (to date). The term ‘experimental work’ is indicative of the successful detection of small solutions through visual inspection and interpretation of eigenspectra. Table 12.1, used in conjunction with Figure 12.1, identifies relevant chapters or sections in the thesis where further details can be found. We indicate where new results have been established, where conjectures (based on

our experimental work) have been formulated and classes of DDE for which automation has been achieved, or is under consideration. For example, if the reader is interested in the scalar DDE with delay and period equal then, following the flow chart, we see that ‘experimental work’ for this equation is referenced (E1). In Table 12.1 E1 refers the reader to chapters 3, 4, etc.

E1	chapters 3, 4, 5 and 9	E5	section 8.6.2
T1	section 2.5.2, chapters 3 and 4	C5	section 8.6.3
A1	chapter 10	R5	sections 6.2.2 and 6.3.2
E2	chapter 7	E6	sections 13.1, 6.4.4 and 10.7.1
T2	section 7.2	C6	sections 6.4.3 and 6.4.5
A2	section 10.6		
E3	chapter 8	E7	chapter 11
T3	section 8.2	C7	section 11.4.3
R3	sections 8.3 and 8.5	T7	section 11.2
A3	section 10.7.1		
E4	chapter 6	E8	section 8.6.1
R4	sections 6.2.1 and 6.3.1		
T4	section 6.1		
C4	sections 6.4.5 and 6.4.3		
A4	section 13.1		

Table 12.1: A key to Figure 12.1. The location of further details in the thesis.

Research into the detection of small solutions continues. We anticipate that a complete characterisation of the eigenspectra for the scalar equation when $b(t)$ is complex (with respect to the existence or otherwise of small solutions to the equation) will inform research into the systems case discussed in chapter 6. Some possible directions for further work have been indicated in earlier chapters (see, for example, 6 and 10). In chapter 13 we raise other issues that we hope to begin to address in future research, suggesting possible approaches or providing evidence of ongoing research where appropriate.

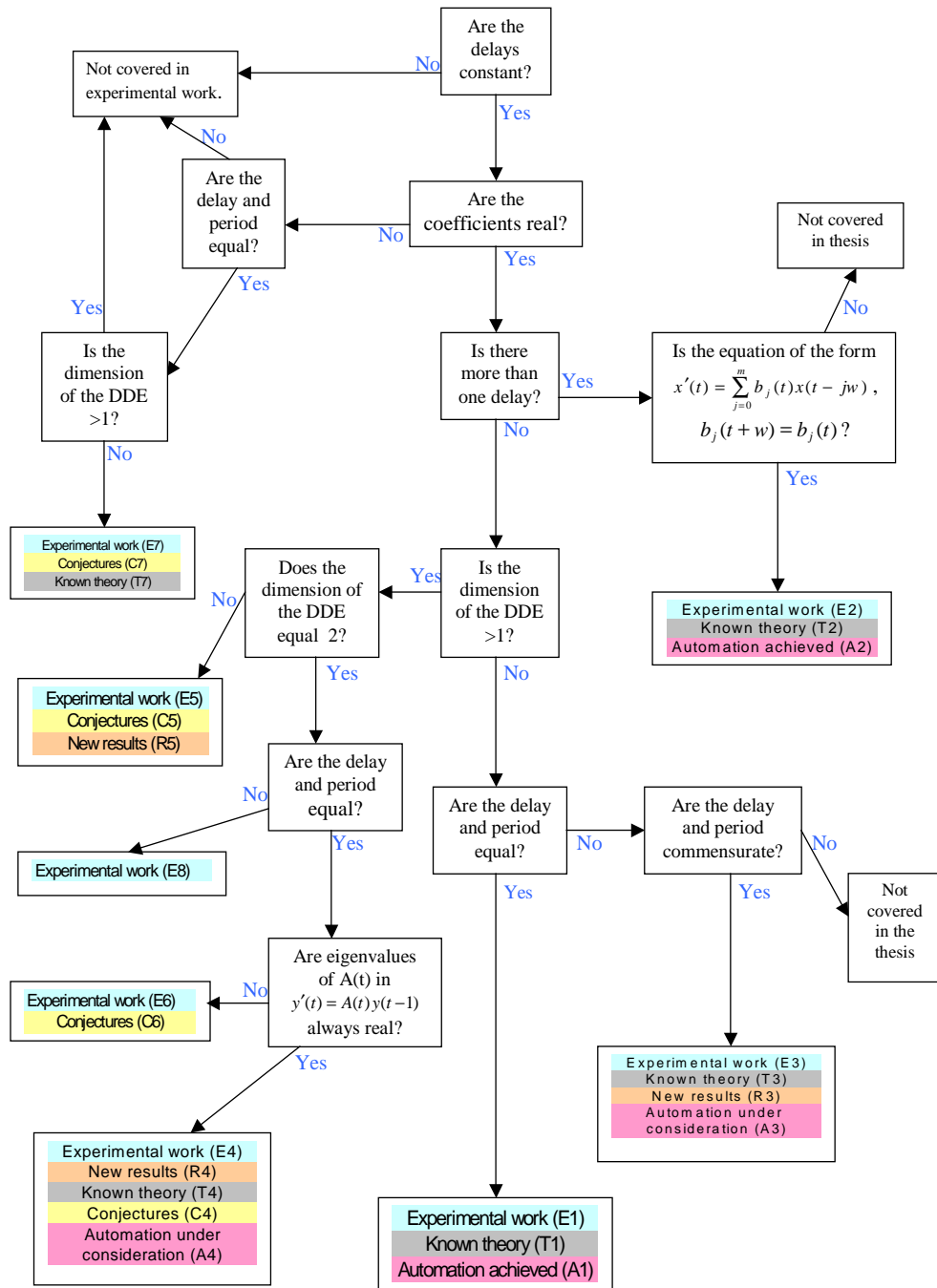


Figure 12.1: Question: “Does my DDE admit small solutions?”
 Can our research help?
 What is known about the existence of small solutions to this DDE?
 Where can details be found?

Chapter 13

Looking to the future

13.1 Is further automation possible?

We present evidence to illustrate how our algorithm might be extended to systems of DDEs of the form (6.2) in which the eigenvalues of $A(t)$ are always real.

Matrix $A(t)$ is triangular (or diagonal)

In section 6.2.1, for the case when $A(t)$ is a diagonal matrix, we noted that the eigenspectra for the two-dimensional case consisted of a superposition of two separate eigenspectra, each arising from a one-dimensional equation. In section 6.3, when $A(t)$ is a triangular matrix, we again noted that the eigenspectra consisted of the superposition of 2 (or more) separate eigenspectra, arising from the functions on the leading diagonal of $A(t)$. This indicates that our algorithm is readily extendable to the case where $A(t)$ is a triangular matrix, with part (ii) of decision tool 9.3.1 replaced by $n(L_1) = k$ for the k -dimensional case. We support this statement with the following illustrative examples.

Example 13.1.1 We consider equation (6.2) with

$$A(t) = \begin{pmatrix} \sin 2\pi t + cuu & 0 \\ 0 & \sin 2\pi t + ctt \end{pmatrix}.$$

In Table 13.1 we present the distribution of the magnitudes of the arguments of the eigenvalues for examples of this equation. The equations in use in the upper section of the table do not admit small solutions whilst those in the lower part do admit small solutions.

Example 13.1.2 We consider equation (6.2) with

$$A(t) = \begin{pmatrix} \sin 2\pi t + cuu & \sin 2\pi t + cut \\ 0 & \sin 2\pi t + ctt \end{pmatrix}.$$

In Table 13.2 we present the distribution of the magnitudes of the arguments of the eigenvalues for examples of this equation. The equations in use in the upper

cuu	ctt	Range of values for α					
		[0, 0.5)	[0.5, 1)	[1, 1.5)	[1.5, 2.5)	[2.5, 3)	[3, π]
2	1.5	2	0	0	252	4	0
1.1	1.8	2	0	0	252	4	0
1.05	1.1	2	0	0	252	4	0
0.5	-0.3	46	54	22	74	44	18
0.1	0.2	36	80	0	88	30	24
1.3	0.7	33	2	0	190	24	9
-1.4	-0.8	25	8	110	88	14	13

Table 13.1: Distribution of eigenvalues associated with example 13.1.1

section of the table do not admit small solutions whilst those in the lower part do admit small solutions.

cuu	cut	ctt	Range of values for α					
			[0, 0.5)	[0.5, 1)	[1, 1.5)	[1.5, 2.5)	[2.5, 3)	[3, π]
1.3	0.4	-1.1	2	4	68	182	2	0
1.9	-3.6	1.5	2	0	0	252	4	0
1.2	0.1	0.6	23	16	0	178	30	11
0.5	-0.3	-1.7	21	28	42	134	24	9
0.3	1.4	0.6	44	46	0	100	46	22
1.3	0.7	33	2	0	190	24	9	
0.05	1.9	-0.1	36	82	14	74	34	18

Table 13.2: Distribution of eigenvalues associated with example 13.1.2

$A(t)$ is not triangular

In the more general case when the eigenvalues of $A(t)$ are always real, but when $A(t)$ is not triangular (or cannot be transformed to a triangular matrix), experimental results suggest that our algorithm can be modified to automate the detection of small solutions for this case. We provide the following illustrative examples.

Example 13.1.3 We consider $y'(t) = A(t)y(t-1)$ with

$$A(t) = \begin{pmatrix} \sin 2\pi t + c & \sin 2\pi t + 0.8 \\ \sin 2\pi t + 1.8 & \sin 2\pi t + 0.7 \end{pmatrix}.$$

We can show that $\det A(t)$ changes sign, and hence the equation admits small solutions, if $c \leq \frac{23}{15}$ or $c \geq \frac{167}{85}$. The distribution of the magnitudes of the arguments, obtained using the approach developed in chapters 6 and 11, is shown

in Table 13.3. For the values of c listed the equation does not admit small

c	Range of values for α					
	[0, 0.5)	[0.5, 1.0)	[1.0, 1.5)	[1.5, 2.5)	[2.5, 3.0)	[3, π]
1.1	27	4	10	194	2	21
1.2	25	2	12	196	2	21
1.3	25	2	14	198	2	17
1.4	21	2	20	198	2	15
1.5	13	4	76	154	2	9
1.6	6	2	78	170	2	0
1.7	6	2	78	170	2	0
1.8	6	4	84	162	2	0
1.9	6	4	120	126	2	0
2.0	23	20	40	152	16	7
2.1	17	36	6	174	16	9
2.2	23	30	0	176	20	9
2.3	23	26	0	178	22	9
2.4	23	22	0	176	28	9
2.5	23	20	0	178	30	7

Table 13.3: Distribution of eigenvalues associated with example 13.1.3

solutions for $c = 1.6, 1.7, 1.8, 1.9$ and admits small solutions for all other values. We make the following observations:

1. The distribution of the eigenvalues is markedly different when the equation admits small solutions.
2. When the equation does not admit small solutions there are no eigenvalues whose arguments have magnitudes in the interval $(3, \pi]$. We note the similarity between this and the scalar case. (Compare with Table 9.6).
3. The number of eigenvalues in the interval $[0, 0.5)$ is constant when the equation does not admit small solutions. The figure of 6 suggests that a refinement of the interval giving the value $n1$ in the algorithm ‘smallsolutiondetector1’, or a modification of the value of $n1$ used in the decision-making process, might be required.

In Tables 13.4 and 13.5 we present examples of the distribution of the magnitudes of the arguments of the eigenvalues arising from discretisation of equations of the form $y'(t) = A(t)y(t - 1)$ (see chapter 6) with

$$A(t) = \begin{pmatrix} \sin 2\pi t + cuu & \sin 2\pi t + cut \\ \sin 2\pi t + ctu & \sin 2\pi t + ctt \end{pmatrix}.$$

In Table 13.5 $\det A(t)$ changes sign and the equation admits small solutions. In Table 13.4 $\det A(t)$ does not change sign and the equation does not admit small solutions.

cuu	cut	ctu	ctt	Range of values for α					
				[0, 0.5)	[0.5, 1)	[1, 1.5)	[1.5, 2.5)	[2.5, 3)	[3, π]
2	0.7	0.5	1.5	2	0	0	252	4	0
1.4	0.8	1.8	0.2	2	6	52	196	2	0
1.6	0.8	1.8	0.7	6	2	78	170	2	0
2	1.3	1.2	0.5	4	4	74	174	2	0
1.5	1.2	1.2	0.5	4	4	60	188	2	0
2	0.8	1.2	1.5	2	0	0	252	4	0

Table 13.4: Distribution of the magnitudes of the arguments of the eigenvalues for examples where the equation does not admit small solutions

cuu	cut	ctu	ctt	Range of values for α					
				[0, 0.5)	[0.5, 1)	[1, 1.5)	[1.5, 2.5)	[2.5, 3)	[3, π]
-2	0.8	1.8	0.7	33	4	36	156	18	11
1.5	0.7	0.5	0.5	25	20	0	182	28	3
2	0.8	1.8	0.7	25	18	40	154	16	5
$\frac{72}{35}$	0.8	1.8	0.7	17	32	17	171	16	5
-0.8	1.2	1.4	0.2	21	24	38	142	22	11
-1.1	0.8	-1.2	1.2	23	6	70	134	16	9

Table 13.5: Distribution of the magnitudes of the arguments of the eigenvalues for examples where the equation admits small solutions

In view of the results of our preliminary experiments, with illustrative examples given in Table 13.4, detecting the non-existence of small solutions based on the number in the last column (0 when no small solutions are admitted) seems very possible. However, in order to detect the existence of small solutions we may need to refine the range of α in the first interval in order to use the value of $n(L_1)$ in our algorithm, or modify the condition $n(L_1) = 1$ in the light of experimental evidence. Further work is needed before automation of the decision-making process in the case when the eigenvalues of $A(t)$ are always real (but $A(t)$ is not triangular) can be considered further.

13.2 Small solutions and other classes of DDE

Non-linear DDEs

For non-linear equations the *usual* approach would be to linearise around zero. However, to do this we often need the condition that there are no small solutions. Cao has shown that, for a particular class of non-linear autonomous scalar equations, if the linearised equation does not admit small solutions then the non-linear equation has no small solutions (see [21] and the references therein).

Question: If the linear periodic DDE has (no) small solutions does this also hold for the non-linear equation if you start near $p(t)$, that is, does $[x(t; \phi) - p(t)]e^{kt} \rightarrow 0$ for all k ? [79]

An open problem [79]

We could consider the equation $x'(t) = b(t)x(t-1)$ with $b(t) = a + \frac{b}{(t+c)^n}$. Analytical knowledge concerning the existence or otherwise of small solutions for this equation is limited.

It is known that:

- the equation does not admit small solutions if $b(t)$ is bounded away from zero.
- if $b(t)$ is real, analytic and approaches zero at infinity then there are no small solutions.

There is no equivalent autonomous problem and the solution map has no eigenvalues. A possible numerical investigation could involve plotting the eigenspectra arising from the product of $N, 2N, 3N$ etc matrices and observing whether there is evidence of changes in the pattern of the eigenspectra.

Mixed equations

For mixed equations of the form $x'(t) = a(t)x(t) + b(t)x(t-1) + c(t)x(t+1)$, that is a functional differential equation involving both retarded and advanced delays, it is known that for no small solutions we need $b(t)c(t) > 0$ for all t [79]. It would be interesting to see whether the ideas and methods used in our research to date could be developed or adapted to gain useful insight into the numerical detection of small solutions of mixed equations.

Appendix A

Matlab code for Smallsolutiondetector1

A.1 Smallsolutiondetector1

The algorithm uses several Matlab m-files. These are:

- definefunctionb
- smallsolutiondetection
- modifiedalgorithm
- reducingtolerance
- decisionchecker
- eigenvaluecalculator

The Matlab codes for these m-files are included as subsections.

```
%This program is called smallsolutiondetector1.
disp('When b(t) is a real-valued periodic function with b(t+p)=b(t)')
disp('this program determines whether the delay differential equation')
disp('x''(t)=b(t)x(t-p) admits small solutions.')
disp('A step length of 1/128 is being used in this algorithm')
disp('NOTE: In your statement of b(t) please enter')
disp('3t as 3*t, -5t as -5*t,')
disp('sin2t as sin(2*t),')
disp('t^2 as t.^2, t^3 as t.^3,')
disp('t(t-1)(t+2) as t.*(t-1).*(t+2), etc.')
```

```
definefunctionb % The user is asked to specify the function b(t)
smallsolutiondetection % The original algorithm is used to make a decision.
modifiedalgorithm % The user is given the option of checking the decision
                  % reached using the modified algorithm.
reducingtolerance % The user is given the opportunity to re-run the modified
                  % algorithm with a reduced tolerance to clarify the decision.
```


A.1.1 definefunctionb

The user is invited to define the function $b(t)$ in their equation.

```
% This program is called "definefunctionb".
N=128;
h=1/N;
n=1:1:N+1;
p=input('State the period/delay p:');
t=p*n*h;
ftext1=input('Give the function b(t):');
ftext=p*ftext1;
```

A.1.2 smallsolutiondetection

A decision is made using the algorithm.

```
% This program is called smallsolutiondetection.
% It uses the original algorithm to make a decision.
for n=1:N
    b=ftext;
end;
for n=N+1:2*N
    b(n)=b(n-N);
end;
A=zeros(N+1,N+1);
A(1,1)=1;
for k=2:N+1
    A(k,k-1)=1;
end;
C=eye(N+1);
for j=1:N
    A(1,N)=h/2*b(j+1);
    A(1,N+1)=h/2*b(j);
    C=A*C;
end;
z=eig(C)+eps*i;
a=angle(z);
m=abs(a);
m1=length(find(m<0.5)); % This establishes the number of eigenvalues
                        % with argument of magnitude less than 0.5
m5=length(find(m<3)); % This establishes the number of eigenvalues
                        % with argument of magnitude less than 3
m6=length(find(m<3.2)); % This establishes the number of eigenvalues
                        % with argument of magnitude less than 3.2
n1=m1;
n6=m6-m5; % This establishes the number of eigenvalues with magnitudes
          % in the range 3 to 3.2
if n6==0
    disp('The equation does not admit small solutions')
    res1=2;
end
```

```

if n6>0
    if n1>1
        disp('The equation admits small solutions')
        res1=1;
    elseif n1==1
        disp('The equation does not admit small solutions.')
        disp('However, you are close to a critical function.')
        res1=3;
    elseif n1==0
        disp('You are close to a critical function.')
        disp('A reliable decision cannot be made by this method.')
        disp('Running the modified algorithm will not be beneficial')
        disp('at the moment.')
        res1=3;
        res2=3;
    end
end
end

```

A.1.3 modifiedalgorithm

The user is given the option of using the modified algorithm to check the answer already given. If the user makes an error in reading and following the instructions at least one opportunity is given to make a correct input before the program proceeds using built-in decisions.

```

% This program is called "modifiedalgorithm".
disp('The modified algorithm can now be used to check the above result')
disp('You can decide whether or not to proceed with the modified algorithm')
disp('Give the answer 1 to proceed with the modified algorithm')
disp('Give the answer 2 if you are satisfied with the above answer')
ans=input('Give your answer:');
if ans==2
    disp('You have decided not to proceed with the modified algorithm')
    res2=0;
elseif ans==1
    disp('You can accept the specified tolerance of 0.0001')
    disp('or set your own tolerance for the problem')
disp('Give the tolerance 1 to accept and 2 to set your own tolerance');
tl=input('Give the tolerance:');
if tl==1
    tol=0.0001;
elseif tl==2
    tol=input('Give your value for the tolerance:');
else
    disp('Your answer must be 1 or 2')
    tl=input('Give the tolerance:');
    if tl==1
        tol=0.0001;
    elseif tl==2
        tol=input('Give your value for the tolerance:');
    else

```

```

disp('Please read the above instructions again.')
disp('Your answer must be 1 or 2')
tl=input('Give the tolerance:');
if tl==1
    tol=0.0001;
elseif tl==2
    tol=input('Give your value for the tolerance:');
else
    disp('The specified tolerance will be used')
    tol=0.0001;
end
end
end
disp('The tolerance being used is')
tolerance=tol
eigenvaluecalculator % calculates the number of eigenvalues with arguments
                    % in the relevant ranges for the functions b(t),
                    % [b(t)+ tolerance] and [b(t)-tolerance].
decisionchecker % checks the decisions made by the algorithm for
                % the three functions mentioned above.
if res2>0
    if res1==res2
        disp('The decisions reached by the two algorithms are the same')
    elseif res1==2
        if res2==3
            disp('The two algorithms are in agreement')
        end
    elseif res1==3
        if res2==2
            disp('The two algorithms are in agreement')
        end
    elseif res2==5
        disp('It is very likely that your function is very near')
        disp('to a critical function')
    else
        disp('The decisions reached by the two algorithms are different.')
        disp('Your function is likely to be near a critical function.')
        disp('A totally reliable decision is not possible using these algorithms')
    end
end
end
else
    disp('Please read the above instructions again.')
    disp('Your answer must be 1 or 2')
    ans=input('Give your answer:');
    if ans==2
        disp('You have decided not to proceed with the modified algorithm')
        res2=0;
    else
        disp('We will proceed with the modified algorithm')
        disp('You can accept the specified tolerance of 0.0001')
        disp('or set your own tolerance for the problem?')
    end
end

```

```

disp('Give the tolerance 1 to accept and 2 to set your own tolerance');
tl=input('Give the tolerance:');
if tl==1
    tol=0.0001;
elseif tl==2
    tol=input('Give your value for the tolerance:');
else
    disp('Your answer must be 1 or 2')
    tl=input('Give the tolerance:');
    if tl==1
        tol=0.0001;
    elseif tl==2
        tol=input('Give your value for the tolerance:');
    else
        disp('Please read the above instructions again.')
        disp('Your answer must be 1 or 2')
        tl=input('Give the tolerance:');
        if tl==1
            tol=0.0001;
        elseif tl==2
            tol=input('Give your value for the tolerance:');
        else
            disp('The specified tolerance will be used')
            tol=0.0001;
        end
    end
end
end
disp('The tolerance being used is')
tolerance=tol
eigenvaluecalculator % calculates the number of eigenvalues with arguments
                    % in the relevant ranges for the functions b(t),
                    % [b(t)+ tolerance] and [b(t)-tolerance].
decisionchecker % checks the decisions made by the algorithm for the
                % three functions mentioned above.

end
if res2>0
    if res1==res2
        disp('The decisions reached by the two algorithms are the same')
    elseif res1==2
        if res2==3
            disp('The two algorithms are in agreement')
        end
    elseif res1==3
        if res2==2
            disp('The two algorithms are in agreement')
        end
    elseif res2==5
        disp('It is very likely that your function is very near to a')
        disp('critical function')
    else

```

```

disp('The decisions reached by the two algorithms are different.')
disp('Your function is likely to be near a critical function.')
disp('A totally reliable decision is not possible using these algorithms')
end
end
end

```

A.1.4 reducingtolerance

The user may be advised to run the program with a reduced tolerance, either one of their own choice or the tolerance built-in to the code.

```

% This program is called "reducingtolerance".
if res2==5
    disp('You can decide whether or not to re-run the modified algorithm')
    disp('Choose one of the following three responses')
    disp('Response 1:- re-run the modified algorithm with the tolerance')
    disp('reduced by a factor of 10')
    disp('Response 2:- re-run the modified algorithm with a tolerance of ')
    disp('your choice')
    disp('Response 3:- do not re-run the modified algorithm')
    rerun=input('Make your choice from the responses 1, 2 or 3:');
    if rerun==3
        disp('You have decided not to re-run the program')
    end
    if rerun==1
        tol=tol/10;
        disp('The tolerance being used is')
    tolerance=tol
    eigenvaluecalculator % calculates the number of eigenvalues with arguments
                        % in the relevant ranges for the functions
                        % b(t), [b(t)+ tolerance] and [b(t)-tolerance].
    decisionchecker % checks the decisions made by the algorithm for the three
                  % functions mentioned above.
if res2>0
    if res1==res2
        disp('The decisions reached by the two algorithms are the same')
    elseif res1==2
        if res2==3
            disp('The two algorithms are in agreement')
        end
    elseif res1==3
        if res2==2
            disp('The two algorithms are in agreement')
        end
    elseif res2==5
        disp('You are advised to re-run the program with a reduced tolerance')
        disp('It is very likely that your function is very near to a ')
        disp('critical function')
    else
        disp('The decisions reached by the two algorithms are different.')
    end
end
end
end

```

```

        disp('Your function is likely to be near a critical function.')
        disp('A totally reliable decision is not possible using these algorithms')
    end
end
end
if rerun==2
    tol=input('Give your value for the tolerance:');
    disp('The tolerance being used is')
    tolerance=tol
    eigenvaluecalculator % calculates the number of eigenvalues with arguments
                        % in the relevant ranges for the functions b(t),
                        % [b(t)+ tolerance] and [b(t)-tolerance].
    decisionchecker % checks the decisions made by the algorithm for the
                  % three functions mentioned above.
    if res2>0
        if res1==res2
            disp('The decisions reached by the two algorithms are the same')
        elseif res1==2
            if res2==3
                disp('The two algorithms are in agreement')
            end
        elseif res1==3
            if res2==2
                disp('The two algorithms are in agreement')
            end
        elseif res2==5
            disp('You are advised to re-run the program with ')
            disp('a reduced tolerance')
            disp('It is very likely that your function is
            disp('very near to a critical function')
        else
            disp('The decisions reached by the two algorithms are different.')
            disp('Your function is likely to be near a critical function.')
            disp('A totally reliable decision is not possible using these algorithms')
        end
    end
end
end
end
end

```

A.1.5 eigenvaluecalculator

```

% This m-file is called eigenvaluecalculator.
% It is used with smallsolutiondetector1
for n=1:N
    b=ftext;
end;
for n=N+1:2*N
    b(n)=b(n-N);
end;
A=zeros(N+1,N+1);
A(1,1)=1;

```

```

for k=2:N+1
    A(k,k-1)=1;
end;
C=eye(N+1);
for j=1:N
    A(1,N)=h/2*b(j+1);
    A(1,N+1)=h/2*b(j);
    C=A*C;
end;
z=eig(C)+eps*i;
a=angle(z);
m=abs(a);
m1=length(find(m<0.5));
m5=length(find(m<3));
m6=length(find(m<3.2));
n1=m1;
n6=m6-m5;
for n=1:N
    b=ftext;
    b1=b+tol;
end;
for n=N+1:2*N
    b(n)=b(n-N);
    b1(n)=b(n)+tol;
end;
A1=zeros(N+1,N+1);
A1(1,1)=1;
for k=2:N+1
    A1(k,k-1)=1;
end;
C1=eye(N+1);
for j=1:N
    A1(1,N)=h/2*b1(j+1);
    A1(1,N+1)=h/2*b1(j);
    C1=A1*C1;
end;
z1=eig(C1)+eps*i;
a1=angle(z1);
mp1=abs(a1);
mpp1=length(find(mp1<0.5));
mpp5=length(find(mp1<3));
mpp6=length(find(mp1<3.2));
np1=mpp1;
np6=mpp6-mpp5;
for n=1:N
    b=ftext;
    b2=b-tol;
end;
for n=N+1:2*N
    b(n)=b(n-N);    b2(n)=b(n)-tol;
end;

```

```

A2=zeros(N+1,N+1);
A2(1,1)=1;
for k=2:N+1
    A2(k,k-1)=1;
end;
C2=eye(N+1);
for j=1:N
    A2(1,N)=h/2*b2(j+1);
    A2(1,N+1)=h/2*b2(j);
    C2=A2*C2;
end;
z2=eig(C2)+eps*i;
a2=angle(z2);
mm1=abs(a2);
mmm1=length(find(mm1<0.5));
mmm5=length(find(mm1<3));
mmm6=length(find(mm1<3.2));
nm1=mmm1;
nm6=mmm6-mmm5;
% disp([nm6 n6 np6 nm1 n1 np1])

```

A.1.6 decisionchecker

```

% This file is called Decisionchecker.
% It is used with Smallsolutiondetector1.
if n6>0
    if np6>0
        if nm6>0
            if nm1>1
                if n1>1
                    if np1>1
                        disp('The equation admits small solutions')
                        res2=1;
                    end
                    if np1==1
                        disp('It is very likely that the equation admits')
                        disp('small solutions')
                        res2=1;
                    end
                end
            end
            if n1==1
                if np1==1
                    disp('It is very unlikely that the equation ')
                    disp('admits small solutions')
                    res2=2;
                end
            end
        end
    end
    if nm1==1
        if n1==1
            if np1>1

```



```

        disp('It is unlikely that the equation admits small solutions')
        res2=2;
        end
    end
end
if nm1==1
    if n1>1
        if np1==1
            disp('It is likely that the equation admits small solutions')
            disp('but you are advised to reduce the tolerance and ')
            disp('re-run the modified algorithm')
            res2=5;
        end
    end
end
if nm1==1
    if n1==1
        if np1==1
            disp('It is unlikely that the equation admits small ')
            disp('solutions but you are near a critical function')
            res2=3;
        end
    end
end
if nm1>1
    if n1==1
        if np1>1
            disp('It is very unliklely that the equation admits ')
            disp('small solutions but you are advised to ')
            disp('re-run the modified algorithm ')
            disp('and reduce the tolerance')
            res2=5;
        end
    end
end
if nm1==1
    if n1>1
        if np1>1
            disp('It is very likely that the equation admits small ')
            disp('solutions but you are near a critical function')
            res2=1;
        end
    end
end
end
end
if n6==0
    if np6==0
        if nm6==0
            disp('The equation does not admit small solutions')
        end
    end
end

```

```

        res2=2;
    end
end
end
if nm6>0
    if n6>0
        if np6==0
            if n1==1
                if np1==1
                    if nm1>1
                        disp('It is unlikely that the equation admits small solutions')
                        disp('but you are near a critical function - a totally ')
                        disp('reliable decision cannot be made by this method')
                        disp('Re-running the modified algorithm with a reduced ')
                        disp('tolerance should clarify the decision')
                        res2=5;
                    end
                    if nm1==1
                        disp('It is very unlikely that the equation admits small ')
                        disp('solutions but you are near a critical function - ')
                        disp('a totally reliable decision cannot be made by ')
                        disp('this method')
                        res2=3;
                    end
                end
            end
        end
        if n1>1
            if np1==1
                if nm1>1
                    disp('It is likely that the equation admits small solutions')
                    res2=1;
                end
                if nm1==1
                    disp('It is likely that the equation admits small solutions')
                    disp('but you are advised to re-run the modified algorithm ')
                    disp('and reduce the tolerance')
                    res2=5;
                end
            end
        end
    end
end
end
if n6>0
    if np6>0
        if nm6==0
            if n1>1
                if np1>1
                    disp('Likely to admit small solutions')
                    disp('but you are near a critical function - a totally ')
                    disp('reliable decision cannot be made by this method')
                end
            end
        end
    end
end

```

```

        res2=1;
    end
    if np1==1
        disp('It is likely to admit small solutions but you are ')
        disp('advised to re-run the modified algorithm and ')
        disp('reduce the tolerance')
        res2=5;
    end
end
if n1==1
    if np1==1
        disp('It is very unlikely that the equation admits small solutions')
        disp('but you are near a critical function - a totally reliable ')
        disp('decision cannot be made by this method')
        res2=3;
    end
    if np1>1
        disp('It is very unlikely that the equation admits small solutions')
        res2=2;
    end
end
end
end
if n6==0
    if nm6==0
        if np6>0
            if np1==1
                disp('It is very unlikely that the equation admits small solutions')
                disp('but you are near a critical function - a totally reliable ')
                disp('decision cannot be made by this method')
                res2=3;
            end
            if np1>1
                disp('It is unlikely that the equation admits small solutions')
                disp('but you are near a critical function - a totally reliable ')
                disp('decision cannot be made by this method')
                res2=3;
            end
        end
    end
end
if n6==0
    if nm6>0
        if np6==0
            if nm1==1
                disp('It is very unlikely that the equation admits small solutions')
                disp('but you are near a critical function - a totally reliable ')
                disp('decision cannot be made by this method')
                res2=3;
            end
        end
    end
end

```

```

        if nm1>1
            disp('It is unlikely that the equation admits small solutions')
            disp('but you are near a critical function')
            disp('you are advised to rerun the modified algorithm and reduce')
            disp('the tolerance to check the decision')
            res2=5;
        end
    end
end
if nm6>0
    if n6==0
        if np6>0
            if n1==1
                if nm1==1
                    if np1==1
                        disp('It is very unlikely that the equation admits ')
                        disp('small solutions')
                        res2=2;
                    else disp('It is unlikely that the equation admits small')
                        disp('solutions but you are advised to re-run the modified')
                        disp('algorithm and reduce the tolerance')
                        res2=5;
                    end
                else disp('It is unlikely that the equation admits small solutions')
                    disp('but you are advised to re-run the modified algorithm ')
                    disp('and reduce the tolerance')
                    res2=5;
                end
            else disp('It is unlikely that the equation admits small solutions')
                disp('but you are advised to re-run the modified algorithm ')
                disp('and reduce the tolerance')
                res2=5;
            end
        end
    end
end
if nm6==0
    if np6==0
        if n6>0
            if n1==1
                disp('It is unlikely that the equation admits small solutions')
                disp('but you are advised to re-run the modified algorithm ')
                disp('and reduce the tolerance')
                res2=5;
            end
            if n1>1
                disp('It is likely that the equation admits small solutions')
                disp('but you are advised to re-run the modified algorithm and ')
                disp('reduce the tolerance')
                res2=5;
            end
        end
    end
end

```

end
end
end
end

Appendix B

Matlab code for 'findanswerchangeoint'

This code was written in connection with testing the reliability of the algorithm `smallsolutiondetector1`. It enables the value of c to be found at which the algorithm's decision changes from 'the equation admits small solutions' to 'the equation does not admit small solutions'.

```
disp('This program finds the interval in which the behaviour of the ')
disp('numerical method changes from producing a YES answer to producing')
disp(' a NO answer to the question ')
disp('"Does the equation admit small solutions?"')
disp('It assumes that we are starting two values, one of which produces ')
disp('the answer YES and the other of which produces the answer NO')
p=160;
for N=32:32:p
h=1/N;
n=1:1:N+1;
format long
cc1=0.4;
cc2=0.5;
while cc2-cc1>0.000000001
ccc=(cc1+cc2)/2;
for n=1:N
    b(n)=sin(2*pi*n*h)+ccc;
end
for n=N+1:2*N
    b(n)=b(n-N);
end
A=zeros(N+1,N+1);
A(1,1)=1;
for k=2:N+1
    A(k,k-1)=1;
end;
C=eye(N+1);
for j=1:N
    A(1,N)=h/2*b(j+1);
```

```

    A(1,N+1)=h/2*b(j);
    C=A*C;
end;
z=eig(C)+eps*i;
a=angle(z);
m=abs(a);
m1=length(find(m<0.5));
m5=length(find(m<3));
m6=length(find(m<3.2));
n1=m1;
n6=m6-m5;
for n=1:N
    bl(n)=sin(2*pi*n*h)+cc1;
end
for n=N+1:2*N
    bl(n)=bl(n-N);
end
Al=zeros(N+1,N+1);
Al(1,1)=1;
for k=2:N+1
    Al(k,k-1)=1;
end;
Cl=eye(N+1);
for j=1:N
    Al(1,N)=h/2*bl(j+1);
    Al(1,N+1)=h/2*bl(j);
    Cl=Al*Cl;
end;
z1=eig(Cl)+eps*i;
a1=angle(z1);
m1=abs(a1);
m11=length(find(m1<0.5));
m15=length(find(m1<3));
m16=length(find(m1<3.2));
n11=m11;
n16=m16-m15;
for n=1:N
    bu(n)=sin(2*pi*n*h)+cc2;
end
for n=N+1:2*N
    bu(n)=bu(n-N);
end
Au=zeros(N+1,N+1);
Au(1,1)=1;
for k=2:N+1
    Au(k,k-1)=1;
end;
Cu=eye(N+1);
for j=1:N
    Au(1,N)=h/2*bu(j+1);
    Au(1,N+1)=h/2*bu(j);

```

```

    Cu=Au*Cu;
end;
zu=eig(Cu)+eps*i;
au=angle(zu);
mu=abs(au);
mu1=length(find(mu<0.5));
mu5=length(find(mu<3));
mu6=length(find(mu<3.2));
nu1=mu1;
nu6=mu6-mu5;
%disp([cc1 n11 n16 ccc n1 n6 cc2 nu1 nu6]);
if n16>0
    if n6>0
        if nu6>0
            if n11>1
                if n1>1
                    if nu1==1
                        cc1=ccc;
                        cc2=cc2;
                    end
                end
            end
        end
        if n11==1
            if n1>1
                if nu1>1
                    cc1=cc1;
                    cc2=ccc;
                end
            end
        end
        if n11>1
            if n1==1
                if nu1==1
                    cc1=cc1;
                    cc2=ccc;
                end
            end
        end
        if n11==1
            if n1==1;
                if nu1>1;
                    cc1=ccc;
                    cc2=cc2;
                end
            end
        end
    end
end
end
end
if n16>0

```



```

    if n6>0
        if nu6==0
            if n1>1
                cc1=ccc;
                cc2=cc2;
            end
            if n1==1
                cc1=cc1;
                cc2=ccc;
            end
        end
    end
end
if n16>0
    if n6==0
        cc1=cc1;
        cc2=ccc;
    end
end
if n16==0
    if n6>0
        if nu6>0
            if n1>1
                cc1=cc1;
                cc2=ccc;
            end
            if n1==1
                cc1=ccc;
                cc2=cc2;
            end
        end
    end
end
if n16==0
    if n6==0
        if nu6>0
            cc1=ccc;
            cc2=cc2;
        end
    end
end
end
end
disp([N cc1 cc2])
end

```

Appendix C

Some relevant theorems

We quote the following theorems (from the reference indicated) for the convenience of the reader.

C.1 Theorem 3.2 from [27]

In the paper the theorem relates to the linear equation

$$(C.1) \quad y'(t) = -\alpha y(t-1), \quad t \geq 0, \quad y(t) = \phi(t), \quad -1 \leq t \leq 0.$$

Let the parameter value $\alpha = \alpha_0$ be fixed and let $z_0 = x_0 + iy_0$ be a characteristic root of equation C.1. With $h > 0$ (chosen so that $h = \frac{1}{m}$ with m some positive integer, as before) we apply a strongly stable linear multistep method (ρ, σ) of order $p \geq 1$ to C.1 to yield a discrete equation that has m characteristic values. Now let $z_h = x_h + iy_h$ be such that $z_h^* = e^{z_h/m}$ is a characteristic value of the discrete equation for which $|e^{z_0} - (z_h^*)^m|$ is minimised. Then $|e^{z_0} - (z_h^*)^m| = O(h^p)$ as $h \rightarrow 0$.

C.2 Theorem 3.1 from [33]

Apply a strongly stable linear multistep method of order $p \geq 1$ to the autonomous delay differential equation

$$(C.2) \quad y'(t) = \alpha y(t - \tau)$$

with characteristic roots that satisfy

$$(C.3) \quad \lambda - \alpha e^{-\tau\lambda} = 0.$$

For each fixed step length $h = (1/m) > 0$ the numerical method has a set S_h of $m + 1$ characteristic roots of the equation

$$(C.4) \quad \lambda^m \rho(\lambda) - h\alpha\sigma(\lambda) = 0,$$

where $\rho(\lambda)$ and $\sigma(\lambda)$ are, respectively, the first and second characteristic polynomials of the linear multistep method being used. Let λ be a root of (C.3) and define d_h to be the distance given by

$$(C.5) \quad d_h = \min_{s \in S_h} |e^\lambda - s^m|$$

then d_h satisfies

$$(C.6) \quad d_h = O(h^p) \text{ as } h \rightarrow 0.$$

Appendix D

Further examples of eigenspectra

In chapter 7 we considered the equation

$$(D.1) \quad \dot{x}(t) = \sum_{j=0}^m b_j(t)x(t - jw).$$

Our eigenspectra resulted from the product of Nmw matrices where w is the period of $b(t)$ and $h = \frac{1}{N}$. Here we use a product of Nw matrices and present eigenspectra arising from some of the problems considered in section 7.3.4, along with two further examples of equation (D.1) with $m = 3$. The following multi-delay equations are used:

Example 1 $x'(t) = (\sin 2\pi t + 0.5)x(t - 1) + (\sin 2\pi t + 1.8)x(t - 2)$.

Example 2 $x'(t) = (\sin 2\pi t + 0.5)x(t - 1) + (\sin 2\pi t + 0.3)x(t - 2)$.

Example 3 $x'(t) = (\sin 2\pi t + 0.6)x(t) + (\sin 2\pi t + 0.3)x(t - 1) + (\sin 2\pi t + 0.2)x(t - 2) + (\sin 2\pi t + 0.7)x(t - 3) + (\sin 2\pi t + 1.4)x(t - 4)$.

Example 4 $x'(t) = (\sin 2\pi t + 1.8)x(t) + (\sin 2\pi t + 1.3)x(t - 1) + (\sin 2\pi t + 1.2)x(t - 2) + (\sin 2\pi t + 1.7)x(t - 3) + (\sin 2\pi t + 0.4)x(t - 4)$.

Example 5 $x'(t) = (\sin 2\pi t + 1.3)x(t - 1) + (\sin 2\pi t + 0.2)x(t - 2) + (\sin 2\pi t + 1.4)x(t - 3)$.

Example 6 $x'(t) = (\sin 2\pi t + 1.4)x(t - 1) + (\sin 2\pi t + 1.8)x(t - 2) + (\sin 2\pi t + 0.3)x(t - 3)$.

We observe the clear presence of additional trajectories in the the right-hand diagrams of Figures D.1, D.2 and D.3 when the equation is known to admit small solutions. We note the correspondence between the value of m and the number of ‘pairs of circles’. In general less computational time is needed and the diagrams are effective as tools for detecting the existence, or otherwise, of small solutions to the equation. However, the diagrams in chapter 7 are symmetrical about the real axis and hence are potentially more useful with regard to an extension of the algorithm that we presented in chapter 10 to multi-delay equations.

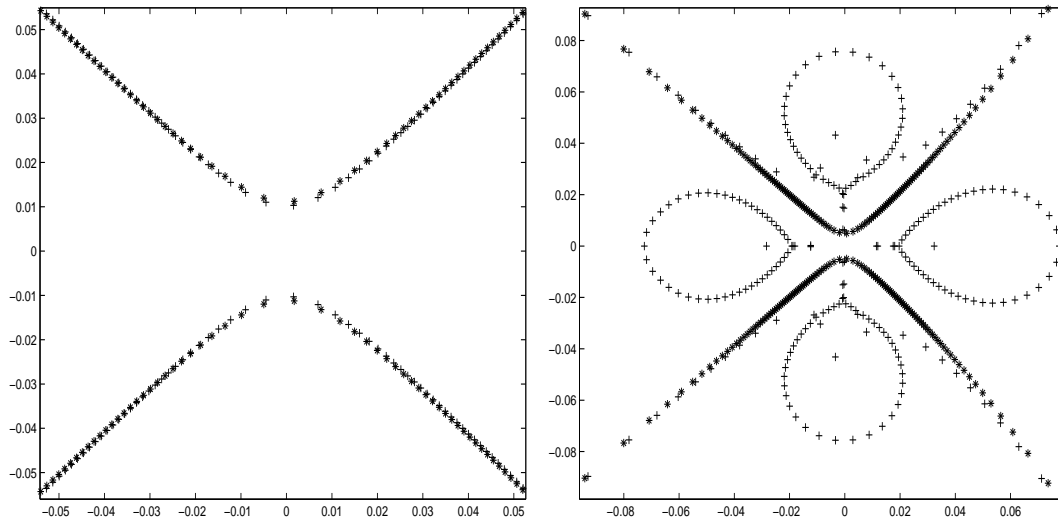


Figure D.1: Left: Example 1 Right: Example 2

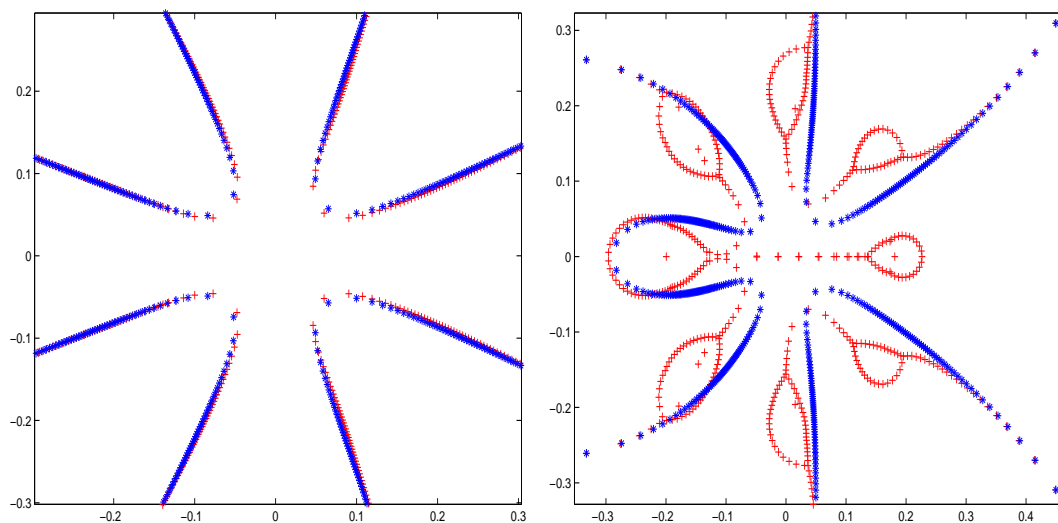


Figure D.2: Left: Example 3 Right: Example 4

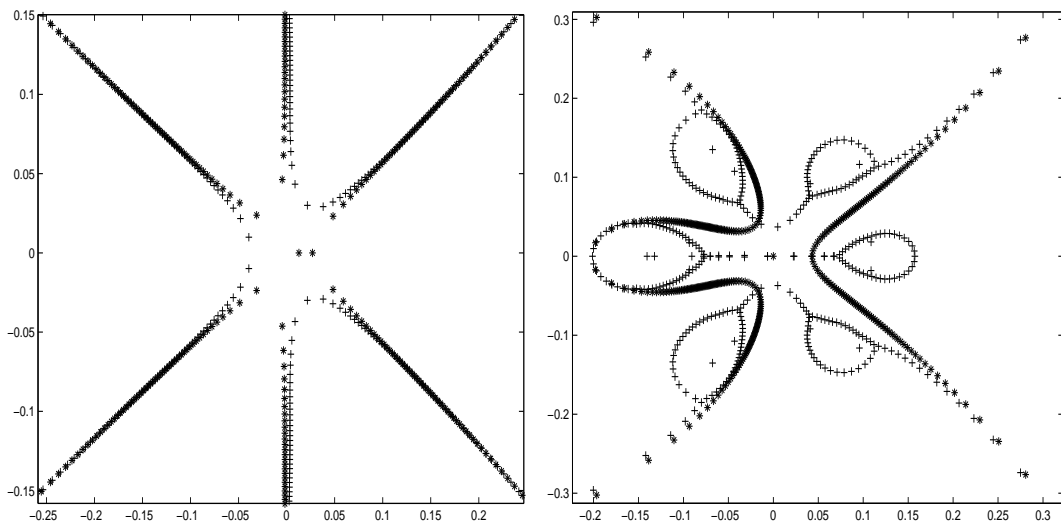


Figure D.3: Left: Example 5 Right: Example 6

Appendix E

The first generation of the algorithm

The first generation of the algorithm was based only on the number of eigenvalues with magnitudes lying in the interval $(3, \pi]$, a result of 0 indicating that the equation does not admit small solutions and a value > 0 indicating that the equation admits small solutions.

The reliability of the algorithm was considered in the same way as was indicated in section 10.4 and, in Table E.1, we present values of c at which the algorithm's decision changes for the same three functions and step lengths.

	$b(t) = t - 0.5 + c$		$b(t) = t(t - 0.5)(t - 1) + c$		$b(t) = \sin(2\pi t) + c$	
CV	$c = 1$		$c = \frac{\sqrt{3}}{36}$		$c = \frac{1}{2}$	
N	Actual	$ Error $	Actual	$ Error $	Actual	$ Error $
32	0.492025	0.007975	0.049943	0.001831	1.022162	0.022162
64	0.496766	0.003234	0.048787	0.000674	1.008510	0.008510
96	0.498115	0.001885	0.048503	0.000390	1.004803	0.004803
128	0.498726	0.001274	0.048376	0.000263	1.003194	0.003194
160	0.499066	0.000934	0.048305	0.000193	1.002327	0.002327

Table E.1: Values of c at which the decision changes for the first generation of the algorithm.

NB. CV = the value of c which gives the critical function.

Remark E.0.1 For the function $b(t) = t - 0.5 + c$ the reliability does actually decrease with the current algorithm. We present Table E.2 to explain why this has occurred. However, we are using the value $n1 = 1$ to help to identify when small solutions do not occur. This is illustrated for the function $b(t) = \sin(2\pi t) + c$ in Table E.3. We note that in the current version of 'smallsolutiondetector1' the

errors for $b(t) = t - 0.5 + c$ decrease as $O(h)$, as would be expected, guaranteeing an improvement in reliability with a decrease in step length. This is not the case with the original version. Overall, we observed a significant increase in reliability using the current version of the algorithm, justifying our decision for the modification.

c	$(\alpha < 0.5)$ n1	$(3 < \alpha \leq \pi)$ n6	Decision of original algorithm	Decision of current algorithm
0.491	2	1	Correct	Correct
0.492	2	1	Correct	Correct
0.493	1	2	Correct	Incorrect
0.494	1	2	Correct	Incorrect

Table E.2: Explanation for decrease in reliability for $b(t) = t - 0.5 + c$

c	$(\alpha < 0.5)$ n1	$(3 < \alpha \leq \pi)$ n6	Decision of original algorithm	Decision of current algorithm
1.000	1	3	Incorrect	Correct
1.001	1	4	Incorrect	Correct
1.002	1	2	Incorrect	Correct
1.003	1	2	Incorrect	Correct
1.004	1	0	Correct	Correct

Table E.3: Explanation for increase in reliability for $b(t) = \sin(2\pi t) + c$

Appendix F

Preservation of the property of admitting small solutions

We are identifying the existence of small solutions by studying the eigenvalues of a matrix C . When the eigenvalues of C are real then small solutions exist if $\det C$ changes sign, that is takes the value zero for some t . Matrix theory states that the eigenvalues of a matrix are invariant under a similarity transformation [81].

We find that the property of possessing, or not possessing, small solutions is preserved by the similarity transformation that transforms the matrix in the autonomous problem (with which we make comparison of the eigenspectra), to a diagonal matrix or a Frobenius matrix (depending upon whether or not the matrix is defective) when applied to the matrix of the non-autonomous problem.

We compare the eigenspectra arising from the equation $y'(t) = A(t)y(t-1)$ with that arising from $y_1'(t) = \hat{A}y_1(t-1)$. If the matrix \hat{A} is non-defective with eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_n$ then we can find a non-singular matrix H such that

$$H^{-1}\hat{A}H = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n).$$

Let $D_1 = H^{-1}\hat{A}H$. Since H is non-singular

$$\det(\hat{A}) = 0 \quad \Leftrightarrow \quad \det(D_1) = 0.$$

If the matrix A is defective then there exists a similarity transformation with matrix H such that the matrix $H^{-1}\hat{A}H$ consists of simple Jordan submatrices isolated along the diagonal with all other elements equal to zero. $H^{-1}\hat{A}H = J$ is a Jordan canonical form.

If

$$H = \begin{pmatrix} H_1 & 0 & \dots & \dots & 0 \\ 0 & H_2 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \dots & \dots & 0 & H_k \end{pmatrix}$$

where each H_i is non-singular for $i = 1, \dots, k$ then $\det(H) \neq 0$ [81].

Let

$$J = \begin{pmatrix} G_1 & 0 & \dots & \dots & 0 \\ 0 & G_2 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \dots & \dots & 0 & G_k \end{pmatrix}$$

be such that $H^{-1}AH = J$.

$\det(J) = 0 \Rightarrow \det(G_i) = 0$ for some i .

$\Rightarrow \lambda_i = 0$ for some i .

Hence, for non-defective matrices:- if the equation admits small solutions then equation $H^{-1}A(t)Hy(t-1) = y'(t)$, where H is the non-singular matrix such that $H^{-1}\hat{A}H$ is $\text{diag}(\lambda_1, \dots, \lambda_n)$, also admits small solutions.

For defective matrices

$$H^{-1}A(t)Hy(t-1) = y'(t)$$

admits small solutions where the non-singular matrix H is such that $H^{-1}\hat{A}H$ is in Jordan canonical form. We illustrate with the following example.

Example F.0.1 If the matrix $A(t)$ is given by

$$A(t) = \begin{pmatrix} \sin(2\pi t) + 0.4 & \sin(2\pi t) - 0.1 \\ \sin(2\pi t) + 0.2 & \sin(2\pi t) + 0.1 \end{pmatrix}$$

then the the matrix \hat{A} is given by

$$\hat{A} = \begin{pmatrix} 0.4 & -0.1 \\ 0.2 & 0.1 \end{pmatrix}.$$

\hat{A} has distinct eigenvalues of 0.2 and 0.3 and we find that the non-singular matrix H such that $H^{-1}\hat{A}H$ is $\text{diag}(0.2, 0.3)$ is $\begin{pmatrix} 1 & 1 \\ 2 & 1 \end{pmatrix}$.

Applying the same similarity transformation to the non-autonomous matrix $A(t)$ gives

$$H^{-1}A(t)H = \begin{pmatrix} 0.2 & 0 \\ 3 \sin(2\pi t) & 2 \sin(2\pi t) + 0.3 \end{pmatrix}.$$

As predicted by the theory we find that the eigenspectra for the non-autonomous equations

$$y'(t) = A(t)y(t-1) \text{ and } y_1'(t) = H^{-1}A(t)Hy_1(t-1)$$

and their related autonomous problems are identical. We note that in this example the equation admits small solutions.

Bibliography

- [1] D. Alboth, Individual Asymptotics of C_0 -Semigroups: Lower Bounds and Small Solutions, *J. Differential Equations* **143** (1998), 221-242.
- [2] U. M. Ascher, L. R. Petzold, *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*, SIAM, 1998.
- [3] N. V. Azbelev, P. M. Simonov, *Stability of Differential Equations with Aftereffect*, Taylor & Francis, 2003.
- [4] C. T. H. Baker, C. A. H. Paul & D. R. Willé, *Issues in the Numerical Solution of Evolutionary Delay Differential Equations*, MCCM Numerical Analysis Report No. 248, Manchester University, 1994.
- [5] C. T. H. Baker, C. A. H. Paul & D. R. Willé, *A Bibliography on the Numerical Solutions of Delay Differential Equations*, MCCM Numerical Analysis Report No. 269, Manchester University, 1995.
- [6] C. T. H. Baker, Retarded differential equations, *J. Computational and Applied Mathematics* **125** (2000), 309-335.
- [7] C. T. H. Baker, G. A. Bocharov, A. Filiz, N. J. Ford, C. A. H. Paul, F. A. Rihan, A. Tang, R. M. Thomas, H. Tian, and D. R. Willé, *Numerical Modelling by Retarded Functional Differential Equations*, MCCM Numerical Analysis Report No 335, Manchester University, 1998. ISSN 1360 1725.
- [8] C. T. H. Baker, G. A. Bocharov & F. A. Rihan, *A Report on the Use of Delay Differential Equations in Numerical Modelling in the Biosciences*, MCCM Numerical Analysis Report No 343, Manchester University, 1999. ISSN 1360 1725.
- [9] C. T. H. Baker, G. A. Bocharov, J. M. Ford, P. M. Lumb, S. J. Norton, P. Krebs, T. Junt, B. Ludewig, C. A. H. Paul, Computational Approaches to Parameter Estimation and Model Selection in Immunology, In preparation.

- [10] J. Bélair, Lifespans in Population Models: Using Time Delays, In S. Busenberg, M. Martelli (Eds), *Differential Equations Models in Biology, Epidemiology and Ecology*, Proceedings, Claremont 1990, Springer-Verlag Berlin Heidelberg, 1991.
- [11] A. Bellen, M. Zennaro, *Numerical Methods for Delay Differential Equations*, Oxford University Press, 2003.
- [12] R. Bellman, K. L. Cooke, *Differential-Difference Equations*, Academic Press, 1963.
- [13] G. A. Bocharov, F. A. Rihan, Numerical modelling in biosciences using delay differential equations, *J. Computational and Applied Mathematics* **125** (2000), 183-199.
- [14] N. Burić, D. Todorović, Dynamics of delay-differential equations modelling immunology of tumor growth, *Chaos, Solitons and Fractals* **13** (2002), 645-655.
- [15] J. C. Butcher, *Numerical Methods for Ordinary Differential Equations*, John Wiley & Sons, Ltd., 2003.
- [16] Y. Cao, The Discrete Lyapunov Function for Scalar Differential Delay Equations, *J. Differential Equations* **87**, (1989), 365-390.
- [17] Y. Cao, The Oscillation and Exponential Decay Rate of Solutions of Differential Delay Equations, In J. R. Graef, J. K. Hale, editors, *Oscillation and Dynamics in Delay Equations*, American Mathematical Society, 1992.
- [18] C. Chicone, S. M. Kopeikin, B. Mashhoon, D. G. Retzloff, Delay equations and radiation damping, *Physics Letters A* **285**, (2001), 17-26.
- [19] T. Cochrane, P. Mitchell, Small Solutions of the Legendre Equation, *Journal of Number Theory*, **70**, (1998), 62-66.
- [20] K. L. Cooke, G. Derfel, On the Sharpness of a Theorem by Cooke and Verduyn Lunel, *J. Math. Anal. Appl.* **197** (1996), 379-391.
- [21] K. L. Cooke, S. M. Verduyn Lunel, Distributional and Small Solutions for Linear Time-Dependent Delay Equations, *J. Differential and Integral Equations*, **6**, 5, (1993), 1101-1117.
- [22] O. Diekmann, S. A. van Gils, S. M. Verduyn Lunel, H.-O. Walther, *Delay Equations, Functional-, Complex- and Nonlinear Analysis*, Springer Verlag, New York, 1995.

- [23] R. D. Driver, *Ordinary and Delay Differential Equations*, Springer Verlag, New York, 1977.
- [24] K. Engelborghs, D. Roose, Numerical computation of stability and detection of Hopf bifurcations of steady state solutions of delay differential equations, *Adv. Comput. Math* **10** (1999), 271-289.
- [25] Y. A. Fiagbedzi, Characterization of Small Solutions in Functional Differential Equations, *Appl. Math. Lett.* **10** (1997), 97-102.
- [26] Y. A. Fiagbedzi, Finite-Dimensional Representation of Delay Systems, *Applied Mathematics Letters* **15** (2002), 527-532.
- [27] N. J. Ford, *Numerical approximation of the characteristic values for a delay differential equation*, MCCM Numerical Analysis Report No 350, Manchester University, 1999. ISSN 1360 1725.
- [28] N. J. Ford, P. M. Lumb, Numerical approaches to delay equations with small solutions, in E. A. Lipitakis(Ed), *Proceedings of HERCMA 2001*, 1, 101-108.
- [29] N. J. Ford, P. M. Lumb, *Systems of delay equations with small solutions: a numerical approach*, In J. Levesley, I. J. Anderson & J. C. Mason (Eds), *Algorithms for Approximation IV*, University of Huddersfield, 2002, 94-101.
- [30] N. J. Ford, P. M. Lumb, *Small solutions to periodic delay differential equations with multiple delays: a numerical approach*, submitted for publication.
- [31] N. J. Ford, P. M. Lumb, *An algorithm to detect small solutions in delay differential equations*, submitted for publication.
- [32] N. J. Ford, P. M. Lumb, *Detecting small solutions for delay differential equations with delay and period commensurate: a numerical approach*, in preparation.
- [33] N. J. Ford, S. M. Verduyn Lunel, Characterising small solutions in delay differential equations through numerical approximations, *Applied Mathematics and Computation*, **131** (2002), 253-270.
- [34] N. J. Ford, S. M. Verduyn Lunel, *Numerical approximation of delay differential equations with small solutions*, Proceedings of 16th IMACS World Congress on Scientific Computation, Applied Mathematics and Simulation, Lausanne 2000, paper 173-3, New Brunswick, 2000. ISBN 3-9522075-1-9.
- [35] H. Gluesing-Luerssen, *Linear Delay-Differential Systems with commensurate Delays: An Algebraic Approach*, Springer-Verlag Berlin Heidelberg, 2000.

- [36] G. H. Golub, C. F. Van Loan, *Matrix Computations*, The John Hopkins University Press, 1996.
- [37] K. Gopalsamy, *Stability and Oscillations in Delay Differential Equations of Population Dynamics*, Kluwer Academic Publishers, 1992.
- [38] N. Guglielmi, Delay dependent stability regions of θ -methods for delay differential equations, *IMA Journal of Numerical Analysis*, **18** (1998), 399-418.
- [39] E. Hairer, S. P. Norsett, G. Wanner, *Solving Ordinary Differential Equations 1 Nonstiff Problems*, Springer-Verlag, 2000.
- [40] A. Halanay, *Differential Equations*, Academic Press, New York and London, 1966.
- [41] J. K. Hale and S. M. Verduyn Lunel, *Introduction to Functional Differential Equations*, Springer Verlag, New York, 1993.
- [42] L. Hatvani, L. Stachó, On Small Solutions of Second Order Differential Equations with Random Coefficients, *ARCHIVUM MATHEMATICUM (BRNO)* **Tomus 34** (1998), 119-126.
- [43] D. Henry, Small Solutions of Linear Autonomous Functional Differential Equations, *J. Differential Equations* **8** (1970), 494-501.
- [44] T. Hong-Jiong, K. Jiao-Xun, The Numerical Stability of Linear Multi-step Methods for Delay Differential Equations with many delays, *SIAM J.Numer.Anal* **33**(3) (1996), 883-889.
- [45] G. D. Hu, G. D. Hu and S. A. Meguid, Stability of Runge-Kutta methods for delay differential systems with multiple delays, *IMA Journal of Numerical Analysis* **19** (1999), 349-356.
- [46] E. L. Ince, *Ordinary Differential Equations*, Dover, 1956.
- [47] K. J. in't Hout, The stability of θ -methods for systems of delay differential equations, *Annals of Numerical Mathematics* **1** (1994), 323-334.
- [48] A. Iserles and A. Zanna, *A scalpel, not a sledgehammer: Qualitative approach to numerical mathematics*, DAMTP Numerical Analysis Report NA 07, University of Cambridge, 1996.
- [49] A. Iserles, Insight, not just numbers, DAMTP Numerical Analysis Report NA 10, University of Cambridge, 1997.
- [50] W. Just, On the eigenvalue spectrum for time-delayed Floquet problems, *Physica D* **142** (2000), 153-165.

- [51] M. A. Kaashoek, S. M. Verduyn Lunel, Characteristic matrices and spectral properties of evolutionary systems, *Transactions of the American Mathematical Society* **334**, 2, (1992), 479-515.
- [52] A. V. Kim, *Functional Differential Equations*, Kluwer Academic Publishers, 1999.
- [53] V. Kolmanovskii and A. Myshkis, *Applied Theory of Functional Differential Equations*, Kluwer Academic Publishers, Dordrecht, 1992.
- [54] V. Kolmanovskii and A. Myshkis, *Introduction to the Theory and Applications of Functional Differential Equations*, Kluwer Academic Publishers, Dordrecht, 1999.
- [55] Y. Kuang, *Delay Differential Equations With Applications in Population Dynamics*, Academic Press, 1993.
- [56] J. D. Lambert, *Numerical Methods for Ordinary Differential Equations The Initial Value Problem*, John Wiley & Sons Ltd., 2000.
- [57] P. M. Lumb, *A Review of the Methods for the Solution of DAEs*, MSc Thesis, University College Chester, UK, 1999.
- [58] T. Luzyanina, K. Engelborghs, S. Ehl, P. Klenerman, G. Bocharov, Low level viral persistence after infection with LCMV: a quantitative insight through numerical bifurcation analysis, *Mathematical Biosciences* **173** (2001), 1-23.
- [59] J. Mallet-Paret and G. Sell, Systems of Differential Delay Equations: Floquet Multipliers and Discrete Lyapunov Functions, *J. Differential Equations* **125** (1996), 385-440.
- [60] A. Manitius, Completeness and F -Completeness of Eigenfunctions Associated with Retarded Functional Differential Equations, *J. Differential Equations* **35** (1980), 1-29.
- [61] I. V. Melnikova, A. Filinkov, *Abstract Cauchy Problems: Three Approaches*, Chapman & Hall/CRC, 2001.
- [62] P. W. Nelson, A. S. Perelson, Mathematical analysis of delay differential equation models of HIV-1 infection, *Mathematical Biosciences* **179** (2002), 73-94.
- [63] C. A. H. Paul, *Developing a Delay Differential Equation Solver*, MCCM Numerical Analysis Report 204, Manchester University, 1991.

- [64] C. A. H. Paul, *A user guide to ARCHI - An explicit (Runge-Kutta) code for solving delay and neutral differential equations*, MCCM Numerical Analysis Report 283, Manchester University, 1995.
- [65] T. L. Saaty, *Modern Nonlinear Equations*, Dover, 1981.
- [66] L. F. Shampine, S. Thompson, Solving DDEs in MATLAB, *Appl. Numer. Math.* **37** (2001), 441-458.
- [67] D. J. Sheskin, *Handbook of Parametric and Non-parametric Statistical Procedures*, Chapman & Hall, CRC, 2000.
- [68] L. Torelli, Stability of numerical methods for delay differential equations, *J. Computational and Applied Mathematics* **25** (1989), 15-26.
- [69] S. M. Verduyn Lunel, Small Solutions and Completeness for Linear Functional and Differential Equations, in John R. Graef, Jack K. Hale, editors, *Oscillation and Dynamics in Delay Equations*, American Mathematical Society, 1992.
- [70] S. M. Verduyn Lunel, A Sharp Version of Henry's Theorem on Small Solutions, *J. Differential Equations* **62** (1986), 266-274.
- [71] S. M. Verduyn Lunel, Series Expansions and Small Solutions for Volterra Equations of Convolution Type, *J. Differential Equations* **85** (1990), 17-53.
- [72] S. M. Verduyn Lunel, The closure of the generalised eigenspace of a class of infinitesimal generators, *Proceedings of the Royal Society of Edinburgh*, **117A** (1991), 171-192.
- [73] S. M. Verduyn Lunel, About Completeness for a Class of Unbounded Operators, *J. Differential Equations* **120** (1995), 108-132.
- [74] S. M. Verduyn Lunel, Series Expansions for Functional Differential Equations, *J. Integral Equations and Operator Theory*, **22(1)** (1995), 93-122.
- [75] S. M. Verduyn Lunel, Small Solutions for Linear Delay Equations, in [CA] Martelli, Mario (ed) et al, *Differential Equations and applications to Biology and Industry*, Proceedings of the Claremont International conference dedicated to the memory of Stavros Busenberg (1941-1993), Claremont, CA, USA, June 1-4, 1994, Singapore: World Scientific, (1996), 531-539.
- [76] S. M. Verduyn Lunel, Parameter identifiability of differential delay equations, *Int. J. Adapt. Cont. Signal Proc.*, **15** (2001), 655-678.

- [77] S. M. Verduyn Lunel, Spectral theory for delay equations, In A. A. Borichev, N. K. Nikolski (Eds), *Systems, Approximatio, Singular Integral Operators, and Related Topics*, International Workshop on Operator Theory and Applications, IWOTA, Operator Theory: Advances and Applications, **129** (2001), 465-508.
- [78] S. M. Verduyn Lunel, Inverse Problems for Nonself-Adjoint Evolutionary Systems, *Fields Institute Communications*, **29** (2001), 321-347.
- [79] Private communication from S. M. Verduyn Lunel.
- [80] P. Waltman, *A Second Course in Elementary Differential Equations*, Academic Press, Inc., 1986.
- [81] J. H. Wilkinson, *The Algebraic Eigenvalue Problem*, Oxford University Press, 1965.
- [82] M. Zennaro, Delay Differential Equations: Theory and Numerics, In M. Ainsworth, J. Levesley, W. A. Light, and M. Marletta (Eds), *Theory and Numerics of Ordinary and Partial Differential Equations*, Oxford University Press, 1995.