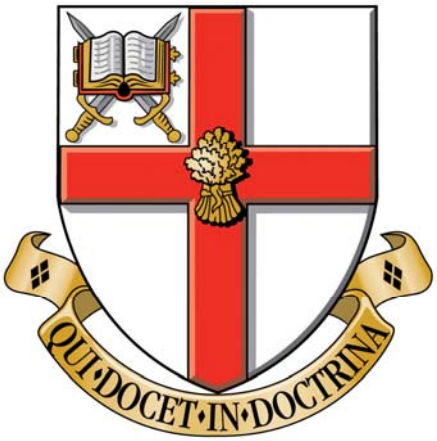


University of Chester

Numerical Analysis of some Integral Equations with Singularities

Thesis submitted in accordance with the requirements
of the University of Liverpool for the degree of
Doctor in Philosophy by Sophy Margaret Thomas

April 2006

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 supervisor, Professor Neville J. Ford, University of Chester, for his help and advice during this project, also Dr. John T. Edwards, Head of Mathematics, University of Chester, for his helpful suggestions, and friends and colleagues at Chester for their support and encouragement.

I would also like to thank Dr. Pedro Lima and Dr. Teresa Diogo of Instituto Superior Tecnico, Lisbon for useful advice, and Professor Ernst Weniger of Universität Regensburg for a most informative discussion regarding extrapolation, at the Algorithms for Approximation V conference at Chester, July 2005.

This work is supported by a College Bursary from the University of Chester.

Abstract

In this thesis we consider new approaches to the numerical solution of a class of Volterra integral equations, which contain a kernel with singularity of non-standard type. The kernel is singular in both arguments at the origin, resulting in multiple solutions, one of which is differentiable at the origin.

We consider numerical methods to approximate any of the (infinitely many) solutions of the equation. We go on to show that the use of product integration over a short primary interval, combined with the careful use of extrapolation to improve the order, may be linked to any suitable standard method away from the origin. The resulting split-interval algorithm is shown to be reliable and flexible, capable of achieving good accuracy, with convergence to the one particular smooth solution.

Contents

Chapter 1

Introduction

1.1 Rationale

This thesis is concerned with a certain class of Volterra integral equations with a non-standard singularity of the kernel. In particular, we are interested in the linear equation

$$u(t) = g(t) + \int_0^t \frac{s^{\mu-1}}{t^{\mu}} u(s)\, ds, \tag{1.1}$$

where we find that the kernel is singular in both arguments.

Equation (1.1) results from a heat conduction problem with non-standard boundary conditions, providing a focus of analytical and numerical development at Instituto Superior Tecnico, Lisbon, and University of Chester, UK, in collaboration. The class of equations includes notably the presence of a singularity in the kernel, of an unusual type, and one which does not conform with the subject as presented by the classical texts.

Our objectives are:

1. to consider the theoretical approach in the context of integral equations,

2. to develop a reliable and accurate means of numerical solution for the case when $\mu < 1$, by separation of the interval of integration into a region close

to the origin, which addresses the singularity, and the remaining extent of the domain, which can be approximated by standard methods.

The main contribution of this thesis is the development of a split-interval approach, which uses extrapolation procedures to raise the order of the primary (first interval) method to match that of the secondary.

Motivation

The numerical solution of equations of the type (1.1) is found for the case $\mu > 1$ in [26] and [64] by means of (a) product integration methods and (b) Hermite collocation, with the expected convergence towards a unique solution. When $0 < \mu < 1$ there is a family of non-smooth solutions, and a single smooth solution, provided the input function g is smooth. Numerical approximation based on the product Euler scheme [44] converges to the smooth solution, but with poor order of convergence, and extrapolation (convergence acceleration) is used to improve the quality of the results. This requires the computation of many trajectories at progressively reduced stepsize, and a more efficient use of computational time is sought which can provide better quality of results.

Singular integral equations

Singularity of the kernel is not a recent phenomenon: probably the earliest integral equation is the Abel equation,

$$g(t) = \int_0^t \frac{y(s)}{\sqrt{t-s}} ds,$$

which we now classify as a Volterra equation of the first kind, (y being the unknown function for which a solution is required). The classical theory on singular integral equations relates to such kernels with weak singularity of the form $(t-s)^{-\alpha},\ 0 < \alpha < 1$, leading to a class of equations which are soluble by use of the Laplace transform. However, equations such as (1.1) do not conform to this pattern, and different treatment is required.

General considerations

The environment of equation (1.1) is straightforward: we are looking at a single equation rather than a system, of linear type, in two dimensions; the variables, dependent and independent, belong to the set of real numbers $\mathbb{R}$, with the independent variable representing time. The integration process is in the main that of the Riemann integral, with reference to the Lebesgue integral when specifically required. We will use the ideas of functional analysis for the more generalised aspects of our discussion.

This class of equations causes us to examine wide-ranging issues, commencing with careful assessment of theorems regarding the existence and uniqueness of solutions, before pursuing the question of applying various numerical methods of solution. We examine the problem in relation to a single equation, linear in form, although much of the theoretical argument can be extended to the non-linear case, to complex values, and to systems of integral equations.

The solution behaviour of this class of equations depends on the value of a parameter $\mu > 0$. When $0 < \mu < 1$, there is a family of solutions, only one of which is of continuity order $m \geq 1$. Previous studies have concentrated on the single smooth solution, whereas our investigation extends to the family of solutions for any given forcing function. We will demonstrate that, once away from the origin, any one of the family of solutions may be uniquely defined as a trajectory whose value is given at some point $t = t_1$, $t_1 \neq 0$. This line of enquiry provides insight into our construction of the split-interval method introduced in chapter 6. Existing approximation methods, as described in [26], [43], [44] or [64] are necessarily restricted in order to overcome the singularity at the origin, but we shall be seeking a neater and more economical approach which may be linked up away from the singularity. A primary aim is to provide a means of solution which allows more flexibility than product integration will allow. Questions of convergence, consistency and stability will be considered, and the effect of

changing the length of the initial interval, and the parameter μ over $0 < \mu < 1$.

The numerical approximation rules which we have selected to apply in this context can be separated into two categories. The first set of schemes is chosen for their potential ability to commence at the origin; the second is a set of standard methods which are well understood in the context of ODE solution, which we apply away from the origin in order to assess whether the order of convergence or any other known characteristics of the method are affected by application to a test equation within the class of equations. The combined scheme employs a carefully constructed acceleration process to improve the quality of the results.

1.2 Derivation

The class of equations which is the topic of this thesis derives originally from a paper by Bartosevich (1975) [9] in which he describes a problem concerning the temperature distribution in two conductors of different lengths and thermal diffusivities. There are certain non-standard boundary conditions attached, resulting in a system of four Volterra-type integral equations with unbounded lower limit. A further transformation yields a representation in operator form, which allows solution by means of operator series. The text of this document is brief, and apart from the concern with the two heat conductors of different diffusivities, we know nothing of the background to which the problem relates. The references consist of two standard texts. This is followed by a further work [10], in which the expansion by means of Watson operators is obtained. The next publication in sequence - also originating in the (then) USSR - is that of Sub-Sizhonenko [63] (1979). The commencement of [63] postulates a single equation in terms of an integral operator, which is a simplified form of one of the Volterra-type equations in [9].

We take as our starting point the equation of Sub-Sizhonenko:

$$\frac{1}{\sqrt{\pi}}\int_{1/x}^{\infty}(xs)^{-1}(\log(xs))^{-1/2}f(s)\,ds+\frac{1}{x}f(1/x)=g(x) \tag{1.2}$$

where $g(x)\in L^2(0,\infty)$. The method of expansion with respect to orthogonal Watson operators suggested by Bartosevich is used to obtain the explicit form of the solution of (1.2) as:

$$f(x)=\frac{d}{dx}\int_0^{1/x}\left\{\int_{-\log xs}^{\infty}\operatorname{erfc}(t^{1/2})dt-\operatorname{erfc}((-\log(xs))^{1/2})\right\}\frac{g(s)}{s}ds+\frac{g(1/x)}{2x},$$

where $f\in L^2[0,\infty]$. The term $\operatorname{erfc}(x)$ has the usual definition of the complementary error function,

$$\operatorname{erfc}(x)=\frac{2}{\sqrt{\pi}}\int_x^{\infty}e^{-t^2}dt.$$

Rooney [56] extends the solution space to $L_{\mu,p},(\mu>0,1\le p<\infty)$ of the Mellin transform space, and provides a simplified solution structure to equation (1.2) as

$$f(x)=\int_x^{\infty}\left[(t/x)\operatorname{erfc}((\log(t/x))^{1/2})-\pi^{-1/2}(\log(t/x))^{-1/2}\right]h(t)dt/t+h(x),$$

$$(x>0).$$

This result is further developed by Lamb [40], who applies a spectral approach to generalised fractional operators in the (complex) Banach space. Up to this time, the methods employed have been highly theoretical, and the equations and their solution formulae unsuited to numerical methods of approximation owing to the singularity of the integrand in each case at the lower limit of integration, and the unbounded upper integration limits.

Tang et al. [64] introduce the substitution $F(t)=f(1/t)$ and $H(t)=h(1/t)$ into equation (1.2), leading to a standard form of the Volterra integral equation

$$F(t)+\frac{1}{\sqrt{\pi}}\int_0^t\frac{1}{\sqrt{\log(t/s)}}.\frac{1}{s}F(s)\,ds=H(t),\qquad(t>0).$$

However, the transformation leads to a further difficulty, in that $F(0)$ is unbounded, except for certain cases of $f(t)$. A further substitution, introducing the arbitrary parameter $\mu > 0$,

$$y(t) := t^{-\mu}F(t), \qquad g_1(t) := t^{-\mu}H(t),$$

leads to the equation form

$$y(t) + \int_0^t \frac{1}{\sqrt{\pi}} \frac{1}{\sqrt{\log(t/s)}} \Big(\frac{s}{t}\Big)^{\mu} . \frac{1}{s} ds = g_1(t). \tag{1.3}$$

We point out that the value of μ is entirely arbitrary in this context, the term $t^{-\mu}$ being a device to remove the limitations on the forcing function.

The derivation and theoretical background of equation (1.3) are given fully in [44] and the references cited therein.

The first use of product integration formulae in this context is applied to the problem of equation (1.3) above. The product Euler and product trapezoidal methods are applied for the case when $\mu > 1$, with the restriction on the input function that $f \in C^m[0, T]$, where $m = 1$ for the product Euler and $m = 2$ for the product trapezoidal scheme, with convergence orders 1 and 2 respectively.

In 1991 Diogo et al. [26] obtain a solution approximation for equation (1.3) when $\mu > 1$ by means of Hermite-type collocation. They show that there is a unique solution to this equation with continuity properties similar to those of the input function g. Further, equation (1.3) is shown in [26] to be analogous to

$$y(t) - \int_0^t \frac{s^{\mu-1}}{t^{\mu}} y(s) ds = g_2(t) \tag{1.4}$$

in the sense that the solution of (1.4) is the same as the solution of equation (1.3) when the input functions g_1 and $g_2$satisfy the following relationship

$$g_1(t) = -\int_0^t \frac{1}{\sqrt{\pi}} \frac{1}{\sqrt{\log(t/s)}} \frac{s^{\mu-1}}{t^{\mu}} g_2(s)\ ds + g_2(t).$$

Our interest is mainly with the simplified form, equation (1.4) (identified with (1.1)), together with the related equation (1.3).

Han (1994) [32] obtains solution formulae for these equations, by means of the related ODE. The cases when $\mu > 1$, $\mu = 1$ and $\mu < 1$ are dealt with separately: the existence of multiple solutions, one particular with at least C^1 continuity is now evident. The theory is extended to include generalised forms of the kernel. These formulae are given in full in Chapter 3.

Lima and Diogo (1997) [43] define an associated equation in which

$$v(t) := t^{\beta} u(t) \tag{1.5}$$

$$\overline{g}(t) := t^{\beta} g(t) \tag{1.6}$$

for the case when $\mu > 1$. Existence and uniqueness of solutions to this are discussed, and the Product Euler method then applied to the transformed equation. The results have a low order of convergence, which is improved by Richardson's extrapolation. The device (1.5/1.6) enables $g(t)$ to take the form $t^{-\alpha} f(t)$, $\alpha > 0$.

A further paper by Lima and Diogo [44] opens up the consideration of numerical methods applied to equation (1.4) when $\mu < 1$. The proofs are realised by dividing the structure of the solution into two parts, separating $g(t)$ into its value $g(0)$ at the origin, and the residual part $g_1(t)$ which passes through the origin. Since the integral equation is linear, the separate solutions may be combined to give the solution of the original equation. This paper and its predecessors provide the foundation for the present investigation.

1.3 Computation

All computer programs for the numerical methods described in this thesis have been written and implemented by the author, using Matlab sv6.5, on a system with Intel Celeron Pentium II processor, and 256 Mb of RAM.

Chapter 2

Integral Equations

Our objective in this chapter is to summarise the existing theory on integral equations, to establish the classifications, to find the criteria for existence and uniqueness of solutions for the Volterra equation, and, where possible, the means of obtaining an analytic solution. The principal sources are Linz [46], Kreyszig [38], Riesz and Sz.Nagy [55], Atkinson [5], Hochstadt [33] and Smithies [61].

The application of integral equations to a wide range of physical problems is well-established, the integral equation allowing greater flexibility than the differential equation, both in structure (for example, the renewal equation), and tolerance of input functions such as the 'step' function, usually described as *generalized functions*. The study of integral equations developed in parallel with that of functional analysis, and the availability of such techniques as normed spaces, measure theory and the Laplace transform, created a comprehensive and adaptable theoretical basis. The background is well-documented, and we do not propose to go into the detail. (see e.g Bernkopf [11]).

The development of integral equations was occurring, more or less simultaneously, with two distinct schools of thought. These have resolved into what we now consider to be main two main classes of integral equation: the Fredholm equation (FIE), and the Volterra equation (VIE).

2.1 Classification

Let $K(t, s, \phi)$ be a known expression (the 'kernel'), and $\psi(t)$ also known (the input term), then we define the Volterra equation of the second kind to be an equation of the form

$$\phi(t) = \psi(t) + \int_a^t K(t, s, \phi(s))ds, \tag{2.1}$$

where ϕ is the unknown function for which a solution is required. The upper limit of integration is variable, and we can usually take the lower limit as zero. If the unknown function does not appear outside the integrand, then the equation is a Volterra equation of the first kind. We make the following further distinctions:

a.) if $K(t, s, \phi(s)) = K(t, s)\phi(s)$, then the equation is termed *linear*

b.) if $K(t, s) = \alpha(t)\beta(s)$ or if $K(t, s) = \sum_i \alpha_i(t)\beta_i(s)$ then the equation is termed *separable* or *degenerate*

Further clarification will be made at the appropriate stage.

We define the Fredholm integral equation of the second kind

$$\phi(t) = \psi(t) + \int_a^b K(t, s, \phi(s))ds, \tag{2.2}$$

with similar description as the Volterra equation, the only difference being that the limits of integration are now fixed. However, the behaviour and treatment of the two is very different, and we shall see that when we require a generalized approach, we find the abstract methods of functional analysis more convenient.

We shall in particular be considering the class of linear Volterra integral equations of the second kind, and the methods we describe in this chapter relate to such equations, except where specifically stated. We draw on the standard methods as described in e.g. [46] and [33], paying particular attention to the conditions, and for now leave aside the implications of the functional analysis approach.

2.2 Existence and Uniqueness of Solutions

The ideal case for the solution of any time-based equation is that there is one and only one solution, which may be expressed as a function, or evaluated numerically to satisfy the discretized equation. The usual result of solving an ODE includes the arbitrary constant of integration, and the unique solution is defined by initial or alternatively boundary values - the initial (IV) (or boundary (BV)) value problem. The constraints placed on an IV problem for a unique solution to exist might be expected to have an exact parallel for the VIE if it is considered as another means of representation of the IV problem. However, the smoothing properties of the integration process, together with the definition of certain function spaces, notably $L^2[a,b]$, have enabled the restrictions to be eased, and the proofs amended accordingly. We take the usual definition of $L^2[a,b]$ as the vector space of all continuous real-valued functions f on $[a,b]$, with norm $\| f \| = \left[\int_a^b | f(x) |^2 \, dx \right]^{1/2}$.

The standard theoretical approach in dealing with the issue of existence and uniqueness of solutions involves restraints on the kernel $K(t,s)$ as well as on the forcing function ψ. The limiting factor may be one of continuity order, or the less restrictive Banach space $L^2[a,b]$, or a Lipschitz condition. We introduce several basic theorems at this early stage, and in Chapter 3 we shall see that certain kernels do not satisfy such requirements, and therefore merit careful attention.

Continuity Conditions

The usual starting point is that, for the linear case, $K(t,s)$ is required to be continuous. This implies no more than C^0 continuity, on the triangle $0 \leq s \leq t \leq T$, $T < \infty$, over the interval $[0,T]$ which leads to the basic proof, the method of successive approximations, using the contraction mapping argument (see e.g. [38], [46], [33]). This leads to a unique *continuous* solution. However, as demonstrated in e.g. [46], there may be other non-continuous solutions. The

proof, together with the conditions on g and K can be extended to the C^m case.

$L^2[a,b]$ Conditions

If we now take those texts which base their proofs on the existence of an $L^2[a,b]$ solution, we find that the kernel $K(t,s)$ is required to be square-integrable, i.e. to satisfy

$$\int_a^b \int_a^b \mid K(t,s) \mid^2 dsdt \leq N^2 < \infty,$$

and we see (e.g. in Smithies [61]) that again we have the existence of a solution, this time in $L^2[a,b]$, but not necessarily its uniqueness. We note also (see [61]) that 'a continuous kernel is also, *a fortiori*, an L^2 kernel' (the implication being again that this applies over the compact interval $[a,b]$).

The Lipschitz Condition

There is, however, a further condition, which if applicable to the kernel, results in the equation having one and only one solution, of appropriate continuity order if the continuity conditions apply, or in $L^2[a,b]$ if that is the relevant function space. This is the Lipschitz condition, applied to the third argument of the kernel, as

$$\mid K(t,s,\phi_1) - K(t,s,\phi_2) \mid \leq L \mid \phi_1 - \phi_2 \mid,$$

where the constant L represents a bound on the partial derivative $\delta K/\delta y$. Where $K(t,s,y)$ is non-linear in y, the Lipschitz condition must be included specifically for the unique solution to exist, and is applied on a local basis in this context.

If the kernel is linear in y, and $K(t,s)$ is continuous and therefore bounded on the interval of integration, the Lipschitz condition will clearly hold, and is used in the proof - even though not necessarily stated as an *a priori* condition of the theorem. For a linear kernel, if $K(t,s)$ is not bounded, then the Lipschitz condition also fails.

If the space involved is $L^2[a,b]$, then the inequality is as defined above, but taking the appropriate norm in place of the modulus. When the problem exists in the form of a system of equations, the Lipschitz constant is represented by a bound on the appropriate norm of the Jacobian matrix $\| J \|$.

Theorems on Existence and Uniqueness

The main theorem is given by Linz [46] as follows:

Theorem 2.2.1 *If $k(t,s)$ is continuous in $0 \leq s \leq t \leq T$ and $\psi(t)$ is continuous in $0 \leq t \leq T$ then the integral equation (2.1) possesses a unique continuous solution for $0 \leq t \leq T$.*

This theorem depends on a contraction mapping argument, which may be extended to include the case for which the kernel is square-integrable, and the input function $\psi \in L^2[0,T]$. We find this developed by Hochstadt [33] (Theorem 6):

Theorem 2.2.2 *Let $\psi(t) \in L^2[0,1]$ and suppose $k(t,s)$ is such that*

$$\int_0^1 \int_0^1 \left| k(t,s) \right|^2 ds\, dt < \infty,$$

then

$$\phi(t) - \lambda \int_0^t k(t,s)\phi(s)ds = \psi(t)$$

has a unique solution for all $\lambda \in L^2[0,1]$.

There are two points of particular note here:

1. Hochstadt derives his results primarily for the Fredholm equation; the transfer to the Volterra equation is straightforward, if we take the limits as $[a,b]$ and then define $k(t,s) = 0$ for $s > t$ ([33] p.31);

2. further, again arising from the Fredholm structure, we find the parameter λ before the integral: the result for the Fredholm equation involves a restriction on the range of values of λ, but for the Volterra equation there is no

such restriction, if the continuity requirements are defined as in Theorem 2.2.1.

These two theorems provide the fundamental basis for existence and uniqueness of solutions. However, even at this early stage we find a point of principle which has important implications in our next chapter. In Theorem 2.2.1 we are told that there is a unique continuous solution: it is made clear subsequently by Linz that there may exist other non-continuous solutions. Hochstadt is more explicit, and states that "...the space in which one chooses to work is significant in determining the resultant theory ...". Hence by defining the space in which certain solutions are admissible, the existence and uniqueness theorems hold, but this does not preclude further solutions occurring outside that space.

The issue for the numerical analyst in such a case is to identify whether an approximation method converges to a unique solution trajectory in the defined space, or to one of the other nonunique trajectories, and what would be the significance of such behaviour.

There is a lifting of these constraints for a class of equations which is weakly singular, and which has been found to have unique solutions: these are equations with kernels of the form $k(t,s) = p(t,s)(t-s)^{-\alpha}$, where $p(t,s)$ is smooth, and $-1 < \alpha < 0$, sometimes known as Abel-type equations. The classical theory developed for the solution of singular equations relates to this structure, which enables the use of the Laplace Transfoms.

The constraints on the Volterra equation are a major issue in our investigation, and in our next chapter we will consider a class of equations which do not comply with the conditions attached to the kernel, nor are they of Abel type, so a careful inspection of the wording is essential.

The Fredholm Alternative

Finally, we have a theorem which is not just a framework for the uniqueness of the solution, but gives insight on the case where a family of solutions exists. This is given in various forms, depending on the context of writing. We take the theorem as stated by Atkinson [5], in the general form of functional operators:

Theorem 2.2.3 *The Fredholm Alternative (in: Atkinson [5])*

Let $\mathcal{X}$ be a Banach space, and let $\mathcal{K} : \mathcal{X} \to \mathcal{X}$ be compact. Then the equation $(\Lambda - \mathcal{K})x = y, \Lambda \neq \emptyset$, has a unique solution $x \in \mathcal{X}$ if and only if the homogeneous equation $(\Lambda - \mathcal{K})z = 0$ has only the trivial solution $z = 0$. In such a case, the operator $\Lambda - \mathcal{K} : \mathcal{X} \to \mathcal{X}$ has a bounded inverse $(\Lambda - \mathcal{X})^{-1}$, ($\Lambda := \lambda I$, I being the identity operator).

If we take this a step further, to the case for which the associated homogeneous equation has non-zero forms of solution, we have the superposition of the particular solution together with solutions to the homogeneous equation providing the full solution set of the equation. This is made clear in the theorem as presented by Riesz and Sz.-Nagy [55], including the complex case:

Theorem 2.2.4 *The Fredholm Alternative (in: Riesz and Sz.-Nagy [55])*

Either the integral equations

$$f - Kf = g \quad (a); \qquad f' - K^* f' = g' \quad (b)$$

with kernels $K(x, y)$, $\quad K^*(x, y) = \overline{K(y, x)}$, have unique solutions f, f', whatever be the given functions g, g', and in particular have the unique solutions $f = 0, f' = 0$ when $g = 0, g' = 0$, or the homogeneous equations

$$\varphi - K\varphi = 0 \quad (c); \qquad \varphi' - K^* \varphi' = 0 \quad (d)$$

also have non-zero solutions, and the number n of linearly independent solutions is finite and the same for the two homogeneous equations.

In the second case, a necessary and sufficient condition that equations (c) and (d) have solutions is that g be orthogonal to all the solutions φ of (c) and that g' be orthogonal to all the solutions φ' of (d).

This topic will have a considerable bearing on our treatment of the class of equations in question. No doubt, originally the Fredholm alternative was considered in the context of the Fredholm equations, with fixed limits on the integration. Later representations, however, such as the two forms of the theorem quoted above, are general results, applied in the context of functional analysis. Since we are considering a class of Volterra integral equations, we are able to take the view that these may be considered as special cases of the Fredholm class of equations, with the limits fixed as $0 \leq a \leq s \leq b$, and further that $K(t,s) = 0$ for $s > t$. The discontinuity created in the kernel is acceptable if we consider the solution $u \in \text{Ł}^2[a,b]$.

Remark 1 *The preceding discussion demonstrates the extreme care which we have to take in considering the use of the word 'unique' in the description of solution(s) of a VIE. The phrase most commonly used in the texts is that* 'there is a unique solution $\in C^m$ '*, and we have to consider that this may not necessarily preclude the existence of further solutions outside C^m. A major feature of the Fredholm alternative is that it takes account of the further solutions which exist when $L(y) \neq 0$.*

2.3 Solution methods

A means of obtaining the analytic solution(s) of a VIE is only available in a restricted number of cases. (See e.g. [46]). We first introduce the definition of the resolvent kernel:

Definition 2.3.1 *Resolvent kernel:*

Let $k_0 = k(t,s)$, $k_n(t,s) = \int_s^t k(t,\tau)k_{n-1}(t,\tau)d\tau$. *Obtaining the successive terms* k_n *by iteration, the resolvent kernel is* $R(t,s)$ *where*

$$R(t,s) = \sum_{n=0}^{\infty} k_n(t,s).$$

Again, $k(t,s)$ is required to be continuous [46] or in $L^2[a,b]$ [33], depending on the space in which we are working. This leads to the following solution formula:

Theorem 2.3.1 *If* $k(t,s)$ *and* $\psi(t)$ *are continuous, then the unique continuous solution of equation (2.1) is given as*

$$\phi(t) = \psi(t) + \int_0^t R(t,s)\psi(s)ds.$$

Degenerate kernels

Re-stating the case for which the kernel is separable, the VIE is of the structure

$$\phi(t) = \psi(t) + \int_0^t \sum_{i=1}^{m} P_i(t)Q_i(s,\phi(s))ds. \tag{2.3}$$

For the linear case, $K(t,s,\phi) = k(t,s)\phi(s)$ where

$$k(t,s) = \sum_{i=1}^{m} P_i(t)Q_i(s)$$

this is called a degenerate kernel of rank m. We find that this is equivalent to the system of ordinary differential equations

$$x_i'(t) = \sum_{j=1}^{m} P_j(t)Q_i(t)x_j(t) + Q_i(t)g(t),$$

whose solutions $x_i(t)$ are characterised by $x_i = 0, \quad i = 1,\ldots,m.$

The solution of equation (2.3) is then found from the formula

$$\phi(t) = \psi(t) + \sum_{i=1}^{m} x_i(t)P_i(t).$$

2.3.1 The Convolution Equation

The classical theory concerning weakly singular Volterra equations relates to those with a convolution kernel, where we define the convolution $a * b$ as

$$a * b = \int_0^t a(t-s)b(s)ds,$$

and the Volterra convolution equation

$$\phi(t) = \psi(t) + \int_0^t k(t-s)\phi(s)ds.$$

The solution of such an equation is readily obtained using the Laplace transform, defined as

$$L(f)(w) = \int_0^\infty e^{-wt} f(t)dt,$$

subject to certain restraints on the domain of w. The convolution theorem then states that

$$L(a * b) = L(a).L(b),$$

which is equivalent to the statement that the Laplace transform of a convolution of two functions is equal to the product of the Laplace transforms of those functions. We are now able to write the convolution equation in the form

$$\phi = \psi + k * \phi,$$

and applying the transform to both sides yields

$$L(\phi) = L(\psi) + L(k).L(\phi)$$

which we can re-arrange to give

$$L(\phi) = \frac{L(\psi)}{1 - L(k)} = L(\psi)\left(1 + \frac{L(k)}{1 - L(k)}\right).$$

If the expression on the right hand side is a known transform, or can be put into a linear combination of such, then the inverse transform can be readily obtained,

from known tables, solving for ϕ; otherwise it is possible, though less convenient, to use the inverse formula

$$L^{-1}(\phi)(t) = \frac{1}{2\pi i}\int_{\alpha-i\infty}^{\alpha+i\infty} e^{wt}u(w)dw.$$

This idea can be extended to the resolvent kernel formula for the solution of the linear convolution equation, so that if the resolvent is known, or can be found, we may use the resolvent equation

$$R(t) = k(t) + \int_0^t k(t-s)R(s)ds.$$

This can then be solved using the formula

$$\phi = \psi + L^{-1}\left(\frac{L(k)}{1-L(k)}\right) * \psi.$$

We note that the expression on the right does not exist when L(k)=1: this and other aspects of the Paley-Wiener theory as applicable to integral equations is dealt with in some depth by Gripenberg, Londen and Staffans [30]. The conditions on the convolution terms are such that these may be weakly singular, but integrable. This allows the use of the convolution method where kernels of the type $(t-s)^{-\beta}$ are involved, $0 < \beta < 1$, as well as the case where the equation is a non-singular difference equation.

The above is a brief summary of the way in which the Laplace transform can be used for such equations. For a fuller description and proofs we refer to Linz [46], from which the above is drawn, and as a main reference on this subject Churchill [17].

Chapter 3

A Class of Integral Equations with Weak Singularity

3.1 Rationale

The aim of this thesis is to explore the theoretical background of, and to further develop the numerical means of solution for a certain class of Volterra integral equations, with weak singularity at the origin.

The class of equations, introduced in Chapter 1, is typified by kernels of the form

$$\begin{aligned} K_1(t,s,u) &= k(t,s,u)\frac{s^{\mu-1}}{t^{\mu}} \\ K_2(t,s,u) &= k(t,s,u)\frac{1}{\sqrt{\pi}}\frac{1}{\sqrt{\log(t/s)}}\frac{s^{\mu-1}}{t^{\mu}} \end{aligned}$$

where $k(t,s,u)$ has "well-behaved" characteristics on the triangular domain $0 \leq s \leq t \leq T$. The discussion which follows will pursue the simplest possible case, where $k(t,s,u)$ is linear in u, and $k(t,s) = 1$. Thus the VIE which is the subject matter of our investigation reduces to

$$u(t) = g(t) + \int_0^t \frac{s^{\mu-1}}{t^{\mu}} u(s)ds. \tag{3.1}$$

This equation derives originally from a problem in heat conduction, and the background has been described in Chapter 1. We now look at the problem in detail, to find why such an apparently innocuous equation cannot be dealt with by the usual methods applicable to integral equations. We find that the conditions appropriate to the various proofs and solution derivations described in the previous chapter are not met in this case: the kernel does not comply with any continuity order, is not in $L^2[0, b]$, nor can a Lipschitz constant be obtained, for any interval which includes the origin. The underlying implication of this aspect is the topic of this chapter.

The structure of solutions to this equation depends on the value of the exponent μ, and three separate categories can be identified, depending on whether $0 < \mu < 1$, $\mu = 1$, or $\mu > 1$. The case when $\mu > 1$ is the most amenable: the integrand is non-singular (except at $t = 0$), and we are able to show that equation (3.1) has a unique solution, provided that the input function g has certain constraints in place. When $\mu = 1$, the uniqueness breaks down, and we have a set of parallel solutions, while when $\mu < 1$, there is a family of solutions, one of which retains certain smoothness properties, the remainder having infinite gradient at the origin. This is the situation in which we are most interested, and in this chapter we will summarize the previous work, and introduce a new approach to describe and develop the theoretical understanding of equation (3.1) before we go on to consider numerical means of solution.

3.2 The Fredholm Alternative

The Fredholm Alternative is a theorem based on a functional analysis approach, which provides further insight into the qualitative behaviour of solutions to equations, which are now considered as integral operators. It goes further than the classical methods, in that we are offered an understanding of the case where multiple solutions are known to exist.

The equation in which we are interested cannot be described as compact, (see e.g. the condition (8.1-3) and related theorems in [55]), but we will examine this equation in the light of the Fredholm alternative, and see whether the results are relevant, even if the conditions are absent.

We are clearly in the second option of the alternative, as given in Chapter 2. Only the real part of the theorem is required, and it is obvious (by substitution) that the homogeneous equation

$$L(u) = u(t) - \int_0^t \frac{s^{\mu-1}}{t^{\mu}} u(s)ds = 0$$

has solutions of the form $c_0 t^{1-\mu}$ where c_0 is an arbitrary constant. A brief inspection shows that any other solution for $L(u) = 0$ is not possible. Hence we have a family of solutions, linearly dependent on each other, but only one such family. This result is consistent with the second part of the Fredholm alternative, provided that for the non-homogeneous equation, the input function is not a linear dependent of this family.

The intriguing nature of equation (3.1) is apparent when we consider the comparison with a Fredholm equation: the Fredholm alternative then relates to the homogeneous equation $(\lambda I - \mathcal{F})x = 0$, where $\mathcal{F}$ is the Fredhom integral operator, I the identity operator, and λ is defined as an eigenvalue: a unique solution to the Fredholm equation exists for values of λ for which there is no 'general' solution of this type (see e.g. [5]). This aspect is usually irrelevant to a Volterra equation for which the kernel is compact: λ is not restricted. If we return to equation (3.1), defining the integral operator as $\mathcal{G}(u)$, we find that for any value of λ other than unity, there are no solutions to the homogeneous equation $(\lambda I - \mathcal{G})u = 0$, and the value $\lambda = 1$ (identified with equation (3.1)) is the only case for which multiple solutions to equation $(\lambda I - \mathcal{G})u = g$, exist.

3.3 Structure and Behaviour of Solutions

3.3.1 Existence and Uniqueness for $\mu > 1$

The existence and uniqueness of the solution to the related equation

$$y(t) = f(t) + \int_0^t K_2(t, s, y)ds$$

is dealt with by Tang et al. [64] for $\mu > 1$. We follow the same reasoning for equation (1.1), as follows:

Theorem 3.3.1 *If $g(t) \in C^m[0, T]$ and $\mu > 1$ then the equation*

$$u(t) = g(t) + \int_0^t \frac{s^{\mu-1}}{t^\mu} u(s)ds$$

possesses a unique solution $u \in C^m[0, T]$.

Proof

Choose an arbitrary function $v \in C^m[0, T]$. Define $u = S(v)$ such that

$$u(t) = g(t) + \int_0^t \frac{s^{\mu-1}}{t^\mu} v(s)ds.$$

Setting $s = \lambda t, ds = t\, d\lambda$,

$$\int_0^t \frac{s^{\mu-1}}{t^\mu} v(s)ds = \int_0^1 \lambda^{\mu-1} v(\lambda t)d\lambda.$$

Since $v, g \in C^m[0, T]$

$$u^{(j)}(t) = g^{(j)}(t) + \int_0^1 \lambda^{\mu-1+j} v^{(j)}(\lambda t)d\lambda \tag{3.2}$$

where $0 \leq j \leq m$ and $u^{(j)} := d^j u/dt^j$.

If $u_1 = S(v_1)$ and $u_2 = S(v_2)$, then from (3.2)

$$\begin{aligned} |u_1^{(j)} - u_2^{(j)}| &\leq \int_0^1 \lambda^{\mu-1+j} |v_1^{(j)}(\lambda t) - v_2^{(j)}(\lambda t)|d\lambda \\ &\leq \int_0^1 \lambda^{\mu-1} d\lambda \parallel v_1 - v_2 \parallel_m \end{aligned}$$

where

$$\| \phi \|_m := max \left| \frac{d^j \phi}{dt^j} \right|, \quad 0 \leq j \leq m, \; 0 \leq t \leq T$$

so that

$$\| u_1 - u_2 \|_m \leq \frac{1}{\mu} \| v_1 - v_2 \|_m .$$

If $\mu > 1$, this is a contraction mapping, and we have a unique solution $u \in C^m[0,T]$. □

For a more informative result, and extending to the case when $\mu \leq 1$, we shall require the solution formula given by Han [32].

3.3.2 The Han Solution

We re-state equation (1.1) for which we consider solutions in this section:

$$u(t) = g(t) + \int_0^t \frac{s^{\mu-1}}{t^\mu} u(s) ds. \tag{3.3}$$

The solution obtained by Han was derived using the conversion from VIE to ODE and in Lemma 2.1 of [32] the solution was formulated as follows:

For $\mu > 1$, $\quad g \in C^m[0,T], \quad (m \geq 0)$, there is a unique solution $u(t) \in C^m[0,T]$,

$$u(t) = g(t) + t^{1-\mu} \int_0^t s^{\mu-2} g(s) ds.$$

For $\mu = 1$, $\quad g \in C^1[0,T], \quad g(0) = 0,$

$$u(t) = c_0 + g(t) + \int_0^t s^{-1} g(s) ds.$$

For $\mu < 1$, $\quad g \in C^1[0,T],$

$$u(t) = c_0 t^{1-\mu} + g(t) + \frac{1}{\mu - 1} g(0) + \int_0^t \frac{s^{\mu-2}}{t^{\mu-1}} [g(s) - g(0)] ds. \tag{3.4}$$

In this section we examine this result, using methods applicable to integral equations, to see whether the conversion to ODE has properly represented the solution set. We take as the underlying hypothesis the premise that in accordance with the theorems stated in Chapter 2 on existence and uniqueness of solutions,

together with the Fredholm alternative, we may expect to obtain solutions of equation (3.3), and hence owing to the connection of equation (1.3), as in the following two statements:

Theorem 3.3.2 *Let $g \in C^m[0,T], m \geq 1$, then there is a single solution of (3.3), $u_0 \in C^m[0,T]$, together with a family of solutions which inclue a further term linearly independent of the smooth solution u_0, non-differentiable when $\mu < 1$.*

Conjecture 3.3.3 *Let $g \in L^2[0,T]$, then there is a single solution of (3.3), $u_0 \in L^2[0,T]$, together with a family of solutions which inclue a further term linearly independent of the smooth solution u_0, non-differentiable when $\mu < 1$.*

Remark 2 *The theorem above is fully supported in this section. The conjecture which follows we believe to be also valid, as the functional analysis approach can be expected to take the $L^2[0,T]$ space into account; however, rigorous proof is not at present available.*

3.3.3 Re-assessment of the solution

We quote from Polyanin and Manzhirov [51], that

"The general solution of a Linear Non-homogenous Integral Equation is the sum of the general solution $Y = Y(x)$ of the corresponding homogeneous equation $L[y] = 0$, and an arbitrary particular solution $\overline{y} = \overline{y}(x)$ of the non-homogeneous equation $L[\overline{y}] = g(x)$, i.e. $y = Y + \overline{y}$." This is the practical application of the Fredholm Alternative, which we have given abstractly in section (2.2).

We consider the structure of the formula (3.4). Taking the terms in order,

1. $c_0 t^{1-\mu}$ will be shown to be the complementary solution, i.e. the general solution of the assocated homogeneous equation $L(u) = 0$, and

$$u(t) = g(t) + \int_0^t \frac{s^{\mu-2}}{t^{\mu-1}} g(s) ds \tag{3.5}$$

a particular solution. See item 4. We note that the complementary solution will be orthogonal to the particular solution derived below, provided $g \notin \left\{\phi : \phi(t) = k\, t^{1-\mu}\right\}$, k arbitrary.

2. g is the input function.

3. The third term could be considered superfluous, as it cancels out against the second term of the integrand. However, if we retain it in place, and subtract out the lower limit of the integrand, this yields the $y(0)$ value as $g(0) + \frac{1}{\mu-1}g(0)$ - enabling us to define the integral term at the lower limit as zero. This gives $y(0) = g(0)\frac{\mu}{\mu-1}$, tying in with the Han result [32] (2.5).

4. While the conditions justifying the use of a resolvent kernel are not present, if we follow the usual procedure for finding such a kernel the result obtained is the form identical to the kernel in the Han solution: $\mathcal{H}(t,s) = \frac{s^{\mu-2}}{t^{\mu-1}}$.

We justify the above in the Lemmas and the Remark which follow.

Lemma 3.3.4 *The solution of the homogeneous equation $L[u] = 0$ corresponding to equation (3.1), is $u = c_0 t^{1-\mu}$, where c_0 is an arbitrary constant.*

Proof

We take the homogeneous equation $L(u) = 0$,

$$u(t) - \int_0^t \frac{s^{\mu-1}}{t^{\mu}} u(s)ds = 0.$$

Substitution of the term $u(t) = c_0 t^{1-\mu}$ demonstrates that this is indeed the solution of the homogeneous equation. Alternatively, differentiating,

$$u'(t) = \frac{-\mu}{t^{\mu+1}} \int_0^t s^{\mu-1} u(s)ds + \frac{1}{t}u(t)$$

which leads to

$$\frac{du}{dt} = \frac{u}{t}(1-\mu)$$

and integration gives the solution of $L(u) = 0$ as

$$u(t) = c_0 t^{1-\mu}.$$

□

The particular solution will depend on the term $g(t)$.

Lemma 3.3.5 *There is a particular solution to (3.3) with the kernel* $s^{\mu-2}/t^{\mu-1}$.

Proof

We take the result from [51] (p.119), changing the use of μ to κ, to avoid confusion.

For a Volterra equation where the structure is of the form

$$y(x) + A\int_a^x x^\lambda t^\kappa y(t)dt = f(x),$$

the solution is given as

$$y(x) = f(x) - \int_a^x \mathcal{Q}(x,t)f(t)dt$$

where

$$\mathcal{Q}(x,t) = Ax^\lambda t^\kappa exp\left\{\frac{A}{\lambda+\kappa+1}(t^{\lambda+\kappa+1} - x^{\lambda+\kappa+1})\right\}$$

for $\lambda + \kappa + 1 \neq 0$ and

$$\mathcal{Q}(x,t) = Ax^{\lambda-A}t^{\kappa+A}$$

if $\lambda + \kappa + 1 = 0$, which applies in the case of equation (1.1).

Applying the above to equation (1.1), with $A = -1$, $\kappa = \mu - 1$, $\lambda = -\mu$, and using t and s as the time variables, this gives the solution form as

$$y(t) = g(t) + \int_0^t \mathcal{Q}(t,s)g(s)ds$$

where $\mathcal{Q}(t,s) = +t^{-\mu+1}s^{\mu-2}$ concluding the Lemma. □

Note that the kernel $\mathcal{Q}(t,s)$ of the solution is identical to the resolvent form $\mathcal{R}(t,s)$ developed below.

To construct $\mathcal{R}(t,s)$, we bear in mind that the customary constraints do not apply at the origin - however, we require a formula which can yield a solution for $t \in [0,T]$, so provided we can specify the value at $t=0$, let us for this purpose temporarily suspend the restriction, and follow the usual procedure:

Remark 3 *If suitable conditions were in place, then the resolvent kernel would take the form* $\mathcal{R} = \frac{s^{\mu-2}}{t^{\mu-1}}$.

Proof

The proof is obtained by creating a Neumann series, by induction. We follow the notation of Linz [46].

$$K_0 = \frac{s^{\mu-1}}{t^{\mu}}$$

$$\begin{aligned} K_1 &= \int_s^t \frac{\tau^{\mu-1}}{t^{\mu}} \frac{s^{\mu-1}}{\tau^{\mu}} d\tau \\ &= \frac{s^{\mu-1}}{t^{\mu}} (\log t - \log s) \end{aligned}$$

$$\begin{aligned} K_2 &= \int_s^t \frac{\tau^{\mu-1}}{t^{\mu}} \frac{s^{\mu-1}}{\tau^{\mu}} (\log \tau - \log s) d\tau \\ &= \frac{s^{\mu-1}}{t^{\mu}} \frac{(\log t - \log s)^2}{2}. \end{aligned}$$

Now take the case for $K_{r+1} : r \in \mathrm{Z}$. Assume that

$$K_r = \frac{s^{\mu-1}}{t^{\mu}} \frac{1}{r!} (\log t - \log s)^r$$

$$\begin{aligned} K_{r+1} &= \int_s^t \frac{\tau^{\mu-1}}{t^{\mu}} \frac{s^{\mu-1}}{\tau^{\mu}} \frac{1}{r!} (\log \tau - \log s)^r \\ &= \frac{s^{\mu-1}}{t^{\mu}} \int_s^t \frac{(\log \tau - \log s)^r}{\tau} d\tau. \end{aligned}$$

Using the substitution $v = \log \tau - \log s$,

$$K_{r+1} = \frac{s^{\mu-1}}{t^{\mu}} \frac{1}{(r+1)!} (\log t - \log s)^{r+1}.$$

We have shown that this is so for $r = 0, 1$, hence true for all $r \in \mathrm{Z}$, which completes the induction. The resolvent kernel is then formally obtained by summation

$$\mathcal{R}(t,s) = \sum_{i=1}^{\infty} K_i(t,s)$$

$$= \frac{s^{\mu-1}}{t^{\mu}} \left\{ 1 + \log\frac{t}{s} + \frac{1}{2!}\left(\log\frac{t}{s}\right)^2 + \frac{1}{3!}(\log\frac{t}{s})^3 + ... \right\}$$

$$= \frac{s^{\mu-1}}{t^{\mu}} \exp\left\{ \log\frac{t}{s} \right\}$$

$$= \frac{s^{\mu-2}}{t^{\mu-1}}.$$

This completes the proof. □

We note that this is identical to $\mathcal{Q}(t,s)$ obtained by Lemma 2, and also the form of resolvent identified by Brunner and Van der Houwen [14] (p.39) for the class of VIE's whose kernels are separable.

So we have been able to verify that the ODE solution for $\mu < 1$ holds good when examined in the context of integral equation treatment, consisting of a single smooth solution, together with a family of non-smooth solutions.

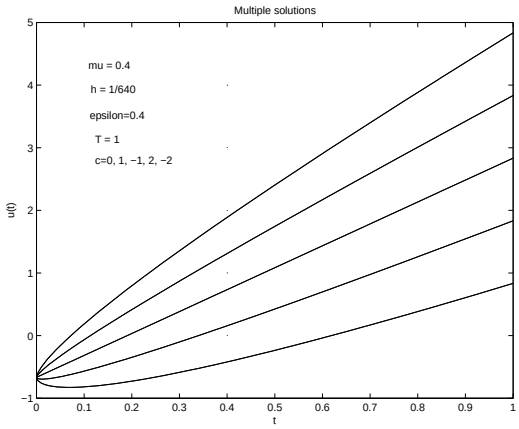


Figure 3.1: $\mu < 1$

Case 1: $\mu \in (0, 1)$

So far, we have not stipulated a range of values for μ, apart from the assumption that $\mu > 0$. In the foregoing arguments, we have looked at the most intricate of the three solution formulae, which applies to the situation when $0 < \mu < 1$, and all elements of this formula have now been accounted for.

Case 2: $\mu \in (1, \infty)$

The solution for $\mu > 1$ may be built up in the same way, with the homogeneous equation $L(u) = 0$ having solutions $c_0 t^{1-\mu}$, and the particular solution with the same resolvent structure $u(t) = g(t) + t^{1-\mu} \int_0^t s^{\mu-2} g(s) ds$.

We note that for $1 < \mu < 2$ this form of solution still has a weak singularity in the integrand, so subtracting out as before,

$$u(t) = g(t) + \frac{g(0)}{\mu - 1} + \int_0^t \frac{s^{\mu-2}}{t^{\mu-1}} [g(s) - g(0)] ds,$$

where the integral is zero at the lower limit.

The general solution is again

$$u(t) = c_0 t^{1-\mu} + g(t) + \frac{g(0)}{\mu - 1} + \int_0^t \frac{s^{\mu-2}}{t^{\mu-1}} [g(s) - g(0)] ds,$$

where the first term gives a family of solutions for arbitrary c_0. The term $c_0 t^{1-\mu}$ is now unbounded at the origin, as well as having an unbounded derivative. So for each $\mu > 1$ there exists a family of solutions converging asymptotically towards the smooth solution.

Remark 4 *This extends the solution set for $\mu > 1$ to include solutions unbounded at the origin. If the problem to be modelled is a physical one, with an initial value implication, these solutions are not relevant. In the wider sense of mathematical abstraction, they are an intrinsic feature of the solution set.*

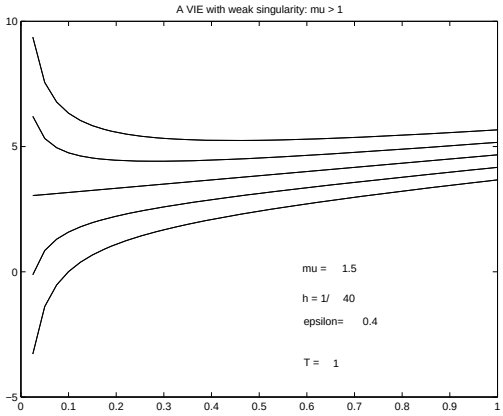


Figure 3.2: $\mu > 1$

Case 3: $\mu = 1$

Finally, we examine the case when $\mu = 1$. The equation now reduces to

$$u(t) = g(t) + t^{-1} \int_0^t u(s)ds \tag{3.6}$$

where the solution derived by Han, in order to be meaningful, required that $g(0) = 0$. However, here we find that the homogeneous equation $L(u) = 0$ yields solution $u(t) = k$, where k is an arbitrary constant. Dealing formally with equation (3.6), as for the case when $\mu < 1$, we obtain a resolvent kernel,

$$\begin{aligned} \mathcal{R}(t,s) = \frac{1}{t}\left\{1 + \log(t/s) + (\log(t/s))^2 + \dots\right\} \\ = \frac{1}{t}e^{\log(t/s)} \qquad = \frac{1}{s}, \end{aligned}$$

giving the general solution

$$u(t) = k + g(t) + \int_0^t s^{-1}g(s)ds$$

for an arbitrary constant k, subject to $g(0) = 0$, and the Han solution is again confirmed, giving a family of parallel solutions.

The restriction that $g(0) = 0$ is necessary for the existence of an initial value, $u(0)$. However, if $g(0)$ is non-zero, solutions may exist away from the origin. As

before, we subtract out the singularity at the lower limit of integration, to obtain a family of solutions all of which are unbounded at the origin:

$$u(t) = k + g(t) + g(0)\log t + \int_0^t s^{-1}(g(s) - g(0))ds.$$

3.3.4 Smoothness of the C^1 solution

We are now in a position to establish the behaviour of the C^1 solution, in particular its differentiability for all positive values of μ. This is an extension of the result in Theorem (3.3.1), but we are now able to use the solution formula obtained by Han, and corroborated in the preceding section. We re-state the solution, in its fundamental form, i.e. taking $c_0 = 0$ to give the particular solution we require:

$$u(t) = g(t) + \frac{g(0)}{\mu - 1} + \int_0^t \frac{s^{\mu-2}}{t^{\mu-1}} \left[g(s) - g(0)\right] ds.$$

Making the substitution $s = t\tau$, we give the following result from Han [32]:

Theorem 3.3.6 *Regularity of the C^1 solution (Han)*

When $\mu > 1$ for any $g \in C^m[0,T]$ ($m \geq 0$ an integer), the integral equation (3.3) has a unique solution u, $u \in C^m[0,T]$ and $\|u\|_m \leq a\|g\|_m$ for some constant a.

When $0 < \mu \leq 1$, for any $g \in C^m[0,T]$ ($m \geq 1$ an integer), with $g(0) = 0$ if $\mu = 1$, the integral equation (3.3) has a family of solutions depending on a parameter. Out of the family of solutions, there is one particular solution u with C^1 continuity. Such a solution is unique, and $\|u\|_m \leq a\|g\|_m$ for some constant a.

Proof The existence of a family of solutions, one of which has C^1 continuity, has been explored in the earlier part of this chapter. Alternatively, the method of Han is to be found in [32].

For the regularity estimate, using the substitution above, we obtain

$$\begin{aligned} u(t) &= g(t) + \int_0^1 \tau^{\mu-2} g(t\tau) d\tau, \quad \mu \geq 1, \\ u(t) &= g(t) + \frac{g(0)}{\mu - 1} + \int_0^1 \tau^{\mu-2} [g(t\tau) - g(0)] d\tau, \quad 0 < \mu < 1 \end{aligned}$$

for the C^1 solution. The regularity property now follows, and we note that this form of the equation is found to be compact.

□

3.4 Further Implications

Uniqueness of trajectory at $t = \alpha$

It may seem obvious that at some point, $t = \alpha$ say, if the value of the solution $u(\alpha)$ is given, this will uniquely define the trajectory $u_c(t)$ passing through this point. This is addressed in more detail in [20], but we can take the simpler proof of *reductio ad absurdum*.

Suppose there are two separate trajectories, u_1 and u_2, with identical input functions g, both of which pass through the same point,

$$\begin{aligned} u_1(t) &= g(t) + \int_0^t \frac{s^{\mu-1}}{t^\mu} u_1(s) ds, \\ u_2(t) &= g(t) + \int_0^t \frac{s^{\mu-1}}{t^\mu} u_2(s) ds. \end{aligned}$$

At $t = \alpha$, these are equal, so that

$$u_1(\alpha) - u_2(\alpha) = \int_0^\alpha \frac{s^{\mu-1}}{\alpha^\mu} [u_1(s) - u_2(s)] ds = 0.$$

Since s is allowed to vary from 0 to α, the only possibility is that $u_1(t) \equiv u_2(t)$ for all t, and hence there is one and only one trajectory satisfying a given value of $u(\alpha)$.

Alternatively, we take the solution formula, where different trajectories are identified by the constant c_0 in the term $c_0 t^{1-\mu}$, so that there is precisely one

solution $u_c(t)$ for each value of c_0. The reasoning applies equally to the case $\mu > 1$. Hence for all $\mu > 0$ if a numerical method is applied, commencing at some positive value of $t = t_1$, based on an *approximate* solution at $t = \alpha$, or on data readings, it is possible that one (and only one) of the adjacent trajectories may be followed.

The uniqueness of the smooth solution is assured in Theorem 3.3.7. The only multiple solutions which can exist have been shown (p.27) to take the form $c_0 t^{1-\mu}$, and the uniqueness of a *particular* solution away from the origin is shown in subsection (6.3.1).

If the forcing function g is given, then this trajectory is uniquely defined by

$$u(t_1) = c_0 t_1^{1-\mu} + g(t_1) + \frac{g(0)}{\mu - 1} + t_1^{1-\mu} \int_0^{t_1} s^{\mu-2}(g(s) - g(0))ds,$$

i.e.

$$c_0 = t_1^{\mu-1}\Big[u(t_1) - g(t_1) - \frac{g(0)}{\mu - 1}\Big] - \int_0^{t_1} s^{\mu-2}(g(s) - g(0))ds.$$

This is an alternative definition for c_0 to the one given by Han, and links a specific trajectory with a specific value of the function u away from the origin. Without loss of generality, we could take $t_1 = 1$, to give

$$c_0 = u(1) - g(1) - \frac{g(0)}{\mu - 1} - \int_0^{1} s^{\mu-2}(g(s) - g(0))ds.$$

Having identified a specific value of c_0 with a given solution value at $t_1 = 1$, we must take account of the overall context as to how the problem is postulated.

If the question is one of initial value significance at $t_0 = 0$, and we restrict the solution set to $u : u \in C^m$, then the further solutions for $\mu > 1$ are irrelevant, as are those for $\mu = 1$ when $g(0) \neq 0$. However, if we lift this restriction, and extend the consideration to the full solution set, we must take account of the behavioural pattern described above.

3.5 Stability

$$u(t) = g(t) + t^{-\mu} \int_0^t s^{\mu-1} u(s)\, ds. \tag{3.7}$$

The stability properties of the equation (3.7) must be considered in the context of the multiple solutions. We recall that out of the multiple solutions which exist for each input function g, one and only one is the 'smooth' solution, and when $g(t) \in C^m[0,T]$, for some non-zero integer m, the single smooth solution $u_0(t)$, which we will call the fundamental solution, is also in $C^m[0,T]$ (see Theorem 3.4.7). The infinite set of solutions are of the form $u_c(t) = u_0(t) + c_0 t^{1-\mu}$. We will develop the stability of equation (1.1) in relation to a small change in the input function g by first considering the response of the smooth solution.

Let Δg be some small perturbation to the input function g, and let Δu_0 be the corresponding change in the fundamental solution. We then have

$$u_0(t) + \Delta u_0(t) = g(t) + \Delta g(t) + t^{-\mu} \int_0^t s^{\mu-1} \{u_0(s) + \Delta u_0(s)\} ds,$$

and subtracting equation (3.7) with $u = u_0$ gives

$$\Delta u_0(t) = \Delta g(t) + t^{-\mu} \int_0^t s^{\mu-1} \Delta u_0(s) ds,$$

which taking the norm $\| \, . \, \| = \max_{t\in[0,T]} |u(t)|$ gives the inequality

$$\begin{aligned} \| \, \Delta u_0(t) \, \| &\leq \| \, \Delta g(t) \, \| + \| \, \Delta u_0(t) \, \| \, t^{-\mu} \int_0^t s^{\mu-1} ds \\ &= \| \, \Delta g(t) \, \| + \| \, \Delta u_0(t) \, \| \, \frac{1}{\mu}, \end{aligned}$$

and we can bound the resulting change in the solution by

$$\| \, \Delta u_0(t) \, \| \leq \left| \frac{\mu}{\mu - 1} \right| \| \, \Delta g(t) \, \| \, .$$

The behaviour of the set of non-smooth solutions depends on the exponential term, and as this forms the solution to the related homogeneous equation, which is unaffected by the change in g, the new fundamental solution will also have its

infinite set of related non-smooth solutions now defined by the arbitrary constant d,

$$u_d(t) + \Delta u_d(t) = u_0(t) + \Delta u_0(t) + d\, t^{1-\mu}.$$

This gives us a restricted interpretation of the stability of the solution set resulting from a small perturbation to the input function.

3.6 The alternative equation

We finish this chapter with some thoughts on the earlier equation described in section (1.1), to confirm and consolidate the way in which the two equations are related. We now re-state equation (1.3) as follows:

$$y(t) + \int_0^t p(t,s)y(s)ds = g_1(t), \tag{3.8}$$
$$p(t,s) := \frac{1}{\sqrt{\pi}} \frac{1}{\sqrt{log(t/s)}} \frac{s^{\mu-1}}{t^{\mu}},$$

for $0 \leq s \leq t \leq T$ and $\mu > 0$. We open this section with the result obtained by Diogo et al. [26] (Lemma 2) that it is possible to show that this is equivalent to equation (3.1), in the following way:

Lemma 3.6.1 *(Diogo, McKee and Tang)*

Equation (3.8) can be transformed into the equivalent equation

$$u(t) = g_2(t) + \int_0^t \frac{s^{\mu-1}}{t^{\mu}} u(s)ds$$

where

$$g_2(t) := -\int_0^t p(t,s)g_1(s)ds + g_1(t).$$

Proof

Consider

$$y(s) + \int_0^t p(s,\lambda)y(\lambda)d\lambda = g_2(\lambda).$$

Multiplying both sides by $p(t,s)$, and integrating with respect to λ, then reversing the order of integration yields

$$\int_0^t p(t,s)y(s)ds + \int_0^t y(\lambda)\int_\lambda^t p(t,s)p(s,\lambda)ds\,d\lambda = \int_0^t p(t,s)g_1(s)ds. \qquad (3.9)$$

We have used Dirichlet's formula, as given in [26], which states that

$$\int_0^t \int_0^s \phi(s,\lambda)d\lambda\,ds = \int_0^t \int_s^t \phi(s,\lambda)ds\,d\lambda.$$

Combining (3.9) and (3.8) gives the required result.

We expand on the proof as follows. The first term in (3.9) is equal to $g_1(t) - y(t)$, from (3.8), and the final term is equal to $g_1(t) - g_2(t)$, by definition. We now consider the middle term I, where

$$I = \int_0^t y(\lambda)\int_\lambda^t \frac{1}{\pi}\frac{1}{\sqrt{(\log(t/s)}}\frac{1}{\sqrt{\log(s/\lambda)}}\frac{s^{\mu-1}}{t^\mu}\frac{\lambda^{\mu-1}}{s^\mu}ds\,d\lambda.$$

We need two substitutions, and for clarity will take these separately.

(1). Let $\log t - \log s = \tau$, so that $-(1/s)ds = d\tau$, and $t/s = e^\tau$.

$$I = \int_0^t y(\lambda)\frac{\lambda^{\mu-1}}{\pi t^\mu}\left\{\int_{\log t/\lambda}^0 \frac{1}{\sqrt{\tau}}\frac{-d\tau}{\sqrt{(\log(t/\lambda)-\tau)}}\right\}d\lambda.$$

(2). Now let $\tau = \sin^2\theta\log(t/\lambda)$, so that $d\tau = 2\sin\theta\cos\theta\log(t/\lambda)d\theta$, to give

$$I = \int_0^t y(\lambda)\frac{\lambda^{\mu-1}}{\pi t^\mu}\left\{\int_{\sin^{-1}1}^0 \frac{-2\sin\theta\cos\theta\log(t/\lambda)d\theta}{\sqrt{\log(t/\lambda)}\sin\theta\sqrt{\log(t/\lambda)}\cos\theta}\right\}d\lambda,$$

$$I = \int_0^t y(\lambda)\frac{\lambda^{\mu-1}}{\pi t^\mu}\Big[-2\theta\Big]_{\pi/2}^0 d\lambda,$$

$$= \int_0^t y(\lambda)\frac{\lambda^{\mu-1}}{t^\mu}d\lambda,$$

or reverting back to the variable of integration s,

$$I = \int_0^t \frac{s^{\mu-1}}{t^\mu}y(s)\,ds.$$

Now we recall that the first and third terms of (3.9) are realised, and we have derived the required equation

$$y(t) = g_2(t) + \int_0^t \frac{s^{\mu-1}}{t^\mu}y(s)ds,$$

where g_1 and g_2 are related as above. □

3.6.1 Additional Results

The assumption was made in [64] that $\int_0^t p(t,s)ds = 1/\sqrt{\mu}$, which is derived in an attenuated form by Rooney [56]. In the following lemma we show that this is so, and use the method to provide two further results. We take the definition of the gamma function in the form

$$\Gamma(z) = \int_0^\infty x^{z-1}e^{-x}dx,$$

and the result that $\Gamma(\frac{1}{2}) = \sqrt{\pi}$. (See e.g. [1]).

Lemma 3.6.2 *If p(t,s) is as defined in (2) above, then* $\int_0^t p(t,s)ds = 1/\sqrt{\mu}$.

Let $\log(t/s) = v$, so that $-(1/s)ds = dv$, and $(t/s) = e^v$.

$$\begin{aligned}\int_0^t p(t,s)ds &= \frac{1}{\sqrt{\pi}}\int_\infty^0 -v^{-1/2}e^{-\mu v}dv \\ &= \frac{1}{\sqrt{\pi}}\int_0^\infty v^{-1/2}e^{-\mu v}dv. \qquad (3.10)\end{aligned}$$

Using the definition of the gamma function, if we allow the variables to be $z = 1/2$ and $x = \mu v$, so that $dx = \mu\, dv$,

$$\begin{aligned}\Gamma(\tfrac{1}{2}) &= \int_0^\infty (\mu v)^{-1/2}e^{-\mu v}\mu\, dv, \\ &= \sqrt{\mu}\sqrt{\pi}\int_0^t p(t,s)ds,\end{aligned}$$

comparing with (3.10) above. But $\Gamma(\frac{1}{2}) = \sqrt{\pi}$, so that

$$\int_0^t p(t,s)ds = \frac{1}{\sqrt{\mu}}.$$

□

Corollary 3.6.3 *The only solutions of the homogeneous equation*

$$y(t) + \int_0^t p(t,s)y(s)ds = 0$$

of the form c_0t^α *is for* $\alpha = 1 - \mu$.

Let $I_2 = \int_0^t p(t,s)y(s)ds$, and assume that a solution $y(t) = c_0 t^\alpha$ exists. Using the same substitutions as before,

$$\begin{aligned} I_2 &= \frac{1}{\sqrt{\pi}} \int_\infty^0 -v^{-1/2} e^{-(\mu+\alpha)v} c_0 t^\alpha dv, \\ &= \frac{c_0 t^\alpha}{\sqrt{\pi}} \int_0^\infty v^{-1/2} e^{-(\mu+\alpha)v} dv. \end{aligned}$$

Now substituting $x = (\mu+\alpha)v$ in the gamma function evaluated at $z = \frac{1}{2}$,

$$\Gamma(\tfrac{1}{2}) = \int_0^\infty [(\mu+\alpha)v]^{-1/2} e^{-(\mu+\alpha)v} (\mu+\alpha) dv$$

so that

$$\Gamma(\tfrac{1}{2})(\mu+\alpha)^{-1/2} = \int_0^\infty u^{-1/2} e^{-(\mu+\alpha)v} dv$$

Hence,

$$\begin{aligned} I_2 &= \frac{kt^\alpha}{\sqrt{\pi}} (\mu+\alpha)^{-1/2} \Gamma(\tfrac{1}{2}) \\ &= kt^\alpha (\mu+\alpha)^{-1/2}. \end{aligned}$$

Setting this into the homogenous equation, we need to know if there are values of t^α which satisfy

$$kt^\alpha + kt^\alpha (\mu+\alpha)^{-1/2} = 0.$$

Clearly, there is the trivial solution $t \equiv 0$, but also possible solutions in $\mathbb{R}$ when the negative root is taken, and $(\mu+\alpha)^{1/2} = -1$. Hence, $\alpha = 1-\mu$ yields solutions to the homogeneous equation of the form $kt^{1-\mu}$. □

Corollary 3.6.4 *If $g(t) = Bt^\beta$ in equation (3.8) above, then for any given $\mu > 0$, there is a unique smooth solution of the form At^β for $\mu + \beta \geq 0$.*

The process is as for the homogeneous equation above. We assume a solution At^β, and let

$$\begin{aligned} I_3 &= \int_0^t \frac{1}{\sqrt{\pi}} \frac{1}{\sqrt{log(t/s)}} \frac{s^{\mu-1}}{t^{\mu}} A s^{\beta} ds, \\ &= \frac{A}{\sqrt{\pi}} \int_0^{\infty} v^{-1/2} e^{-(\mu+\beta)v} t^{\beta} dv, \\ &= \frac{At^{\beta}}{\sqrt{\pi}} (\mu+\beta)^{-1/2} \Gamma(\tfrac{1}{2}) \\ &= At^{\beta}(\mu+\beta)^{-1/2}, \end{aligned}$$

so that

$$I_3 = At^{\beta}(\mu+\beta)^{-1/2}.$$

Returning to the main equation, and substituting for the assumed solution and the integral term,

$$At^{\beta} + At^{\beta}(\mu+\beta)^{-1/2} = Bt^{\beta},$$

and the solution is valid for a unique value of A when B is given,

$$A = B \frac{(\mu+\beta)^{1/2}}{(1+(\mu+\beta)^{1/2})}.$$

□

3.7 Transformation

No single transformation enables us to deal with all aspects of equation (1.1); however there are possible substitutions which enable better understanding of its behaviour, one of which has already been mentioned.

1. The elementary shift of origin $\tau = t + a$, $\sigma = s + a$ for some $a > 0$ does not improve the situation analytically: the lower limit of integration, and the singularity, are still in place at $t = -a$; we shall, however, find this a convenient device when considering the convergence analysis in chapter 6.

2. Let $\lambda = s/t$, $ds = t\, d\lambda$, so that equation (3.1) becomes

$$u(t) = g(t) + \int_0^1 \lambda^{\mu-1} u(t\lambda) d\lambda,$$

a Fredholm-type equivalent which we have seen in section (3.2).

3. Let $s^\mu = x$, and $t^\mu = w$, so that $\mu s^{\mu-1} ds = dx$, hence equation (3.1) is transformed into

$$u(w^{1/\mu}) = g(w^{1/\mu}) + (\mu w)^{-1} \int_0^w u(x^{1/\mu}) dx.$$

This has interesting possibilities for numerical approximation, in that applying the uniform grid $\mathcal{T}$ to the discrete version of the transformed equation, we may obtain a stepsize h which performs well close to the origin. However, for $t > 1$, the solution will grow exponentially.

4. Finally, there is the logarithmic substitution $\sigma = \log s$, $\tau = \log t$, $d\sigma = (1/s)ds$. The transformed equation becomes

$$u(e^\tau) = g(e^\tau) + \int_{-\infty}^{\tau} e^{\mu(\sigma-\tau)} u(e^\sigma) d\sigma.$$

We can separate the interval of integration, and let $v(\tau) = u(e^\tau)$ to give

$$v(\tau) = \overline{g}(\tau) + \int_0^\tau e^{-\mu(\tau-\sigma)} v(\sigma) d\sigma,$$

$$\overline{g}(e^\tau) = g(e^\tau) + \int_{-\infty}^{0} e^{-\mu(\tau-\sigma)} v(\sigma) d\sigma,$$

where $\overline{g}$ now includes the relevant information defining the initial trajectory over the interval $[-\infty, 0]$, or in the original coordinates the interval $[0, 1]$. The transformed equation is a convolution equation in powers of e, which is the subject of the paper [21]. The classical Paley-Wiener theory, and the related Lubich method for the discrete case, are extended to include an exponential type resolvent, hence allowing the use of Laplace transform techniques. We refer the interested reader to [21] and references cited therein.

3.8 Reprise

In this chapter we have reached certain conclusions regarding the behavioural pattern of equation (3.1). We believe that the foregoing sections 3.1 - 3.6 establish a unified approach in our dealings with this class of equations, based on the theory of integral equations and the related methods of functional analysis, which we feel is appropriate in this context. Although the sufficient conditions attached to the standard theorems on existence and uniqueness are not present, we find that the solution set is compatible with the results of those theorems, provided the specific detail of admissible function spaces is taken into account. In particular, it conforms to the postulates of the Fredholm alternative. We have a unique solution *within the appropriate space* ($C^m[0,T]$ or $L^2[0,T]$), and a family of solutions orthogonal to this, arising from the homogeneous equation $Lu = 0$.

We find an analogy in the frequently-cited case (see e.g [46]) of the equation

$$\phi(t) - \int_0^t s^{t-s}\phi(s)ds = 0,$$

which has the general solution

$$\phi(t) = ct^{t-1},$$

singular at the origin, although the analogy is not complete, since the kernel function in this case is "well-behaved". We have used methods directly applicable to integral equations, to confirm and extend results previously obtained, in particular to functions $\phi, \psi \in L^2[a,b]$.

Section 3.7 provides a more detailed aspect on the connection between the two linked equations in which we are interested, and finally in section 3.8 we assemble the several possible ways in which equation (3.1) could be amended to find a more accommodating means of approach.

Chapter 4

Numerical Methods of Solution of Volterra Integral Equations.

4.1 Preliminary

In this chapter, we will look at the various means of finding numerical solutions of second-kind VIE's,

$$u(t) = g(t) + \int_0^t K(t,s)u(s)ds. \tag{4.1}$$

This is not intended to be a complete picture of all possible methods, but rather an overview of the general structures involved, supported by detailed examples. The material is drawn from the standard literature, in particular Brunner and van der Houwen [14], Butcher [16], Evans [28], Hairer, Norsett and Wanner [31], Iserles [35], Lambert [41], and Linz [46]. We concentrate on the constructional technique, and refer to the standard texts for the relevant convergence proofs. We look to identify ways of dealing with a kernel which is singular at the origin, and to provide the algorithms to support the split-interval scheme developed in Chapter 6, and the programs used to obtain the results in Chapter 7.

To a great extent, methods for VIE's are developed from their counterparts in

the solution of ODE's: such methods may be based on quadrature rules, the linear multistep process, collocation or spline methods, sometimes in combination in the context of the VIE. We shall assume throughout this chapter that the forcing function, g, is non-singular and continuous on the interval [0, T] for some finite T.

Before considering how the quadrature is achieved, we state some basic ideas and define the terms which will be used throughout. The horizontal axis, instead of representing a continuum, is treated as a mesh of points, $t_n, n = 0...N$. Unless stated otherwise, this will form a uniform grid $\mathcal{T}$ of width h, containing N abscissae, such that:

$0 \leq t_i \leq T, \qquad t_{i+1} - t_i = h, \qquad N = \frac{T}{h} + 1.$

The term $u(t)$ will be reserved for the analytic form of solution of the equation. Define $u(t_n)$ as the value of the analytic solution at $t = t_n$, and $\tilde{u}(t_n)$ as the numerical evaluation of the solution at $t = t_n$. We define a general scheme of numerical approximation as $Q(w, h)$ such that

$$\tilde{u}(t_n) = g(t_n) + h \sum_{j=0}^{n} w_j K(t_n, t_j, \tilde{u}_n(t_j)), \tag{4.2}$$

where the w_j are appropriate weights as designated or calculated for the method.

Methods of solution for the ODE are widely available and there is a considerable range of texts available. We now consider how these are developed for the integral equation, drawing on the texts by Linz [46] and Brunner and van der Houwen [14].

The approximation is based on construction of the composite rule at each step, with the quadrature over $0 \leq t_j \leq t_n$ replacing the integration of s over $[0, t]$. Hence the rule for any single-step method is invoked n times to achieve the increment of quadrature to advance the solution from $\tilde{u}(t_{n-1})$ to $\tilde{u}(t_n)$. The application of a single-step quadrature rule to the Volterra equation is essentially a triangular implicit system of equations, which allows solution to be evaluated for each step in turn. This is in contrast to the Fredholm equation (2.2), where

both limits of integration are fixed: the discretization then results in a fully implicit system of equations, which have to be solved simultaneously.

Returning to the Volterra equation, if the rule required is one of simple quadrature, such as the trapezoidal, then the same formula (in terms of the weights) is applied over the first $n-1$ steps (for which the solution is known) as for the final 'new' solution value. If, however, the scheme is of greater complexity, or if a multistep method is used, then a suitable quadrature rule $Q_0(w_j, h)$ is used for calculation of the discrete kernel $K(t_n, t_j, \tilde{u_j})$ over the first $n-1$ terms, with the designated scheme $Q_1(h)$ used to evaluate the new approximation $\tilde{u}(t_n)$. By 'suitable' we imply a method of similar convergence order, and compatible with Q_1. The quadrature $Q_0(w_{nj}, h),\ j = 0, \ldots, n-1$ is usually referred to as the 'tail' or 'lag' expression.

It is probable that $Q_1(h)$ will not be a straightforward question of weighted values at the interval endpoints t_{n-1} and t_n: in the case of a linear multistep method, further values are taken at $t_{n\mp k}$ for a k-step rule, and the Runge-Kutta class of methods use *nodes (or abscissae)*, which are values of t internal to the grid $\mathcal{T}$.

4.2 Basic Methods of Quadrature

The simplest form of quadrature is the rectangle rule, otherwise known as Euler's method, where the VIE solution is given by

$$\tilde{u}(t_n) = g(t_n) + \sum_{j=0}^{n-1} hK(t_n, t_j)\tilde{u}(t_j), \tag{4.3}$$

which is a scheme explicit in $\tilde{u}(t_n)$; to apply the Backward Euler method, the summation is over $K(t_n, t_j)\tilde{u}(t_j),\ j = 1, \ldots, n$, which is now triangularly implicit.

If we include the trapezium rule, also triangularly implicit, these three may

be combined using the θ convention, in the form defined by e.g. [8]

$$\tilde{u}(t_n) = g(t_n) + \sum_{j=0}^{n-1} h\left[(1-\theta)K(t_n, t_{j-1})\tilde{u}(t_{j-1}) + \theta K(t_n, t_j)\tilde{u}(t_j)\right], \qquad (4.4)$$

which allows us to consider the Euler, Backward Euler, and Trapezium Rules taking θ =0, 1 or 0.5. Convergence for the Euler and Backward Euler methods is of order 1, and for the trapezium rule order 2. We note that explicit schemes such as the forward Euler are less stable than the implicit counterpart, and in general a θ-method of quadradure " ... is A-stable if and only if $0 \leq \theta \leq \frac{1}{2}$." (Iserles [35] p.59.)

4.3 Newton-Cotes Methods

Developing from the trapezoidal rule, which is a linear approximation between two points, the Newton-Cotes methods employ approximating polynomials of degree $m-1$, taken over m points.

We introduce the Lagrange polynomials, $\ell_{n,j}$, defined by

$$\ell_{n,j}(x) = \prod_{\substack{i=0 \\ i \neq j}}^{n} \frac{(x - x_i)}{(x_j - x_i)}.$$

If $a_j = hw_j$ where the $\{w_j\}$ are the weights in the discretisation 4.2, the coefficients $\{a_j\}$ are found by integrating $\ell_{n,j}$ between the limits x_0 and x_n, on a uniform grid of points $x_i = x_0 + ih, \quad i = 1 \ldots n$. The linear case ($n = 1$) yields the trapezium rule, and the quadratic ($n = 2$) Simpson's rule.

Further weights are given in the table below, for closed Newton-Cotes schemes, i.e. where the end points of the interval occur at grid points. If they do not coincide, the method is termed 'open', and a different set of weights must be applied ([14] p. 57) Here, the trivial first case corresponds to the mid-point rule. The construction of these schemes, and the remainder term of the expansion in each case, is shown in [14] and [28], and a more extensive list up to the 11-point closed

rule, and 9-point open rule, is given in [1]. The closed m-point construction has error of order h^{m+1} when m is odd, or h^m when m is even.

Weights for Newton-Cotes methods [12]							
m	w_0	w_1	w_2	w_3	w_4	w_5	w_6
2	$\frac{1}{2}$	$\frac{1}{2}$					
3	$\frac{1}{3}$	$\frac{4}{3}$	$\frac{1}{3}$				
4	$\frac{3}{8}$	$\frac{9}{8}$	$\frac{9}{8}$	$\frac{3}{8}$			
5	$\frac{14}{45}$	$\frac{64}{45}$	$\frac{24}{45}$	$\frac{64}{45}$	$\frac{14}{45}$		
6	$\frac{95}{288}$	$\frac{375}{288}$	$\frac{250}{288}$	$\frac{250}{288}$	$\frac{375}{288}$	$\frac{95}{288}$	
7	$\frac{41}{240}$	$\frac{216}{240}$	$\frac{27}{240}$	$\frac{272}{240}$	$\frac{27}{240}$	$\frac{216}{240}$	$\frac{41}{240}$

This class of methods however has a difficulty: it can be seen that the value of the weights is successively increasing, as the system is extended to include each extra step, and from the 8-point scheme contain negative values. There is a requirement for a sequence of formulae to be convergent in the sense that $\sum_{j=0}^{n} |w_{n,j}| < K$ for all $n \in N$, but this has been shown to be invalid for Newton-Cotes schemes (references cited in [14]). The trapezium rule is known for its stability and reliability. Simpson's rule however does not have A-stability, but is included by Linz [46], Evans [28] and Brunner and van der Houwen [14] as acceptable for the solution of a Volterra equation, provided the construction is carried out according to the following considerations.

Starting Values and Intermediate Values

When $m = 2$, we have the trapezoidal rule, and given the initial value of the solution, the VIE scheme is triangularly implicit, and the subsequent values are obtained immediately. However, if we consider the 3-point Simpson's rule in the VIE context, a further solution value is necessary, and $m-2$ values for the general case. These starting values can be arrived at in various ways: it is possible to use

the trapezoidal rule over the first step, Simpson's for the second, and so on up to the requisite number of values, this may be combined with subdivision of the steplength to achieve the necessary convergence; alternatively, a block-by-block process can be applied to solve for the first $m-1$ values simultaneously.

This is linked to the further issue when implementing the Newton-Cotes formulae which is first demonstrated at this stage. When Simpson's rule is used for quadrature, the process takes place over multiples of 2 steps: we recall the usual formula

$$\int_{t_0}^{t_2} f(x)dx = (h/3)[f(x_0) + 4f(x_1) + f(x_2)] + O(h^5)$$

which is based on approximation at three points. When used directly for quadrature of an integral over n points this is implemented over a pair of steps at a time to give

$$\int_{t_0}^{t_n} f(x)dx = h\sum_{i=0}^{n} w_i f(x_i) + O(h^4),$$

where $\{w_i\} = \{\frac{1}{3}, \frac{4}{3}, \frac{2}{3}, \ldots \frac{4}{3}, \frac{1}{3}\}$ with n taken to be odd, and an even number of steps. When used as a VIE method, we have to increment by one step at a time, and consider quadrature of the alternate steps when i is odd. This may be done using the trapezoidal rule, the 3/8 rule, or a 2-stage block-by-block method [46]. The higher order methods require additional treatment at the intervening $m-2$ (non-multiplicative) lines, for an $m-1$ step method. Such incremental changes are best installed at the end of each relevant line, in order to comply with the *repetition factor* constraint, where

Definition 4.3.1 *Repetition factor*

A VIE quadrature method with weights $\{w_{ij}\}$ is said to have a repetition factor of ρ if ρ is the smallest integer such that

$$w_{n+\rho,i} = w_{n,i}, \qquad n = 0, 1, \ldots n-k,$$

where k is an integer independent of n.

This definition is taken from Linz [46], followed by proof that methods with repetition factor 1 such as the trapezoidal, fourth order Gregory and Simpson's method 2 (which uses the 3/8 rule over the final three points) are stable, while those with repetition factor of 2 or greater can have unstable properties due to the accumulated starting error. For a more detailed exposition we refer to McKee and Brunner [49], in which we find specific reference to Simpson's rule, supplemented by the 3/8 rule at the beginning or end of each alternate row, the latter having repetition factor of 1, this being the preferred scheme. However, the inconvenience of this hybrid construction renders such schemes not impossible, but certainly less than ideal in the VIE context. The block-by-block approach might be taken, or another option is to use the linear multistep analogy of the appropriate rule, which increments one step at a time.

4.4 Gregory methods

Although not featuring prominently in the current texts on Volterra equations, the derivation of the Gregory methods is of interest, and raises issues which we consider again in the following chapter, in developing the product trapezoidal rule commencing at the origin.

We return again to the classic trapezoidal rule, where order 2 is obtained by means of the simple weighting pair $[\frac{1}{2}, \frac{1}{2}]$ at the endpoints of each step in the quadrature, repeated over the region of integration. This scheme has a well-known error formula in the Euler-McLaurin expansion (see e.g. [19]), given in its theoretical form for $h = (b-a)/n$ and assuming $f(x) \in C^{2k+1}[a,b]$ by

$$\begin{aligned} h\left[\tfrac{1}{2}f(a) + f(a+h) + \cdots + f(a+(n-1)h) + \tfrac{1}{2}f(b)\right] \\ = \int_a^b f(x)dx + \frac{B_2}{2!}h^2[f'(b) - f'(a)] + \frac{B_4}{4!}h^4[f'''(b) - f'''(a)] + \ldots \\ + \frac{B_2 k}{2k!}h^2 k[f^{(2k-1)}(b) - f^{(2k-1)}(a)] + h^{2k+1}\int_a^b C_k f^{(2k+1)}(x)dx, \quad (4.5) \end{aligned}$$

where the B_{2k} are the Bernouilli numbers, defined by the generating formula

$$\mathcal{B}(t) = \frac{t}{e^t - 1} = \sum_{i=0}^{\infty} \frac{B_i(t)}{i!} t^i,$$

the coefficients of odd powers being zeros.

When the Euler-McLaurin expansion is expressed in terms of differences, it is possible to exploit the error terms to derive higher order methods of order $2p,\ p = 2, 3, \ldots$; we find, however, that this does not continue indefinitely, as the expansion does not converge for higher Bernouilli numbers. Allowing $f_0 = f(a)$, $f_i = f(a + ih)$, $i = 1, \ldots, n-1$, $f_n = f(b)$, Gregory methods of order 4, 6 and possibly higher are available, and we have

$$\int_a^b f(x)dx = \frac{h}{2}(f_0 + 2f_1 + \cdots + 2f_{n-1} + f_n) - \frac{h}{12}(\bigtriangledown f_n - \bigtriangleup f_0) - \frac{h}{24}(\bigtriangledown^2 f_n - \bigtriangleup^2 f_0) - \frac{19h}{720}(\bigtriangledown^3 f_n - \bigtriangleup^3 f_0) + \cdots + R_p,$$

(see [28], [6] or [57]), where R_p represents the remainder term in (4.5) above such that $R_p = 0(h^{2k+1})$ if $f \in C^{2k+1}[a, b]$, and $R_p = 0(h^{2k+2})$ if $f \in C^{2k+2}[a, b]$ [6].

Collecting terms in f_n, f_{n-1} and f_{n-2} we obtain the final three weights w_{n-i}, $i = 0, 1, 2$ for the quadrature as $[3/8, 7/6, 23/24]$, and the first three w_i reflect the same values; intervening weights $w_i,\ i = 3 : n - 3$ are unity, as with the composite trapezoidal. The disadvantage of such a scheme is that it requires four starting values (in addition to the initial value) before the rule can be applied, more for the higher order Gregory rules, and it may be this which limits their usefulness. Baker wrote in 1977 that "Gregory's formula may yet be a useful method". [6]

4.5 Runge-Kutta Methods

The foregoing methods compute the quadrature of the integral term using a grid of uniformly spaced points which define the step structure, $t_i = a + ih, i =$

$0, 1, \ldots, N$. The evaluation takes place at these grid values. The accuracy may be improved considerably by taking intermediate values (abscissae or nodes) along the t-axis, and for an m-stage method evaluating m approximation function values, which are then combined with suitable weighting into the solution value for the step. There are four issues for consideration:

- The number of intermediate node values, which will also define the number m of approximation stages required;
- How are the nodes $\{c_j\}$ to be defined;
- Choice of weights $\{a_{ij}\}$ for the approximation stages;
- Choice of weights $\{b_i\}$ for the final combination.

We can summarise this in the Butcher array, also known as the RK tableau, as shown by e.g. Iserles [35].

$$\begin{array}{c|c} \mathbf{c} & \mathbf{A} \\ \hline & \mathbf{b^T} \end{array}$$

where $\mathbf{c} = \{c_j\}$ is the node spacing, $\mathbf{b} = \{b_i\}$ is the final weighting, and $\mathbf{A} = \{a_{i,j}\}$ is the matrix of intermediate weight values $(i, j = 1 \ldots s)$.

We give the formula for solution of the ordinary differential equation $y'(t) = f(t, y)$ as shown in e.g. [16] or [35]:

$$Y_i = y_{n-1} + h \sum_{j=0}^{s} a_{ij} f(t + c_j h, Y_j) \qquad \text{i=1,..., s,}$$

$$y_n = y_{n-1} + h \sum_{i=0}^{s} b_i f(t + c_i h, Y_i) \qquad \text{n=1, ..., N-1,}$$

where $\{Y_i\}$ are the intermediate solutions, and y_n is the current step approximation. The many ways in which this may be constructed give us a wide choice of methods at our disposal. Explicit RK schemes are represented by the lower triangular matrix $\mathbf{A}$ where $a_{ij} = 0,\ j \geq i$, exemplified by the classical RK method

(see e.g. [35]):

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{2} & & \\ 1 & -1 & 2 & \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array} .$$

To achieve the necessary order of convergence, the values of the a_{ij} have to satisfy certain conditions (see e.g. [14], [31]).

The ERK scheme above is 3-stage, and of order 3, and such methods with m stages may be constructed of order m for $m \leq 5$, but thereafter the order ceases to match the number of stages (the Butcher barrier), and further stages are needed in order to achieve a given order of accuracy. (e.g. Theorem 3.1 in [31]). We must look at the implicit Runge-Kutta methods to find the real benefit in terms of accuracy which is expected from this approach. For a m-stage IRK method, we can achieve accuracy of order $2m$ for each $m \geq 2$.

4.5.1 RK methods for Volterra integral equations

The implementation of a Runge-Kutta structure for solution of the Volterra integral equation is given in detail by Brunner and van der Houwen [14], and may be achieved in several different ways. The RK tableau is extended to include two further sets of parameters, $\mathbf{d} = \{d_{ji}\}$ and $\mathbf{e} = \{e_j\}$, which are required to construct methods of *Beltukov* or BVRK type. However, when the requirement is for PVRK or *Pouzet*-type methods, we find that the $\{d_{ji}\} = c_j$, where $\{c_j\}$ are as defined above, and $\{e_j\} = 1$ for all j. So the structure of the PVRK scheme derives directly from its ODE counterpart.

We have a further aspect to consider: as each step is implemented, the effect of the previous evaluations needs to be included to form a 'tail'or 'lag' term $\tilde{F}$ which may be constructed either from straightforward quadrature weight schemes, to give a *mixed VRK method*, or by re-utilising the terms from the intermediate stages to give an *extended VRK method*.

In chapter 7 we will consider applying PVRK schemes, with the extended lag term, which we construct below for an m-stage method, as in [14]:

$$Y_{n,j} = \tilde{F}_n(t_n + c_j h) + h \sum_{i=1}^{m} a_{ji} k(t_n + c_j h, t_n + c_i h, Y_{n,i}), \tag{4.6}$$

$$j = 1 \dots m,$$

$$y_{n+1} = \tilde{F}_n(t_n + h) + h \sum_{j=1}^{m} b_j k(t_n + h, t_n + c_j h, Y_{n,j}), \tag{4.7}$$

$$n = 0 \dots N - 1,$$

$$\tilde{F}_n(t_n) = g(t_n) + h \sum_{l=0}^{n-1} \sum_{j=1}^{m} b_j k(t_n + h, t_l + c_j h, Y_{l,j}). \tag{4.8}$$

In constructing such methods, we find that values of the abscissae in the second argument of the kernel of the Volterra equation are required to exceed those in the first, during the implicit evaluation of the m equations involved at the intermediate phase. This challenges the basic definition of the Volterra equation, in which the process of integration occurs over the triangle $0 \leq s \leq t, \quad 0 \leq t \leq T$. We find that we are justified in allowing this, as demonstrated by Baker [8], who defines a smooth extension of the kernel

$$K_{ext}(t, s, y) := \begin{cases} K(t, s, y), & 0 \leq s \leq t \\ K_{new}(t, s, y), & s > t \end{cases}$$

for the purposes of completing an implicit RK scheme.

More sophisticated schemes within the Runge-Kutta hierarchy are available, in particular if an element of error control is employed. We mention in particular the schemes of Fehlberg, also Verner, and Dormand and Prince, (see e.g. [16]) which are outside the scope of this investigation.

4.6 Gaussian Quadrature

Gaussian methods are based on polynomial collocation at carefully chosen nodes for which a system of implicit equations is solved at each step. The particular

Table 4.1: Gaussian Quadrature Structure [12]

Gaussian Quadrature Structure			
Method	Integral type	$g(x)$	Interval
Gauss-Legendre	$\int_{-1}^{1} f(x)dx$	1	$(-1, 1)$
Gauss-Chebyshev	$\int_{-1}^{1} \frac{f(x)dx}{\sqrt{1-x^2}}$	$\frac{1}{\sqrt{1-x^2}}$	$(-1, 1)$
Gauss-Laguerre	$\int_{0}^{\infty} x^q e^{-x} f(x)dx$	$x^q e^{-x}$	$(0, \infty)$
Gauss-Hermite	$\int_{-\infty}^{\infty} e^{-x^2} f(x)dx$	e^{-x^2}	$(-\infty, \infty)$

method used depends on the choice of these nodes, decided by the type of integrand under consideration. The particular class of Gaussian methods best suited to our needs is based on the Legendre polynomials, to yield the Gauss-Legendre schemes. By this means,we are able to construct a scheme of order $2m$ with uniquely defined elements in the Butcher array for each an m-stage method.

4.6.1 Gauss-Legendre methods

The choice of nodes is based on the roots of the Legendre polynomials $P_n(x)$. (The following detail is widely available, see e.g. [54].) These may be defined as the (orthogonal) polynomials satisfying the differential equation

$$(1 - x^2)y'' - 2xy' + n(n + 1)y = 0,$$

or alternatively by the generating function

$$(1-2xt-t^2)^{-1/2} = \sum_{n=0}^{\infty} P_n(x)t^n, \quad |t|<1,\ |x|\le 1.$$

One (of several) recurrence formulae gives

$$(n+1)P_{n+1}(x) = (2n+1)xP_n(x) - nP_{n-1}(x),$$

so given that $P_0(x) = 1$, $P_1(x) = x$, we see that $P_2(x) = (3x^2-1)/2$, and $P_3(x) = (5x^3-3x)/2$, which we utilise in the 2-stage and 3-stage schemes in chapter 7. With the nodes chosen as the roots of the m-degree polynomial, we find the order attained can be extremely good - for an m-stage GL method, the order is $2m$; the appropriate choice of the intermediate weights $\{a_{ij}\}$ is critical, and the order can only be achieved by a unique combination of the $\{a_{ij}\}$, $\{b_i\}$ and $\{c_j\}$.

With the c_j determined, we now construct the b_i and a_{ij}, using the Lagrange polynomials $\ell_{m,i}$ of degree m, with the interpolation points at the nodes c_i. Thus to complete the array we have:

$$b_i = \int_0^1 \ell_{m,i}(\tau)d\tau, \qquad a_{ij} = \int_0^{c_i} \ell_{m,i}(\tau)d\tau.$$

The related proof depends on the following *'simplifying assumptions'* (see [31] and references cited therein, and [14] for the indvidual listing):

$$B(p) \quad : \qquad \sum_{k=1}^{s} b_i c_i^{r-1} = \frac{1}{r}, \quad r = 1,\dots,p, \tag{4.9}$$

$$C(\eta) \quad : \qquad \sum_{j=1}^{m} a_{ij} c_j^{r-1} = \frac{c_i^r}{r}, \quad i = 1,\dots,m,\ r = 1,\dots,\eta, \tag{4.10}$$

$$D(\zeta) \quad : \qquad \sum_{i=1}^{m} b_i c_i^{r-1} a_{\ ij} = \frac{b_j}{r}(1-c_j^r), \quad j = 1,\dots,\nu,\ r = 1,\dots,\eta. \tag{4.11}$$

The proof (by Butcher (1964)) is shown in [31], based on the 'tree' notation defined in [15] or [16]. By letting $p = 2\nu$, $\eta = \nu$ and $\zeta = \nu$, conditions B and C provide $2m + m^2$ equations which enables the unique set of values determining the m-stage Butcher array, and D is now implied combining B and C.

It is possible to construct the rules for $m > 3$, which are, however, considerably more expensive in computational terms (the 4-stage method is shown in [31]). While Gaussian methods are given separate and special attention in the texts owing to the quality of their approximation capability, their construction is still in the format of the Butcher array, and we find that the application of the Gauss-Legendre scheme to a specific problem (in this case, a Volterra equation) follows the same pattern as that described for Runge-Kutta methods.

4.7 Linear Multstep Methods

The linear multistep (LM) method applied to the ODE $y' = f(t, y), \quad y(0) = y_0$ is usually given in the form of the difference equation over a uniform grid:

$$\sum_{i=0}^{k} \alpha_i y_{n-i} = h \sum_{i=0}^{k} \beta_i f(t_{n-i}, y_{n-i}),$$

where the number of steps k and choice of $\alpha_i,\ \beta_i$ determines the particular type of LM method involved. The simpler schemes are based on the supposition that the approximation is obtained from

$$y_n = y_{n-1} + h \sum_{i=0}^{k} \beta_i f(t_{n-i}, y_{n-i}),$$

so that $\alpha_0 = 1$ and $\alpha_1 = -1$, the summation takes place over k previous values of the solution, and the β_i depend on the interpolating polynomial. Even though Lagrange polynomials are again used as the approximating functions, the coefficients β_i do not reflect the Newton-Cotes weights, being obtained less directly for the multistep process. We define a set of intermediate values $\gamma_i,\ i = 0, \ldots, k$ where

$$\gamma_i = \int_{t_n}^{t_{n+1}} \ell_i(t) dt,$$

ℓ_i being the Lagrange polynomial of degree i located at the points t_{n-i}. When the integrals are evaluated, apart from a single term h, all other terms in n and h

vanish, and a numerical fraction emerges, the value of γ_i. The process is lengthy for $i \geq 3$, and it is quicker and easier to use the equivalent formula, the explicit case being given as

$$\gamma_i = \int_0^1 (-1)^i \begin{pmatrix} -s \\ i \end{pmatrix} ds.$$

It is now most convenient to represent the approximation process in terms of backward differences, the coefficients being the appropriate γ_i, hence if $\bigtriangledown$ is the usual backward difference operator, we have

$$y_n = y_{n-1} + h[\gamma_0 \bigtriangledown^0 + \gamma_1 \bigtriangledown^1 \gamma_2 \bigtriangledown^2 + \dots] f_{n-1}$$

for an explicit scheme, or the same formula concluded with f_n if implicit. Setting the appropriate values of γ_i, and collecting terms in $f_{n-1},\ f_{n-2}, \dots$ yields the LM method required.

To illustrate this, we take the 3-step explicit method to obtain $\gamma_0 = 1$, $\gamma_1 = 1/2$, $\gamma_2 = 5/12$, $\gamma_3 = 3/8$ and hence

$$\begin{aligned} y_n = y_{n-1} + h\Big[f_{n-1} + \frac{1}{2}(f_{n-1} - f_{n-2}) + \frac{5}{12}(f_{n-1} - 2f_{n-2} + f_{n-3}) \\ + \frac{3}{8}(f_{n-1} - 3f_{n-2} + 3f_{n-3} + f_{n-4})\Big] \\ = y_{n-1} + h\Big[\frac{55}{24} f_{n-1} - \frac{59}{24} f_{n-2} + \frac{37}{24} f_{n-3} - \frac{9}{24} f_{n-4}\Big]. \quad (4.12) \end{aligned}$$

Better quality results are, as usual obtained using implicit methods, and the interval of integration of the interpolating polynomial now includes the new point being approximated. The shortcut formula only needs a single adjustment, to give

$$\overline{\gamma}_i = \int_0^1 (-1)^i \begin{pmatrix} -s+1 \\ i \end{pmatrix} ds,$$

where the $\overline{\gamma}_i$ are the coefficients of the backward differences for the implicit methods. Setting these coefficients in the formula based on f_n, we obtain the corresponding LM coefficients $\overline{\beta}_i$

The following are the main categories of basic schemes of this type:

Scheme	Property
Adams-Bashforth	$\alpha_1 = -\alpha_0 = 1, \quad \beta_0 = 0$ (explicit)
Adams-Moulton	$\alpha_1 = -\alpha_0 = 1, \quad \beta_0 \neq 0$ (implicit)
Milne-Simpson, Nystrom	$\alpha_2 = -\alpha_0 = 1,$
Backward differentiation formulae	$\beta_0 \neq 0, \quad \beta_i = 0, \quad i = 1 \ldots k.$

Associated with these formulae, we define the characteristic equation

$$\rho(z) = h\sigma(z),$$

where ρ and σ are the first and second characteristic polymomials in z,

$$\rho(z) = \sum_{i=0}^{k} \alpha^i z^{k-i}, \qquad \sigma(z) = \sum_{i=0}^{k} \beta^i z^{k-i}.$$

For such methods to be stable, we have the requirement that the strong root condition applies to the first polynomial $\rho(z)$: the roots must lie within or on the unit circle, and any roots on the unit circle must be simple.

More intricate schemes are developed combining the Adams-Bashforth (AB) with the Adams-Moulton (AM) to construct *predictor-corrector* methods, where an initial solution is predicted by the AB, and fed back into the AM formula, with a considerable improvement in accuracy; a repetition of the feedback produces even better results, and there are numerous ways in which this can be achieved. Except for the backward differentiation scheme, the LM methods do not figure strongly in the literature on numerical means of solution of the Volterra equation. We find various reasons why this is so: there is the computational cost involved - we shall describe the ways of constructing the Volterra analogy of a LM scheme, and further there are questions of order and of stability. We define the Dahlquist first and second barriers in the form of theorems:

Theorem 4.7.1 (Dahlquist first barrier)

For a k-step linear multistep method of order p,

$p \leq k+2$ *if* k *even,*

$p \leq k+1$ *if* k *odd,*

$p \leq k$ *if* $\beta_k/\alpha_k \leq 0$ *(in particular for the explicit case). [31].*

Theorem 4.7.2 (Dahlquist second barrier)

The highest order of an A-stable LM method is 2. [35].

If we are prepared to accept these limitations, we must consider the construction of the VLM (Volterra linear multistep) scheme. We recall that to solve each step of a discrete VIE, we apply the appropriate method to solve that step, together with the lag terms which precede it. It is certainly possible to use the difference equation over the precursive terms, but each time we need the starting scheme over $k-1$ steps. There are two further possibilities: one is to use the rather intricate implementation offerred by Brunner and van der Houwen [14], as shown in equation (4.13) for a k-step method with $n = T/h$:

$$\sum_{i=0}^{k} \alpha_i y_{n-i} + \sum_{i=0}^{k} \sum_{j=-k}^{k} \beta_{i,j} \tilde{F}_{n-i}(t_{n+j}) = h \sum_{i=0}^{k} \sum_{j=-k}^{k} \gamma_{i,j} k_{n-i}(t_{n+j}), \tag{4.13}$$

where the lag terms consist of the quadrature

$$\tilde{F}_n(t) = g(t) + h \sum_{i=0}^{n} w_{n,\ell} k_\ell(t),$$

$$k_n(t) = k(t, t_n, y_n).$$

The alternative is to make use of the property of certain methods, which are defined as (ρ, σ)-reducible. The significance of this is, that we are able to find suitable quadrature weights which can replace the LM-recurrence scheme over the lag terms as the means of approximating the integral.

4.7.1 BDF

These formulae come within the general definition of linear multistep methods, however their construction, and behaviour, are somewhat different. All the coefficients $\beta_i,\ i = 1, \ldots, k$ are zero, and the only evaluation of $f(t_n, y_n)$ is at the current point to be approximated. The values of the α_i are obtained by *differentiation* of the interpolating Lagrange polynomial, to yield a class of methods known for their good stability properties. The order of these methods corresponds directly to the number of steps used, and we obtain the following set of rules up to order 4 ([3]):

BDF coefficients							
p	k	β_0	α_0	α_1	α_2	α_3	α_4
1	1	1	1	-1			
2	2	$\frac{2}{3}$	1	$\frac{-4}{3}$	$\frac{1}{3}$		
3	3	$\frac{6}{11}$	1	$\frac{-18}{11}$	$\frac{9}{11}$	$\frac{-2}{11}$	
4	4	$\frac{12}{25}$	1	$\frac{-48}{25}$	$\frac{36}{25}$	$\frac{-16}{25}$	$\frac{3}{25}$

We have mentioned the idea of (ρ, σ)-reducible formulae, and for this class of methods, in the particular context of the Volterra equation, a substantial amount of work has been done by P.H.M. Wolkenfelt [71], who introduced the following notation. The matrix of weights $\{w_{nj}\}$ is treated as a composite of three elements, S, Σ and Ω, where S gives the weights for the starting procedure, Σ the initial weights for each step calculation, and Ω the diagonally implicit triangular matrix for completion of each step:

$$\begin{bmatrix} S & \\ \Sigma & \Omega \end{bmatrix}.$$

A means of constructing the relevant weights for the 2-step method with the trapezoidal as the starting rule is given in [71] and [14] by the following formulae:

$$\begin{aligned} S: &= \{\tfrac{1}{2}, \tfrac{1}{2}\}, \\ \Sigma: \quad w_{nj} &= \tfrac{3}{4}(1-(\tfrac{1}{3})^n), \quad n \geq 1, j = 0, 1, \\ \Omega: \quad w_{nj} &= 1-(\tfrac{1}{3})^{n-j+1}, \quad n-j \geq 0,\ j \geq 2. \end{aligned}$$

From this (essentially a discrete convolution expression) we can obtain the weights required, but an explicit (i.e. directly applicable) formula is only available for the 2-step method. A means of developing the weights for $k \geq 3$ is shown by Wolkenfeldt, but involves rather detailed computation.

Stability of BDF methods

The BDF formulae were originally developed in response to the need for solution of so-called 'stiff' ordinary differential equations, notably by Gear (1971) (cited in e.g. [31]). We can therefore expect useful stability properties, but there is an upper limit to the number of steps, and hence the order which can be achieved. The following approach is based on Hairer, Norsett and Wanner [31].

Using the backward difference representation, the formula is given as

$$\sum_{j=1}^{k} \frac{1}{j} \nabla^j y_{n+1} = h f_{n+1},$$

so the first characteristic polynomial is

$$\rho(z) = \sum_{j=1}^{k} \frac{1}{j} z^{k-j}(z-1)^j.$$

For convenience, we let $w = 1/(1-z)$ and consider the polynomial

$$p(w) = (1-w)^k \rho\left(\frac{1}{(1-w)}\right) = \sum_{j=1}^{k} \frac{w^j}{j},$$

which is the partial sum of $-\log(1-w)$.

This leads to the BDF equivalent of the strong root condition, that the k-step formula is stable if and only if all roots of the polynomial $p(w)$ lie *outside* the unit circle $w : |w - 1| \leq 1$, with only simple roots occurring on the perimeter. For $k \leq 6$ this condition holds, but not for higher values of k, and hence we have

Theorem 4.7.3 *A k-step BDF method is stable for $k \leq 6$, and unstable for $k \geq 7$.*

This also restricts the possible order attainable for the ODE to six, but for the reasons described earlier, the 2-step scheme is most useful for the Volterra equation, when the stability consideration is of greater importance than the order of convergence.

4.8 Product Integration

Finally, we consider a class of methods which is particularly suitable for the case in which the interval of integration includes a singularity, as evaluation of the function at the unbounded term is replaced by means of a weighting function, which includes the analytic integral over the step. Product integration is considered in [14], [39], and in more detail by Linz [46] and Atkinson [5] .

The integrand is separated into its well-behaved and singular parts. Suppose that the integral term of a Volterra integral equation is

$$I = \int_0^t p(t,s)\, q(t,s)\, y(s)\, ds,$$

where $p(t,s)$ is smooth but $q(t,s)$ is integrable, but weakly singular. The numerical approximation may be given as

$$\int_0^t p(t,s)q(t,s)y(s)ds \approx \sum_{i=0}^{n-1} \Big\{ p(t_n,t_i)y(t_i) \int_{t_i}^{t_{i+1}} q(t_n,s)ds \Big\},$$

and the problem reduces to the non-singular form

$$I \approx \sum_{i=0}^{n-1} w_{n,i}\, p(t_n,t_i)\, y(t_i)$$

where

$$w_{n,i} = \int_{t_i}^{t_{i+1}} q(t_n, s)\, ds.$$

This is the product Euler rule, which is found to be at best of order $O(h)$, although this may not always be achieved, as we shall see in chapter 7.

The generalised form of the product integration rules is obtained by treating the weight function $w := \{w(n,i)\}$ as integration of the product of the singular term and the appropriate Lagrange polynomial.

Define the space S_r as follows:

$$S_r := \{y : y(t) = \sum_{i=1}^{r} \ell_{n,r}\, y_{n-i+1}\},$$

where the $\ell_{n,r}$ are Lagrange polynomials of degree r, evaluated at the points $t_{n+1-i},\ i = 0, \ldots, r$ such that

$$\ell_{n,r} = \prod_{\substack{j \neq k \\ j,k=1:r}} \Big\{ \frac{t - t_{n+1-k}}{t_{n+1-j} - t_{n+1-k}} \Big\}.$$

The specific cases we are interested in are the product Euler scheme described above, for which $\ell_{n,1} = 1$ for all n, and the product trapezoidal, where

$$\begin{aligned} \ell_{n,1} &= \frac{t - t_n}{t_{n-1} - t_n} \\ \ell_{n,2} &= \frac{t - t_{n-1}}{t_n - t_{n-1}} \qquad t \in [t_{n-1}, t_n]. \end{aligned}$$

Further product integration schemes are obtained using the appropriate Newton-Cotes structure (product Simpson, etc) to extend the number of points and the order of the method accordingly. As in section (4.3), attention must be given to the starting procedures, and the final step calculations on intermediate rows. Linz [46] suggests the use of block-by-block methods to avoid the potential weakness of employing lower-order methods to complete the procedure.

When we apply the product trapezoidal method to the integrand of a Volterra

integral equation, this yields the algorithm as given by Linz [46]:

$$K(t,s,f(s)) \approx \frac{s-t_{n-1}}{h}p(t,t_n,f(t_n)) + \frac{t_n-s}{h}p(t,t_{n-1},f(t_{n-1}),$$
$$s \in [t_{n-1},t_n],$$

so that

$$\int_0^{t_n} q(t_n,s)p(t_n,s,f(s))ds \approx \alpha_{n,1}p(t_n,t_0,f(t_o)) +$$
$$+\sum_{i=1}^{n-1}(\alpha_{n,i+1}+\beta_{n,i})p(t_n,t_i,f(t_i)) + \beta_{n,n}p(t_n,t_n,f(t_n))$$

where

$$\begin{aligned}\alpha_{n,i+1} &= \frac{1}{h}\int_{t_i}^{t_{i+1}}(t_{i+1}-s)p(t_n,s)ds,\\ \beta_{n,i+1} &= \frac{1}{h}\int_{t_i}^{t_{i+1}}(s-t_j)p(t_n,s)ds\end{aligned}$$

In chapter 6 we develop the convergence proof relating to the product trapezoidal and higher order product methods for a VIE with the type of singularity under consideration.

4.9 Hermite Interpolation

This method is essentially a collocation method, being based on the third order spline polynomials of Hermite. We follow the construction developed in [26] for the case when $\mu > 1$. The scheme is based on that given by Prenter [53], using the notation based on that source.

Let Π_N denote a uniform mesh $t_n = nh, 0 \le n \le N, h = T/N$ and S_3 be the space of piecewise cubic Hermite polynomials

$$S_3 := \{u : u(t) = \phi_{1n}(t)u_n + \phi_{2n}(t)u_{n+1} + \psi_{1n}(t)u_n + \psi_{2n}(t)u_{n+1}$$
$$t \in [t_n,t_{n+1}], \quad 0 \le n \le N-1\},$$

where $u_j = u(t_j),\ u'_j = u'(t_j),\ j = n, n+1$, and

$$\phi_{1n}(t) = (t - t_{n+1})^2[h + 2(t - t_n)]/h^3, \tag{4.14}$$

$$\phi_{2n}(t) = (t - t_n)^2[h + 2(t_{n+1} - t)]/h^3, \tag{4.15}$$

$$\psi_{1n}(t) = (t - t_n)(t - t_{n+1})^2/h^2, \tag{4.16}$$

$$\psi_{2n}(t) = (t - t_n)^2(t - t_{n+1})/h^2. \tag{4.17}$$

Definition 4.9.1 *Let $f \in C^m[0,T], m > 1$ be a given function. Then $p(t) \in S_3$ is the Hermite cubic interpolant to f if*

$$p(t_n) = f(t_n), \quad p'(t_n) = f'(t_n), \quad 0 \le n \le N.$$

Lemma 4.9.1 *(Prenter, 1975)*

Assume $f \in C^4[0,T]$. Then

$$\|f^{(\lambda)} - p^{(\lambda)}\|_\infty = O(h^{4-\lambda}), \quad \lambda = 0, 1.$$

□

We need to construct an approximate solution u_n of equation (4.1) which satisfies

$$u(t_n) - \int_0^{t_n} k_1(t_n, s)u(s)\,ds = g(t_n), \tag{4.18}$$

$$u'(t_n) - \int_0^{t_n} k_2(t_n, s)u(s)\,ds = g'(t_n), \tag{4.19}$$

(4.19) being obtained by making the substitution $\lambda = s/t$ in (4.18) and differentiating with respect to t. We can now take the piecewise approximations of equations (4.18) and (4.19) as

$$u(t_n) - \sum_{i=0}^{n-1} \int_{t_n}^{t_{n+1}} k_1(t_n, s)u(s)\,ds = g(t_n),$$

$$u'(t_n)) - \sum_{i=0}^{n-1} \int_{t_n}^{t_{n+1}} k_2(t_n, s)u'(s)\,ds = g'(t_n).$$

Using the change of variable $s = t_i + \nu h,\ 0 \leq i \leq n-1$ we obtain

$$\begin{aligned} u(t_n) - \sum_{i=0}^{n-1} h \int_0^1 k_1(t_n, t_i + \nu h) u(t_i + \nu h)\, d\nu &= g(t_n), \\ u'(t_n) - \sum_{i=0}^{n-1} h \int_0^1 k_2(t_n, t_i + \nu h) u'(t_i + \nu h)\, d\nu &= g'(t_n), \end{aligned}$$

where

$$\begin{aligned} u(t_i + \nu h) = l_1(\nu) u_i + l_2(\nu) u_{i+1} + d_1(\nu) h u'_i + d_2(\nu) h u'_{i+1}, \\ 0 \leq \nu \leq 1 \\ u'(t_i + \nu h) = \frac{1}{h} l'_1(\nu) u_i + l'_2(\nu) u_{i+1} + d'_1(\nu) h u'_i + d'_2(\nu) h u'_{i+1}, \\ 0 \leq \nu \leq 1 \end{aligned}$$

The four polynomials l_1, l_2, d_1 and d_2 are obtained setting the appropriate values in equations (4.14)-(4.17) above, to yield

$$\begin{aligned} l_1(\nu) &= 2\nu^3 - 3\nu^2 + 1, \\ l_2(\nu) &= -2\nu^3 + 3\nu^2, \\ d_1(\nu) &= \nu^3 - 2\nu^2 + \nu, \\ d_2(\nu) &= \nu^3 - \nu^2. \end{aligned}$$

We are now in a position to construct the recursive algorithm, as a pair of simultaneous equations to be solved for the two unknowns, u_n and hu'_n. The detail of this for equation (1.1) is shown in Chapter 5.

4.10 Extrapolation Methods

We are interested in the methods of extrapolation in the specific sense of convergence acceleration. That is, not as methods of numerical approximation which can work independently, but as a means of improving the convergence of the numerical results of some scheme, arbitrary for the time being, for which the

order of convenrgence is inadequate. This section is drawn from the references Brezinski and Redivo-Zaglia [13] and Weniger [68], and we refer the interested reader to these for more information on this subject.

What follows is a brief summary of the way in which these processes are constructed, following four well-known schemes, which will enable us to justify the method we use in chapter 6.

There are three means by which such methods may be constructed: the first, and original format is as the ratio of determinants, the structure of the determinants defining the method used. While this is closest to the fundamental derivation of the method, it can quickly lead to excessive computer operations, with cumulative rounding errors; secondly, for methods which may be applied directly, a straightforward algebraic construction can be used; and thirdly, there is the iterative process.

Richardson

Probably the best known scheme, usually the introductory text book method. We include in this the Romberg formula (occasionally the two names are used interchangeably), which is a basic extrapolation applied to the trapezoidal rule. The process of extrapolation is developed in columnar form, each column showing a better convergence order than the previous.

Let us suppose we have a set $\{S_n\}$ of solution approximations to a given problem, evaluated at $h_n, \quad n = 1, 2, \ldots$; let S be the true solution, and assume that this is the limiting case, as $h_n \to 0$, and $\{x_n\}$ be a subsidary sequence, for our purposes such that $\{x_n\} \equiv \{h_n\}$. We consider $\{S_n\}$ to be of the form

$$S_n = S + a_1 x_n + a_2 x_n^2 + \cdots + a_k x_n^k.$$

In determinant form, the Richardson scheme is defined to be the transformation

T where

$$T = \frac{\begin{vmatrix} S_n & \dots & S_{n+k} \\ x_n & \dots & x_{n+k} \\ \vdots & & \vdots \\ x_n^k & \dots & x_{n+k}^k \end{vmatrix}}{\begin{vmatrix} 1 & \dots & 1 \\ x_n & \dots & x_{n+k} \\ \vdots & & \vdots \\ x_n^k & \dots & x_{n+k}^k \end{vmatrix}}.$$

This can also be represented as the recursive formula

$$T_k^{(n)} = \frac{x_{n+k} T_{k-1}^{(n)} - x_n T_{k-1}^{(n+1)}}{x_{n+k} - x_n}.$$

Thirdly, the algebraic solution of the system of equations

$$S_{n+i} = T_k^{(n)} + a_1 x_{n+i} + \cdots + a_k x_{n+i}^k$$

yields the same result. [13].

Aitken/Shanks

The original form of this transformation is known as the Aitken Δ^2 process, based on the forward differences $\Delta S_n = S_{n+1} - S_n$, defined in its numerically stable form by the transformation T_n where

$$T_n = S_n - \frac{(S_{n+1} - S_n)^2}{(S_{n+2} - 2S_{n+1} + S_n)}, \qquad n = 0, 1, \dots$$

This is generalised by Shanks in the determinantal form

$$e_k(S_n) = \frac{\begin{vmatrix} S_n & \dots & S_{n+k} \\ \Delta S_n & \dots & \Delta S_{n+k} \\ \vdots & & \vdots \\ \Delta S_{n+k-1} & \dots & \Delta S_{n+2k-1} \end{vmatrix}}{\begin{vmatrix} 1 & \dots & 1 \\ \Delta S_n & \dots & \Delta S_{n+k} \\ \vdots & & \vdots \\ \Delta S_{n+k-1} & \dots & \Delta S_{n+2k-1} \end{vmatrix}},$$

and the recursive form is given by the ϵ-algorithm of Wynn,

$$\epsilon_{-1}^n = 0, \qquad \epsilon_0^n = S_n, \qquad n = 0, 1, \dots$$

$$\epsilon_{k+1}^n = \epsilon_{k-1}^{n+1} + \frac{1}{\epsilon_k^{n+1} - \epsilon_k^n}, \qquad k, n = 0, 1, \dots$$

This process does not assume dependence on a subsequence such as that used in the Richardson method, and is directly applicable to the case where the expansion is in the form of the differences ΔS_n. [13].

E-algorithm

This is the most general process for the situation in which an expansion in the form of a subsequence $\{x_n\}$ is known. The structure of the sequences S_n is now taken to be of the form

$$S_n = S + a_1 g_1(n) + \cdots + a_k g_k(n),$$

and the transformation is given in its determinantal structure by

$$E_k^{(n)} = \frac{\begin{vmatrix} S_n & \dots & S_{n+k} \\ g_1(n) & \dots & g_1(n+k) \\ \vdots & & \vdots \\ g_k(n) & \dots & g_k(n+k) \end{vmatrix}}{\begin{vmatrix} 1 & \dots & 1 \\ g_1(n) & \dots & g_1(n+k) \\ \vdots & & \vdots \\ g_k(n) & \dots & g_k(n+k) \end{vmatrix}}.$$

The resulting iterative formula consists of the main rule:

$$E_k^{(n)} = E_{k-1}^n - \frac{E_{k-1}^{(n+1)} - E_{k-1}^n}{g_{k-1,k}^{(n+1)} - g_{k-1,k}^{(n)}},$$

and the subsidiary rule

$$g_{k,i}^{(n)} = g_{k-1,i}^{(n)} - \frac{g_{k-1,i}^{(n+1)} - g_{k-1,i}^{(n)}}{g_{k-1,k}^{(n+1)} - g_{k-1,k}^{(n)}} . g_{k-1,k}^{(n)},$$

where the starting values are now

$$E_0^{(n)} = S_n, \qquad n = 0, 1, \dots,$$
$$g_{0,i}^{(n)} = g_i(n), \quad n = 0, 1, \dots, \quad i = 1, 2, \dots.$$

We have shown the construction of these methods, and refer to the texts cited above for the underlying theory and related proofs.

Chapter 5

Existing Algorithms

There are three main algorithms which have been used to obtain a numerical approximation to the solution of (1.1), which we now discuss in detail.

5.1 Approximation methods

5.1.1 Product Euler

We recall the product Euler rule from section (4.8) as

$$\int_0^t p(t,s)q(t,s)y(s)ds \approx \sum_{i=0}^{n-1} \Big\{ p(t_n,t_i)y(t_i) \int_{t_i}^{t_{i+1}} q(t_n,s)ds \Big\}.$$

In [64] the earlier form of equation (1.1), with the kernel

$$K(t,s) = \frac{1}{\sqrt{\pi}} \frac{1}{\sqrt{\log(t/s)}} \frac{s^{\mu-1}}{t^{\mu}}$$

(cf. eq. (1.3)) is to be approximated, for the case when $\mu > 1$. The schemes for the product Euler and product trapezoidal (see below) are applied, and convergence $O(h^1)$ is shown to hold for the product Euler.

The process used in [43] for $\mu > 1$ and [44] for $\mu < 1$ in the approximation of equation (1.1) is implemented by setting $q(t_n, t_i) = t_i^{\mu-1}$ to obtain the weights

for the discretisation $w_i = (t_{i+1}^{\mu} - t_i^{\mu})/\mu$, resulting in the explicit formula

$$\tilde{u}(t_n) = g(t_n) + t_n^{-\mu} \sum_{i=0}^{n-1} w_i \tilde{u}_i. \tag{5.1}$$

In [43], they further require the input function g to contain an algebraic singularity. The equation is transformed by a factor of t^{β}, and proof of convergence $O(h^1)$ is again given. Richardson extrapolation (in its standard format) is used to accelerate the convergence, but aspects of the results indicate that the error expansion requires more detailed investigation.

The extension to the case when $0 < \mu < 1$, and a considerably refined convergence proof, together with the asymptotic error expansion for specific ranges of μ is found in [44]. The product Euler scheme is now supported by the more powerful extrapolation process known as the E-algorithm of Brezinski (described in section 4.10).

The main result, which we use in the following chapter, is given as follows:

Theorem 5.1.1 *Lima/Diogo*

Consider equation (1.1) with $g \in C^2[0,T]$ and $g'(0) \neq 0$. If $0 < \mu < 1$ then the approximate solution $\tilde{u}(t_n)$ defined by (5.1) satisfies the error estimate

$$u(t_n) - \tilde{u}(t_n) = C_{\mu} t_n^{1-\mu} h^{\mu} + O(h),$$

where C_{μ} does not depend on h.

The case where g is non-smooth is placed in a separate corollary.

They further find that the error expansion when $1 < \mu < 2$ also has terms in h^{μ}. Here, the leading error term is, as expected, $O(h)$, but if extrapolation is required for μ occurring in this range, then the full expansion is required.

The special cases when $\mu = 1$ and $\mu = 2$ include a logarithmic term in the error expansion.

5.1.2 Product Trapezoidal

This scheme is used in [64] for the approximation of equation (1.3), taking $\mu > 1$. The scheme is implemented in the usual way (see section (4.8)):

$$u(t_n) + \sum_{i=0}^{n-2} \int_{t_i}^{t_{i+1}} K(t_n, s)u(s)ds + \int_{t_{n-1}}^{t_n} K(t_n, s)\ell_{1,n-1}(s)u(t_{n-1})ds + \int_{t_{n-1}}^{t_n} K(t_n, s)\ell_{2,n-1}(s)u(t_n)ds = g(t_n),$$

where $K(t, s)$ is as given by (5.1.1), and $\ell_{i,n},\ i = 1, 2$ are the linear Lagrange interpolants

$$\begin{aligned} \ell_{1,n} &= (t - t_{n+1})/(t_n - t_{n+1}), \\ \ell_{2,n} &= (t - t_n)/(t_{n+1} - t_n). \end{aligned}$$

The expected convergence $O(h^2)$ is shown, however this is not directly relevant for our purposes.

5.1.3 Hermite-type Collocation

The scheme described in section (4.9) is developed in [26] for the solution of equation (1.1) with $\mu > 1$, setting

$$k_1(t, s) = \frac{s^{\mu-1}}{t^\mu} \quad \text{and} \quad k_2(t, s) = \frac{s^\mu}{t^{\mu+1}}.$$

In the discretisation, the kernels k_1 and k_2 are represented by

$$\begin{aligned} k_1(t_n, t_i + \nu h) &= \left(\frac{\nu + i}{n}\right)^\mu \frac{1}{t_i + \nu h}, \\ k_2(t_n, t_i + \nu h) &= \left(\frac{\nu + i}{n}\right)^{\mu+1} \frac{1}{t_i + \nu h}. \end{aligned}$$

Define for $r = 1, 2$

$$L_{0,r}(n,i) := \frac{1}{n^{\mu}} \int_0^1 (\nu + i)^{\mu-1} l_k(\nu)\, d\nu,$$
$$L_{1,r}(n,i) := \frac{1}{n^{\mu+1}} \int_0^1 (\nu + i)^{\mu} l'_k(\nu)\, d\nu,$$
$$D_{0,r}(n,i) := \frac{1}{n^{\mu}} \int_0^1 (\nu + i)^{\mu-1} d_k(\nu)\, d\nu,$$
$$D_{1,r}(n,i) := \frac{1}{n^{\mu+1}} \int_0^1 (\nu + i)^{\mu} d'_k(\nu)\, d\nu,$$

and our pair of simultaneous equations is

$$u_n - L_{0,2}(n, n-1)u_n - D_{0,2}(n, n-1)hu'_n =$$
$$L_{0,1}(n, n-1)u_{n-1} + D_{0,1}(n, n-1)hu'_{n-1} + \sum_{i=0}^{n-2} \big[L_{0,1}(n,i)u_i +$$
$$L_{0,2}(n,i)u_{i+1} D_{0,1}(n,i)hu'_i + D_{0,2}(n,i)hu'_{i+1}\big] + g_n,$$

$$hu'_n - L_{1,2}(n, n-1)u_n - D_{1,2}(n, n-1)hu'_n =$$
$$L_{1,1}(n, n-1)u_{n-1} + D_{1,1}(n, n-1)hu'_{n-1} + \sum_{i=0}^{n-2} [L_{1,1}(n,i)u_i +$$
$$L_{1,2}(n,i)u_{i+1} + D_{1,1}(n,i)hu'_i + D_{1,2}(n,i)hu'_{i+1}] + g'_n,$$

where $g_n = g(t_n)$ and $g'_n = g'(t_n)$. This simultaneous pair has to be solved at each step, so the method is expensive in terms of computer operations, but the results are a good order of accuracy, and can be justified on that account.

The application of this method to the solution of equation (1.1) is shown in detail in chapter 7. The convergence result for $\mu > 1$ is found in [26].

5.2 Recent work

A further development in the use of the higher order product integration is to be found in [24]: the product trapezoidal rule is applied to equation (1.1), and

the use of graded meshes is employed to improve the quality of the results. Convergence of order $p : p = \mu + \alpha + 1$ is obtained experimentally, for nonsmooth $g(t) := t^{\alpha}\overline{g}(t)$, and possibly order 2 might be attainable, if the appropriate mesh grading is applied.

In [25], with $\mu < 1$, the integration term in (1.1) is separated into two intervals, in a sense anticipating the work of the following chapter: however, the first integral is assumed to be known exactly, for a chosen member of the family of solutions, and the product Euler rule is applied away from the origin. Convergence of order one is obtained, but if the initial interval ϵ is less than h, and expressed as a power of h: $\epsilon = h^{\theta},\ 0 < \theta < 1$, the error is found to be $O(h^{1-\theta})$.

A further paper [20] shows the uniqueness of a specific trajectory away from the origin, and standard numerical methods (Euler, backward Euler and trapezoidal rules) when used away from the origin, are shown to converge to the particular solution $u_c(t)$ defined by the given point $(r, u_c(r))$, where $r \neq 0$.

Analytically, the development of series solutions for various input functions g is shown by Poole [52].

5.3 Perspective

So far, the methods for $\mu < 1$ consist of the low order product Euler rule, which with the use of extrapolation would require many trajectories to obtain any required level of convergence, or suitable accuracy, also the product trapezoidal, with only preliminary evidence of convergence order $1 + \mu$. From the above, it would seem that there is a need for a reliable means of solution of (1.1) which is able to commence at the origin, but provide good quality results with better convergence order and better use of computational time than has hitherto been the case. The concept of separation of the interval of integration is one way of

achieving this: the use of a lower order method commencing at the origin may be combined with extrapolation, and then the (smooth) part of the integral may be implemented by any of the standard methods available.

The way in which this achieved, the selection of primary and secondary methods, and the manner in which extrapolation needs to be used to produce the final convergence result necessary is the subject of the next chapter.

Chapter 6

A Split-interval Scheme

6.1 An Algorithm for Singularity

The idea of separating the approximation into two processes is not competely new: Brunner and van der Houwen suggest that "in actual computation one might apply product integration formulas only in regions where the integrand is not smooth and not in the whole integration interval" [14]; however we believe that it has not previously been used for the solution of a Volterra equation with a kernel weakly singular at the origin, where non-integer terms occur in the error expansion. We also believe that the use of extrapolation processes at the end of the first interval is a new approach, and the consolidated scheme represents a reliable and accurate means of obtaining an approximation to the smooth solution of equation (1.1), and of other equations in this class (i.e. with algebraic singularity at the origin).

The underlying principles of the scheme are introduced in the two papers [20] and [22]: in [20] we show that the uniqueness of a specific trajectory is defined over any interval which does not include the origin, with separation at the rate of $at^{1-\mu}$; in [22] the split-interval method is described, and to this we now add the structured use of the extrapolation process, which ensures the reliability of

the results.

We return to equation (1.1), and take the interval of integration over $[0, \alpha]$ and $[\alpha, T]$,$(t > \alpha)$:

$$u(t) = g(t) + \int_0^\alpha \frac{s^{\mu-1}}{t^\mu} u(s)\, ds + \int_\alpha^T \frac{s^{\mu-1}}{t^\mu} u(s)\, ds, \tag{6.1}$$

and its discrete counterpart

$$\tilde{u}(t_n) = g(t_n) + \sum_{i=0}^{q} h w_i k(t_n, t_i)\tilde{u}(t_i) + \sum_{i=q}^{n} h v_i k(t_n, t_i)\tilde{u}(t_i) + E_{\alpha,\mu}, \qquad n = q+1, \ldots, T,$$

where $E_{\alpha,\mu}$ is the error term and $\alpha = qh$. On closer examination, this is not strictly accurate, as the first interval ought to be implemented by the usual Volterra triangular implicit set of values: we need to re-state (6.1) a little more carefully, as

$$u(t) = g(t) + \int_0^{t_1} \frac{s^{\mu-1}}{t^\mu} u(s)\, ds + \int_\alpha^{t_2} \frac{s^{\mu-1}}{t^\mu} u(s)\, ds,$$

where $t_1 = t \in [0, \alpha]$, and $t_2 = t \in [\alpha, T]$.

Over the first interval, we need a numerical approximation method suitable to commence at the origin, and by this means an approximate value of the solution $\tilde{u}(\alpha)$ is obtained. We may assume that this is not exactly the smooth solution at α, but will lie on *one of the adjacent non-smooth solutions*. So knowing $\tilde{u}(\alpha)$, and using the invariance of the equation for non-smooth solutions, we define

$$\tilde{I}_\alpha = \int_0^\alpha s^{\mu-1}\tilde{u}(s)\, ds,$$

where $\tilde{u}(t)$ is the specific non-smooth solution passing through $\tilde{u}(\alpha)$. Returning to the original equation, we now have an explicit representation of $\tilde{I}_\alpha$ as

$$\tilde{I}_\alpha = \alpha^\mu(\tilde{u}(\alpha) - g(\alpha)).$$

Now for $t_n > \alpha$ we can replace the first summation with $\tilde{I}_\alpha / t_n^\mu$, to obtain the split-interval discretisation

$$\tilde{u}(t_n) = g_n(t_n) + \frac{\tilde{I}_\alpha}{t_n^\mu} + \sum_{i=q}^{n} h v_i \frac{t_i^{\mu-1}}{t_n^\mu} \tilde{u}_i(t_i) + E_{\alpha,\mu},$$
$$n = q+1, \ldots, T,$$

where the v_i are the weights of the method chosen to approximate the second interval. The error term $E_{\alpha,\mu}$ is the subject of section (6.3).

When we consider the extrapolation process applied at α, the construction above remains unchanged, except that the value of $\tilde{I}_\alpha$ will alter, becoming closer to the analytic value I_α.

In previous pages we have discussed certain methods for the solution of VIE's, to assess their suitability for either phase one or phase two of our algorithm. For reasons previously stated, these tend not to coincide: the requirements for the primary rule are highly restrictive, and are governed by the need to find a method which can successfully tackle the singularity in the arguments of the kernel, while for the secondary we can select from the wide range of standard methods available, depending on requirements.

6.1.1 Primary interval

The 'closed' Newton-Cotes methods take the end points of each step as the defining system of quadrature, and are therefore invalid at $t = 0$ for the class of equations in question. The 'open' methods require additional function evaluations, which would need a starting scheme to establish the procedure (as would the closed n-point schemes for $n \geq 3$).

Moving on to the Runge-Kutta schemes, we have to reject any which contain the first end-point of the interval. This excludes many of the classical forms, a few typical examples of which are the Radau I and the Lobatto. The Radau II construction could be of interest, but its implementation in the solution of

equation (1.1) yields a singular matrix over the first step calculation in the two-stage scheme, and near-singular for the three- and four-stage versions.

Next, we take the Gaussian quadrature methods. Here, when the relevant components of an ODE or a VIE are well-behaved, we can expect a high order of accuracy. Of these, the Gauss-Legendre method appears to be the most promising, for several reasons: the nodes are so placed as to give optimal accuracy of order $2m$ for an m-stage method; the first node does not coincide with the end point; and the interval of integration over which the method is constructed enables an afffine transformation of the interval of integration onto the step being processed. However, disappointingly, we find that, as with the Radau II algorithms, when applied to equation (1.1) the two-stage rule results in a singular matrix for the first step, and the three- and four-stage variants are near-singular.

The class of linear multistep methods is also unsuitable, as these require a starting formula, which precludes any of the above methods in the construction, and to utilise any of the possible ideas considered below as a starting scheme for a LM process, while not impossible, would render the resulting error analysis of the tri-formate scheme very unwieldy. The same issue occurs in the case of the open Newton-Cotes rules.

Product Integration

These methods have already been successfully applied to the class of singular equations which is the topic of our investigation, specifically the product Euler rule. Its ability to operate in the context is unquestioned, even if the input function g is also singular at $t = 0$. However, the rate of convergence is low - when $\mu < 1$, the product Euler method is only of order μ (see Lima and Diogo [44]). Further, results although technically convergent are poor, particularly over a long time interval. The extrapolation procedures of section 4.11 need to be used in order to achieve any significant improvement in the accuracy.

The product trapezoidal rule is found in practice to be of much greater ac-

curacy, and is of order $(\mu + 1)$. The convergence analysis is described at the end of this chapter, and owing to the general nature of the cited sources, this is now suitable for the development of product integration methods of higher order. Again, acceleration can be applied to improve the order, to the required level.

Hermite Collocation

In chapter 4 we suggested the method of Hermite as a suitable alternative to deal with the singularity at the origin. It was used previously in this context for $\mu > 1$ [26], and we will again consider its merits as a means of constructing the solution of (1.1) when $\mu < 1$ over the α-interval. It is a collocation method in the fullest sense. The gradient is employed at every step, as well as the function value, a factor which gives us a very clear indication that the path to be pursued will be the smooth solution. Its construction is intricate, but the convergence resulting is of a very high accuracy: we obtain the order 4 for standard application, providing an exceptionally close approximation to the true solution. When this method is applied to equation (1.1) with $\mu = 0.4$, we find the convergence order to be in the region of 3.6, slightly better than the $3 + \mu$ predicted by the work of Lyness and Ninham [48], although it is too early to generalise from this. The step size does not need to be small - good results are obtained, comparable to the findings of [26], with step sizes 0.1, 0.05 and 0.025. We consider this to be a suitable method for the α-interval, if highly accurate results are required.

6.1.2 Convergence Acceleration

As we remarked earlier, this is not an alternative numerical method for obtaining a solution, but works to improve schemes whose rate of convergence is inadequate in the context for which it is to be used. We have principally used the work of Brezinski and Redivo-Zaglia [13], and we also find in Weniger [68] a first class readable explanation of the extrapolation schemes and their construction, with

the emphasis on nonlinear methods. The recent book by Sidi [59] provides a very detailed analysis, with a considerable section devoted to Richardson extrapolation, and the generalisations of this scheme.

To contruct the split-interval algorithm, we can consider the alternative options: (a) to apply the extrapolation process at the end of the first interval, $t = \alpha$, or (b) to apply it at the final time, $t = T$. There are several arguments in favour of the former:

- only one set of error expansion terms has to be considered in contructing the extrapolation algorithm;
- the approximate solution is restrained as early as possible to pursue the required trajectory over the second stage;
- economy of computer time, in the reduction of trajectories required to continue to $t = T$.

In order to decide which method of extrapolation to use, we have to assess what is to be achieved. Here, we are looking for a way to eliminate the successive terms of an error expansion of the form

$$|e| = a_1 h^{\mu} + a_2 h + a_3 h^{\mu+1} + a_4 h^2 + \dots \tag{6.2}$$

for $\mu < 1$, which is the appropriate expansion for the product Euler method, and for the product trapezoidal where $a_1 = a_2 = 0$.

The Richardson scheme is the best known and usually the first choice; but as it applies to an expansion of integer powers of h, we are unable to use it in this case, and other standard methods are equally unsuitable. We refer to the references cited above for a full account of the various procedures.

There is an alternative, in the E-algorithm of Brezinski, which may be applied to any case where there is a known expansion for the error term. However, it is very detailed to construct, and as stated by Brezinski [13] (p.55).

"... the drawback of such a generality is that, in a particular case, it will be less powerful (in terms of number of arithmetic operations and storage requirements) than an algorithm particularly adapted to that case."

The E-algorithm is used in [44], applied to the results from the product Euler method applied to equation (1.1) over the interval $t = [0, 1]$. While it does obtain a marked improvement in the error, we feel that this might be achieved by other means. We shall employ a simpler method, which is constructed upon the basis of a modification of the Richardson scheme, and which we find is comparable to the accuracy of the E-algorithm. Our new scheme cannot be a fully recursive algorithm, as the terms are not in uniformly ascending powers of h. We introduce the system of equations

$$Y_1^{(n)} = Y_0^{(n)} + a_1 h_n^{\mu} + a_2 h_n + a_3 h_n^{\mu+1} + a_4 h_n^2 + \ldots$$

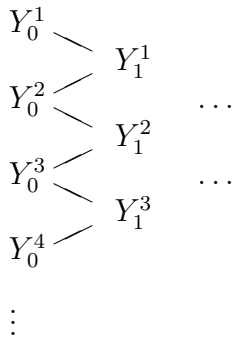
where $n = 1, 2, \ldots, p$ for a $(p-1)$-stage application. The $Y_0^{(n)}$ are the approximate solutions evaluated at stepsize h_n where $h_n = h_{n-1}/2$, and as with Richardson we may eliminate the a_1 term, and reduce the number of equations by one, the difference now being that we require to factor by $(1 - 2^{\mu})$. The next stage is to obtain the terms $Y_2^{(n)}$ with a standard Richardson format eliminating a_2, the principal error term being O(h). This can now be developed into a sequential process,

$$Y_i^{(n)} = Y_{i-1}^{(n)} + e_i$$

where the e_i terms are obtained by eliminating the most significant error term, which will alternate between $h^{\mu+k}$ $(k = (i-1)/2)$ and h^k $(k = i/2)$. Clearly this is unsuitable for a recursive formula, but we can use the semi-recursive process:

$$Y_n^{(j+1)} = \begin{cases} \dfrac{Y_n^{(j)} - 2^{j+\mu} Y_{n+1}^{(j)}}{1 - 2^{j+\mu}}, & j = 0, 2, 4, \ldots, \\ \dfrac{Y_n^{(j)} - 2^{j} Y_{n+1}^{(j)}}{1 - 2^{j}}, & j = 1, 3, 5, \ldots. \end{cases}$$

As with other similar schemes, we can portray this in columnar form, with the number of values reducing by one at each stage:



The usual texts describing Richardson extrapolation present the related convergence proofs which apply solely to the series expansion which contains integer powers. However, we find in Sidi (2003) [59] the application to non-integer powers, and this provides the rigorous foundation for our construction. Alternatively, the proofs presented for the integer case by Brezinski and Redivo-Zaglia ([13], Theorems 2.15 and 2.17) may be extended to the non-integer case by separating the expansion into two series, and using induction.

6.1.3 Secondary schemes

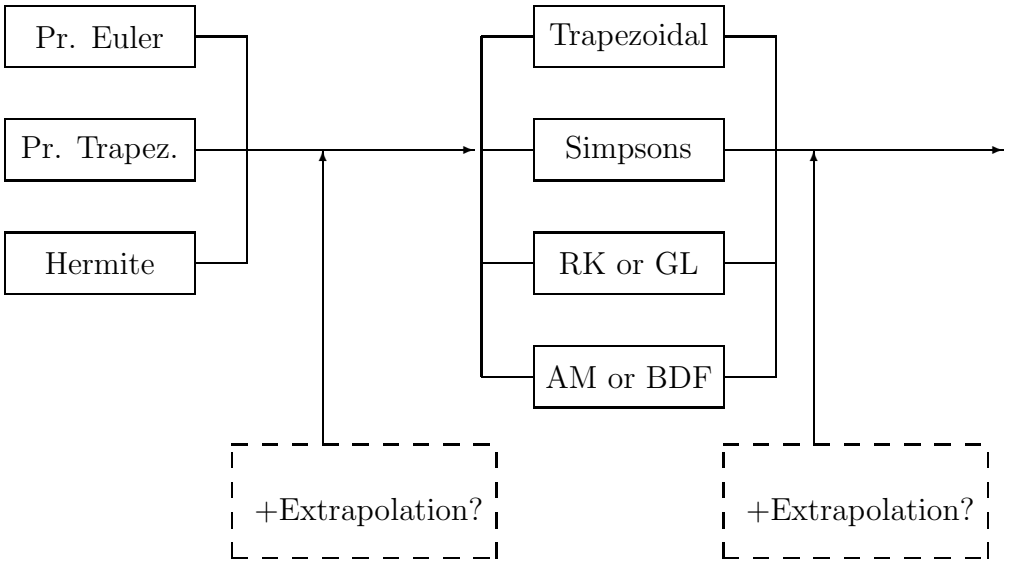
The choice of secondary scheme is much wider, and depends on the properties required: this may be simple construction, economical in terms of floating point operations, or higher order and better accuracy, or perhaps a preferred method of the user. While the explicit and implicit Euler schemes do give reliable results in terms of the order $O(h)$, the actual error values are unacceptably large. When we move to the trapezoidal rule, however, not only do we have the order $O(h^2)$, but also a marked increase in accuracy. Simpson's rule is also a possibility if order $O(h^4)$ is required, but there would be no advantage in proceeding to the

three-eighth rule, which is of the same order. However, this order can be better achieved by other means.

We can also consider any of the lower order Runge-Kutta schemes: the restraint on the first endpoint is now lifted. If a high accuracy is required over the second interval, the Gauss-Legendre rules, dismissed for the α-interval, give excellent results of orders 4 and 6 for the 2- and 3- stage schemes respectively.

Of the linear multistep methods, when used in the context of the VIE, there appears to be a question as to the stability of the Adams-Bashforth and Adams-Moulton processes. The particular case of the backward differentiation formulae has been shown to be exceptionally stable, even in problems where there is a high Lipschitz constant [14]. We shall give examples of the AM and the BDF construction.

Our possibilities for the combined algorithm have narrowed down to the following representation:



Can the two stages be linked up without special treatment? Provided both schemes are single-step methods, we do not see any problem. The solution value of the first scheme at $t = \alpha$ (whether extrapolated or not) is taken up as the starting value for the second stage. This would also apply if points internal to

the step are used, as in the Runge-Kutta methods. If, however, the choice of secondary methods is of linear multistep type, then the single solution value at α is insufficient. Two or more solution values are required if the transition is to be seamless, alternatively the use of a starting procedure is required. This will happen if an extrapolation method is used directly after the first stage. An alternative would be the multiple use of acceleration processes - extrapolation could be taken at two, three or more points, spaced on the basis of the stepsize of the secondary scheme.

6.2 Construction of the split-interval algorithm

We have outlined in Section (6.1) the possible methods for the primary and secondary stages of our dual method, together with the need for improving the order by means of acceleration methods. We will now see how it is possible to construct a numerical solution to some given accuracy, say $O(h^m), m \in Z$. We recall that the error expansions we are dealing with are in powers of h in the set $P\{\mu, 1, 1+\mu, \dots\}$

(a) The simplest idea is to take primary and secondary methods both of order m. We note that either method may be of order $p > m$, but this would still result in a final solution of order m, such as would be achieved by the use of product integrational methods, which we have shown to be of non-integer order. Here we suggest the optimal value to be $p : p \in P, \lfloor p \rfloor = m$.

(b) We can take a primary method order p and a secondary method order $q : q \in Z, q > p$ with extrapolation at $t = T$ to improve the order to m, and allow the extrapolation process to take care of terms $h^p \dots h^m$ in the error expansion. If, as in the product Euler method, $p = \mu (< 1)$, then we require $2m-1$ stages of extrapolation, or for the general case, $2(m-\lfloor p \rfloor)-1$ stages.

(c) The extrapolation may be used at the point α, at which the primary method ends. We again need $2(m - \lfloor p \rfloor) - 1$ acceleration stages, if the secondary method is of order m.

(d) We might use extrapolation both at α and at T. This is advantageous if a very high accuracy is required in the final solution, or if the final time T is large. Let us suppose the required order is M, and as before, we have a primary method of order p, and a secondary method order m. The extrapolation at α would be $2(m - \lfloor p \rfloor) - 1$ stages, eliminating error terms below h^m, and a further $2(M - m)$ stages at $t = T$.

We summarize this in the table

Sch.	*Primary Order*	*Extrap. stages*	*Secondary Order*	*Extrap. stages*	*Final Order*
a	$p : p < m$		m		p
b	$p : p < m$		$q : p < q \leq m$	$2(m - \lfloor p \rfloor) - 1$	m
c	$p : p < m$	$2(m - \lfloor p \rfloor) - 1$	m		m
d	$p : p < m < M$	$2(m - \lfloor p \rfloor) - 1$	m	$2(M - m)$	M

Of these four schemes, (a), (b) and (d) (described earlier in this section) were investigated during the developmental process, and we will now look at (c) in detail, implemented using the product Euler method over the first interval, and the trapezoidal rule for the second, with $g(t) = 1 + t$, $\alpha = 0.5$ and $\mu = 0.4$. Consider the approximation over the first interval, taken over m stepsizes, in this case $m = 4$. Define $h^{(1)}_{max} = 1/20$, say, and further values $h = h^{(1)}_{max}/2^i$, $i = 1, 2, 3$. The results at $t = \alpha$ are then accelerated using the modified Richardson scheme described above, and the second interval is then implemented using the trapezium rule at stepsize $h = h^{(1)}_{max}$; this process is then repeated twice more, using $h^{(2)}_{max} = 1/40$ and $h^{(3)}_{max} = 1/80$, and the extrapolated result in each case is now the starting point for the trapezoidal rule at $h = h^{(2)}_{max}$ and $h = h^{(3)}_{max}$. Clearly, this involves some repetition of calculation, and becomes in effect a

five-level extrapolation, curtailed at the third level, as tabulated below, with $e_{i,n},\ i = 0, \ldots, 3,\ n = 1, \ldots, 6$ defined as the error after i levels of extrapolation.

Extrapolation errors				
h	$e_{0,n}$	$e_{1,n}$	$e_{2,n}$	$e_{3,n}$
$1/20$	$-7.1472e-1$	$-1.1564e-1$	$-9.6285e-3$	$-1.6226e-5$
$1/40$	$-5.6966e-1$	$-6.2637e-2$	$-3.6586e-3$	$4.3660e-6$
$1/80$	$-4.4689e-1$	$-3.3148e-2$	$-1.3836e-3$	$2.6690e-6$
$1/160$	$-3.4670e-1$	$-1.7266e-2$	$-5.2264e-4$	
$1/320$	$-2.6693e-1$	$-8.8941e-3$		
$1/640$	$-2.0445e-1$			

Now we have the values $Y_3^{(n)}$ as the given starting values for the second interval, and continuing with the trapezoidal rule, we find as expected that the final results at $T = 10$ are of order 2:

Errors at t=T	
h	*Error*
$1/20$	$-2.6394e-3$
$1/40$	$-6.0993e-4$
$1/80$	$-1.4302e-4$

6.3 Error analysis

6.3.1 Second interval

In order to create a rigorous framework for applying an arbitrary method over the second interval, we recall that for the Volterra equation in which all terms are of an acceptable order of continuity, the following holds: if the integration term of a VIE is approximated by a method of order p, then the approximation

to the solution of the equation is convergent to the true solution, the convergence also being of order p.

Thus, following the treatment by Linz [46] for the generic Volterra equation

$$\phi(t) = \psi(t) + \int_0^t K(t, s, \phi(s))ds, \tag{6.3}$$

we require that

1. $\psi(t)$ be continuous over the interval $0 \leq t \leq T$,

2. $K(t, s, \phi)$ be continuous over $0 \leq s \leq t \leq T$,

3. the Lipschitz condition on the third argument of the kernel

$$|K(t, s, \phi_1) - K(t, s, \phi_2)| \leq L|\phi_1 - \phi_2|$$

be satisfied, for all $0 \leq s \leq t \leq T$, and all $\phi_1,\ \phi_2$.

If these conditions are met, then uniqueness of the solution to (6.3) is assured, and further, "...the analysis of the numerical methods will utilize these assumptions and, strictly speaking, holds only when they are satisfied." [46].

We now apply an approximation rule based on a quadrature method of order p, and weights w_i to the Volterra equation (6.3) to obtain the related discrete equation

$$\tilde{\phi}_n = \psi(t_n) + h \sum_{i=0}^{n} w_{ni} K(t_n, t_i, \tilde{\phi}_i), \tag{6.4}$$

where $t_n = nh, \quad n = 0, 1, \ldots T/h$ and $\tilde{\phi}_n$ is the approximation to the solution evaluated at t_n. We note that inclusion of the second subscript to the weights indicates that they now have to be applied (and in more accurate methods calculated) over the summation in i for each step increment in the approximation $\tilde{\phi}_n$.

Convergence: the traditional approach

For the second interval, we are now in a position to use the standard arguments relating to the convergence of an arbitrary approximation method (5.11) to the solution of the conventional Volterra equation (5.10). We summarize the results as given in [46], with the main theorem and associated definitions:

Definition 6.3.1 *Let ε_i be the discretisation error, where*

$$\varepsilon_i = \tilde{\phi}_i - \phi(t_i).$$

Definition 6.3.2 *A VIE approximation method is convergent if*

$$\lim_{h\to 0}\left\{\max_{0\le i\le N}|\varepsilon_i|\right\} = 0.$$

Definition 6.3.3 *A VIE method is convergent of order p if*

$$\max_{0\le i\le N}|\varepsilon_i| \le Mh^p$$

for some $M < \infty$ and for the maximum possible value of p.

Definition 6.3.4 *For a rule whose weights are w_i, the local consistency error is $\delta(h, t_n)$ where*

$$\delta(h,t_n) = \int_0^{t_n} K(t_n, s, \phi(s))ds - h\sum_{i=0}^{n} w_{ni}K(t_n, t_i, \phi(t_i)).$$

Definition 6.3.5 *For a given class of equations of form (6.3), if*

$$\lim_{h\to 0}\max_{0\le i\le N}|\delta(h,t_n)| = 0,$$

then the approximation method is said to be consistent.

Definition 6.3.6 *If there is a constant C which exists for a particular class of equations such that*

$$\max_{0\le i\le N}|\delta(h,t_n)| \le Ch^p,$$

then the method is said to be consistent of order p for such equations.

Theorem 6.3.1 *General Convergence Theorem (in Linz [46])*
Taking the approximate solution of (6.3) by (6.4), with the assumptions

1. *the solution $\phi(t)$ of (6.3) and the kernel $K(t, s, \phi)$ are such that the approximation method is consistent of order p with (6.3),*

2. *the weights satisfy*
$$\sup_{n,i} |w_{ni}| \leq W < \infty,$$

3. *the starting errors $\tilde{\phi}_i - \phi(t_i)$, $i = 0, 1, \ldots, r-1$ go to zero as $h \to 0$, and hence*
$$\lim_{h \to 0} \sum_{i=0}^{r-1} |\tilde{\phi}_i - f(t_i)| = 0,$$

then the method is a convergent approximation method. Also, in the absence of starting errors, the order of convergence is at least p.

In arriving at this conclusion, the formula for the discretisation error ε_i is given as
$$|\varepsilon_i| \approx \frac{hWL}{1 - hWL} \left\{ \max_{0 \leq i \leq n} |\delta(h, t_n)| + hWL \sum_{i+0}^{r-1} |\tilde{\phi}_i - \phi(t_i)| \right\} \tag{6.5}$$
for $h < 1/WL$, $W = max\{w_i\}$, L is the Lipschitz constant for a particular class of equations, h is the step length, and the other terms as defined above. The formula (6.5) as presented by Linz includes an expression for the starting error, shown as the second term in braces, but we must emphasize that this is the error due to a starting scheme which is required for some method (such as the Newton-Cotes or linear multistep) where the rule itself requires solution values at more than one point to enable the first application of such a rule. If the stepsize h changes, then so does the interval over which the starting method is effective. This is in contrast to the split-interval method we use, for which the initial interval $[0, \alpha]$ remains constant while the stepsize is progressively reduced.

In the above, we have a general theorem which proves convergence, but does not assist in explaining any connection between the error and the parameter

α of our split-interval scheme, nor with the parameter μ of equation (6.3). In appendix (A) we adopt a less orthodox approach, which neverthless allows an understanding of these questions.

Application

We return to equation (1.1), and make the following substitutions: $\tau = t - \alpha$, $\sigma = s - \alpha$ to obtain

$$\begin{aligned} u(\tau+\alpha) &= g(\tau+\alpha) + \frac{I_\alpha}{(\tau+\alpha)^\mu} + \int_\alpha^{\tau+\alpha} \frac{s^{\mu-1}}{(\tau+\alpha)^\mu} u(s)\, ds, \\ &= g(\tau+\alpha) + \frac{I_\alpha}{(\tau+\alpha)^\mu} + \int_0^{\tau} \frac{(\sigma+\alpha)^{\mu-1}}{(\tau+\alpha)^\mu} u(\sigma+\alpha)\, d\sigma. \end{aligned}$$

Now define

$$\begin{aligned} \overline{y}(\tau) &= u(\tau+\alpha), \\ \overline{f}(\tau) &= g(\tau+\alpha) + \frac{I_\alpha}{(\tau+\alpha)^\mu}, \end{aligned}$$

to give the Volterra equation

$$\overline{y}(\tau) = \overline{f}(\tau) + \int_0^{\tau} \frac{(\sigma+\alpha)^{\mu-1}}{(\tau+\alpha)^\mu} \overline{y}(\sigma)\, d\sigma,$$

where $0 \le \tau \le T - \alpha$.

The integral is still non-zero at the lower limit, so as a final precaution let $y(t) = \overline{y}(t) - \overline{y}(0)$ and $f(t) = \overline{f}(t) + y(0)/\mu$, to give

$$\begin{aligned} \overline{y}(\tau) &= \overline{f}(\tau) + \frac{\overline{y}(0)}{\mu} + \int_0^{\tau} \frac{(\sigma+\alpha)^{\mu-1}}{(\tau+\alpha)^\mu} (\overline{y}(\sigma) - \overline{y}(0)) d\sigma, \\ y(\tau) &= f(\tau) + \int_0^{\tau} \frac{(\sigma+\alpha)^{\mu-1}}{(\tau+\alpha)^\mu} y(\sigma)\, d\sigma. \end{aligned}$$

These substitutions do not alter the behaviour of equation (1.1) over $t \in [\alpha, T]$, involving a straightforward shift of origin by α and $u(\alpha)$ in the original ordinates.

The smooth region of equation (1.1) now conforms fully to the requirements above, the Lipschitz constant being $1/\alpha$, and we note that if I_α is the analytic integral over the first interval, the solution to (6.4) is the smooth solution, whereas

if it is replaced by an approximation, $\tilde{I}_\alpha$, then the solution represented will be a specific member of the non-smooth family. This is dealt with more fully in subsection (6.3.3) below.

6.3.2 First interval

We use the product Euler method as the primary scheme for many of the examples in the next chapter. While the method is elementary of its type, its inclusion as part of the split-interval scheme is by no means trivial. The issues dealt with at this level enable the development of a secure and reliable approach, capable of extension to higher orders.

We now extend the analysis to include the implementation of the product Euler method over the first interval. The weights w_i are the first order product integration expressions

$$w_i = \int_{t_i}^{t_{i+1}} k(t,s)ds,$$

and the second interval weights v_i are, as above, the trapezoidal weights, to give

$$\tilde{u}(t_n) = g(t_n) + \sum_{i=0}^{m} w_i\tilde{u}(t_i), + \sum_{i=m}^{n} v_i k(t_n,t_i)\tilde{u}(t_i) \qquad n = m+1,\ldots,T.$$

The approximate solution of equation (1.1) by the product Euler method has been developed by Lima and Diogo (see [43] and [44]), and its convergence behaviour is known, and the proofs well established. We present the main results the first of which we introduced earlier in Chapter 3:

Theorem 6.3.2 (Existence and uniqueness) *If $0 < \mu \leq 1$ then equation (1.1) has a family of solutions $u \in C^0[0,T]$ given by the formula*

$$u(t) = c_0 t^{1-\mu} + g(t) + \gamma + t^{1-\mu}\int_0^t s^{\mu-2}(g(t)-g(0))ds, \tag{6.6}$$

where $\gamma = 0$ if $\mu = 1$, or $\gamma = g(0)/(\mu-1)$ for $0 < \mu < 1$.

Theorem 6.3.3 (Convergence) *Assume that $g \in C^1[0,T]$,for $\mu > 0$ and $\mu \neq 1$, then the approximate solution defined by the product Euler scheme*

$$\tilde{u}(t_n) = g(t_n) + t^{-\mu} \sum_{i=0}^{n-1} \tilde{u}(t_i) \int_{t_i}^{t_{i+1}} s^{\mu-1} ds \tag{6.7}$$

converges to the particular true solution 6.6 of (1.1) for which $c_0 = 0$.

Theorem 6.3.4 *Define $e_n = |u(t_n) - \tilde{u}(t_n)|$. For $0 < \mu < 1$ the approximate solution defined in 6.7 above satisfies the error estimate*

$$e_n = C_\mu t_n^{1-\mu} h^\mu + 0(h),$$

where C_μ is independent of h.

These results are extended to the cases for which $\mu = 1$, when the error term contains a logarithmic element, and for the case when $g(t)$ is such that the solution is not sufficiently smooth.

Hence we establish the viability of the product Euler method as a suitable approximation over the primary interval. Similar convergence proof for the product trapezoidal has not previously been available. We show in section (6.4), using the results of de Hoog and Weiss [34], based on the earlier work of Lyness and Ninham [48] that convergence order of $1 + \mu$ is obtained. The Hermite-type method used in [64] for $\mu > 1$ does not at the time of writing have a convergence proof for $\mu < 1$, but empirical results show convergence of order at least $3 + \mu$.

6.3.3 Propagated error

We can now express the error term at the end of the first interval $t = \alpha = qh$ as

$$e_\alpha = C_\mu \alpha^{1-\mu} h^\mu + O(h),$$

and the equation for this particular solution over $t \in [\alpha, T]$ becomes

$$u_1(t) = g(t) + t^{-\mu} \tilde{I}_\alpha + t^{-\mu} \int_\alpha^t s^{\mu-1} [u(s) + e_\alpha(h, \mu, s)] ds,$$

where $e_\alpha(h,\mu,s)$ is the error term arising from the first interval, representing the asymptotic error expansion (6.3), and the dependency on s is of the form $s^{1-\mu}$. If this particular trajectory is followed accurately, we have

$$u_1(t) = g(t) + t^{-\mu}\tilde{I}_\alpha + t^{-\mu}\int_\alpha^t s^{\mu-1}[u(s) + s^{1-\mu}e_\alpha(h,\mu)]ds,$$

to give

$$u_1(t) = g(t) + t^{-\mu}\tilde{I}_\alpha + t^{-\mu}\int_\alpha^t s^{\mu-1}u(s)ds + t^{-\mu}(t-\alpha)e_\alpha(h,\mu).$$

As the asymptotic expansion $e_\alpha(h,\mu)$ is independent of the time variable, (the subscript is there to indicate that the term is generated at α, not any relation involving subsequent time values), it is this expression which is propagated as a result of the error at the end of the first interval, defining the non-smooth term over the second interval, and we further note that

$$t^{-\mu}(t-\alpha)e_\alpha(h,\mu) \leq t^{1-\mu}e_\alpha(h,\mu).$$

6.3.4 Combined Error

This yields the combined error estimation E at $t = t_j \in [\alpha, T]$ as

$$E = e_\alpha(h,\mu)t_j^{1-\mu} + (t_j - \alpha)\sum_{i=m}^{\infty} b_i h^i,$$

where the first term includes the asymptotic expansion of the primary scheme, and the second term is the error of the secondary scheme of order m, converging to the non-smooth solution $u_1(t)$ identified above.

Suppose that the primary scheme is the product Euler, and the secondary is the trapezoidal rule. Then we have

$$e_\alpha(h,\mu) = a_1h^\mu + a_2h + a_3h^{1+\mu} + a_4h^2 + \ldots$$

where the multipliers a_r and b_i are independent of t_j, h and α, and the second series commences at $m = 2$.

As it stands, the scheme retains the weakness of the first interval approximation, and the solution at α needs to be sufficiently improved, at least to match the order m, (or higher), of the second interval method. We introduce the extrapolation applied to the solution set at α over successively reduced values of h: the error term at the end of the first interval is now, by construction, of order m, and this is in part the reason behind our choice of a simple linear form of extrapolation, since we require to know, not only the resulting power of h, but also how the overall scheme behaves in relation to α.

6.3.5 The effect of extrapolation

We selected a modification of the Richardson process mainly for its particular application to the series which we wish to accelerate. A further benefit is that it is a linear method, which will retain other factors we might be interested in. The work of Lima and Diogo provides the leading term of the error expansion, and the assumption is made - borne out by numerical evidence during this investigation - that the further terms of the expansion contain the structure of a Frobenius series (as well as Taylor). This is supported in the work of Lyness and Ninham [48], which will be examined fully later.

By construction, being linear, the extrapolation does not alter the nature of this series, but what does change is the value of the coefficients, and in particular the leading residual coefficients. If we take the unprocessed error expansion for the product Euler, as the series to be extrapolated, we have

$$S_h = S + a_1 h^{\mu} + a_2 h + a_3 h^{1+\mu} + a_4 h^2 + \ldots,$$

and $h_i = h_{i-1}/2$, we obtain the first coefficient (of h) of the new series as $a_2(1 - 2^{\mu-1})/(1 - 2^{\mu})$. As μ approaches unity, we no not have a problem, the numerator tending to zero as the denominator nears -1. However, when μ is small - say 0.1 - the magnitude of the denominator is small at .07, while the numerator approaches unity, so the effect on the leading term after the first extrapolation

is considerable. However, provided at least one more level is taken, except in extreme cases, the offending term can be eliminated.

Subsequent levels of extrapolation do not have this problem - the denominators are successively (-1), $(1-2^{1+\mu})$, $(-3), \ldots$ and nor would an acceleration process for any primary method (such as the product trapezoidal) where the leading power is greater than unity.

We find at each level, that the correlation of the coefficients $\{a_i\}$ with other values , such as α or μ, are unaffected by the extrapolation process, and hence the propagated error itself will retain its dependence on such parameters.

6.3.6 Summary

In conclusion of this section, we have considered the error of a split-interval scheme for solution of equation (1.1). We have taken results already in place for the product Euler method from Lima and Diogo [44]. In the next section we show, in principle, that product integration methods of higher order can be introduced, commencing at the origin, using the generic system constructed by Lyness and Ninham [48], developed for product integration rules applied to integral equations by de Hoog and Weiss [34]

We have shown that the trapezoidal rule is a suitable choice for the second interval approximation, and traced the error term from the primary method, and the way in which it is propagated into the second interval. The second interval error in isolation is treatable by the normal convergence arguments we have given, but the very general nature of this approach leaves several key questions unanswered: in particular, can we establish a link between the error and the length of the initial interval, α, and further, between the error and the parameter μ? In Appendix (A) we introduce a less orthodox method, which investigates the particular case of the second interval for equation (6.1), and we find that the terms h and α are linked in such a way that the order of a method as a power of

h is matched with the same inverse power in α; further, we are able to establish a link between the error and the value of μ.

We have introduced a convergence acceleration process at the end of the first interval, the related error analysis being available in e.g. [59] and the behaviour of the effect caused by α and μ can be traced through the extrapolation process.

In the next chapter, we will give some examples to illustrate these results.

6.4 Product Trapezoidal Rule

6.4.1 Lyness and Ninham

Our understanding of the analysis of the product trapezoidal rule as applied to equation (1.1) depends on the key paper by Lyness and Ninham [48] (1967), which is quoted as the authoritative source for the error expansion relating to the presence of certain forms of singularity as recently as 1997 [67] and mentioned in Sidi (2003). We summarize the content, simplifying to include only one (of four) types of singularity. Specific references included are Lighthill [42], for the expansion of the Fourier transform of the generalized function, and Whittaker and Watson [69] concerning the generalized *zeta*-function, and the Fourier theorem in the manner described below.

One form of the Fourier theorem may be expressed as follows (subject to certain conditions):

$$f(x) = \sum_{r=-\infty}^{\infty} \exp(-2\pi irx) \int_0^1 f(t) \exp(2\pi irt) dt.$$

No condition of bounded variation applies to $f(x)$. Re-arrangement yields

$$f(t_j) - \int_0^1 f(t)dt = \sum_{r=-\infty}^{\infty}{}' \exp(-2\pi irt_j) \int_0^1 f(t) \exp(2\pi irt) dt,$$

where the prime denotes omission of the term for which $r = 0$.

Adding linear combinations with weights $a_j,\ j = 1, 2, \ldots, m$ we have

$$\sum_{j=1}^{m} a_j f(t_j) - \int_0^1 f(t)dt = \sum_{r=-\infty}^{\infty}{}' \left\{ \sum_{j=1}^{m} a_j \exp(-2\pi i r t_j) \right\} \int_0^1 f(t) \exp(2\pi i r t) dt,$$

where

$$\sum_{j=0}^{m} a_j = 1.$$

This is described as the fundamental summation formula, and it is noted that the Poisson formula is a special case.

For a particular rule, denoted Rf,

$$Rf(x) = \sum_{j=1}^{m} a_j f(t_j), \qquad \sum_{j=1}^{m} a_j = 1,$$

the error functional is then $Rf - If = Ef$, where

$$Ef = Rf - If = \sum_{r=-\infty}^{\infty}{}' d_r(R) \int_0^1 f(t) \exp(2\pi i r t) dt,$$

$$d_r(t) = \sum_{j=1}^{m} a_j \exp(-2\pi i r t_j) = R(\exp(-2\pi i r x)),$$

a coefficient which depends only on the rule.

The next section of [48] (not shown here) develops Ef for a generalized trapezoidal rule, and shows how the construction applies to more intricate methods.

Expansions of Euler-McLaurin type

The expansion follows from the modified Poisson summation formula

$$\int_0^1 f(t) \exp(2\pi i r t) dt = \frac{f(1) - f(0)}{2\pi i r} - \frac{f'(1) - f'(0)}{(2\pi i r)^2} + \ldots + (-1)^w \frac{f^{(w)}(1) - f^{(w)}(0)}{(2\pi i r)^{w+1}} + (-1)^{w+1} \int_0^1 \frac{f^{(w+1)}(t) e^{2\pi i r t}}{(2\pi i r)^{w+1}} dt. \quad (6.8)$$

Inserting this into the error formula for the end-point trapezoidal rule

$$R^{(m,1)}f - If = \sum_{r=-\infty}^{\infty}{}' \int_0^1 f(t)\exp(2\pi irmt)dt$$

and summing over r yields the traditional form of the Euler-MacLaurin summation formula.

They then define formally the *zeta*-function, generalized *zeta*-function and periodic generalized *zeta*-function.

Equation (4.23) in [48] then gives the traditional form of the Euler-McLaurin formula:

$$R^{(m,1)}f - If = 2\sum_{n=1}^{\infty}(-1)^{n-1}\frac{\zeta(2n)}{(2\pi m)^{2n}}[f^{(2n-1)}(1) - f^{(2n-1)}(0)].$$

Section 5(a) deals with numerical analysis of integration rules of specified degree, and 5(b) with stochastic processes, which are not directly relevant for our purposes.

Section 6 of [48] ('Lighthill's Procedure') deals specifically with singularities. If $f(x)$ or its early derivatives are discontinuous in the interval $0 \leq x \leq 1$ then the theory of sections 3,4 and 5(a) of [48] is no longer valid. Section 2, however, is still applicable. They look to develop asymptotic expansions more suited to computational purposes when $f(x)$ has a simple algebraic or algebraico-logarithmic singularities in the interval.

They consider singularities of the form

$$f(x) = x^{\beta}(1-x)^{\omega}|x-t_k|^{\gamma}sgn(x-t_l)|x-t_l|^{\delta}h(x),$$

where $h(x)$ and its derivatives are continuous in the interval $0 \leq x \leq 1$, and β, ω, γ and δ are non-integers. In the following, we extract only those parts of each formula which relate to singularities of the form t^{β}, i.e. $\omega = \gamma = \delta = 0$.

We require an asymptotic expansion for the Fourier transform

$$g(r) = \int_0^1 f(t)\exp(-2\pi irt)dt$$

as the expansion (6.8) [(4.1) of [48]], which was obtained by integration by parts, is not valid for this function. They utilise the 'powerful, simple and systematic' method of Lighthill [42], who uses generalized function theory.

We require the FT of the generalized function

$$\phi(x) = f(x)H(x)H(1-x),$$

where

$$\begin{aligned} H(x) \quad &= 0, \qquad && x < 0 \\ &= 1/2, && x = 0 \\ &= 1, && x > 0. \end{aligned}$$

$\phi(x)$ coincides with $f(x)$ on the interval of integration, and is zero elsewhere.

Corresponding to each singularity t_j they construct an 'approximating' function $F_j(x)$ with the following properties:

(i) $\phi(x) - F_j(x)$ has an absolutely integrable Nth derivative in a neighbourhood of $x = t_j$

(ii) $F(x)$ is a linear combination of functions of types $|x-t_j|^\alpha$, $|x-t_j|^\alpha sgn|x-t_j|$, $|x-t_j|^\alpha ln|x-t_j$, $|x-t_j|^\alpha ln|x-t_j|sgn|x-t_j|$ and $|x-t_j|^\alpha ln|x-t_j|sgn|x-t_j|$ (the first of these only is required for equation (1.1)),

Theorem 6.4.1 *If $F_j(x)$ $j = 0, 1, \ldots, m$ satisfy these conditions and have as their Fourier transforms $G_j(r)$:*

$$G_j(r) = \int_{-\infty}^{\infty} F_j(t) \exp(-2\pi irt) dt,$$

then an asymptotic expansion for $g(r)$ is

$$g(r) = \sum_{j=0}^{m} G_j(r) + O(|r|^{-N}) \text{ as } r \to \infty. \tag{6.9}$$

We limit the singularities in $F(x)$ above to that at $t = 0$. Define a function $\phi_0(x)$, continuous and with continuous derivatives at $t = 0$, as follows:

$$f(x) = x^\beta \phi_0(x).$$

The approximating function $F_0(x)$ (satisfying conditions (i) and (ii) above) is then constructed by retaining only the first N terms in the power series expansion of $\phi_0(x)$ about $x = t_0 = 0$. Thus specifically

$$F_0(x) = \sum_{s=0}^{N-1} \frac{\phi_0^{(s)}(0)}{s!} x^{\beta+s} H(x).$$

The Fourier transform of the individual terms are given by Lighthill [42], p.43. Let

$$h(\beta, r) = \frac{\beta!}{(2\pi i r)^{\beta+1}} \qquad \text{where} \qquad \beta! = \Gamma(\beta + 1).$$

Then we have

$$\int_{-\infty}^{\infty} x^{\beta+s} H(x) \exp(-2\pi i r x) dx = h(\beta + s, r),$$

and other singularity types are developed in a similar way. Substitution into equation (6.9) yields

$$\int_0^1 f(x) \exp(-2\pi i r x) dx = \sum_{s=0}^{N-1} \frac{1}{s!} \{\phi_0^{(s)}(0) h(\beta + s, r)\} + O(|r|^{-N}).$$

This result provides the basis for the subsequent analysis. Section 7 deals with the Euler-McLaurin formula for endpoint singularities, with the aim of finding an asymptotic expansion for the error function associated with

$$f(x) = x^{\beta}(1 - x)^{\omega} h(x),$$

where $h(x)$ and its derivatives are continuous over the appropriate interval of integration.

Again allowing $\omega = 0$, they obtain the error function $E^{[m,\alpha]}$ for an arbitrary trapezoidal rule $R^{[m,\alpha]}$, leading to the endpoint trapezoidal rule such that

$$E^{[m,1]} = \sum_{s=1}^{N-1} \frac{\phi_0^{(s)}(0)}{s!} \frac{\zeta(-\beta - s)}{m^{\beta+s+1}} + O(m^{-N}),$$

at which point we leave the Lyness/Ninham paper, and consider specifically product integration methods in the following.

6.4.2 de Hoog and Weiss

We follow a similar pattern with this paper [34], summarizing to include only one (of four) types of singularity. Note the change of notation - now $f(t)$ is smooth, and $g(t)$ is absolutely integrable on $0 \leq t \leq 1$.

In section 2 they define the product integration rule, where

$$\omega(t) = \prod_{k=1}^{n}(t - u_k),$$

and the Lagrange polynomials of degree n:

$$L_k(t) = \omega(t)/(\omega'(u_k)(t - u_k)), \qquad k = 1, \ldots, n.$$

The u_k are a set of points, which may be chosen optimally, but for our purposes (the product trapezoidal rule), $u_1 = 0$, and $u_2 = 1$.

On $t_l \leq t \leq t_{l+1}, \quad l = 0, \ldots, m-1$ the approximation to $f(t)$ is

$$\overline{f}(t) = \sum_{k=1}^{n} L_k\left(\frac{(t - t_l)}{h}\right) f(t_{lk}),$$

and hence,

$$\begin{aligned} I_g(\overline{f}) &= \sum_{l=0}^{m-1} \int_{t_l}^{t_{l+1}} g(s)\overline{f}(s)ds, \\ &= \sum_{l=0}^{m-1}\sum_{k=1}^{n} f(t_{lk}) \int_{t_l}^{t_{l+1}} g(s)L_k\left(\frac{(s - t_l)}{h}\right) ds, \\ &= \sum_{l=0}^{m-1}\sum_{k=1}^{n} hf(t_{lk}) \int_{0}^{1} g(t_l + sh)L_k(s)ds. \end{aligned}$$

This is the nm-point quadrature rule with which the paper is concerned. $n = 2$ defines the product trapezoidal rule, which we require to analyse. The error functional for this rule is

$$E_g(f) = I_g(\overline{f}) - I_g(f) = I_g(\overline{f} - f).$$

Lemma 6.4.2 *If $f(t) \in C^{p+1}[0, T]$, $p \geq n$, then*

$$E_g(f) = h\sum_{r=0}^{p-n} h^{n+r} \int_0^1 \omega_r(s)h \sum_{l=0}^{m-1} g(t_l + sh)f^{(n+r)}(t_l + sh)ds + O(h^{p+1}), \quad (6.10)$$

where $\omega_r(t) = \omega(t)p_r(t)$, *and* $p_r(t)$ *is a polynomial*

$$p_r(t) = \frac{(-1)^r}{(n+r)!}\sum_{q=0}^{r}\sum_{k=1}^{n}\binom{n+r-1}{n+q-1}(-1)^{q-1}\frac{u_k^{n+q-1}}{\omega'(u_k)}s^{r-q}.$$

For any fixed s, $0 \le s \le 1$, the sum

$$h\sum_{l=0}^{m-1} g(t_l+sh)f^{(n+r)}(t_l+sh) \qquad \text{(2.11) in [48]} \tag{6.11}$$

is a generalized Euler approximation to $\int_0^1 g(s)f^{(n+r)}(s)ds$.

Section 3 of [34] deals with smooth $g(t)$, which we omit at this time.

Summation formulae for (6.11) have been investigated by Lyness and Ninham, and the application of their results to $g(t)f^{(n+r)}(t)$ is the basis of section 4, of [34], where $g(t)$ has a finite number of algebraic or logarithmic singularities. Again, we select from the four possible singular types, that of the form $g(t) = t^\beta$, such that $\beta > -1$.

Expansions of the form

$$h\sum_{l=0}^{m-1} g(t_l+xh)z(t_l+xh),$$

where z is a smooth function, are required. Such expansions have been obtained by Lyness and Ninham [48], who use Lighthill's procedure to obtain asymptotic expansions for the integral terms in Poisson's formula

$$\begin{aligned} h\sum_{l=0}^{m-1} g(t_l+xh)z(t_l+xh) - \int_0^1 g(s)z(s)ds, \\ = \sum_{q=-\infty}^{+\infty}{}' (-1)^q \exp(-\pi i(2x-1)q)\int_0^1 g(s)z(s)\exp(2\pi iqms)ds, \\ = \sum_{q=-\infty}^{+\infty}{}' \exp(-2\pi iqx)\int_0^1 g(s)z(s)\exp(2\pi iqs/h)ds. \end{aligned} \tag{6.12}$$

Applying the generalized Euler-Mclaurin expansion (as given in [48]) to $g(t)f^{(n+r)}(t)$,

we find that

$$h\sum_{l=0}^{m-1} g(t_l+xh)f^{(n+r)}(t_l+xh)$$
$$= \int_0^1 g(s)f^{(n+r)}(s)ds + \sum_{q=0}^{p-n-r} \frac{h^{q+1}}{q!}\left\{h^\beta \tilde{\zeta}(-\beta-q,x)\psi_{0r}^{(q)}(0)\right\}$$
$$+ O(h^{p-n-r+1}), \qquad r=0,\ldots,p-n,$$

where

$$\psi_{0r}(t) = f^{(n+r)}(t)t^\beta,$$

again taking the other terms as irrelevant for our purpose, and $\tilde{\zeta}$ as the periodic generalized zeta function.

Substitution of (6.12) into (6.10) yields

$$E_g(f) = \sum_{r=0}^{p-n} h^{n+r}\int_0^1 \omega_r(s)ds \int_0^1 g(s)f^{(n+r)}(s)ds$$
$$+ \sum_{r=0}^{p-n} h^{n+r+\beta+1}\sum_{l=0}^{r}\frac{\psi_{ol}^{(r-l)}(0)}{(r-l)!}\int_0^1 \omega_l(s)\tilde{\zeta}(-\beta-r+l,s)ds$$
$$+ O(h^{p+1}).$$

This is the desired Euler-McLaurin expansion for $g(t)$ given by t^β selecting the appropriate terms. For the important case of endpoint singularities (which includes $g(t)=t^\beta$) terms of the form $\int_0^1 \omega_l(s)\tilde{\zeta}(\alpha,s)ds$ can be reduced to sums of ordinary zeta functions.

The remainder of this section includes singularities of logarithmic type.

Application to Integral Equations

Section 5 of [34] deals with the application to integral equations, specifically to the Fredholm equation. (The transition to the VIE is straightforward). Further, they consider singularities of the type $|t-s|^\gamma$, which we now replace with t^β, and adjusting related terms accordingly. We commence with the Fredholm equation

$$y(t) = G(t) + \lambda\int_0^1 K(t,s)y(s)ds, \quad 0\le t\le 1, \tag{6.13}$$

and also define the Volterra equation

$$y(t) = G(t) + \int_0^t K(t,s)y(s)ds,$$

in both cases with

$$K(t,s) = P(t,s)Q(t,s)$$

(simplifying the kernel by dropping the summation), and where the relevant conditions apply, $Q(t,s)$ being smooth, and $\int_0^1 P(t,s)ds$ bounded. The application of product integration to the integral term in (6.13) yields the numerical scheme

$$Y_{ij} = G(t_{ij}) + \lambda \sum_{l=0}^{m-1} \sum_{k=1}^{n} W_{lk}(t_{ij})Q(t_{ij}, t_{lk})Y_{lk}, \qquad (6.14)$$
$$j = 1, \ldots, n; \quad i = 0, \ldots, m-1,$$

where

$$W_{lk}(t) = \int_{t_l}^{t_{l+1}} P(t,s)L_k\left(\frac{s-t_i}{h}\right)ds,$$

and Y_{ij} denotes the numerical approximation to $y(t_{ij})$. They cite the result from Atkinson, that if λ is not an eigenvalue of (6.13), then (6.14) has a unique solution for sufficiently small h, with a bound on the error,

$$\max_{\substack{j=1:n \\ i=0:m-1}} |y(t_{ij}) - Y_{ij}| = O(E),$$

where

$$E = \max_{\substack{j=1:n \\ i=0:m-1}} |\sum_{l=0}^{m-1} \sum_{k=1}^{n} W_{lk}(t_{ij})Q(t_{ij}, t_{lk})y(t_{lk}) - \int_0^1 K(t_{ij}, s)y(s)ds|.$$

This is the error formulation for the Fredholm equation, as shown in [34]. Adjustment of the subscripts and the first summation limit, and taking the upper limit of integration to be h (see below), yields the required expression for the Volterra equation.

Remark 5 *The application of the error term $E_g(f)$, (4.4) in [34], and in its condensed form (6.10) above, indicates that the dominant term in the error expansion is an integer power of h. This does not match the numerical results,*

which give convergence of a lower noninteger order. We observe that the theoretical approach used involves integration of $g(s)f^{(n+r)}(s)$ *over the limits 0 to 1. The Volterra equation is solved one step at a time, and we need the upper integration limit to be* h*. If we scale the integrand by the substitution* $\sigma = sh$*, and with* $g(s) = s^{\beta}$*, a factor of* h^{β} *is introduced, and the leading error term becomes* $O(h^{n+\beta})$*, which reflects the numerical findings.*

Chapter 7

Numerical Results

In this chapter we are looking to answer some basic questions for the solution of equation (1.1), with the integration process taken over the two intervals $[0, \alpha]$ and $[\alpha, t]$:

$$u(t) = g(t) + \int_0^\alpha \frac{s^{\mu-1}}{t^\mu} u(s)ds + \int_\alpha^t \frac{s^{\mu-1}}{t^\mu} u(s)ds. \tag{7.1}$$

Commencing *away* from the origin, can we rely on the efficacy of a standard method to perform as accurately as would be expected, if the kernel were fully continuous in all arguments? How closely can we approach the origin while maintaining the order of convergence of the method? What methods are suitable to commence at the origin, and are there others as yet untried? Finally, we must consider the way in which one method, which we call the primary method, can commence at the origin, and after a short interval be linked to an alternative method, the secondary method, which can match the primary method in terms of accuracy, while allowing greater flexibility over the remainder of the full time interval. The matching process needs to take account of the length of the initial interval, and its effect on both methods, and considerations of the step size between the first and second stages. Much of the preliminary discussion is covered in the previous chapter. We have examined the various algorithms, giving reasons for selecting those which we now consider in detail.

7.1 Test Equations

We are looking at a wide range of methods, and it is not always the case that a test equation suitable in one context may also be suitable in another. The purpose of choosing a test equation is to quantify the error at some given point $t = t_1$, by comparison between the true analytic solution and the numerical approximation. In certain cases, a simple input function g may be adequate, while in other circumstances more thought is necessary.

Our input functions are selected from the set of power terms in t. It is readily established that if $g(t) = \sum_{k=0}^{m} b_k t^{\beta_k}$ for integer k and $\beta > 0$, the smooth solution is available, in the form

$$u_0(t) = \sum_{k=0}^{m} b_k t^{\beta_k} \left(\frac{\mu + \beta_k}{\mu + \beta_k - 1} \right),$$

provided $\mu + \beta_k - 1 \neq 0$. We seek to approximate the smooth solution, and also determine whether, once a non-smooth trajectory $u_c(t)$ is established, such that $u_c(t) = u_0(t) + c_0 t^{1-\mu}$, the numerical approximation shows convergence to that particular trajectory. In some cases, we have used the simple input $g(t) = 1 + t$ so that $u_0(t) = \mu/(\mu - 1) + (\mu + 1)t/\mu$, but certain methods were found to have a very high degree of accuracy, although without convergence. We recall that a quadrature method is expected to be completely accurate in approximation of polynomials of degree up to the order of the method, so when this occurs, we use either a higher degree polynomial, or a non-integer power of t, with satisfactory results.

When dealing with non-integer powers, a further precaution is to consider the smoothness of the function. We see above that the differentiability of the unknown function will be of the same order as that of the input function, and when using methods of greater complexity, we may require more stringent restrictions. An example of this is the Hermite collocation described earlier, where we are working in the space $C^4[0, T]$. A slight alleviation in this context is that the differentiability we have to consider does not depend on the power of the function

alone, but on the combined power of $\mu + \lambda$, where $u \in C^{\lambda}[0, T]$, (Corollary 4.1 in [44]).

Taking the intervals in reverse order, we first show the results obtained from various standard methods over the second interval, commencing at the analytic value of $u(\alpha)$ - the convergence pattern is predictable, but we can also show the link between the error and α, predicted by the analysis in Appendix (A); the primary methods are then considered in isolation, followed by the extrapolation process, and finally the set of results for the split-interval scheme, taking various alternative primary and secondary rules.

7.2 Secondary Methods

In order to evaluate the behaviour of the numerical approximation over the second interval, we begin by treating it in isolation, that is, we will assume for the time being that the solution over the primary interval is the true analytic solution up to a defined point $t = \alpha \neq 0$, and the secondary method will be applied over the interval $[\alpha, T]$.

Since the equation is "well-behaved" away from the origin, we anticipate that the order of a method will be maintained, and further we will be looking for evidence to support the results of the analysis in Appendix (A), that the error is also inversely proportional to the same power of α. We will also consider the effects of altering μ and T, and how small we can take α. It will not be possible to include every case, but we give some typical examples to illustrate our findings, and at the end of this section give a generalized summary of further trials.

We take the discrete version of equation (7.1) in the form:

$$\tilde{u}(t_n) = g(t_n) + t_n^{-\mu} \int_0^{\alpha} s^{\mu-1} u(s) ds + \sum_{i=q}^{N} w_n K(t_n, t_i) u_i, \qquad n = q+1, \ldots, N,$$

where the function required over the initial interval is solved analytically by

means of the integral term, I_α, where

$$I_\alpha = \int_0^\alpha s^{\mu-1} u(s) ds,$$

which will be used throughout this section.

There is now a Lipschitz condition applicable to the third argument of the kernel, with Lipschitz constant $L = 1/\alpha$. If we apply the restriction of Theorem (6.3.1) on the stepsize, that $h \leq 1/LW$, where $W = max\{w_n\}$, w_n being the weights for the appropriate rule, we obtain the limitation on α in terms of h: for the trapezoidal rule, this becomes $\alpha \geq h/2$, and if we are to take α in terms of integer multiples of h, then $\alpha = h$ is the safe effective minimum value.

7.2.1 Basic Methods

We use the discretisation :

$$\tilde{u}(t_n) = g(t_n) + t_n^{-\mu} I_\alpha + t_n^{-\mu} \sum_{i=0}^{n} w_i t_i^{\mu-1} \tilde{u}(t_i) ds,$$

where $w_i = 1 - \theta$.

Example 7.2.1

We start with the simplest of quadrature methods, the explicit Euler rule, for which $\theta = 0$. We take $\mu = 0.4, \quad T = 10, \quad \alpha = 0.2 * k, \quad k = 1, \ldots, 5$, and $h = 1/(10 * 2^m), \quad m = 1, \ldots, 6$, and set $g(t) = 1 + t$. Since we are applying the quadrature over the region for which the kernel is smooth, we would expect the convergence to be of order 1, and results indicate that this is the case. When we take the logarithm of the error and plot against log(h) for each value of α, the linearity is good; the same matrix of results can then be used to plot log(error) against log(α), and we find gradients lie between -0.9 and -0.95.

When we take the implicit Euler method, where $\theta = 1$, we reach similar conclusions. We include these cases for completeness, but the methods are not sufficiently accurate for our purposes.

Example 7.2.2

The trapezoidal rule is found to be highly reliable, simple to construct, and a useful tool in a wide range of experiments. The predicted order of 2 is achieved, we have the inverse relationship with α: error $\propto \alpha^{-2}$, and error $\propto (T - \alpha)$. The main results are displayed in figures (7.1) and (7.2) where we have again taken $\mu = 0.4$, $T = 10$, and $g(t) = 1 + t$. The logarithmic error values are plotted (a) against $\log(h)$, for different values of α, where we obtain gradients of 2, and (b) against $\log(\alpha)$ for various h, where the gradients are -2. We have used the order approximations A_0 and A_1 in this and all subsequent cases, where

$$A_0 = \left| \frac{\log e_{0,k} - \log e_{0,1}}{\log h_k - \log h_1} \right|$$

$$A_1 = \left| \frac{\log e_{1,\ell} - \log e_{1,1}}{\log \alpha_\ell - \log \alpha_1} \right|$$

taking the approximations over k values of h, and ℓ values of α. Since we have good linearity of the results, the average of the gradient of the logarithmic graphs gives a reliable measure of the convergence of the method. This is the first definite indication we have that there is a connection between the error and the magnitude of the initial interval α, and we shall be looking to see whether this is shown to be the case when other methods are considered below.

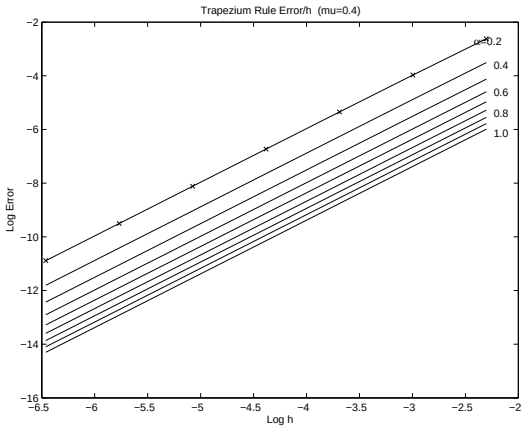
The connection between the error and μ is more obscure. The analysis of Appendix A indicates that *error* $\propto 1/(\mu - 1)(\mu - 2)$, and this is confirmed by the logarithmic plot (not shown) of error against the expression in μ, if h and α remain constant.

Taking extreme values of μ makes little difference to the validity - the results for $\mu = 0.02$ are included in Appendix (B). We further find that if we extend the time interval T progressively to $T = 100$, the usual linear link between error and elapsed time holds good. In case this might be as result of the initial interval being a significant proportion of the total time, we can extend the values of T,

now up to T=1000; and these relations still hold. When we pursue this through the error analysis, we find that the inverse square term in α does indeed emerge. When $\mu = 0.4$, the actual values of error at T=10 are shown for values of h at $\alpha = 0.4$ and for values of α when $h = 1/80$.

h	$\|\mathbf{e}\|$	α	$\|\mathbf{e}\|$
1/20	2.0227e-002	0.2	5.4934e-002
1/40	5.0761e-003	0.4	1.1864e-002
1/80	1.2702e-003	0.5	7.4023e-002
1/160	3.1764e-004	0.6	5.0761e-003
1/320	7.9415e-005	0.8	2.8393e-003
1/640	1.9854e-005	1.0	1.8287e-003

The graphs of these results for $\mu = 0.4$ are shown in figures (7.1) and (7.2).



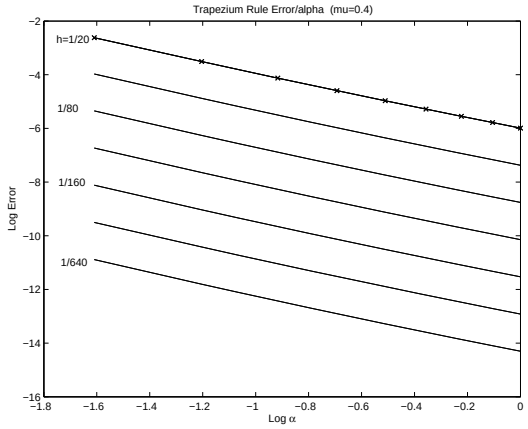


Figure 7.1:
Trapez. Rule: Error/h

Figure 7.2:
Trapez. Rule: Error/α

We obtain the order of the method from the gradients of the logarithmic plots, making the assumption that these plots are linear in form, an assumption which is supported by the results achieved.

Non-smooth solutions: The above results are all based on the premise that the smooth solution has been selected, and this is the trajectory which is followed, as we defined earlier, taking the case when $c = 0$. The path is defined by the value of the term I_α, which has so far been taken as representing the integration of the smooth solution over the initial interval $[0, \alpha]$. We now take the value of I_α to be amended by including the term $c_0 t^{1-\mu}$ in the integration process to give a different trajectory over the initial interval. We now define I_c in terms of our test equation to be

$$I_c = \int_0^\alpha s^{\mu-1}(a + bs + c_0 s^{1-\mu})ds, \tag{7.2}$$

$$= \frac{t^\mu}{\mu - 1} + \frac{t^{\mu+1}}{\mu} + c_0 t. \tag{7.3}$$

Our objective is to ascertain whether the secondary numerical scheme continues on the present path, returns to the smooth solution, or pursues some third way, which we would not have anticipated. Taking $\mu = 0.4$, $\alpha = 0.4$, $T = 10$ and $c_0 = 0.5$ we obtain the following results, where e_t is the error in the approximation to the true solution ($c_0 = 0$), and e_c is the error in the approximation to the non-smooth trajectory for $c_0 = 0.5$.

h	e_t	e_c
1/20	-0.00407346979360	-0.00407346979360
1/40	-0.00102071217663	-0.00102071217664
1/80	-0.00025532586443	-0.00025532586445
1/160	-0.00006384072516	-0.00006384072514
1/320	-0.00001596076036	-0.00001596076039
1/640	-0.00000399022635	-0.00000399022634

So in each case we have the appropriate convergence pattern of $O(h^2)$, and the magnitude of the error in the non-smooth case resembles that of the smooth approximation.

7.2.2 Newton-Cotes

Simpson's Rule

The natural sequence after the trapezium rule is to look at Simpson's Rule (further Newton-Cotes methods can be constructed similarly, extending the process as indicated in Chapter 5). We need to establish the convergence of order four, and the nature of the link with the term α. The issues discussed here affect all methods of Newton-Cotes type, applied to Volterra equations.

For straightforward quadrature, these rules are implemented over two steps at a time (three for the 3/8 rule etc.), and it is straightforward to divide the region of integration into subintervals in multiples of two (three, etc) to carry out the approximation. However, as described earlier, for a numerical solution to the Volterra equation, we need to increment the process a step at a time. When we follow the scheme described in section (4.3), we require two starting values before Simpson's rule can be applied, (three for the 3/8 rule, etc.) and there are various ways of constructing these starting values, and the intermediate step values. Provided this is done with care, we should obtain the expected convergence of order 4. The criteria we have decided on are as follows (see refs. in Chapter 5):

1. the 3-point Simpson's rule is augmented by the 4-point 3/8 rule on alternate lines,

2. the repetition factor of unity is maintained for $n_0 \geq 5$, by including the 3/8 rule as the *final* values of intermediate rows;

3. a starting method consisting of the trapezium rule over the first half-step and Simpson's over two half-steps is applied.

This results in the weights $\{w_{n,j}\}$ for Simpson's rule (cf. Brunner and van der

Houwen [14])

$$\left[\begin{array}{ccccccc}
\frac{1}{2} & \frac{1}{2} & & & & & \\
\frac{1}{6} & \frac{4}{6} & \frac{1}{6} & & & & \\
\frac{1}{6} & & \frac{4}{6} & \frac{1}{6} & & & \\
\frac{3}{8} & & \frac{9}{8} & \frac{9}{8} & \frac{3}{8} & & \\
\frac{1}{6} & & \frac{4}{6} & \frac{2}{6} & \frac{4}{6} & \frac{1}{6} & \\
\frac{1}{6} & & \frac{4}{6} & \frac{7}{24} & \frac{9}{8} & \frac{9}{8} & \frac{3}{8} \\
\cdots & \cdots & & & & &
\end{array}\right]$$

where the second column indicates the weights at $t_0 + h/2$. We seek to keep the scheme as simple as possible without compromising on the accuracy. Other alternatives for the starting method and for the main body of the scheme might be to use half- and quarter-steps with the trapezoidal rule ([8]), or a block-by-block method ([46]).

We will not pursue the Newton-Cotes methods of higher order: the extra accuracy can be better achieved in other ways, and they are unsuitable for dealing with the singularity if applied from $t = 0$.

Simpson's rule is, however, an alternative possibility to the trapezoidal rule away from the origin, if convergence of order four is required, and we have in addition the inverse relationship with α of order four to recommend it.

7.2.3 Runge-Kutta

The construction of the various RK methods over the second interval has been implemented using the method described in [14]. The final or 'incremental' step is calculated using the appropriate RK formula, and the tail or lag terms are also calculated using the same RK formula, rather than some arbitrary method of quadrature. Thus we retain the integrity of the method for a particular VRK scheme, and by implication, the order of the lag quadrature is the same as that of the newly calculated step.

As before, the initial interval is assumed to be known, so letting $q = \alpha/h$, we obtain the formula with intermediate stages:

$$Y_{n,i} = \tilde{F}_n(t_n + c_i h) + h \sum_{j=1}^{m} a_{ij} \frac{(t_n + c_j h)^{\mu-1}}{(t_n + c_i h)^{\mu}} Y_{n,j},$$

$$i = 1 \ldots m,$$

$$\tilde{F}_n(t_n + c_i h) = g(t_n + c_i h) + \frac{I_\alpha}{(t_n + c_i h)^{\mu}} + h \sum_{i=q}^{n} \sum_{j=1}^{m} a_{ij} \frac{(t_i + c_j h)^{\mu-1}}{(t_n + c_i h)^{\mu}} Y_{i,j},$$

and the $(n+1)$th step evaluation:

$$y_{n+1} = \tilde{F}_n(t_n + h) + h \sum_{j=1}^{m} b_j \frac{(t_n + c_j h)^{\mu-1}}{t_{n+1}^{\mu}} Y_{n,j},$$

$$n = 0 \ldots N - 1,$$

$$\tilde{F}_n(t_n + h) = g(t_{n+1}) + \frac{I_\alpha}{t_{n+1}^{\mu}} + h \sum_{i=q}^{n} \sum_{j=1}^{m} a_{ij} \frac{(t_i + c_j h)^{\mu-1}}{t_{n+1}^{\mu}} Y_{i,j}.$$

As the kernel is separable, we are able to retain the intermediate summation for subsequent use, so avoiding repeated calculation.

The values $a_{i,j}$, b_j and c_i are obtained from the appropriate Butcher array, and we are now able to employ any of the wide range of RK methods available. We give three examples below, and a further selection is included in Appendix (C). The choice has been influenced by the requirement to demonstrate that the RK explicit and impicit methods retain their order of convergence, to discover whether the relationship between the error and α is altered in any way, and to assess whether it would be possible to use any of these methods starting from the origin ($\alpha = 0$).

Example 7.2.3 *RK classic explicit (3-stage)*

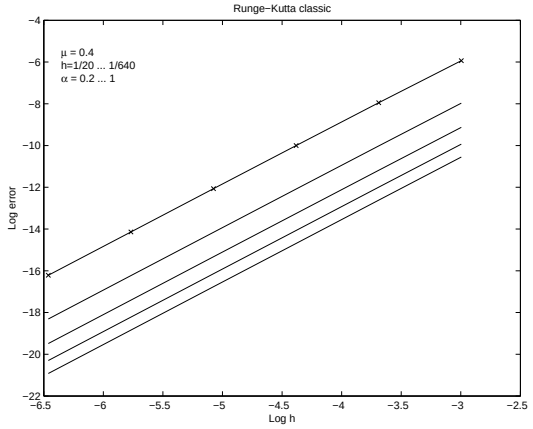


Figure 7.3:
RK classic: Error/h

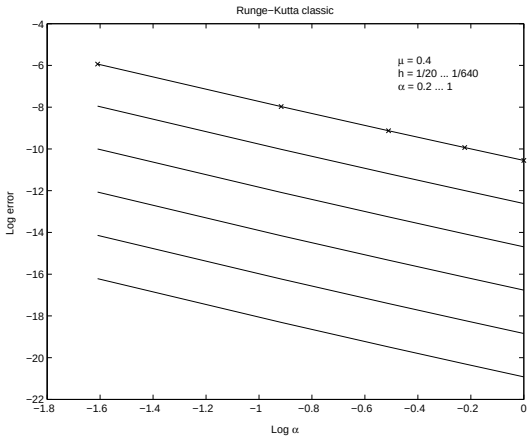


Figure 7.4:
RK classic: Error/α

The Butcher array for this example is

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{2} & \frac{1}{2} & & \\ 1 & -1 & 2 & \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}$$

Again, we take $\mu = 0.4$, $h = 1/(10 * 2^i)$, $\quad i = 1 \ldots 6$, $\quad \alpha = 0.2 * j$, $\quad j = 1 \ldots 5$. We find the expected convergence rate of $O(h^3)$, and that the inverse relation with α is $O(\alpha^{-3})$.

Example 7.2.4 *RK Implicit (2-stage)*

We now take the Butcher array as

$$\begin{array}{c|cc} 0 & \frac{1}{4} & \frac{-1}{4} \\ \frac{2}{3} & \frac{1}{4} & \frac{5}{12} \\ \hline & \frac{1}{4} & \frac{3}{4} \end{array}$$

Again, we have convergence $O(h^3)$ and the same inverse relation with α.

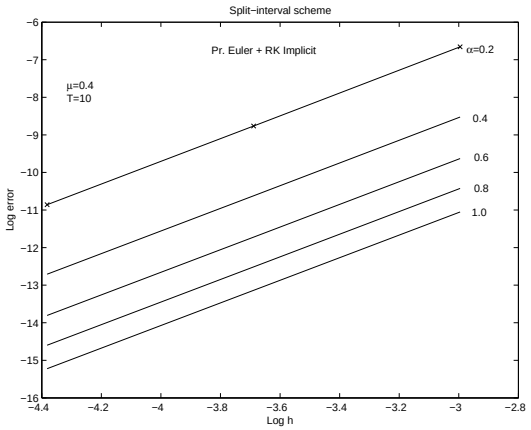


Figure 7.5:
RK Implicit: Error/h

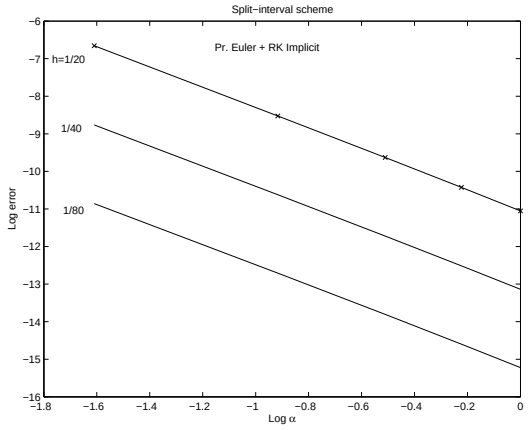


Figure 7.6:
RK Implicit: Error/α

Example 7.2.5 *RK Nystrom*

We take the same parameters and input function as before, and using the Butcher array

$$\begin{array}{c|ccc} 0 & & & \\ \frac{2}{3} & \frac{2}{3} & & \\ \frac{2}{3} & 0 & \frac{2}{3} & \\ \hline & \frac{1}{4} & \frac{3}{8} & \frac{3}{8} \end{array}$$

we obtain the expected order of 2. However, in this example we do not find the uniformity of the relation between error and α. Between the lowest values taken,

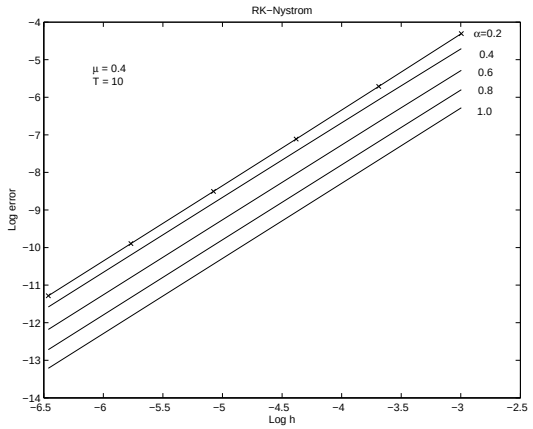


Figure 7.7:
RK Nystrom: Error/h

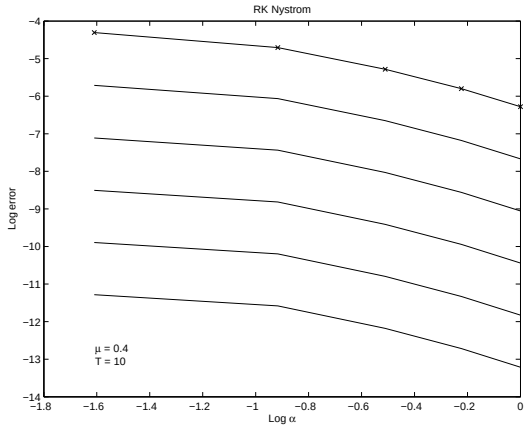


Figure 7.8:
RK Nystrom: Error/α

the gradients A_1 are less than unity, the average figure being approximately 1.2. Over the subsequent interval, where $\alpha \in [0.4, 1.0]$ the values of the gradients are more in line with the general picture obtained from other methods, approximating 1.7, as we see from figure (7.8) and the tables in Appendix (B). However, due to this discrepancy we do not include the RK Nystrom method of order 2 in the combined split-interval examples.

Example 7.2.6 *Radau and Lobatto*

We apply the Radau Ia, Radau II, Radau IIa and Lobatto 3c methods, all of which attain their respective expected convergence orders. In general, the 2-stage methods achieve the same inverse power when the error is expressed in terms of α, but the 3-stage methods tend to be approximately 0.5 lower.

Example 7.2.7 *Gauss-Legendre*

The Butcher array for the two-stage method is

$$\begin{array}{c|cc} \frac{3-\sqrt{3}}{6} & \frac{1}{4} & \frac{3-2\sqrt{3}}{12} \\ \frac{3+\sqrt{3}}{6} & \frac{3+2\sqrt{3}}{12} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

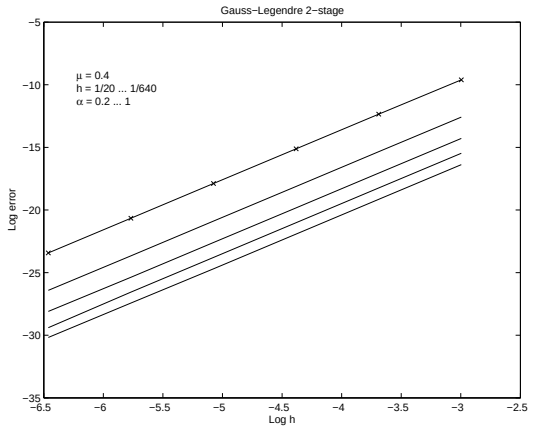


Figure 7.9:
Gauss-Legendre 2-stage:
Error/h

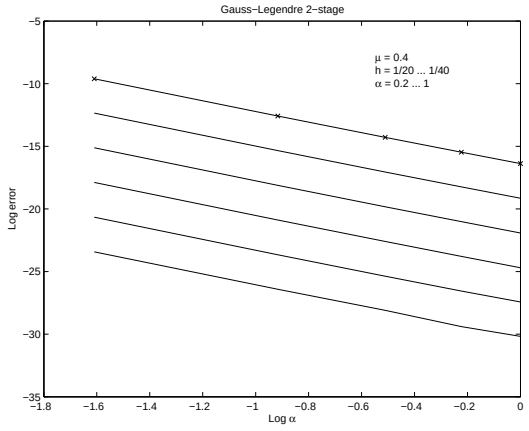


Figure 7.10:
G-L 2-stage: Error/α

The expected order $O(h^4)$ is achieved, and the inverse relation with α is slightly better than the inverse $O(\alpha^{-4})$

The Gauss-Legendre schemes would seem to present a strong possibility of providing an approximate solution to equation (1.1), to include the singularity. The nodes upon which the approximation is based do not include either end of the step interval. The method is based upon integration over the finite interval [-1,1], which may easily be transformed to accommodate the step interval of the VIE. The accuracy of the method is extremely good - the order is $2m$ for an m-stage method, and the coefficient of the leading error term appears low. We have constructed the two, three and four-stage rules for equation (1.1) starting away from the origin, and each of these schemes behaves in accordance with expected orders of accuracy. Further, we again have the link with the α-interval, finding that the relationship between error and α is inversely of the same order as the method. The results for the 3-stage method would require to be constricted within a narrower range of step size. We find that the high level of accuracy at small step sizes (1/320, 1/640) is subject to rounding error, affected by the computer accuracy $eps = 2.220446049250313e - 016$. If we scale the values of h

to larger step sizes, we need to relate to values of α which are multiples of the enlarged scale.

For both these systems, we can now get to one step away from the origin without losing convergence. However, when we attempt to implement Gauss-Legendre from the origin, a problem arises. For the 2-stage method, the matrix constructed for solution of the implicit system is singular. This is shown up when we attempt to run the code, and it is straightforward to check the individual terms of the matrix to find that this is indeed the case. When we run the code for the 3-stage Gauss-Legendre scheme, the matrix is almost singular. It is usual to consider the 3-stage version as providing sufficient accuracy for normal use, but feeling that this is a special case, we must also look at the 4-stage scheme. This might be more expensive, but as we are only looking to find a result over the initial short interval, it is worth looking at. However, disappointingly, we find that here, too, the matrix is almost singular.

7.2.4 Linear Multistep schemes

Again, there is a wide variety of such schemes available, but here, in the presence of a singularity, we are particularly interested in the backward differentiation 2-step method to illustrate a means of approximating the second interval, with its known stability properties. First, though we consider the Volterra analogy of the Adams-Moulton 2-step method, of order 3.

Example 7.2.8 *Adams-Moulton 2-step*

The analogue of LM methods in the context of the Volterra equation has been assessed in the previous chapter, and several means of implementation are considered. We apply an iterative scheme based on the 2-step AM method as applied to the ODE, described in full in Appendix (D). We find that higher order algorithms can be constructed, but in this context it would not be useful to pursue this further: higher order results of greater stability are better achieved

by e.g. the Gauss-Legendre schemes, or for reliability and ease of construction, the trapezoidal. We include this example to demonstrate that a VLM scheme is possible here.

Applying the algorithm described in Appendix (D), we find that the results give a good convergence of order 3, and also the connection between the error and α is of order -3. However, this is based on a starting scheme of *two* steps of the trapezoidal rule. If this is reduced to a single step, then the order of 3 is maintained, but the relation between error and α now approximates to -2. Other possibilities exist, such as taking the trapezium weights over two half-steps as the starting rule, or using a method of order 3 over the first step.

Example 7.2.9 *BDF 2-step*

We take the 2-step backward differentiation formula, with $a_0 = 3$, $a_1 = -4$, $a_2 = 1$ and $b_0 = 2$. We use the property that this formula is (ρ, σ)-reducible, and apply the weights as calculated by the formula derived by Wolkenfeldt [71], commencing at $t_q = \alpha$, with the trapezoidal weights over the first step:

$$\begin{aligned} w_{1,1} &= w_{1,2} = \tfrac{1}{2}, \quad n = 1, \\ w_{n,1} &= w_{n,2} = \tfrac{3}{4}(1 - (\tfrac{1}{3})^n, \qquad n \geq 2, \\ w_{n,j} &= 1 - (\tfrac{1}{3})^{n-j+1} \qquad n - j \geq 0, \quad j \geq 3. \end{aligned}$$

The weights are applied to the quadrature formula

$$\tilde{u}(t_n) = g(t_n) + t^{-\mu} I_\alpha + h \sum_{i=0}^{n} w_i k(t_n, t_i) \tilde{u}(t_i),$$

where the $\{w_i\}$ are now calculated as above. We obtain the results that indicate convergence of order 2, and also the inverse power in α (see Appendix (A)).

The obliging link between the convergence order and the inverse relation between error and α is not invariably present. Further numerical results are given in Appendix (B), and we find that in certain cases, for a method of order

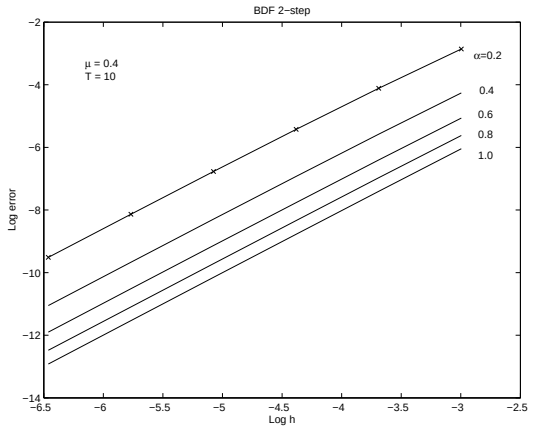


Figure 7.11:
BDF 2-step: Error/h

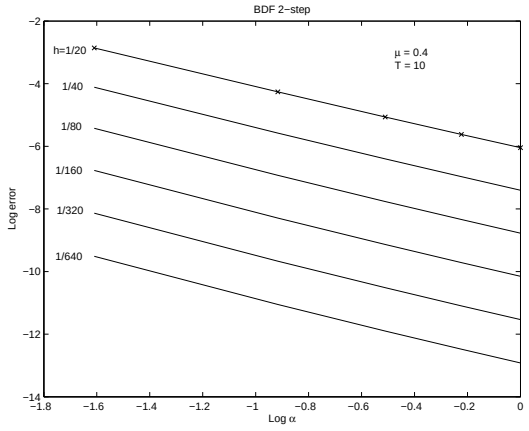


Figure 7.12:
BDF 2-step: Error/α

p, the error is proportional to α^{-r}, where $p-1 < r < p$. This discrepancy has not been fully explained.

7.3 Primary Methods

Product integration is a highly suitable means of commencing at the origin, for the reasons already discussed. The product Euler method is already well established, with order of convergence h^{μ} when $\mu < 1$, and when the scheme is used alone, the error is proportional to $c_0 t_N^{1-\mu}$. In the remainder of the chapter, we shall be curtailing the product Euler method at $t_q = \alpha = qh$, so this will influence the cumulative error of the combined scheme.

We include some results to illustrate that when $\mu < 1$, the convergence of the product trapezoidal method is $h^{1+\mu}$. We also confirm that the error is again proportional to $c_0 t_N^{1-\mu}$

We further include evidence that the method of Hermite-type collocation, developed previously in [64] for the case when $\mu > 1$ where it was shown to have order h^4, now has convergence order $h^{3.5}$ approximately.

7.3.1 Product Euler

We define the scheme to be as described in chapter 5, which we now develop using the trivial Lagrange zero degree polynomial $\ell_0 = 1$, and the kernel $K(t,s) = t^{-\mu}s^{\mu-1}$ to obtain the explicit method:

$$u_n(t_n) = g(t_n) + \sum_{j=0}^{n-1} t_n^{-\mu} u_j \int_{t_j}^{t_{j+1}} s^{\mu-1} ds$$

to yield

$$u_n(t_n) = g(t_n) + \sum_{j=0}^{n-1} t_n^{-\mu} u_j \frac{(t_{j+1}^{\mu} - t_j^{\mu})}{\mu}.$$

Although this is a well-established method, and the convergence is weak but known and proved, the actual magnitude of the errors is not insignificant, as the following plots demonstrate. At $\mu = 0.2$, $T = 10$, and for $h = 1/20$, the error approaches 50%, and it is a remarkable fact that, even with this error of totally unacceptable proportion, we can still obtain the requisite convergence, with a uniform linearity over the stepsizes used. Figure (7.13) is the solution plots for various h taken at $\mu = 0.4$, while figure (7.14) shows the error values plotted logarithmically against h, for $\mu = 0.2,\ 0.4,\ 0.6$ and 0.8.

Pr. Euler	
μ	A_0
0.2	0.1982
0.4	0.3930
0.6	0.5784
0.8	0.7419

We find in the following section that with the use of extrapolation, it is possible to obtain reliable results of an acceptable degree of accuracy.

7.3.2 Product Trapezoidal

We construct the product trapezoidal method as indicated in the previous chapter, with the first degree Lagrange polynomial $\ell_1 = (t - t_0)/(t_1 - t_0)$. We retain

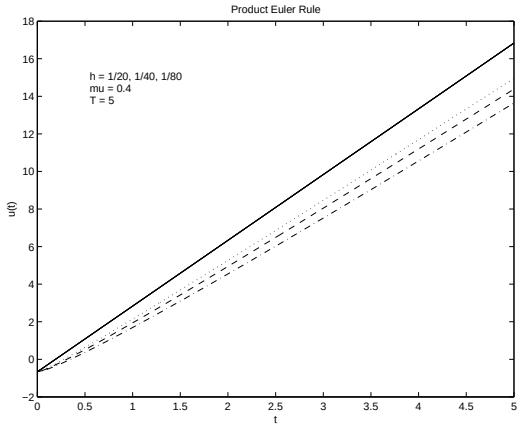


Figure 7.13:(A) Product Euler solutions

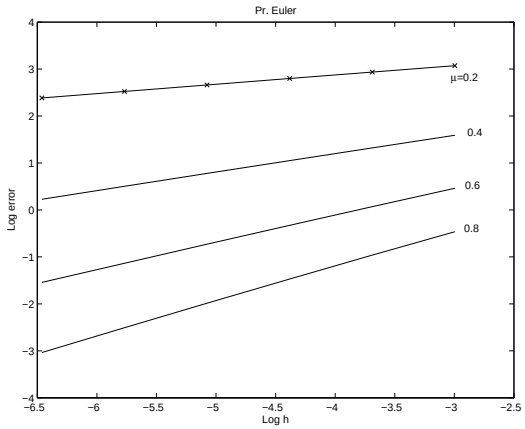


Figure 7.14:(B) Product Euler conv.

$\mu = 0.4$ and $T = 10$, but the input function is now $g : g(t) = t - t^2/10$.

Pr.	Trapezoidal
μ	A_0
0.2	1.1997
0.4	1.3978
0.5	1.4950
0.6	1.5909
0.8	1.7596

So we have a fairly close convergence order of $1+\mu$, with a slight weakness when $\mu = 0.8$, but with good linearity of the results.

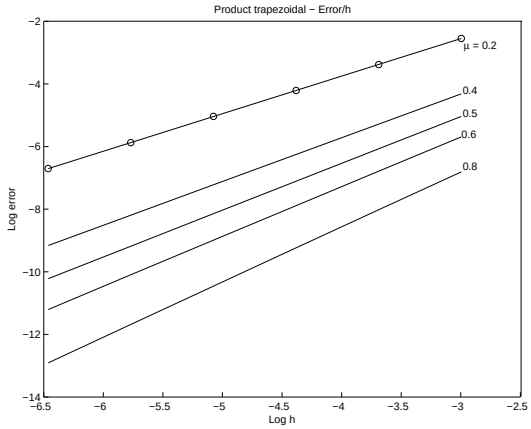


Figure 7.15:(C) Product trapez. conv.

7.3.3 Hermite-type collocation

For a much higher level of accuracy, we can use the method of Hermite-type collocation, previously applied for the case when $\mu > 1$, where it was found to be of order 4. However the case for which $\mu < 1$ does not have the necessary bound, and as we have seen there is a multiplicity of solutions for any given input function g. One further piece of information is required to identify which of these solutions is required, which has already been considered by identifying a point away from the origin by which a given solution is determined. If the unique solution required is to be the smooth solution, then we may take the heuristic argument that the gradient of the solution defined at $t = 0$ is also sufficient to determine the trajectory which the numerical rule is required to approximate. As the above method utilises the given starting value of $u'(0)$, we may consider this a suitable starting method for our split-interval scheme.

Now with $\mu < 1$, we find evidence that commencing from the origin, the convergence order p is such that $3 < p < 4$. However, unlike the product

integration methods, there appears to be little change when different values of μ are selected. There are inconsistencies at the smallest stepsize, almost certainly due to machine error, and a slight shift in the convergence when $\mu = 0.8$. The table of gradients A_0 is calculated excluding the case when $h = 1/640$.

μ	A_0
0.2	3.5597
0.4	3.6768
0.5	3.6447
0.6	3.5924
0.8	3.3825

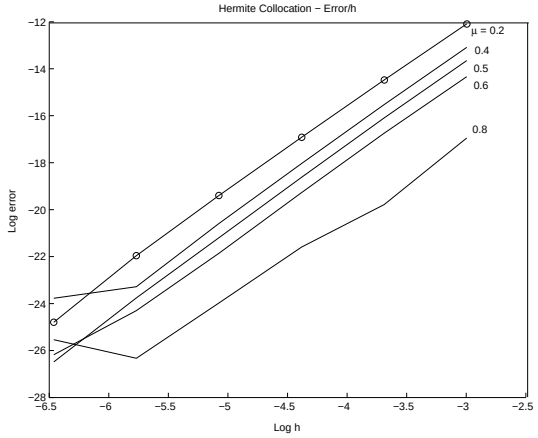


Figure 7.16:Hermite collocation

7.3.4 Comparison of primary methods

To conclude this section, we compare the three methods not only for convergence, but for their overall accuracy. If each of these plots is extended to the vertical line $\log h = 0$, this gives an insight into the values of the leading error factor in each case: with $\mu = 0.4$ and $T = 5$, these are 0.105 for the product Euler, 0.024 for the product trapezoidal, and 0.129 for the Hermite scheme. The figure

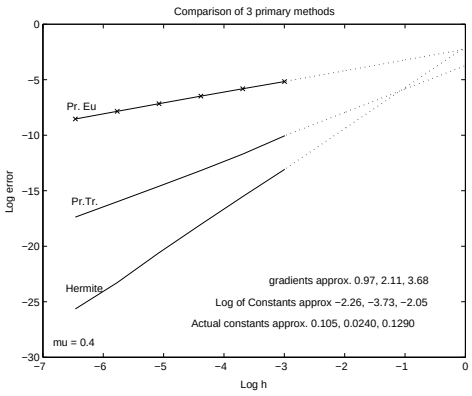


Figure 7.17:Comparison of primary methods

is based on the following actual error values:

Comparison of errors			
h	$Pr.Euler$	$Pr.Trap.$	$Hermite$
$1/20$	$-5.7125e-3$	$4.2796e-5$	$-2.0697e-6$
$1/40$	$-2.9857e-3$	$8.2734e-6$	$-1.8088e-7$
$1/80$	$-1.5302e-3$	$1.8838e-6$	$-1.4696e-8$
$1/160$	$-7.7527e-4$	$4.5729e-7$	$-1.1415e-9$
$1/320$	$-3.9029e-4$	$1.1333e-7$	$-7.738e-11$
$1/640$	$-1.9582e-4$	$2.8273e-8$	$-4.727e-11$

and we clearly see the relative behaviour, not only of the order of convergence, but the overall accuracy comparison.

7.4 Extrapolation

7.4.1 Modified Richardson

We use the modified Richardson scheme described in the previous chapter, and apply the acceleration process to a set of results obtained by using the product Euler method defined above, setting $g(t) = 1 + t$, over $[0, 10]$, with $\mu = 0.4$,

and stepsizes $h = 1/(10 * 2^i), \quad i = 1, \ldots, 6$. The error values are in the first column, and we take five levels of extrapolation, eliminating error terms up to and including $h^{2+\mu}$. The true solution is $34.3333\ldots$. The actual process is carried out on the solution approximation values, and the table below shows the errors at each level of extrapolation.

Modified Richardson						
h	$e_{0,n}$	$e_{1,n}$	$e_{2,n}$	$e_{3,n}$	$e_{4,n}$	$e_{5,n}$
1/20	-4.9018e+0					
1/40	-3.7496e+0	-1.4342e-1				
1/80	-2.8594e+0	-7.3154e-2	-2.8869e-3			
1/160	-2.1760e+0	-3.7123e-2	-1.0915e-3	3.9725e-6		
1/320	-1.6536e+0	-1.8768e-2	-4.1290e-4	1.1141e-6	1.6135e-7	
1/640	-1.2555e+0	-9.4621e-3	-1.5627e-4	3.0133e-7	3.0391e-8	-2.2097e-10

7.4.2 E-algorithm

The E-algorithm of Brezinski is applied to the same inputs as above, to yield the errors in the extrapolation stages to be

E-algorithm						
h	$e_{0,n}$	$e_{1,n}$	$e_{2,n}$	$e_{3,n}$	$e_{4,n}$	$e_{5,n}$
1/20	-4.9018e+0					
1/40	-3.7496e+0	-1.4342e-1				
1.80	-2.8594e+0	-7.3154e-2	-2.8869e-3			
1/160	-2.1760e+0	-3.7123e-2	-1.0915e-3	3.9725e-6		
1/320	-1.6536e+0	-1.8768e-2	-4.1290e-4	1.1141e-6	1.6135e-7	
1/640	-1.2555e+0	-9.4621e-3	-1.5627e-4	3.0133e-7	3.0391e-8	-2.2087e-10

Other schemes for convergence acceleration are noticeably less accurate for this problem. Making the comparison of these two sets of calculations, we note the following:

1. There is negligible difference between the two ways of achieving convergence

acceleration;

2. both methods require the asymptotic error expansion to be known;

3. the modified Richardson method requires fewer calculations, and is tailored to the requirements of this particular situation;

4. adapting to different circumstances, e.g. product trapezoidal error expansion, is also simpler with the modified Richardson.

We conclude that the modified Richardson extrapolation is the preferred method.

7.5 The Split-interval scheme

We have tested the separate components of the split-interval scheme defined in the previous chapter, and we now need to show how these are combined to form a reliable algorithm for the approximate solution of equation (1.1). In the majority of cases we use the product Euler as the primary method - its weakness provides a very real test of the acceleration process, and illustrates the importance of the correct construction of the combined method. The process is best described through examples.

Example 7.5.1 *Product Euler and trapezoidal rule (a)*

For our first example, we take the product Euler scheme as the primary, and the trapezoidal rule as secondary, using extrapolation at α to improve the primary result to order 2. We commence with a 'base' value of the stepsize, $h_0^{(1)}$, and three further values, each half of the one before. This provides a set of four approximate solutions at α, to which we apply 3 levels of extrapolation, and the single value from this is used as the starting point for the trapezoidal rule *at stepsize* $h_0^{(1)}$.

We then commence with a new 'base' value of h, $h_0^{(2)}$, and obtain four values of the approximation, again applying three levels of extrapolation, and implementing the second interval with stepsize $h_0^{(2)}$; this process is continued for a further level, and so we achieve a final set of three approximations $u_n(T)$ which provide the overall convergence of order 2. Allowing $h_0^{(3)} = h_0^{(2)}/2 = h_0^{(1)}/4$ avoids repetitive calculation, and we effectively have a 5-level extrapolation table, curtailed at the third level.

In this example we take $\alpha = 0.5$ and $\mu = 0.4$. The values $e_{m,n}$ are the errors of each of the approximate solutions $Y_{m,n}$ of the extrapolation table.

Extrapolation at α				
h	$e_{0,n}$	$e_{1,n}$	$e_{2,n}$	$e_{3,n}$
$\frac{1}{20}$	$-7.0918e-1$	$-1.1938e-1$	$-1.1066e-2$	$-1.6947e-4$
$\frac{1}{40}$	$-5.6637e-1$	$-6.5223e-2$	$-4.2984e-3$	$-3.8971e-5$
$\frac{1}{80}$	$-4.4502e-1$	$-3.4761e-2$	$-1.6530e-3$	$-9.1210e-6$
$\frac{1}{160}$	$-3.4568e-1$	$-1.8207e-2$	$-6.3203e-4$	
$\frac{1}{320}$	$-2.6638e-1$	$-9.4195e-3$		
$\frac{1}{640}$	$-2.0416e-2$			

Instead of the more conventional layout, we align the values to the top of the table, to emphasise the appropriate stepsize for the secondary method.

After setting the trapezoidal rule over the remainder of the interval $[\alpha, T]$ we obtain the final set of results

Final Error values	
h	*Error*
1/20	-0.00249769500631
1/40	-0.00060425726332
1/80	-0.00014733377138

which clearly demonstrates order 2 for the combined scheme.

Example 7.5.2 *Product Euler and trapezoidal rule (b)*

Having demonstrated the construction of the method, we now take a range of values for α, and obtain a more general set of results.

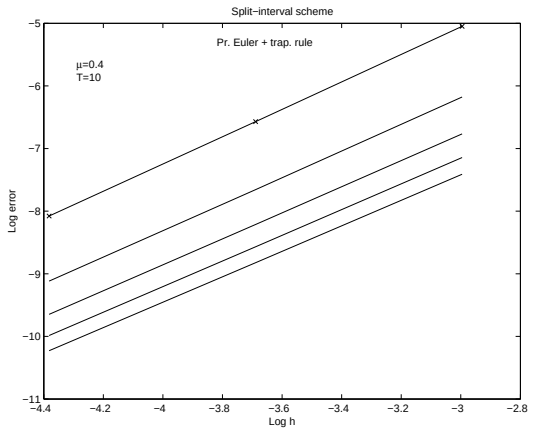


Figure 7.18:
Pr. Euler + trapezium rule:
Error/h

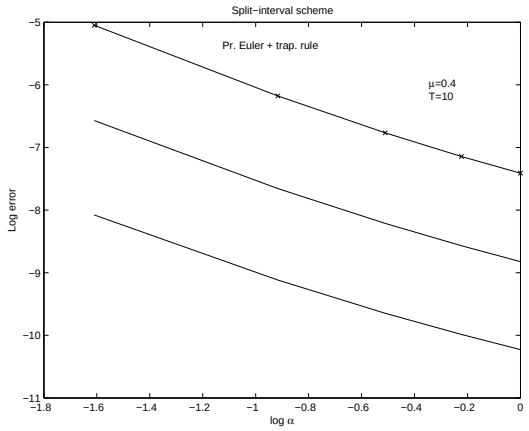


Figure 7.19:
Pr. Euler + trapezium rule:
Error/α

This yields the following convergence figures, based on the gradients as before:

P.Euler + Trap. rule			
α	A_0	h	A_1
0.2	2.1847	1/20	−1.4676
0.4	2.1203	1/40	−1.3992
0.6	2.0783	1/80	−1.3358
0.8	2.0506		
1.0	2.0316		

with a good $O(h^2)$ for the combined scheme, and we find that the error is of order $-1.3 \cdots -1.5$ in α. It is too early to draw conclusions from this.

Example 7.5.3 *Product Euler + Runge-Kutta classic*

The RK classic is of order 3, so now we require 5 levels of extrapolation to bring the primary order to the same as the secondary. We let $g(t) = t - t^2/10$, otherwise, the construction is as for the previous example. Again, we allow 3 trajectories to continue over the second interval, which yield the expected order of 3, and the error relates to α in terms of order approximately -2.7.

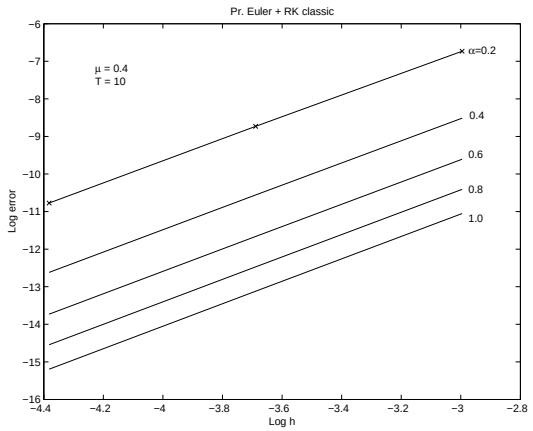


Figure 7.20:
Pr. Euler + RK classic:
Error/h

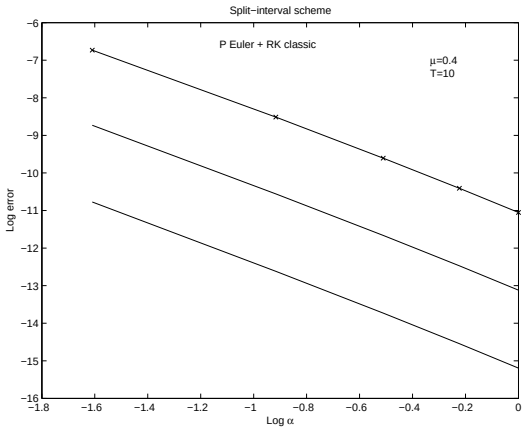


Figure 7.21:
Pr. Euler + RK classic:
Error/α

P.Euler + RK classic			
α	A_0	h	A_1
0.2	2.9160	1/20	−2.6851
0.4	2.9606	1/40	−2.7263
0.6	2.9749	1/80	−2.7459
0.8	2.9821		
1.0	2.9866		

Example 7.5.4 *Product Euler + RK Implicit*

The RK Implicit being also a method of order 3, we retain 5 levels of extrapolation at α, and $g(t) = t - t^2/10$. The convergence is now marginally greater than 3, and we again find the order of the error in α around -2.7.

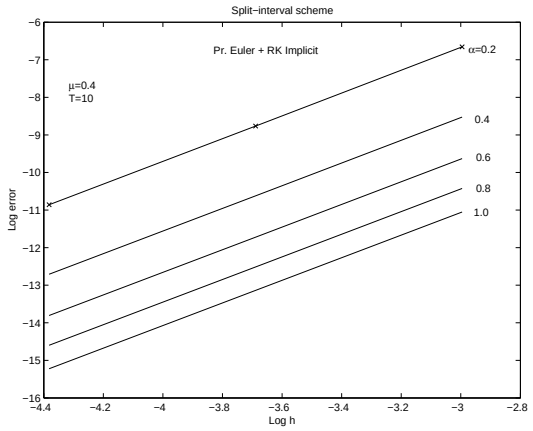


Figure 7.22:
Pr. Euler + RK Implicit:
Error/h

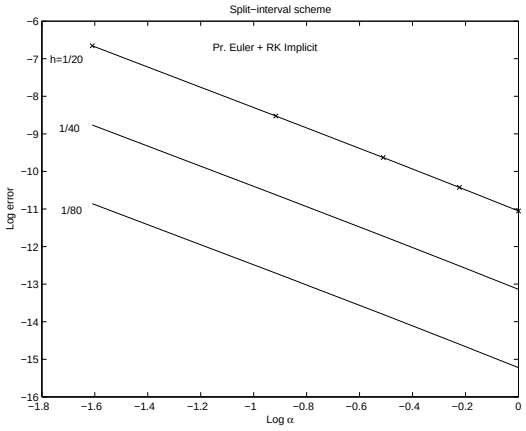


Figure 7.23:
Pr. Euler + RK Implicit:
Error/α

P.Euler + RK Implicit			
α	A_0	h	A_1
0.2	3.0322	1/20	−2.7319
0.4	3.0180	1/40	−2.7190
0.6	3.0127	1/80	−2.7114
0.8	3.0099		
1.0	3.0083		

Example 7.5.5 *Product Euler + GL 2-stage*

The Gauss-Legendre 2-stage is of order 4, so we now need 10 stepsize values at each α, followed by 7 levels of extrapolation, to obtain 3 trajectories over the second interval. Again, $\mu = 0.4$, $T = 10$ and $g(t) = t - t^2/10$. The results are a

close match to the order, and we find a slight weakening of the connection with α, at around -3.5.

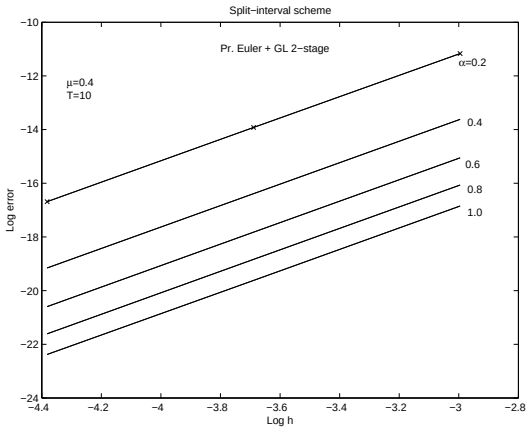


Figure 7.24:
Pr. Euler + GL 2-stage:
Error/h

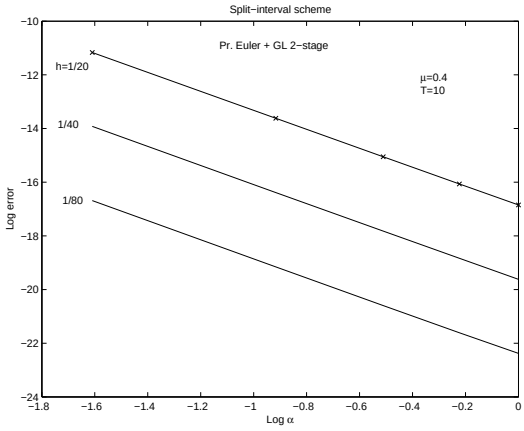


Figure 7.25:
Pr. Euler + GL 2-stage:
Error/α

P.Euler + GL 2-stage			
α	A_0	h	A_1
0.2	3.9810	1/20	-3.5289
0.4	3.9954	1/40	-3.5416
0.6	3.9983	1/80	-3.5379
0.8	3.9993		
1.0	3.9916		

Example 7.5.6 *Product Euler + Adams-Moulton*

The implementation is similar to that of the method described in section (7.2.4), but as with other methods we now commence the AM 2-step method at $\tilde{u}(\alpha)$. The formula for the second interval is the same as that in App. E, and with $g(t) = 1 + t$, we obtain order 3 in h, and order -3 in α.

P.Euler + AM 2-step			
α	A_0	h	A_1
0.2	2.7012	1/20	−2.9308
0.4	2.8390	1/40	−3.0576
0.6	2.8912	1/80	−3.1321
0.8	2.9183		
1.0	2.9349		

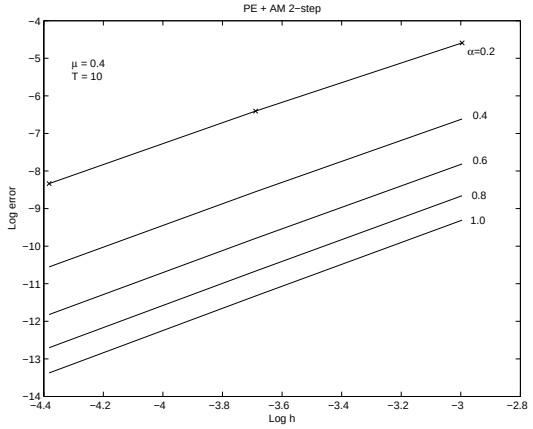


Figure 7.26:
Pr. Euler + AM 2-step
Error/h

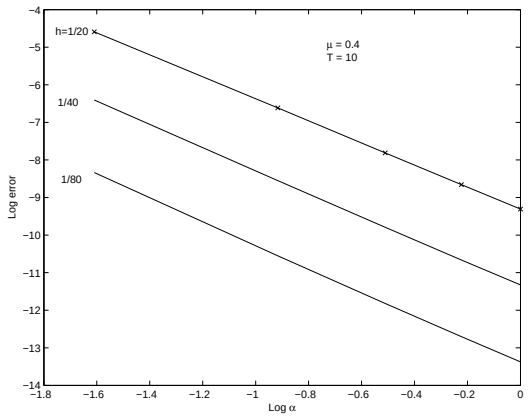


Figure 7.27:
Pr. Euler + AM 2-step:
Error/α

Example 7.5.7 *BDF 2-step*

Again, with the appropriate 3 levels of extrapolation, we find that the combined method is of order 2, matching the expected order for a 2-step method applied to a Volterra equation where all terms are smooth. Further, the relation with α is again $error \propto \alpha^{-2}$. The construction is the same as we used in section (7.2.4) commencing at the true solution at α. As the BDF method is of special interest in the solution of Volterra equations, we include the full representation of the extrapolation process when $\alpha = 1$:

Extrapolation at $\alpha = 1$				
h	$e_0^{(n)}$	$e_1^{(n)}$	$e_2^{(n)}$	$e_3^{(n)}$
$\frac{1}{20}$	-1.1393	$-1.2527e-1$	$-7.3172e-3$	$8.7321e-6$
$\frac{1}{40}$	$-8.9377e-1$	$-6.6295e-2$	$-2.7673e-3$	$5.3380e-6$
$\frac{1}{80}$	$-6.9341e-1$	$-3.4531e-2$	$-1.0453e-3$	$1.9241e-6$
$\frac{1}{160}$	$-5.3386e-1$	$-1.7788e-2$	$-3.9489e-4$	
$\frac{1}{320}$	$-4.0890e-1$	$-9.0916e-3$		
$\frac{1}{640}$	$-3.1209e-1$			

We obtain the final error values at $T = 10$:

Final errors at T=10					
h	$\alpha = 0.2$	0.4	0.6	0.8	1.0
$\frac{1}{20}$	$-5.9322e-2$	$-1.4346e-2$	$-6.3533e-3$	$-3.6028e-3$	$-2.3340e-3$
$\frac{1}{40}$	$-1.6541e-2$	$-3.7643e-3$	$-1.6303e-3$	$-9.1414e-4$	$-5.8827e-4$
$\frac{1}{80}$	$-4.3903e-3$	$-9.6255e-4$	$-4.1150e-4$	$-2.2927e-4$	$-1.4700e-4$

and the gradient values of the logarithmic plots

P.Euler + BDF 2-step			
α	A_0	h	A_1
0.2	1.8781	1/20	−2.0103
0.4	1.9488	1/40	−2.0730
0.6	1.9743	1/80	−2.1105
0.8	1.9870		
1.0	1.9944		

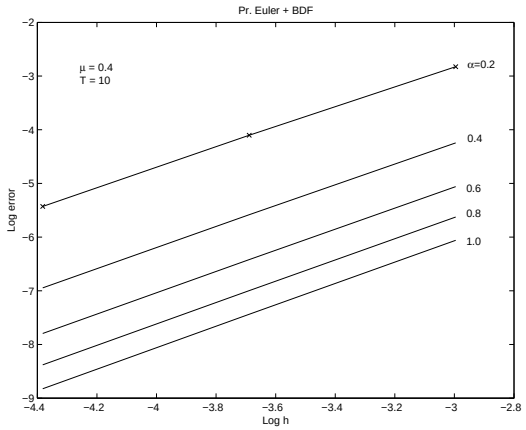


Figure 7.28:
Pr. Euler + BDF 2-step:
Error/h

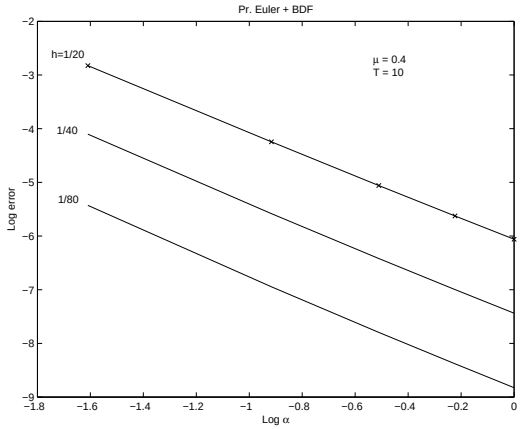


Figure 7.29:
Pr. Euler + BDF 2-step:
Error/α

7.6 Summary

We have shown that a wide range of methods is possible away from the origin, and the standard convergence pattern occurs when such methods are applied, starting from the true analytic solution at $t = \alpha$. Further, that the order of convergence in each case is echoed by a pattern of inverse order in α.

We have examined several methods which are suitable for commencing at the origin. The product Euler method is already known to be reliable, and we also can now include the product trapezoidal, and the collocation method of Hermite. Higher order product integration is theoretically possible. Other potential methods, such as Gauss-Legendre, have been constructed, but found to be unworkable.

We have tested two suitable methods of convergence acceleration - the E-algorithm of Brezinski, and a modification of the Richardson scheme. The performance of these two in terms of accuracy is virtually identical, so we select the modified Richardson as being specifically constructed for the particular expansion involved, and having fewer calculations.

Finally, these separate components have been combined into the split-interval scheme described in the preceding chapter. The construction of this is described through examples, and by careful management of the procedure at α, we find that the expected convergence rates can be achieved. The assumption is that the primary method will have a lower convergence rate than the secondary, and in this case the acceleration process and link with the secondary method must be tightly controlled.

The relationship between error and α is still a matter for concern. For a secondary method of integer order m, and acceleration of the primary method to the same order, we find that the error is proportional to a negative power of α which lies between $m-1$ and m.

Remark 6

If the convergence rate of the primary scheme is higher than that of the secondary, then the order of the combined method is expected to be that of the latter. Extrapolation might be considered at the final time $t = T$, or used on non-integer terms at α, and the integer terms at T. Such a case might arise if the method of Hermite is applied over the first interval.

Chapter 8

Conclusions

We have been looking at the solution structures of a Volterra integral equation with a singularity at the origin which is not of the type considered in standard texts on the subject. The objective has been to develop an improved means of numerical approximation to the solution, but there are also certain theoretical developments which have emerged.

In this thesis we have concentrated on the most elementary member of a particular class of Volterra integral equations. The methods and proofs described are, in principle, applicable in general to the class of Volterra integral equations with a singularity at the origin of algebraic type, such as those which have (a) linear kernels of the form

$$K(t,s) = \frac{s^{\mu-1}}{t^{\mu}} k(t,s), \qquad 0 < \mu < 1,$$

where $k(t,s)$ is 'well-behaved' in the sense of continuity order, (b) the non-linear case where the kernel is now

$$K(t,s,u) = \frac{s^{\mu-1}}{t^{\mu}} k(t,s,u),$$

and (c) systems of such equations. Cases (a) and (b) have been studied theoretically by Han [32], and the underlying theory of function analysis of Chapter 2 can be extended to include these examples, as well as the case when $\mu = 1$, already dealt with.

Previous work described in Chapter 1 has reduced the original equation from an expression of considerable complexity to its innocuous-looking form in equation (1.1). A further possible substitution referred to in section (3.8) would remove the singularity from the integrand, but replace it with an unbounded integral as the input function. This led to an interesting extension to the classical approach, but we do not find the numerical approximation process made any easier.

The construction of numerical approximation to the true smooth solution described in Chapter 6 is suitable for extension to these cases. When the equation is simplified by setting $\mu = 1$, we can use the methods obtained by Lima and Diogo [43] for (a) above. When the singularity is of logarithmic type, we find that the work of Lyness and Ninham [48] and of de Hoog and Weiss [34] contains the necessary convergence results to support the application of product integration methods to such equations.

8.0.1 Theoretical Conclusions

The existing body of knowledge prior to this project showed that equation (1.1) has a unique solution for the parameter $\mu > 1$, and when $\mu < 1$ there is a single smooth solution, provided the input function is smooth, and a family of non-smooth solutions with infinite gradient at the origin. The case when $\mu = 1$ and $g(0) = 0$ has a single solution such that $u(0) = 0$, and a family of parallel solutions. This was developed by Han [32] using the related ordinary differential equation (also singular), and hence obtaining the solution formulae.

In Chapter 3 we consolidate and extend our understanding of the behaviour of this class of equations, laying the foundation for a more generalized approach. We apply the methods directly relating to integral equations, to obtain a more uniform approach to the existence and uniqueness of solution(s). In this context it is apparent that the conditions which may be placed upon the input and output

functions are of the utmost importance, and the defining of the problem is a question of defining the admissible space for the solution. If we seek an analogy to the initial-value differential equation, then the solution set determined by Han [32] is sufficient; if, however, we require the full solution set, including the case for which the initial value $u(0)$ is unbounded, then the methods of Chapter 3 are required.

Next, we consider the case when the initial first derivative of the solution $u'(0)$ may be unbounded: this was included in the solution formulae of Han, but is now treated more economically, and as a part of the overall framework. The Fredholm Alternative, as stated above, has clearly a relevance here, although equation (1.1) does not have the compactness required, and we find that the solution set is compatible with the second part of the Fredholm Alternative.

Defining the problem may now be achieved by a limitation on the function space for which the solution is defined, so that the case when the solution u : $u(t) \in C^m[0,T], \quad t \in [0,T], \quad m \geq 1$ is unique, in that it is the sole solution within that particular space. If, however, $m = 0$, then for $\mu < 1$ the set of non-smooth solutions has to be included, and uniqueness fails. We have set out during our investigation to clarify which of these situations is relevant in any particular context.

8.0.2 Numerical Approximation

We are looking specifically at the solution of equation (1.1), when $\mu < 1$, so that the kernel is singular in both arguments. Our objective is to obtain a means of numerical approximation consisting of two elements: the role of the first is to approximate over a short interval close to the origin, and the second may be any method of choice.

The primary method must have the ability to approximate from the initial singularity, so options are limited, while the secondary may be selected from any

of the methods available for VIE solution, since the evaluation of the kernel is no longer singular for the relevant values of arguments s and t, and the restrictions imposed by the singularity are now lifted.

Since the primary method is usually of lower order than the secondary, we use convergence acceleration at the end of the first interval. This process requires careful construction, in order to match the convergence of the (accelerated) primary scheme with that of the secondary.

Primary Method

We have taken the product Euler method as our main constructional element, owing to its previous use in this context, and the availability of the related convergence proofs. The disadvantage of this is that it has a poor rate of convergence, particularly for $\mu < 1$, when it is $O(h^\mu)$, but that rate of convergence is known and reliable. Further, we assume that the subsequent terms in the error expansion are of the form $O(h^k) + O(h^{k+\mu}), \quad k \in Z$, for the purpose of the extrapolation. We have introduced the product trapezoidal as an alternative for the initial interval. Using the methods of Lyness and Ninham [48], and de Hoog and Weiss [34], we have shown that this is of leading order $1 + \mu$, with further error expansion of the form $O(h^k) + O(h^{k+\mu}), \quad k = 2, 3, \ldots$. The quality of the solution approximation is shown in the examples to be a considerable improvement on the product Euler.

A further method for use over the primary interval is the Hermite-type collocation, used previously for $\mu > 1$ when it was shown to be $O(h^4)$. We find that with $\mu < 1$ we obtain empirical convergence of order $p : 3 < p < 4$, and the quality of the results is further improved.

Extrapolation

The extrapolation process is applied at the end of the first interval ($t = \alpha$), for the reasons described in section (6.1). The aim is to bring the lower order of the

primary method in line with the order of the secondary.

Remark 7 *If further levels of extrapolation are required to raise the order of the combined scheme, then this may be done at the end of the second interval.*

The choice of extrapolation scheme depends on the series expansion which is required to be accelerated. Here, the expansion is known to be of a particular form, and we have considered the schemes best suited to this case. One possibility is the E-algorithm of Brezinski, which is the most general algorithm for a known series expansion. However, this is intricate to construct, and we find that a modification of the Richardson process produces equally satisfactory results, but with the simplicity of a scheme constructed for the purpose.

The extrapolation process to raise a scheme of order p (non-integer) to order m (integer) requires $2(m-\lfloor p \rfloor)-1$ levels, hence $2(m-\lfloor p \rfloor)$ input values; however, this is an intermediate point in the overall scheme, so we take further input values to allow assessment of the convergence at the end of the second interval. The resulting extrapolation table (shown on p.93) is thus effectively a $2(m-\lfloor p \rfloor)+1$ level acceleration, curtailed at the $2(m-\lfloor p \rfloor)-1$ level, leaving the starting values for 3 trajectories using the secondary scheme.

Secondary method

In general, the order of the secondary method will be higher than that of the primary. The second interval may be approximated by any of the many schemes available for solution of Volterra equations, as here the kernel is a continuous function in both arguments, subject to the input function also complying with the appropriate continuity restriction. Our examples cannot extend to all possible cases, but we include a representative selection to demonstrate that the choice of method is not constrained in any way.

8.0.3 Combined Scheme

The split-interval scheme is constructed in such a way as to ensure a secure and reliable link between the primary and secondary methods. The extrapolation process described above plays a crucial role in this, matching the order of the primary with that of the secondary. We have set in place the analysis for the combined scheme, considered principally on the basis of the product Euler, taking three levels of extrapolation, and continuing with the trapezoidal rule. This is capable of development to involve higher order primary and secondary schemes, by extending the extrapolation accordingly. We believe in principle that this can be extended to include cases (a), (b) and (c) above.

8.0.4 Further work

We now have in place the basic precepts of a process which is capable of development in a number of directions, in particular

1. implementation of the suggestions in the preceding subsection;
2. further extension to higher-order product integration as the primary method when $\mu < 1$;
3. error analysis for the Hermite-type collocation as the primary method when $\mu < 1$;
4. the third transform suggested in section (3.8).

These are direct extensions of the work in this investigation. There are also alternative means of dealing with the initial interval, in particular

4. Clenshaw-Curtis methods;
5. the IMT transform (Iri, Moriguti and Takasawa), described in [67], [5] and [19].

Appendix A

Error analysis

A.1 Second interval: An Alternative Approach

We aim to evaluate the *local consistency error* using the definition of Linz [46] as $\delta(h, t_j)$, where

$$\delta(h, t_j) = \int_0^{t_j} K(t_j, s, f(s))ds - \sum_{i=0}^{j} w_{ji} K(t_j, t_i, f(t_i)).$$

First, assume that we have the accurate solution of $\tilde{u}(t_{j-1}) = u(t_{j-1})$ of the VIE at some point $t_{j-1} > \alpha$, and that we are looking for the error of the final step in the s dimension $[t_{j-1}, t_j]$. given by the approximation obtained by the trapezoidal rule

$$\tilde{u}(t_j) = g(t_j) + \frac{I_\alpha}{t_j^\mu} + \int_\alpha^{t_{j-1}} \frac{s^{\mu-1}}{t_j^\mu} u(s)ds + \frac{h}{2t_j^\mu}[t_{j-1}^{\mu-1}\tilde{u}_{j-1} + t_j^{\mu-1}\tilde{u}_j]$$

to the VIE solution at t_j

$$u(t_j) = g(t_j) + \frac{I_\alpha}{t_j^\mu} + \int_\alpha^{t_{j-1}} \frac{s^{\mu-1}}{t_j^\mu} u(s)ds + \int_{t_{j-1}}^{t_j} \frac{s^{\mu-1}}{t_j^\mu} u(s)ds,$$

where

$$I_\alpha = \int_0^\alpha s^{\mu-1} u(s)ds.$$

Assuming that u is smooth, using the Mean Value Theorem for integrals, for some $\xi_j \in [t_{j-1}, t_j]$ and for $j = q+1,\, q+2, \ldots$

$$\begin{aligned}
\mathbf{e_j} = \tilde{u}_j - u(t_j) &= \frac{h}{2t_j^{\mu}}\big[t_{j-1}^{\mu-1}\tilde{u}_{j-1} + t_j^{\mu-1}\tilde{u}_j\big] - \frac{1}{t_j^{\mu}}\int_{t_{j-1}}^{t_j} s^{\mu-1}u(s)ds \\
&= \frac{h}{2t_j^{\mu}}\big[t_{j-1}^{\mu-1}\tilde{u}_{j-1} + t_j^{\mu-1}\tilde{u}_j\big] - \frac{u(\xi_j)}{t_j^{\mu}}\int_{t_{j-1}}^{t_j} s^{\mu-1}ds \\
&= \frac{h}{2h^{\mu}j^{\mu}}\Big[h^{\mu-1}(j-1)^{\mu-1}\tilde{u}_{j-1} + h^{\mu-1}j^{\mu-1}\tilde{u}_j\Big] \\
&\qquad - \frac{u(\xi_j)}{h^{\mu}j^{\mu}}\frac{[h^{\mu}j^{\mu} - h^{\mu}(j-1)^{\mu}]}{\mu} \\
&= \frac{1}{2}\left[\frac{1}{j}\big(1-\frac{1}{j}\big)^{\mu-1}\tilde{u}_{j-1} + \frac{1}{j}\tilde{u}_j\right] - \frac{u(\xi_j)}{\mu}\left[1 - \big(1-\frac{1}{j}\big)^{\mu}\right] \\
&= \frac{1}{2}\left[\frac{1}{j}\left(1 - \frac{(\mu-1)}{j} + \frac{(\mu-1)(\mu-2)}{2j^2} + \ldots\right)\tilde{u}_{j-1} + \frac{1}{j}\tilde{u}_j\right] \\
&\qquad - \frac{u(\xi_j)}{\mu}\left[\frac{\mu}{j} - \frac{\mu(\mu-1)}{2j^2} + \frac{\mu(\mu-1)(\mu-2)}{6j^3} + \ldots\right]
\end{aligned}$$

Again assuming that u is smooth, we have

$$\lim_{h\to 0}\left[\frac{\tilde{u}_{j-1} + \tilde{u}_j}{2} - u(\xi_j)\right] = 0.$$

Now we assemble the remaining terms in powers of j^{-1}:

$$\begin{aligned}
\mathbf{e_j} \approx &\Big[\frac{(\mu-1)}{2j^2} + \frac{(\mu-1)(\mu-2)}{4j^3}\Big]\tilde{u}_{j-1} \\
&\qquad - \Big[\frac{(\mu-1)}{2j^2} + \frac{(\mu-1)(\mu-2)}{6j^3}\Big]u(\xi_j) + O(j^{-4}), \quad \text{(A.1)}
\end{aligned}$$

which at first sight does not look promising as a basis for local error approximation. We could use the methods of e.g. [44], but applying (A.1) to the interval of integration finds a more convenient outcome.

We now consider the error for the repeated trapezium rule over the interval $[\alpha, t_j]$, assuming that the first interval solution is accurately known, so that $j > q$, $q = \alpha/h$.

$$\tilde{u}(t_j) = g(t_j) + \frac{I_{\alpha}}{t_j^{\mu}} + \sum_{i=q}^{j-1}\frac{h}{2t_j^{\mu}}[t_i^{\mu-1}\tilde{u}_i + t_{i+1}^{\mu-1}\tilde{u}_{i+1}]$$

and the true solution may be represented as

$$u(t_j) = g(t_j) + \frac{I_\alpha}{t_j^\mu} + \sum_{i=q}^{j-1} \int_{t_i}^{t_{i+1}} \frac{s^{\mu-1}}{t_j^\mu} u(s) ds.$$

The error for $t_j \in [\alpha, T]$ is now given as

$$\begin{aligned}
E = \tilde{u}_j - u(t_j) \approx \\
&\sum_{i=q}^{j-1} \frac{h}{2t_j^\mu} \left[t_i^{\mu-1} \Big(\tilde{u}_i + \mathbf{e_i} \Big) + t_{i+1}^{\mu-1} \Big(\tilde{u}_{i+1} + \mathbf{e_{i+1}} \Big) \right] - \sum_{i=q}^{j-1} \frac{u(\zeta_i)}{t_j^\mu} \int_{t_i}^{t_{i+1}} s^{\mu-1} ds \\
&= \sum_{i=q}^{j-1} \frac{h}{2t_j^\mu} \left[t_i^{\mu-1} \Big(\tilde{u}_i + \mathbf{e_i} \Big) + t_{i+1}^{\mu-1} \Big(\tilde{u}_{i+1} + \mathbf{e_{i+1}} \Big) \right] - \sum_{i=q}^{j-1} \frac{u(\zeta_i)}{t_j^\mu} \frac{(t_{i+1}^\mu - t_i^\mu)}{\mu} \\
&= \sum_{i=q}^{j-1} \frac{h}{2h^\mu j^\mu} \left[h^{\mu-1} i^{\mu-1} \Big(\tilde{u}_i + \mathbf{e_i} \Big) + h^{\mu-1} (i+1)^{\mu-1} \Big(\tilde{u}_{i+1} + \mathbf{e_{i+1}} \Big) \right] \\
&\qquad - \sum_{i=q}^{j-1} \frac{u(\zeta_i)}{h^\mu j^\mu} \frac{(h^\mu (i+1)^\mu - h^\mu i^\mu)}{\mu}
\end{aligned}$$

for some $\zeta_i \in [t_i, t_{i+1}]$, by the Mean Value Theorem for integrals (again assuming the smoothness of u). Expanding the various terms using equation A.1 gives

$$\begin{aligned}
E = \tilde{u}_j - u_j \approx \sum_{i=q}^{j-1} \frac{h}{2h^\mu j^\mu} \Big[h^{\mu-1} i^{\mu-1} (\tilde{u}_i + \mathbf{e_i}) + h^{\mu-1} (i+1)^{\mu-1} (\tilde{u}_{i+1} + \mathbf{e_{i+1}}) \Big] \\
- \sum_{i=q}^{j-1} \frac{u(\zeta_i)}{h^\mu j^\mu} \frac{(h^\mu (i+1)^\mu - h^\mu i^\mu)}{\mu} \\
= \sum_{i=q}^{j-1} \frac{i^{\mu-1}}{2j^\mu} \Big[\tilde{u}_i + \tilde{u}_{i+1} \big(1 + \frac{(\mu-1)}{i} + \frac{(\mu-1)(\mu-2)}{2i^2} + \dots \big) \\
+ \mathbf{e_i} + \mathbf{e_{i+1}} \big(1 + \frac{(\mu-1)}{i} + \frac{(\mu-1)(\mu-2)}{2i^2} + \dots \big) \\
- \frac{u(\zeta_i)}{j^\mu} \frac{i^\mu}{\mu} \left(1 + \frac{\mu}{i} + \frac{\mu(\mu-1)}{2i^2} + \cdots - 1 \right) \Big]
\end{aligned}$$

$$= \sum_{i=q}^{j-1} \frac{i^\mu}{2j^\mu} \Bigg[\frac{\tilde{u}_i + \tilde{u}_{i+1}}{i} + \tilde{u}_{i+1} \Bigg\{ \frac{(\mu-1)}{i^2} + \frac{(\mu-1)(\mu-2)}{2i^3} + \dots \Bigg\}$$
$$\frac{\mathbf{e_i} + \mathbf{e_{i+1}}}{i} + \mathbf{e_{i+1}} \Bigg\{ \frac{(\mu-1)}{i^2} + \frac{(\mu-1)(\mu-2)}{2i^3} + \dots \Bigg\}$$
$$- \frac{u(\zeta_i)}{\mu} \Bigg\{ \frac{\mu}{i} + \frac{\mu(\mu-1)}{2i^2} + \dots \Bigg\} \Bigg]. \quad \text{(A.2)}$$

Taking the approximation

$$\lim_{h \to 0} \frac{\tilde{u}_i + \tilde{u}_{i+1}}{2} - u(\zeta_i) \to 0$$

so that the leading terms in equation (A.2) again cancel out, we have

$$E \approx \sum_{i=q}^{j-1} \frac{i^\mu}{2j^\mu} \Bigg[\tilde{u}_{i+1} \Bigg\{ \frac{\mu-1}{i^2} + \frac{(\mu-1)(\mu-2)}{2i^3} + \dots \Bigg\}$$
$$+ \tilde{u}_i \Bigg\{ \frac{(\mu-1)}{2i^3} + \frac{(\mu-1)(\mu-2)}{4i^4} \Bigg\} - u(\xi_i) \Bigg\{ \frac{(\mu-1)}{2i^3} + \frac{(\mu-1)(\mu-2)}{6i^4} \Bigg\}$$
$$+ \tilde{u}_{i+1} \Bigg\{ \frac{(\mu-1)}{2(i+1)^3} + \frac{(\mu-1)(\mu-2)}{4(i+1)^4} \Bigg\} - u(\xi_{i+1}) \Bigg\{ \frac{(\mu-1)}{2(i+1)^3} + \frac{(\mu-1)(\mu-2)}{6(i+1)^4} \Bigg\}$$
$$+ \mathbf{e_{i+1}} O(i^{-2}) - \frac{u(\zeta_i) i^\mu}{j^\mu} \Bigg\{ \frac{(\mu-1)}{2i^2} + \frac{(\mu-1)(\mu-2)}{6i^3} + \dots \Bigg\} \Bigg].$$

As $h \to 0$ we take the approximations

$$\begin{aligned} \tilde{u}_i - u(\xi_i) &\longrightarrow 0 \qquad \text{as} \qquad h \to 0 \\ \tilde{u}_{i+1} - u(\xi_i) &\longrightarrow 0 \qquad \text{as} \qquad h \to 0 \\ \tilde{u}_{i+1} - u(\zeta_i) &\longrightarrow 0 \qquad \text{as} \qquad h \to 0 \end{aligned}$$

which yields the error approximation

$$E \approx \sum_{i=q}^{j-1} \frac{i^\mu}{2j^\mu} \Bigg[\tilde{u}_i \frac{(\mu-1)(\mu-2)}{4i^4} + \tilde{u}_{i+1} \Bigg\{ \frac{(\mu-1)(\mu-2)}{2i^3} + \frac{(\mu-1)(\mu-2)}{4(i+1)^4} \Bigg\}$$
$$- u(\xi_i) \Bigg\{ \frac{(\mu-1)(\mu-2)}{6i^4} \Bigg\} - u(\xi_{i+1}) \Bigg\{ \frac{(\mu-1)(\mu-2)}{6(i+1)^4} \Bigg\}$$
$$- 2u(\zeta_i) \frac{(\mu-1)(\mu-2)}{6i^3} + O(i^{-4}). \Bigg]$$

$$= \sum_{i=q}^{j-1} \frac{i^\mu}{2j^\mu} \Bigg[\frac{\tilde{u}_{i+1}}{2i^3} - \frac{u(\zeta_i)}{3i^3} \Bigg] (\mu-1)(\mu-2) + O(i^{-4})$$

Since $q \leq i < j$, we have as $h \to 0$

$$E \leq \sum_{i=q}^{j-1} \frac{\|u\|}{2} \frac{(\mu-1)(\mu-2)}{6i^3}$$

where $\|.\|$ denotes the maximum norm $\|u\| = \max_{t\in[0,T]} |u(t)|$. We take the summation, letting $j = T/h$, T being constant. Using the fact that $q \leq i \leq j$ and since $(j-q)/q^3 = (j-q)h/hq^3 = (T-\alpha)/\alpha q^2$ we have

$$E \leq \frac{(T-\alpha)}{12q^2}\|u\|(\mu-1)(\mu-2) + O(q^{-3})$$

$$= \frac{(T-\alpha)}{12}\left(\frac{h}{\alpha}\right)^2 \|u\|(\mu-1)(\mu-2) + O\big((h/\alpha)^{-3}\big).$$

Our definition of E is equivalent to the local consistency error $\delta(h, t_j)$ since $g(t)$ and I_α/t^μ are unchanged over a single step in the t direction. We observe the way in which the terms in h and α are related. Provided $T >> \alpha$, this gives us

Lemma A.1.1 *If the true solution is known at some point α away from the origin, then the use of the trapezoidal rule in the solution of the VIE (1) has convergence of order 2. Further, the error is also related to α inversely such that $E \propto \alpha^{-2}$.*

Numerical results from higher order schemes, together with the pattern of the error expansions, indicate that this result can be generalised - i.e. whatever the order of the quadrature method used to implement the solution of the VIE (1.1) away from the origin, with the true solution applied over the initial interval $[0, \alpha]$ the order of the method is applicable to the solution of (1.1), and further there is a corresponding inverse relationship between the error and the value of α, of the same order.

The connection between the error and μ is also now plain - for the trapezoidal rule, or indeed any rule of order 2, we find that the inverse proportionality with $(\mu-1)(\mu-2)$ is now apparent, and is confirmed by numerical comparison. We believe this can be extended to rules of order 3, where the factor is $(\mu-1)(\mu-2)(\mu-3)$, and order 4, where the factor becomes $(\mu-1)(\mu-2)(\mu-3)(\mu-4)$.

Finally, we note that if the value of $\tilde{u}(\alpha)$ is other than that satisfying the smooth solution for which $c = 0$, the above analysis will still apply, giving convergence to the non-smooth solution $u_c(t) = u_0(t) + c_0 t^{1-\mu}$, the only change being to the value of I_α. This has important implications for the propagated error.

Appendix B

Tabulated Results

B.1 Secondary Methods

$\mu = 0.4, \quad T = 10, \quad g(t) = 1 + t$

Gradients of log(error) against log(h) and log(α)							
Euler methods							
Fwd Euler				Bwd Euler			
α	A_0	h	A_1	α	A_0	h	A_1
0.2	0.9887	1/20	−0.9056	0.2	1.0106	1/20	−0.9425
0.4	0.9941	1/40	−0.9153	0.4	1.0058	1/40	−0.9339
0.6	0.9959	1/80	−0.9201	0.6	1.0041	1/80	−0.9294
0.8	0.9968	1/160	−0.9224	0.8	1.0032	1/160	−0.9271
1.0	0.9973	1/320	−0.9236	1.0	1.0026	1/320	−0.9259
		1/640	−0.9242			1/640	−0.9253

Trapezium Rule							
Tr. rule $\mu = 0.4$				Tr. rule $\mu = 0.02$			
α	A_0	h	A_1	α	A_0	h	A_1
0.2	1.9962	1/20	−2.1141	0.2	1.9967	1/20	−2.0400
0.4	1.9991	1/40	−2.1200	0.4	1.9992	1/40	−2.0451
0.6	1.9996	1/80	−2.1216	0.6	1.9996	1/80	−2.0464
0.8	1.9998	1/160	−2.1219	0.8	1.9998	1/160	−2.0467
1.0	1.9999	1/320	−2.1220	1.0	1.9999	1/320	−2.0468
		1/640	−2.1221			1/640	−2.0468

Simpson's rule							
Trap.rule for non-smooth trajectory				Simpson's			
α	A_0	h	A_1	α	A_0	h	A_1
0.2	1.9962	1/20	−2.1141	0.2	4.0006	1/20	−3.4869
0.4	1.9991	1/40	−2.1200	0.4	4.0656	1/40	−3.5378
0.6	1.9996	1/80	−2.1216	0.6	4.0670	1/80	−3.5298
0.8	1.9998	1/160	−2.1219	0.8	4.0594	1/160	−3.5082
1.0	1.9999	1/320	−2.1220	1.0	4.0505	1/320	−3.4915
		1/640	−2.1221			1/640	−3.6170

Runge-Kutta							
RK classic				RK implicit			
0.2	2.9667	1/20	−2.8675	0.2	3.0153	1/20	−3.0350
0.4	2.9830	1/40	−2.8990	0.4	3.0093	1/40	−3.0253
0.6	2.9884	1/80	−2.9128	0.6	3.0066	1/80	−3.0183
0.8	2.9911	1/160	−2.9192	0.8	3.0051	1/160	−3.0143
1.0	2.9928	1/320	−2.9223	1.0	3.0041	1/320	−3.0121
		1/640	−2.9238			1/640	−3.0109

Radau 1a							
Radau 1a 2-stage				Radau 1a 3-stage			
α	A_0	h	A_1	α	A_0	h	A_1
0.2	3.0153	1/20	−3.0414	0.2	4.9708∗	1/20	−4.4225∗
0.4	3.0093	1/40	−3.0317	0.4	5.0011	1/40	−4.4655
0.6	3.0066	1/80	−3.0248	0.6	4.9960	1/80	−4.4665
0.8	3.0051	1/160	−3.0207	0.8	5.0234	1/160	−4.4350
1.0	3.0042	1/320	−3.0186	1.0	4.9805	1/320	
		1/640	−3.0174			1/640	
∗ Highly accurate results - see text for detail							

Radau2 2-stage				Radau2 3-stage			
α	A_0	h	A_1	α	A_0	h	A_1
0.2	2.9744	1/20	−2.9632	0.2	4.9073	1/20	−4.3190
0.4	2.9884	1/40	−2.9889	0.4	4.9710	1/40	−4.4104
0.6	2.9925	1/80	−3.0000	0.6	4.9664	1/80	−4.4434
0.8	2.9944	1/160	−3.0051	0.8	4.8358	1/160	−4.4452
1.0	2.9956	1/320	−3.0075	1.0	4.7233	1/320	−4.0021
		1/640	−3.0087			1/640	∗
∗ Rounding errors affected result.							

Radau2A 2-stage $g(t) = 1 + t$				Radau2A 3-stage $g(t) = t - t^2.5/10$			
α	A_0	h	A_1	α	A_0	h	A_1
0.2	2.9983	1/20	−3.1825	0.2	4.9053	1/20	−4.5054
0.4	3.0013	1/40	−3.1892	0.4	4.9785	1/40	−4.5699
0.6	3.0014	1/80	−3.1897	0.6	4.9926	1/80	−4.5846
0.8	3.0013	1/160	−3.1893	0.8	4.9970	1/160	
1.0	3.0011	1/320	−3.1889	1.0	4.9972	1/320	
		1/640	−3.1884			1/640	

Lobatto							
Lobatto3c 2-stage				Lobatto3c 3-stage			
α	A_0	h	A_1	α	A_0	h	A_1
0.2	2.0322	1/20	−2.0104	0.2	4.0182	1/20	−3.4658
0.4	2.0171	1/40	−1.9837	0.4	4.0107	1/40	−3.4559
0.6	2.0116	1/80	−1.9694	0.6	4.0842	1/80	−3.4478
0.8	2.0088	1/160	−1.9620	0.8	4.1299	1/160	−3.4411
1.0	2.0072	1/320	−1.9583	1.0	4.2494	1/320	−3.4498
		1/640	−1.9564			1/640	−3.9638

Gauss-Legendre							
GL 2-stage $g(t) = 1 + t$				GL 3-stage $g(t) = t - t^{2.5}/10$			
0.2	3.9875	1/20	−4.2051	0.2	5.8344	1/20	−5.4434
0.4	3.9905	1/40	−4.2241	0.4	5.9510	1/40	−5.5442
0.6	3.9861	1/80	−4.2291	0.6	5.9658	1/80	−5.5434
0.8	4.0187	1/160	−4.2301	0.8	5.8806	1/160	
1.0	3.9821	1/320	−4.2074	1.0	5.9505	1/320	
		1/640	−4.1934			1/640	

Multistep methods							
BDF $O(h^2)$				AM $O(h^3)$			
0.2	1.9199	1/20	−1.9796	0.2	2.8388	1/20	−2.9313
0.4	1.9587	1/40	−2.0437	0.4	2.9150	1/40	−3.0580
0.6	1.9725	1/80	−2.0807	0.6	2.9430	1/80	−3.1325
0.8	1.9794	1/160	−2.1007	0.8	2.9574	1/160	−3.1732
1.0	1.9836	1/320	−2.1112	1.0	2.9661	1/320	−3.1945
		1/640	−2.1166			1/640	−3.2054

Appendix C

Runge-Kutta Methods

Examples of RK arrays up to 3-stage.

Sources: [3], [14], [15] [16], [31], [35], [41].

A comprehensive derivation, and listing up to 4-stage rules are given in [15].

m = number of stages in the method, p = order of the method

C.1 RK standard methods

RK Classical

$m = 3,\ p = 3$

0			
$\frac{1}{2}$	$\frac{1}{2}$		
1	-1	2	
	$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$

RK Implicit

$m = 2,\ p = 3$

0	$\frac{1}{4}$	$\frac{-1}{4}$
$\frac{2}{3}$	$\frac{1}{4}$	$\frac{5}{12}$
	$\frac{1}{4}$	$\frac{3}{4}$

RK Nystrom

$m = 3,\ p = 3$

0			
$\frac{2}{3}$	$\frac{2}{3}$		
1	-1	2	
	$\frac{1}{4}$	$\frac{3}{8}$	$\frac{3}{8}$

C.2 Radau I

p = 2m − 1

$m = 2,\ p = 3$

0	0	0
$\frac{2}{3}$	$\frac{1}{3}$	$\frac{1}{3}$
	$\frac{1}{4}$	$\frac{3}{4}$

$m = 3,\ p = 5$

0	0	0	0
$\frac{6-\sqrt{6}}{10}$	$\frac{9+6\sqrt{6}}{75}$	$\frac{24+\sqrt{6}}{120}$	$\frac{168-73\sqrt{6}}{600}$
$\frac{6+\sqrt{6}}{10}$	$\frac{9-6\sqrt{6}}{75}$	$\frac{168+73\sqrt{6}}{600}$	$\frac{24-\sqrt{6}}{120}$
$\frac{1}{9}$	$\frac{16+\sqrt{6}}{36}$	$\frac{16-\sqrt{6}}{36}$	

C.3 Radau IA

$m = 2,\ p = 3$

0	$\frac{1}{4}$	$\frac{-1}{4}$
$\frac{2}{3}$	$\frac{1}{4}$	$\frac{5}{12}$
	$\frac{1}{4}$	$\frac{3}{4}$

$m = 3,\ p = 5$

0	$\frac{1}{9}$	$\frac{-1-\sqrt{6}}{18}$	$\frac{-1+\sqrt{6}}{18}$
$\frac{6-\sqrt{6}}{10}$	$\frac{1}{9}$	$\frac{88+7\sqrt{6}}{360}$	$\frac{88-43\sqrt{6}}{360}$
$\frac{6+\sqrt{6}}{10}$	$\frac{1}{9}$	$\frac{88+43\sqrt{6}}{360}$	$\frac{88-7\sqrt{6}}{360}$
	$\frac{1}{9}$	$\frac{16+\sqrt{6}}{36}$	$\frac{16-\sqrt{6}}{36}$

C.4 Radau IIA

$m = 1,\ p = 1$

1	1
	1

$m = 2,\ p = 3$:

$\frac{1}{3}$	$\frac{5}{12}$	$\frac{-1}{12}$
1	$\frac{3}{4}$	$\frac{1}{4}$
	$\frac{3}{4}$	$\frac{1}{4}$

$m = 3,\ p = 5$:

$\frac{4-\sqrt{6}}{10}$	$\frac{88-7\sqrt{6}}{360}$	$\frac{296-169\sqrt{6}}{1800}$	$\frac{-2+3\sqrt{6}}{225}$
$\frac{4+\sqrt{6}}{10}$	$\frac{296+169\sqrt{6}}{1800}$	$\frac{88+7\sqrt{6}}{360}$	$\frac{-2-3\sqrt{6}}{225}$
1	$\frac{16-\sqrt{6}}{36}$	$\frac{16+\sqrt{6}}{36}$	$\frac{1}{9}$
	$\frac{16-\sqrt{6}}{36}$	$\frac{16+\sqrt{6}}{36}$	$\frac{1}{9}$

C.5 Lobatto IIIC

$m = 2,\ p = 2$:

0	$\frac{1}{2}$	$\frac{-1}{2}$
1	$\frac{1}{2}$	$\frac{1}{2}$
	$\frac{1}{2}$	$\frac{1}{2}$

$m = 3,\ p = 4$:

0	$\frac{1}{6}$	$\frac{-1}{3}$	$\frac{1}{6}$
$\frac{1}{2}$	$\frac{1}{6}$	$\frac{5}{12}$	$\frac{-1}{12}$
1	$\frac{1}{6}$	$\frac{16+\sqrt{2}}{3}$	$\frac{1}{6}$
	$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$

C.6 Gauss-Legendre

$s = 1,\ p = 2$:

$\frac{1}{2}$	$\frac{1}{2}$
	1

$s = 2,\ p = 4$:

$\frac{3-\sqrt{3}}{6}$	$\frac{1}{4}$	$\frac{3-2\sqrt{3}}{12}$
$\frac{3=\sqrt{3}}{6}$	$\frac{3+2\sqrt{3}}{12}$	$\frac{1}{4}$
	$\frac{1}{2}$	$\frac{1}{2}$

$s = 3,\ p = 6$:

$\frac{5-\sqrt{5}}{10}$	$\frac{5}{36}$	$\frac{10-3\sqrt{15}}{45}$	$\frac{25-6\sqrt{15}}{180}$
$\frac{1}{2}$	$\frac{10+3\sqrt{15}}{72}$	$\frac{2}{9}$	$\frac{10-3\sqrt{15}}{72}$
$\frac{5+\sqrt{5}}{10}$	$\frac{25+6\sqrt{15}}{180}$	$\frac{10+3\sqrt{15}}{45}$	$\frac{5}{36}$
	$\frac{5}{18}$	$\frac{4}{9}$	$\frac{5}{18}$

Appendix D

A VLM Method

We describe in Chapter 5 several alternative means of implementing the linear multstep methods in the context of a Volterra equation (VLM methods). We now set out the algorithm we use as a possible secondary method for our split-interval scheme. Essentially, this is based on an iterative process applied to the discretisation of the equation

$$u(t) = g(t) + t^{-\mu} I_\alpha + t^{-\mu} \int_0^t s^{\mu-1} u(s) ds$$

which we now represent as

$$u_n(t_n) = g(t_n) + t_n^{-\mu} I_\alpha + \Phi_n, \tag{D.1}$$

our objective being to construct the term Φ_n to be of normalised k-step Adams-Moulton type,

$$y_n = y_{n-1} + h \sum_{i=0}^{k} b_i y_{n-i},$$

where $a_o = -a_1 = 1$. Let

$$\Phi_n = \Phi_{n-1} + \sum_{i=0}^{k} b_i f_{n-i}, \qquad f_{n-i} = \frac{t_{n-i}^{\mu-1}}{t_n^{\mu}} u_{n-i}$$

which is implicit for D.1 when $i = 0$, to give

$$u_n(t_n) = \left[g(t_n) + t^{-\mu} I_\alpha + \Phi'_n \right] / \left[1 - h b_0 / t_n \right]$$

where the prime $'$ denotes exclusion of the f_0 term. Now, we look at Φ_{n-1}, noting that in constructing the code, this has been separated during the implicit calculaton, and needs re-installing here, with the denominator now updated to t_n^μ, and we introduce the additional subscript, to give

$$\Phi_{n,n-1} = \Phi'_{n,n-1} + b_0 h t_{n-1}^{\mu-1}/t_n^\mu.$$

With these issues dealt with, we are now able to construct the VLM method, AM3:. For starting values, we use two steps of the trapezium rule, using the usual implicit quadrature, with weights w_s:

$$\{w_s\} = \begin{bmatrix} 1/2 & 1/2 & \\ 1/2 & 1 & 1/2 \end{bmatrix}$$

We can now define the recursive VLM method to be

$$u_n(t_n) = \left[g(t_n) + t_n^{-\mu} + \Phi_{n,n-1} + h \left\{ \sum_{1=1}^{k} b_i u_{n-i} \right\} \right] \frac{1}{(1 - b_0 h/t_n)},$$

where

$$\Phi_{n,n-1} = \Phi_{n,n-1} + \frac{b_0 h t_{n-1}^\mu}{t_n^\mu}$$

This is the formula we use in chapter 7, and we note that the expression usually described as 'lag' or 'tail' terms are now incorparated in $\Phi_{n,n-1}$ and that the implicit term in b_0 has to be recovered in each calculation.

We illustrate this using the 2-step Adams-Moulton scheme, where $\{\alpha; \beta\} = \{12,\ -12;\ 5\ 4\ -1\}$, so that setting $a_0 = -a_1 = 1,\ b_0 = \frac{5}{12},\ b_1 = \frac{2}{3}\ b_2 = \frac{-1}{12}$, we have

$$u_n(t_n) = \left[g(t_n) + \frac{I_\alpha}{t_n^\mu} + \Phi_{n,n-1} + h \left\{ \frac{2t_{n-1}^{\mu-1} u_{n-1}}{3t_n^\mu} - \frac{t_{n-2}^{\mu-1} u_{n-1}}{12 t_n^\mu} \right\} \right] \frac{1}{1 - \frac{5h}{12 t_n}}$$

where

$$\Phi_3 = \frac{t_q^{\mu-1} u_q}{2t_{q+3}^\mu} + \frac{t_{q+1}^{\mu-1} u_{q+1}}{t_{q+3}^\mu} + \frac{t_{q+2}^{\mu-1} u_{q+2}}{2t_{q+3}^\mu};$$

$$\Phi_n = \Phi_{n-1} \frac{t_{n-1}^\mu}{t_n^\mu} + \frac{5}{12} \frac{t_{n-1}^{\mu-1}}{t_n^\mu} u_{n-1} + h \left[\frac{2}{3} \frac{t_{n-1}^{\mu-1}}{t_n^\mu} u_{n-1} - \frac{1}{12} \frac{t_{n-2}^{\mu-1}}{t_n^\mu} u_{n-2} \right]$$

The core of the code for this iteration is a mere 5 lines.

Appendix E

Smoothness Properties

During this investigation, we looked for a comparison between the various categories of smoothness properties which can be applied. Davis and Rabinowitz [19] give the following list, in ascending order of smoothness:

Smoothness of functions and approximate integration

1. Functions that are bounded and Riemann-integrable over $[a, b]$;
2. Functions that are of bounded variation over $[a, b]$;
3. Functions that are piecewise continuous over $[a, b]$;
4. Functions that are continuous over $[a, b]$;
5. Functions that satisfy a Lipschitz or Holder condition of order $\alpha \leq 1$ over $[a, b]$;
6. Functions that have a continuous first derivative over $[a, b]$;
7. Functions that have a continuous nth derivative over $[a, b]$;
8. Functions that are analytic in a region B containing the interval in its interior;

9. Functions that are entire: i.e. have a Taylor expansion convergent in $|z| < \infty$.

10. Functions that are polynomials of degree $\leq n$.

A class that does not fit into this scheme is the class $L^p[a,b]$ or $L[B]$, $1 \leq p < \infty$, the set of all Lebesgue measurable functions f on $[a,b]$ or B such that

$$\int_a^b |f(x)|^p dx \qquad \text{or} \qquad \int_B |f|^p \, dV$$

is finite, as the case may be. [19].

Appendix F

Glossary of Terms

ϕ	the unknown function of the general Volterra integral equation
ψ	the input function of the general Volterra integral equation
$g(t)$	the input function of the Volterra integral equation (1.1)
$u(t)$	the unknown function for solution of the main equation, as a continuum
$\tilde{u}(t_n)$	the numerical approximation to $u(t_n)$ at points t_n
μ	the parameter occurring in the above equations:
	our principal concern is with $\mu \in (0, 1)$
c_0	arbitrary constant which defines a non-smooth solution of the main equation
α	initial interval over which a primary approximate solution is found;
	the value of t which terminates such an interval
q	the value of n such that $t_n = \alpha$
I_α	the value of the integral term at α: $\int_0^\alpha s^{\mu-1} u(s)\, ds$
$\tilde{I}_\alpha$	approximation for I_α
$E_k^{(n)}$	element of E-algorithm array
$T_k^{(n)}$	element of Richardson extrapolation array
$Y_k^{(n)}$	element of modified Richardson array

Bibliography

[1] M. Abramowitz and I.E. Stegun. *Handbook of Mathematical Functions*. Dover, N.Y. 1965 (rep. 1972).

[2] T.M. Apostol. *Mathematical Analysis*. Addison-Wesley, Reading, Mass. 1957 (rep. 1965). 1957.

[3] U.M. Ascher and L.R. Petzold. *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. SIAM, Philadelphia. 1998.

[4] K.E. Atkinson. *An Introduction to Numerical Analysis*. 2nd ed. Wiley, N.Y. 1989.

[5] K.E. Atkinson. *The Numerical Solution of Integral Equations of the Second Kind*. Cambridge University Press. 1997.

[6] C.T.H. Baker. *The Numerical Treatment of Integral Equations*. Oxford University Press. 1977 (rep. 1978).

[7] C.T.H. Baker. *Numerical Analysis of Volterra Functional and Integral Equations*. in: The State of the Art in Numerical Analysis. (eds. I.S. Duff and G.A. Watson). Oxford University Press, 1997. p.193-222.

[8] C.T.H. Baker. *A Perspective on the Numerical Treatment of Volterra Equations*. J.C.A.M. **125**, 2000. p.217-249.

[9] M.A. Bartosevich. *On a heat conduction problem*. Inz. Fiz. Z. **28**(2), 1975. p.340-346 (in Russian).

[10] M.A.Bartosevich. *Expansion in one orthogonal system of Watson Operators for solving heat conduction problems*. Inz. Fiz. Z. **28**(3), 1978. p.516-522 (in Russian).

[11] M. Bernkopf. *The Development of Function spaces with Particular Reference to their Origins in Integral Equation Theory*. Arch. Hist. Exact Sci. **3**, 1966. p.1-96.

[12] G.J. Borse. *Numerical Methods with Matlab.* PWS, Boston. 1997.

[13] C. Brezinski and M. Redivo-Zaglia. *Extrapolation Methods - Theory and Practice*. NorthHolland, Amsterdam. 1991.

[14] H. Brunner and P. van der Houwen. *The Numerical Solution of Volterra Equations.* NorthHolland, Amsterdam. 1986.

[15] J.C.Butcher. *The Numerical Analysis of Ordinary Differential Equations: Runge-Kutta and General Linear Methods.* Wiley, Chichester, U.K. 1987.

[16] J.C. Butcher. *Numerical Methods for Ordinary Differential Equations*. Wiley, Chichester, U.K. 2003.

[17] R.V. Churchill. *Operational Mathematics.* McGraw-Hiill, Boston, Mass. (3rd ed.) 1972.

[18] C. Corduneanu. *Integral Equations and Applications.* Cambridge University Press. 1991.

[19] P.J. Davis and P. Rabinowitz. *Methods of Numerical Integration*. Academic Press Inc. (London) Ltd. (2nd ed.) 1984.

[20] T. Diogo, J.T. Edwards, N.J. Ford, P. Lima and S.M. Thomas. *Numerical Analysis of a Singular Integral Equation*. Appl. Math. Comput. (to appear).

[21] T. Diogo, J.M. Ford, N.J. Ford and P. Lima *Non-integrable resolvent kernels and qualitative behaviour for exact and approximate solutions to some convolution integral equations*. (preprint).

[22] T. Diogo, N. J. Ford, P. Lima, S. M. Thomas. *Solution of a singular integral equation by a split-interval method*. Comm. Pure Appl. Anal. (to appear).

[23] T. Diogo, N. B. Franco, P. Lima. *High order product integration methods for a Volterra integral equation with logarithmic singular kernel*. Comm. Pure Appl. Anal. **3**, 2004. n.2, p.217-235.

[24] T. Diogo, P. Lima. *Comparative study of numerical methods for a class of integral equations with weakly singular kernel*. HERCMA2001, ed. Lipitakis. LEA, 2002. p.574-581.

[25] T. Diogo, P. Lima, S. Valtchev and N.J. Ford. *Numerical Methods for a nonuniquely solvable Volterra integral equation*. CILAMCE, XXIV Iberian Latin-American Congress on Computational Methods in Engineering, 2003.

[26] T. Diogo, S. Mckee, T. Tang. *A Hermite-Type Collocation Method for the solution of an Integral Equation with a Certain Weakly Singular Kernel*. IMA J. Numer. Anal. **11**, 1991. p.595-605.

[27] D. Elliott. *An asymptotic analysis of two algorithms for certain Hadamard finite-part integrals*. IMA J. Numer. Anal. **13**, 1993. p.445-462.

[28] G. Evans. *Practical Numerical Analysis*. Wiley, Chichester, UK. 1995.

[29] G. Evans. *Practical Numerical Integration*. Wiley, Chichester, U.K. 1993.

[30] G. Gripenberg, S-O. Londen and R. Staffans. *Volterra Integral and Functional Equations*. Cambridge University Press. 1990.

[31] E. Hairer, S.P. Norsett and G. Wanner. *Solving Ordinary Differential Equations I: Non-stiff Problems*. Springer-Verlag, Berlin. 1993.

[32] W. Han. *Existence, Uniqueness and Smoothness Results for Second-Kind Volterra Equations with Weakly Singular Kernels.* J. Integral Equations Appl. **6,** 5. 1994. p.365-384.

[33] H. Hochstadt. *Integral Equations.* Wiley, N.Y. 1973.

[34] F. de Hoog and R. Weiss. *Asymptotic Expansions for Product Integration.* Math. Comp. **27**, 1973. p.295-306.

[35] A. Iserles. *A First Course in Numerical Analysis of Differential Equations.* Cambridge University Press. 1996.

[36] A.J. Jerri. *Introduction to Integral Equations with Applications. 2nd ed.* Wiley, Chichester. 1999.

[37] A.N. Kolmogorov and S.V. Lomin. *Functional Analysis. vol. 1: Metric and Normed Spaces.* Graylock Press, Rochester, N.Y. 1957. (Tr. from 1st Russian edition (1954) by Leo F. Bevan).

[38] E. Kreyszig. *Introductory Functional Analysis with Applications.* Wiley, N.Y. 1978.

[39] A.J. Krommer and C.W. Ueberhuber. *Computational Integration.* SIAM, Philadelphia. 1998.

[40] W. Lamb. *A Spectral Approach to an Integral Equation.* Glasgow Math. J. **26**, 1985. p.83-89.

[41] J.D. Lambert. *Numerical Methods for Ordinary Differential Systems.* Wiley, Chichester, U.K. 1991.

[42] M.J. Lighthill. *Fourier Analysis and Generalised Functions.* Cambridge University Press. 1958 (Rep.1978).

[43] P. Lima and T. Diogo. *An extrapolation method for a Volterra integral equation with weakly singular kernel.* Appl. Num. M. **24**, 1997. p.131-148.

[44] P. Lima and T. Diogo. *Numerical Solution of a non-uniquely solvable Volterra integral equation using extrapolation methods.* JCAM **140**, 2002. p.537-557.

[45] P. Linz. *Numerical Methods for Voltera Equations with Singular Kernels.* SIAM J. Numer. Anal. **6**, 1969. p.365-374.

[46] P. Linz. *Analytical and Numerical Methods for Volterra Equations.* SIAM, Philadephia. 1985.

[47] E.R. Localzo. *in:* T.N.E Greville. *Theory and Applications of Spline Functions*, Academic Press, N.Y. 1969. p.37-44.

[48] J.N. Lyness and B.W. Ninham. *Numerical Quadrature and Asymptotic Expansions.* Math. Comp. **21**, 1967. p.162-178.

[49] S. McKee and H. Brunner. *The Repetition Factor and Numerical Stability of Volterra integral equations.* Comp. Math. Appl. **6**, 1980. p.339-347.

[50] S. McKee, T. Tang and T. Diogo. *An Euler-type Method for two dimensional Volterra Integral Equations of the first kind.* IMA J. Numer. Anal. **20**, 2000. p.423-44.

[51] A.D.Polyanin and A.V.Manzhirov. *Handbook of Integral Equations* CRC, Boca Raton. 1998.

[52] C.D. Poole. *A Basic Analysis of a second-kind Volterra Integral Equation with non-Lipschitz kernel.* M.Sc. Thesis. Chester College, 2002.

[53] P.M. Prenter. *Splines and Variational Methods.* Wiley-Interscience, N.Y. 1975.

[54] L. Rade and B. Westergren. *Mathematics Handbook for Science and Engineering*. 5th ed. Springer-Verlag, Berlin. 2004.

[55] F. Riesz and B. Sz-Nagy. *Functional Analysis*. Dover, N.Y. 1990.

[56] P.G. Rooney. *On an Integral Equation by Sub-Sizhonenko*. Glasgow Math. J. **24**, 1983. p.201-210.

[57] F. Scheid. *Numerical Analysis*. McGraw-Hill, New York. 1988.

[58] L.F. Shampine. *Numerical Solution of Ordinary Differential Equations*. Chapman and Hall, London. 1994.

[59] A. Sidi. *Practical Extrapolation Methods - Theory and Applications*. Cambridge University Press. 2003.

[60] V.L. Smirnov. *Course of Higher Mathematics*. Vol.IV. Pergamon, London. 1964.

[61] F. Smithies. *Integral Equations*. Cambridge University Press. 1962.

[62] A.M. Stuart and A.M. Humphries. *Dynamical Systems and Numerical Analysis*. Cambridge University Press. 1998.

[63] Y.A. Sub-Sizhonenko. *Inversion of an Integral Operator by the method of Expansion with respect to Orthogonal Watson Operators*. Siberian Math. Journal. **20**, 1979. p.318-321.

[64] T. Tang, S. McKee and T. Diogo. *Product Integration Methods for an Integral Equation with a weakly singular kernel*. Appl. Num. M. **9**, 1992. p.259-266.

[65] E.C. Titchmarsh. *The Theory of Functions*. Oxford University Press. 2nd ed. 1939 (repr. 1997).

[66] F.G. Tricomi. *Integral Equations.* Dover, New York. 1985. (orig. pub. Interscience, New York. 1957.)

[67] C.W. Ueberhuber. *Numerical Computation: Methods, Software and Analysis.* Springer-Verlag, Berlin. 1997. (2 vols.)

[68] E.J. Weniger. *Nonlinear Sequence Transformations for the Acceleration of Convergence and the Summation of Divergent Series.* Computer Physics Reports. **10**, 1989. p.189-371.

[69] E.T.Whittaker and G.N. Watson. *A Course of Modern Analysis* (4th ed.). Cambridge University Press. 1963.

[70] D.V.Widder. *The Heat Equation.* Academic Press Inc. (London) Ltd. 1975.

[71] P.H.M.Wolkenfeldt. *The Construction of Reducible Quadrature Rules for Volterra Integral and Integro-differential Equations.* IMA J. Numer. Anal. **2.** 1982. p.131-152.