Approximations for Singular Integral Equations

Darwin T. Herdman

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements of the degree of

Masters of Science
In
Mathematics

John A. Burns, Chair
Jeff Borggaard
Terry L. Herdman

12 May 1999
Northern Virginia Campus

Keywords: Differential Equations, Aeroelastic, Volterra

Approximations for Singular Integral Equations

Darwin T. Herdman

(ABSTRACT)

This work is a numerical study of a class of weakly singular neutral equations. The motivation for this study is an aeroelastic system. Numerical techniques are developed to approximate the singular integral equation component appearing in the complete dynamical model for the elastic motions of a three degree of freedom structure, an airfoil with trailing edge flap, in a two dimensional unsteady flow. The flap can be viewed as an active control surface to dampen vibrations that contribute to flutter. The goal of this work is to provide accurate approximations for weakly singular neutral equations using finite elements as basis functions for the initial function space. Several examples are presented in order to validate the numerical scheme.

Acknowledgements

I would like to acknowledge the hard work and assistance that many people have contributed to the completion of this work. First, I would like to thank my advisor Dr. John Burns for all the guidance and assistance he has provided throughout my academic career. His complete and thorough understanding of both academia and industry has provided valuable motivation and insight into the utility of applied mathematics.

Special thanks to the members of my committee for their continuous support over the past three years. Particular thanks to Dr. Jeffrey Borggaard for contributing his assistance and subject matter expertise. Furthermore, his constructive reviews have provided valuable comments and recommendations to this manuscript.

Sincere thanks to Melissa Chase, program support technician of ICAM, for her assistance.

Finally, I would like to thank my parents. To my father and long time mentor, Dr. Terry Herdman, I can not thank you enough for all that you have provided. Whether in sports, academics, or life you have provided valuable guidance and challenged me to succeed. I will forever be your biggest fan and admirer. To my mother and best friend, Carol Herdman, I would like to thank you for the unconditional love and support you have provided throughout my life. I could not have accomplished anything without you.

Table of Contents

List of Figures

Chapter 1

1. INTRODUCTION

In [5, Section 4] Burns, Herdman and Stech developed a solution representation for the scalar neutral functional differential equation (NFDE)

$$\frac{d}{dt}Dx_t = 0, \quad t > 0, \tag{1.1}$$

with initial function

$$x_0 = \varphi. \tag{1.2}$$

on [-1,0]. The non-atomic (no atom at zero) operator D is defined on the space of continuous functions C[-1,0] by

$$D\psi = \int_{-1}^{0} (-s)^{-\alpha}\psi(s)ds \tag{1.3}$$

with α satisfying $0<\alpha<1$. In this treatment of (1.1)-(1.2) the authors considered the associated neutral equation, integrated form of (1.1),

$$Dx_t = \eta \tag{1.4}$$

with $\eta \in R$, $x_0 = \varphi$. Given φ in C[-1,0] and η in R the unique integrable solution of the initial value problem has the representation

$$\begin{aligned} x(t) = {} & \frac{\sin(\alpha\pi)}{\pi}\int_{-1}^{0}\frac{1}{t-s}\left|\frac{t}{s}\right|^{\alpha}\varphi(s)ds \\ & + \frac{\sin(\alpha\pi)}{\pi}\int_{0}^{t}\frac{(t-s)^{\alpha-1}}{(t-s)+1}\varphi(s-1)ds + [\eta - D\varphi]t^{\alpha-1} \qquad 0 < t \le 1. \end{aligned} \tag{1.5}$$

It is to be noted that a necessary condition for the solution x, given by (1.5), to be continuous at t = 0 is that $D\varphi = \eta$. In that case we have that the initial value problem is well-posed on C[-1,0] and the unique continuous solution is given by (1.5) with $D\varphi = \eta$. The method of steps can be employed to obtain the unique continuous solution of (1.4)-(1.5) on $[-1,\infty)$.

The case of $\alpha = 1/2$ is of particular interest in that a neutral equation having a weakly singular kernel, order of singularity ½, arises in mathematical models of aerodynamics, (see [4 and references within]); we present this model in Chapter 2. In particular, the complete dynamical model of the motion of an airfoil in an unsteady flow developed in [4] has a singular neutral equation component of the form

$$\frac{d}{dt}\int_{-1}^{0}\left(\frac{Us-2}{Us}\right)^{1/2} x(t+s)ds = h(t), \quad t>0, \quad x_0=\varphi \ on\ [-1,0] \tag{1.6}$$

where the function h is a known integrable function and the constant U denotes the stream velocity of the flow. The kernel function for (1.6) can be viewed as $c(s)(-s)^{-1/2}$ where the function c is defined by

$$c(s) \equiv \left(-s+\frac{2}{U}\right)^{1/2}, \quad -1 \le s \le 0. \tag{1.7}$$

It is to be noted that the function c defined by (1.7) is continuous and has continuous derivatives of all orders on [-1,0]. The integrated form for (1.6), integrated over [0,t], is given by

$$\int_{-1}^{0} c(s)(-s)^{-1/2} x(t+s)ds = g(t), \ t>0, \ x_0=\varphi \tag{1.8}$$

with

$$g(t) \equiv \int_{-1}^{0} c(s)(-s)^{-1/2}\varphi(s)ds + \int_{0}^{t} h(s)ds$$

Here we have implicitly assumed that the resulting integrals exist and that the integration performed is valid. Also we used the fact that $x(s) = \varphi(s)$, $-1 \le s \le 0$.

In this presentation we will provide numerical approximations for unique solutions of a general class of singular neutral equations which include equation (1.8) as a special case. In particular, we shall investigate solution representation for equations of the form

$$\int_{-1}^{0} k(s)x(t+s)ds = g(t), \ t>0, \ x_0=\varphi \ on\,[-1,0] \tag{1.9}$$

where

$$k(s) \equiv c(s)(-s)^{-\alpha} \qquad -1 \le s \le 0 \tag{1.10}$$

with $0<\alpha<1$, $c \in C^n[-1,0]$ for all natural numbers n. The singular neutral equation (1.8) is a special case of (1.9)–(1.10) with $\alpha = 1/2$ and c given by (1.7). The scalar neutral equation (1.1)-(1.2) found in [5] is a special case of (1.9) – (1.10) with $c(s) \equiv 1$, $g(t) = \eta$.

Cao, Cerezo, Herdman and Turi [8] have studied solution representations for systems of the form (1.9)-(1.10). Their work was based on transforming (1.9)-(1.10) into a Volterra integral equation of the second kind. The authors first established that the solution x for

(1.9) is the unique solution of an associated Volterra equation. Once this was established, results from the theory of singular Volterra integral equations of the second kind were employed to obtain representation results for the singular NFDE (1.9).

We will restrict our attention to developing approximations for solutions to singular integral equations of the form

$$\frac{d}{dt}\int_{-1}^{0} k(s)x(t+s)ds = f(t) \quad t>0 \tag{1.11}$$

with initial function

$$x_0 = \varphi \quad on \quad [-1,0] \tag{1.12}$$

where $f \in C[0,\infty)$, $\varphi \in C[-1,0]$ and the kernel k has the representation

$$k(s) = c(s)(-s)^{-\alpha} \quad 0<\alpha<1, \quad c \in C^n[-1,0], \; n = 1,2,.... \tag{1.13}$$

The initial value problem (1.11)-(1.13), with $c \equiv 1$ has been extensively studied concerning well-posedness (on continuous and L^p spaces) [5,6,7,16,18], numerical approximations [9,17] and parameter identification [3,10].

In this presentation we will have particular interest in the cases: $c(s)=1, -1 \le s \le 0$, $\alpha = \frac{1}{2}$; $c(s) = (2-s)^{\alpha}$, $0<\alpha<1$. The first case, with $f(t) \equiv 0$, corresponds to the initial value problem found in [5] while the second case includes (1.6), with U = 1, as a special case, $\alpha = 1/2$.

Chapter 2 contains a brief presentation of a mathematical model developed by Burns, Cliff and Herdman [4] to study the elastic motions of a three degree-of-freedom structure, airfoil with a flap, placed into a two-dimensional unsteady flow. We include this presentation as motivation for studying the singular integro-differential equations of neutral type given by (1.11)-(1.13). An approximation scheme, finite elements, is developed in Chapter 3. Numerical examples of interest to this study are given in Chapter 4. We include Matlab codes developed to approximate the solutions as Chapter 5 and provide plots in Chapter 6. The conclusions for this study are given in Chapter 7.

Chapter 2

2. MATHEMATICAL MODEL FOR THE ELASTIC MOTION OF AN AIRFOIL

Burns Cliff and Herdman [4], coupled the evolution equation for the circulation on an airfoil with the rigid body dynamics of the airfoil to obtain a functional differential equation that provides a mathematical model for the composite system. The mathematical model of Burns, Cliff and Herdman provides a means to study the elastic motions of a three degree-of-freedom airfoil, with a trailing edge flap, placed in a two-dimensional flow, also see [1,2,11,13]. The governing functional equation, in reality, has an infinite delay due to effect that the entire past history of the circulation, over (-∞,0], has on the pressure (lift) of the airfoil. For this presentation we consider a finite delay model for the aeroelastic system described above. The model includes a forcing function f, which could be considered as a control, and has the form

$$\frac{d}{dt}\left[Ax(t) + \int_{-r}^{0} A(s)x(t+s)ds \right] = Bx(t) + \int_{-r}^{0} B(s)x(t+s)ds + f(t) \tag{2.1}$$

for t > 0, where

$$x(t) = (h(t), \theta(t), \beta(t), \dot{h}(t), \dot{\theta}(t), \dot{\beta}(t), \Gamma(t), \dot{\Gamma}(t))^{T}$$

The functions h, θ, β denote the plunge, pitch angle, and flap angle respectively. The 8×8 constant matrix A is singular, each entry in the last row is zero, while the 8×8 matrix function A(s) is weakly singular; the component $A_{88}(s)$ has the form

$$A_{88}(S) = \left[\frac{US-2}{Us}\right]^{1/2}$$

where the constant U denotes the undistributed stream velocity.

The function Γ represents the total airfoil circulation. The state of the system includes the past history of $\dot{\Gamma}$ which may be observed over a finite time interval, say [-r,0], and is given as the known initial function φ on [-r,0]. We will make use of the special structure of the system (2.1). In particular, system (2.1), see [16], can be viewed as

$$\begin{aligned} &\frac{d}{dt}\widetilde{D}(z(t), y_t) = \widetilde{L}(z(t), y_t) + f(t) \\ &\frac{d}{dt}Dy_t = \overline{L}x(t) \end{aligned} \tag{2.2}$$

where y_t:[-r,0]→R is defined by $y_t(s) = y(t+s);\ for\ t \geq 0$. In (2.2), z represents the first seven components of the state x, and y represents the last component of x. The corresponding initial conditions are given by

$$z(0) = \eta, \quad y_0 = \varphi \tag{2.3}$$

for some η in R^7 and φ∈ C[-r,0].

The linear operators $\widetilde{D}, D, \widetilde{L}, \overline{L}$ appearing in (2.2) have the following representations for $(\eta, \varphi) \in R^7 \times C[-r,0]$

$$\widetilde{D}(\eta, \varphi) = I\eta + \int_{-r}^{0} A_{12}(s)\varphi(s)ds\,, \tag{2.4}$$

$$\widetilde{L}(\eta, \varphi) = B_{11}\eta + B_{12}\varphi(0) + \int_{-r}^{0} B_{12}(S)\varphi(s)ds\,, \tag{2.5}$$

$$D\varphi = \int_{-r}^{0} k(s)\varphi(s)ds\,, \tag{2.6}$$

$$\overline{L}(\eta) = B_{21}\eta \tag{2.7}$$

where I is a 7×7 identity matrix, B_{11}, B_{12}, B_{21}, $A_{12}(\cdot)$, $B_{12}(\cdot)$ denote non-zero blocks in the matrices A, B, A(·), and B(·) appearing in (2.1). That is,

$$A = \begin{bmatrix} I_{7X7} & 0 \\ 0 & 0 \end{bmatrix} \qquad A(s) = \begin{bmatrix} 0_{7X7} & A_{12}(s) \\ 0 & g(s) \end{bmatrix}$$

$$B = \begin{bmatrix} B_{117x7} & B_{12} \\ B_{21} & 0 \end{bmatrix} \qquad B(s) = \begin{bmatrix} 0_{7X7} & B_{12}(s) \\ 0_{1x7} & 0_{1x1} \end{bmatrix}.$$

The function $A_{12}(\cdot)$, $B_{12}(\cdot)$ are smooth functions and the function k appearing in the operator D is defined by

$$k(s) = \left(\frac{Us-2}{Us}\right)^{1/2} \tag{2.8}$$

for s in [-r,0).

The representations given above allow the system to be viewed as an integrodifferential Volterra equation, the first equation in (2.2), coupled with a singular neutral equation, the second equation in (2.2). Several studies concerning the well-posedness of this model have replaced the weakly singular kernel function k of (2.8) by

$$\hat{k}(s) = (-s)^{\frac{-1}{2}} \tag{2.9}$$

The "true" kernel (2.8) is related to the kernel (2.9). Both kernels are weakly singular, with a singularity of order ½, and

$$k(s) = c(s)(-s)^{\frac{-1}{2}} \tag{2.10}$$

where c belongs to $C^n[-1,0]$ for all non-negative integers n. Here the function $C(s) = [\frac{(2-Us)}{U}]^{\frac{1}{2}}$.

The first component of (2.2), an integrodifferential Volterra equation, can be numerically solved using well-established techniques for Volterra equations. On the other hand, the second component of (2.2), a singular neutral equation, takes more care in constructing numerical approximations. We will devote the next Chapter to the development of a numerical scheme for approximating the solution of a class of singular neutral equations that contains the singular integral component of (2.2) with the right hand side

$$\overline{L}x(t) \quad \textit{replaced by a given function } f(t).$$

In order to study only the singular neutral component of (2.2) we are in some sense uncoupling the two components. That is, we direct our attention to the second component.

In Chapter 3, we set r = 1 and assume with out loss of generality that the constant U = 1.

Chapter 3

3. FINITE ELEMENT APPROXIMATION

The primary focus of this Chapter is to develop a numerical scheme for the approximation of solutions for a class of singular neutral equations. The aeroelastic model in Chapter 2 contains a Volterra integrodifferential equation component and a singular integral equation component. Since there are numerous prescribed schemes that can be successfully employed to approximate Volterra equations, our interests lie in finding an accurate scheme for approximating solutions for the weakly singular integral equations appearing in the aeroelastic model.

We now focus our attention on the equation

$$\frac{d}{dt}\int_{-1}^{0} k(s)x(t+s)ds = f(t), \quad t \geq 0 \tag{3.1}$$

together with the initial condition

$$x_0(s) = \varphi(s), \quad -1 \leq s \leq 0 \tag{3.2}$$

where, $f \in C[0,\infty)$ and $\varphi \in C[-1,0]$ are known functions and the kernel k has the form

$$k(s) = c(s)(-s)^{-\alpha} \quad 0 < \alpha < 1, \quad c \in C^n[-1,0],\ n = 1,2,\cdots. \tag{3.3}$$

First we will approximate the solution on the interval [0,1]. Our numerical scheme can be extended to [0,T] for any T≥0 using the method of steps. For our numerical approximations we select the finite element method using B-splines, β_j, j = 0,1,2,…,N where N is a non-negative integer. We will partition the interval [-1,0] into a uniform mesh; N subintervals (each having length 1/N) using the mesh points

$$\tau_j^N = \frac{-j}{N}, j = 0,1,2,...,N$$

The functions β_j. j = 0,1,2,…,N are defined on[-1,0] as the piecewise linear functions

$$\beta_0^N = \begin{cases} 0 \; on \; [-1, \tau_1^N] \\ Ns+1 \; on \; [\tau_1^N, 0] \end{cases}$$

$$\beta_j^N = \begin{cases} 0 & on\,[-1, \tau_{j+1}^N] \\ N(s-\tau_{j+1}^N) & on\,[\tau_{j+1}^N, \tau_j^N] \\ -N(s-\tau_{j-1}^N) & on\,[\tau_j^N, \tau_{j-1}^N] \\ 0 & on\,[\tau_{j-1}^N, 0] \end{cases} \qquad j = 1,2,\ldots,N-1$$

$$\beta_N^N = \begin{cases} -N(s - \dfrac{-(N-1)}{N}) & on\,[-1, \tau_{N-1}^N] \\ 0 & on\,[\tau_{N-1}^N, 0] \end{cases}$$

The functions β_j^N, j = 0,1,2,…,N are differentiable almost everywhere with derivatives given by

$$\left(\beta_0^N\right)' = \begin{cases} 0 & on\,(-1, \tau_1^N) \\ N & on\,(\tau_1^N, 0) \end{cases}$$

$$\left(\beta_j^N\right)' = \begin{cases} 0 & on\,(-1, \tau_{j+1}^N) \\ N & on\,(\tau_{j+1}^N, \tau_j^N) \\ -N & on\,(\tau_j^N, \tau_{j-1}^N) \\ 0 & on\,(\tau_{N-1}^N, 0) \end{cases} \qquad j = 1,2,\ldots,N-1$$

$$\left(\beta_N^N\right)' = \begin{cases} -N & on\,(-1, \tau_{N-1}^N) \\ 0 & on\,(\tau_{N-1}^N, 0) \end{cases}$$

Our task is to approximate the solution x(t) for t in [0,1]. The solution x appears in the integral equation in the form x(t+s) where the variable s is in [-1,0] and t is greater than or equal to zero. We approximate the solution x in the following fashion:

Find functions $a_j^N(t), j = 0,1,2,\ldots,N$ defined on [0,1], such that the solution x has the approximation

$$x(t+s) \cong w^N(t,s) \equiv \sum_{j=0}^{N} a_j^N(t)\beta_j^N(s), \quad -1 \le s \le 0, 0 \le t \le 1 \qquad (3.4)$$

That is, we want to approximate the solution with a finite sum using the $\beta_j, j = 1,2,\ldots,N$ as the basis functions. The coefficients a_j^N are functions of t, t≥0, while the βj are functions of s, s in [-1,0]. Using the special structure of our Beta functions (the only Beta function that is non-zero at s = 0 is the Beta function having index j = 0) we have the following representation of the solution x on [0,1]

$$x(t) = x(t+0) \cong w^N(t,0) \equiv \sum a_j^N(t)\beta_0^N(0) = a_0^N(t),\ 0 \le t \le 1$$

That is, we will have our approximation for x(t) if we can find $a_0^N(t)$.

Due to the special nature of our solution in the integral equation (3.1) and our representation (3.4) we begin our task by noting that

$$\frac{\partial}{\partial t}x(t+s) = \frac{\partial}{\partial s}x(t+s)$$

or for our approximation

$$\frac{\partial}{\partial t}w^N(t,s) = \sum_{j=0}^{N}\dot{a}_j^N(t)\beta_j^N(s) = \sum_{j=0}^{N}a_j^N(t)\left(\beta_j^N\right)'(s) = \frac{\partial}{\partial s}w^N(t,s) \qquad (3.5)$$

where $\dot{a}_j$ is the derivative with respect to time t and ${\beta_j^N}'$ is the derivative with respect to s. In order to construct N+1 equations for the unknown functions $a_j(t)$ we use the weak form of the solution. That is, we use each of the β_j, j = 0,1,…, N, as a test function and take the inner product of both sides of (3.5) with β_j, j = 0,1,…,N to obtain N+1 equations. We need an inner product defined on the linear space of continuous functions C[-1,0]. Since C[-1,0] is a subspace of $L_2[0,1]$ we will use the $L_2[-1,0]$ inner product defined by

$$\langle f,g\rangle = \int_{-1}^{0} f(s)g(s)ds\,.$$

We note that this inner product can be extended to include functions that are piecewise constant and have a finite number of discontinuities on [-1,0].

Prior to constructing the desired N+1 equations we point out that the following properties hold for the inner products of β_j and β_j'.

$$\left\langle \beta_0^N,\beta_0^N\right\rangle = \int_{-1}^{0}\beta_0^N(s)\beta_0^N(s)ds = \int_{-\frac{1}{N}}^{0}(Ns+1)^2ds = \left.\frac{(Ns+1)^3}{3N}\right|_{s=-\frac{1}{N}}^{s=0} = \frac{1}{3N}$$

$$\left\langle \beta_j^N,\beta_j^N\right\rangle = \int_{\frac{-(j+1)}{N}}^{\frac{-j}{N}}(Ns+j+1)^2ds + \int_{\frac{-j}{N}}^{\frac{-(j-1)}{N}}(-Ns-j+1)^2ds = \frac{2}{3N},$$

$$for\ j = 1,2,\ldots,N-1$$

$$\left\langle \beta_N^N,\beta_N^N\right\rangle = \int_{-1}^{\frac{-(N-1)}{N}}(Ns-N+1)^2ds = \frac{1}{3N}$$

$$\left\langle \beta_j^N, \beta_{j-1}^N \right\rangle = \left\langle \beta_{j-1}^N, \beta_j^N \right\rangle = \int_{\frac{-j}{N}}^{\frac{-(j-1)}{N}} (-Ns - j + 1)(Ns + j)ds = \frac{1}{6N},$$
$$for\ \ j = 1,2,...N$$

$$\left\langle \beta_j^N, \beta_k^N \right\rangle = \left\langle \beta_k^N, \beta_j^N \right\rangle = 0 \quad for \left| j - k \right| \geq 2;\ j,k = 1,2,...,N$$

$$\left\langle \beta_0^N, \beta_0^{N\prime} \right\rangle = \left\langle \beta_0^{N\prime}, \beta_0^N \right\rangle = \int_{-\frac{1}{N}}^{0} N(Ns + 1)ds = \frac{1}{2}$$

$$\left\langle \beta_j^N, \beta_j^{N\prime} \right\rangle = \left\langle \beta_j^{N\prime}, \beta_j^N \right\rangle = \int_{\frac{-(j+1)}{N}}^{\frac{-j}{N}} N(Ns + j + 1)ds + \int_{\frac{-j}{N}}^{\frac{-(j+1)}{N}} -N(-Ns - j + 1)ds = 0,$$
$$for\ \ j = 1,2,..., N-1$$

$$\left\langle \beta_N^N, \beta_N^{N\prime} \right\rangle = \left\langle \beta_N^{N\prime}, \beta_N^N \right\rangle = \int_{-1}^{\frac{-(N-1)}{N}} -N(-Ns - N + 1)ds = -\frac{1}{2}$$

$$\left\langle \beta_j^{N\prime}, \beta_{j-1}^N \right\rangle = \left\langle \beta_{j-1}^N, \beta_j^{N\prime} \right\rangle = \int_{\frac{-j}{N}}^{\frac{-(j-1)}{N}} -N(Ns + j)ds = -\frac{1}{2}, \quad for\ \ j = 1,2,..., N$$

$$\left\langle \beta_j^{N\prime}, \beta_{j+1}^N \right\rangle = \left\langle \beta_{j+1}^N, \beta_j^{N\prime} \right\rangle = \int_{\frac{-(j+1)}{N}}^{\frac{-j}{N}} N(-Ns - j)ds = \frac{1}{2}, \quad for\ \ j = 1,2,..., N-1$$

$$\left\langle \beta_j^N, \beta_k^{N\prime} \right\rangle = \left\langle \beta_k^{N\prime}, \beta_j^N \right\rangle = 0 \quad for \left| j - k \right| \geq 2;\ j,k = 0,2,3,..., N$$

Now back to constructing the N+1 equations for $a_j(t)$. The identity (3.5) implies that for each k = 0,1,2,…,N, we have

$$\left\langle \dot{a}_j^N(t)\beta_j^N, \beta_k^N \right\rangle = \left\langle a_j^N(t)\beta_j^{N\prime}, \beta_k^N \right\rangle$$

which is equivalent to

$$\dot{a}_j^N(t)\left\langle \beta_j^N, \beta_k^N \right\rangle = \sum_{j=1}^{N} a_j^N(t)\left\langle \beta_j^{N\prime}, \beta_k^N \right\rangle.$$

Using the properties of inner products for k = 0, we have the following

$$\sum_{j=0}^{N} \dot{a}_j^N(t)\left\langle \beta_j^N, \beta_0^N \right\rangle = \dot{a}_0^N(t)\left\langle \beta_0^N, \beta_0^N \right\rangle + \dot{a}_1^N(t)\left\langle \beta_1^N, \beta_0^N \right\rangle + 0 = \sum_{j=0}^{N} a_j^N(t)\left\langle {\beta_j^N}', \beta_0^N \right\rangle$$
$$= a_0^N(t)\left\langle {\beta_0^N}', \beta_0^N \right\rangle + a_1^N(t)\left\langle {\beta_1^N}', \beta_0^N \right\rangle + 0$$

while for k = 1,2, …, N-1 we have

$$\sum_{j=0}^{N} \dot{a}_j^N(t)\left\langle \beta_j^N, \beta_0^N \right\rangle = 0 + \dot{a}_{k-1}^N(t)\left\langle \beta_{k-1}^N, \beta_k^N \right\rangle + \dot{a}_k^N(t)\left\langle \beta_k^N, \beta_k^N \right\rangle + \dot{a}_{k+1}^N(t)\left\langle \beta_{k+1}^N, \beta_k^N \right\rangle + 0$$

$$= a_{k-1}^N(t)\left\langle {\beta_{k-1}^N}', \beta_k^N \right\rangle + a_{kk}^N(t)\left\langle {\beta_k^N}', \beta_k^N \right\rangle + a_{k+1}^N(t)\left\langle {\beta_{k+1}^N}', \beta_k^N \right\rangle$$

$$= \sum_{j=0}^{N} a_j^N(t)\left\langle {\beta_j^N}', \beta_k^N \right\rangle$$

and for k = N we have

$$\sum_{j=0}^{N} \dot{a}_j^N(t)\left\langle \beta_j^n, \beta_N^N \right\rangle = 0 + \dot{a}_{N-1}^N(t)\left\langle \beta_{N-1}^N, \beta_N^N \right\rangle + \dot{a}_N^N(t)\left\langle \beta_N^N, \beta_N^N \right\rangle = \sum_{j=0}^{N} a_j^N(t)\left\langle {\beta_j^n}', \beta_N^N \right\rangle$$
$$= 0 + a_{N-1}^N(t)\left\langle {\beta_{N-1}^N}', \beta_N^N \right\rangle + a_N^N(t)\left\langle {\beta_N^N}', \beta_N^N \right\rangle.$$

We will replace the inner products with the values that we calculated earlier and write the above N+1 equations as the matrix differential equation

$$\frac{1}{6N}\begin{bmatrix} 2 & 1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 1 & 4 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 4 & 1 & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 4 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 4 & 1 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 & 2 \end{bmatrix}\begin{bmatrix} \dot{a}_0^N \\ \dot{a}_1^N \\ \dot{a}_2^N \\ \dot{a}_3^N \\ \vdots \\ \dot{a}_{N-2}^N \\ \dot{a}_{N-1}^N \\ \dot{a}_N^N \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 1 & -1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 1 & -1 & -1 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 & -1 \end{bmatrix}\begin{bmatrix} a_0^n \\ a_1^n \\ a_2^n \\ a_3^n \\ \vdots \\ a_{N-2}^n \\ a_{N-1}^n \\ a_N^n \end{bmatrix}.$$

At this point we have only used the fact that the partial of our solution with respect to t is equal to the partial of our solution with respect to s; we have not considered what conditions $a_j^N(t)$ must satisfy for x(t+s) (really our approximations) to be a solution of the integral equation. We now turn our attention to the integral equation. We have

$$\frac{d}{dt}\int_{-1}^{0} k(s)x(t+s)ds = \int_{-1}^{0} k(s)\frac{d}{dt}x(t+s)ds = \int_{-1}^{0} k(s)\frac{d}{ds}x(t+s)ds$$
$$\cong \int_{-1}^{0} k(s)\frac{d}{ds}\sum_{j=0}^{N} a_j^N(t)\beta_j^N(s)ds = \int_{-1}^{0} k(s)\sum_{j=0}^{N} a_j^N(t)\beta_j^{N\prime}(s)ds$$
$$= \sum_{j=0}^{N} a_j^N(t)\int_{-1}^{0} k(s)\beta_j^{N\prime}(s)ds \cong f(t).$$

In order to calculate the necessary integrals we need to consider particular kernels k(s). The derivatives of the beta functions are piecewise constant, N, -N or 0 on [-1,0]. For convenience of the presentation we will introduce the notation

$$J_j^N = \int_{-1}^{0} k(s)\beta_j^{N\prime}(s)ds.$$

For the case c(s) ≡ 1 and α = 1/2 we have

$$J_0^N = \int_{-1}^{0}(-s)^{-1/2}\beta_0^{N\prime}(s)ds = \int_{\frac{-1}{N}}^{0}(-s)^{-1/2}N ds = 2\sqrt{N}$$
$$J_j^N = \int_{-1}^{0}(-s)^{-1/2}\beta_j^{N\prime}(s)ds = \int_{\frac{-(j+1)}{N}}^{\frac{-j}{N}}(-s)^{-1/2}N ds + \int_{\frac{-j}{N}}^{\frac{-(j-1)}{N}}(-s)^{-1/2}(-N)ds$$
$$= 2\sqrt{N}\left(\sqrt{j+1} - 2\sqrt{j} + \sqrt{j-1}\right) \quad for\, j = 1,2,\cdots,N-1$$
$$J_N^N = \int_{-1}^{0}(-s)^{-1/2}\beta_N^{N\prime}(s)ds = \int_{-1}^{\frac{-(N-1)}{N}}(-s)^{-1/2}(-N)ds = 2\sqrt{N}\left(\sqrt{N-1} - \sqrt{N}\right).$$

For the case α = 1/2 and c(s) = $(2\text{-}s)^{1/2}$, that is k(s) = $[(s\text{-}2)/s]^{1/2}$, the int(c,s) command in Matlab yields

$$K(s) \equiv \int k(s)ds = \int [\frac{s-2}{s}]^{1/2} ds = [\frac{s-2}{s}]^{1/2} s - \ln\left|-1 + s + [\frac{s-2}{s}]^{1/2} s\right|, \quad s < 0.$$

We note that

$$K(0) \equiv K(0^-) = \lim_{s\to 0^-} K(s) = 0$$

thus we can use K to calculate all necessary integrals for this case. It follows that

$$J_0^N = N[K(0^-) - K(\frac{-1}{N})] = -NK(\frac{-1}{N})$$

$$J_j^N = N[K(\frac{-j}{N}) - K(\frac{-(j+1)}{N})] - N[K(\frac{-(j-1)}{N}) - K(\frac{-j}{N})], \quad j = 1,2,\cdots,N-1$$

$$J_N^N = -N[K(\frac{N-1}{N}) - K(-1)]$$

where the function K is defined above.

In the case that $0 \le \alpha \le 1$, $c(s) = [2-s]^{\alpha}$ then $k(s) = [(s-2)/s]^{\alpha}$. The anti-derivative of k can be obtained using Mathematica. The anti-derivative K(s) is given in the following form

$$K(s) \equiv \int k(s) = \int [\frac{s-2}{s}]^{\alpha} ds = -[\frac{1}{\alpha-1}][\frac{2-s}{2}]^{-\alpha}[\frac{s-2}{s}]^{\alpha} sF[1-\alpha,-\alpha,2-\alpha,\frac{s}{2}]$$

where the F = Hypergeometric2F1 and has the integral representation

$$F[a,b,c,z] = \int_0^1 t^{b-1}(1-t)^{c-b-1}(1-tz)^{-a}\,dt$$

and the series representation

$$F[a,b,c,z] = \sum_{k=0}^{\infty} \frac{(a)_k (b)_k}{(c)_k}\left(\frac{z^k}{k!}\right)$$

In particular we have

$$F[1-\alpha,-\alpha,2-\alpha,\frac{s}{2}] = \int_0^1 t^{-\alpha-1}(1-t)^1(1-t\frac{s}{2})^{\alpha-1}\,dt$$

and

$$F[1-\alpha,-\alpha,2-\alpha,\frac{s}{2}] = \sum_{k=0}^{\infty} \frac{(1-\alpha)_k(-\alpha)_k}{(2-\alpha)_k}\frac{(s/2)^k}{k!}.$$

As in the special case of $\alpha = 1/2$ considered above we have that

$$K(0) \equiv K(0^-) = \lim_{s\to 0^-} K(s) = 0$$

which allows us to use the anti-derivative K(s) to evaluate the necessary integrals. The J's are as given above where, of course, we use the appropriate anti-derivative K(s).

Consequently we have

$$J_0^N a_0^N(t) + J_1^N a_1^N(t) + J_2^N a_2^N(t) + \cdots + J_N^N a_N^N(t) \cong f(t).$$

Our next step is to couple this identity (in the approximation sense) with the N+1 $\times$ N+1 ODE system that we gave earlier. We note that the function a_0^N only appears in the first two equations of the ODE system, we could replace the first two equations of the system with a linear combination which would not have a $\dot{a}_0^N$ term and we can solve for a_0^N from the above identity in terms of a_j^N, J_j^N for j = 0,2,…, N, and f(t). Indeed these are the steps that we take to couple the above identity to the ODE system. The first two equations of the ODE system are given by

$$\frac{1}{3N}\dot{a}_0^N(t) + \frac{1}{6N}\dot{a}_1^N(t) = \frac{1}{2}a_0^N(t) - \frac{1}{2}a_1^N(t)$$
$$\frac{1}{6N}\dot{a}_0^N(t) + \frac{2}{3N}\dot{a}_1^N(t) + \frac{1}{6N}\dot{a}_2^N(t) = \frac{1}{2}a_0^N(t) - \frac{1}{2}a_2^N(t)$$

or equivalently by (multiply the second equation by -2)

$$\frac{1}{3N}\dot{a}_0^N(t) + \frac{1}{6N}\dot{a}_1^N(t) = \frac{1}{2}a_0^N(t) - \frac{1}{2}a_1^N(t)$$
$$\frac{-1}{3N}\dot{a}_0^N(t) + \frac{-4}{3N}\dot{a}_1^N(t) + \frac{-1}{3N}\dot{a}_2^N(t) = -a_0^N(t) + a_2^N(t)$$

which yields (when the two equations are added together)

$$\frac{-7}{6N}\dot{a}_1^N(t) + \frac{-1}{3N}\dot{a}_2^N(t) = -\frac{1}{2}a_0^N(t) - \frac{1}{2}a_1^N(t) + a_2^N(t).$$

From above we have the following representation for $a_0^N(t)$,

.

$$a_0^N(t) = -(J_0^N)^{-1}J_1^N a_1^N(t) - (J_0^N)^{-1}J_2^N a_2^N(t) - \cdots - (J_0^N)^{-1}J_N^N a_N^N(t) + (J_0^N)^{-1}f(t). \qquad (3.6)$$

Consequently we can eliminate the $a_0^N(t)$ term from our linear combination of the first two equations of the ODE system, in particular we have

$$\frac{-7}{6N}\dot{a}_1^N(t) + \frac{-1}{3N}\dot{a}_2^N(t) = -\frac{1}{2}a_0^N(t) - \frac{1}{2}a_1^N(t) + a_2^N(t)$$
$$= (\frac{1}{2}(J_0^N)^{-1}J_1^N - \frac{1}{2})a_1^N(t) + (\frac{1}{2}(J_0^N)^{-1}J_2^N + 1)a_2^N(t) + \frac{1}{2}(J_0^N)^{-1}J_3^N a_3^N(t) + \cdots + \frac{1}{2}(J_0^N)^{-1}J_N^N a_N^N(t)$$
$$-\frac{1}{2}(J_0^N)^{-1}f(t).$$

We now rewrite the N+1 × N+1 ODE system with the N× N system we get by replacing the first two equations with the one equation (eliminating the $a_0^N(t)$ term) given above. The resulting ODE system is given by

$$\frac{1}{6N}\begin{bmatrix} -7 & -2 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 1 & 4 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 4 & 1 & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 4 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 4 & 1 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 & 2 \end{bmatrix}\begin{bmatrix} \dot{a}_1^N \\ \dot{a}_2^N \\ \dot{a}_3^N \\ \dot{a}_4^N \\ \vdots \\ \dot{a}_{N-2}^N \\ \dot{a}_{N-1}^N \\ \dot{a}_N^N \end{bmatrix}$$

$$=\frac{1}{2}\begin{bmatrix} (J_0^N)^{-1}J_1^N-1 & (J_0^N)^{-1}J_2^N+2 & (J_0^N)^{-1}J_3^N & \cdots & (J_0^N)^{-1}J_{N-2}^N & (J_0^N)^{-1}J_{N-1}^N & (J_0^N)^{-1}J_N^N \\ 1 & 0 & -1 & \cdots & 0 & 0 & 0 \\ \bullet & \bullet & \bullet & \ddots & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \cdots & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \cdots & \bullet & \bullet & \bullet \\ 0 & 0 & 0 & \cdots & 1 & 0 & -1 \\ 0 & 0 & 0 & \cdots & & 1 & -1 \end{bmatrix}\begin{bmatrix} a_1^n(t) \\ a_2^n(t) \\ \bullet \\ \bullet \\ \bullet \\ a_{N-1}^n(t) \\ a_N^n(t) \end{bmatrix}$$

$$-\frac{1}{2}\begin{bmatrix} (J_0^N)^{-1}f(t) \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

The above system can be written in matrix form as

$$C\dot{a}(t) = Da(t) + F(t)$$

where the N×N matrices C and D and the N×1 functions a and F are defined in an obvious way. The special structure of the C matrix guarantees the existence of C^{-1} thus we can write the system in standard form

$$\dot{a}(t) = C^{-1}Da(t) + C^{-1}F(t)\,. \tag{3.7}$$

It is to be noted that our unknown function a(t) here is given by

$$a(t) = \left| \begin{matrix} a_1^N(t) \\ a_2^N(t) \\ \vdots \\ a_N^N(t) \end{matrix} \right|$$

that is $a_0^N(t)$ (which we recall gives us the solution for t greater than or equal to zero) does not appear in a(t). However we can solve for $a_0^N(t)$ once we know a(t), since f(t) is known and we can use (3.6).

We only lack initial conditions a(0) in order to be assured of a unique solution a(t) for the ODE system, hence an approximation for the unique solution x for our weakly singular integral equation. In order to construct initial conditions we again consider

$$x(t+s) \cong \sum_{j=0}^{N} a_j^N(t)\beta_j^N(s) \tag{3.8}$$

where $t \geq 0$ *and* $-1 \leq s \leq 0$. Using our initial condition (2.2) together with setting t = 0 in (3.8) we have

$$x(s) = x(0+s) \cong \sum_{j=0}^{N} a_j^N(0)\beta_j^N(s) \overset{set}{=} \varphi(s),\ for\ -1 \leq s \leq 0\,. \tag{3.9}$$

We let s = -i/N for i = 0,1,2,…,N and use the properties of our β - functions, we obtain the following

$$x(0) = x(0+0) \cong \sum_{j=0}^{N} a_j^N(0)\beta_j^N(0) = a_0^N(0)\beta_0^N(0) = a_0^N(0) \overset{set}{=} \varphi(0)$$

and for i = 1,2,… , N

$$x(\frac{-i}{N}) = x(0+\frac{-i}{N}) \cong \sum_{j=0}^{N} a_j^N(0)\beta_j^N(\frac{-i}{N}) = a_i^N(0) \overset{set}{=} \varphi(\frac{-i}{N})\,.$$

This gives the initial condition for the first order system (3.7).

The procedure that we follow to obtain the approximation for our solution x(t) for (3.1) - (3.3) on [0,1] is as follows:

- Use Matlab (ode45) to find the approximate solution for $a(t) = [\,a_1(t) \ldots a_N(t)\,]$
- Find $a_0(t)$ for t in [0,1] using (3.6)
- x(t) = a_0(t) for t in [0,1]

In order to solve (3.1)-(3.3) on [n, n+1], n a positive integer, the ODE system is the same as presented above. However, we must set out initial conditions for the ODE system. We note that we shall use the method of steps, that is, we solve the system on [n-1,n] and then consider [n,n+1]. We will need a(n), the initial condition for [n, n+1], which can be obtained from

$$x(n + \frac{-i}{N}) \cong \sum_{j=0}^{N} a_j^N(n)\beta_j^N(\frac{-i}{N}) = a_i^N(n)$$

since we already have x(t) for t in [n-1,n].

Several examples are presented in the next Chapter, which are special cases of the kernel functions that we discussed above.

Chapter 4

4. NUMERICAL EXAMPLES

We consider the following examples to validate our numerical scheme. We start with the special case c(s) = 1 and α = 1/2 and only consider the interval [0,1]. This special case serves as a natural starting point since we know the true solution corresponding to several different initial functions φ and forcing functions f. In other cases, we do not know the true solution thus we are not able to compare it to our approximation.

For our examples we denote the forcing function by f, the initial function by φ and the number of element use by N. The gamma function is denoted by Γ(t) and is defined by

$$\Gamma(a) = \int_0^\infty t^{a-1} e^{-a} dt \qquad a \geq 0 .$$

4.1 Special Case α=1/2, c(s)=1: k(s)=(-s) $^{-1/2}$

4.1.1 Let f(t)=1, φ(s)=0 and N=16

For this case we know that the true solution x has the representation $x(t) = (2/\pi)t^{1/2}$ for $0 \leq t \leq 1$. Both the true solution (red) and the approximation (green) are given in Figure 6.1.1. We have an excellent approximation except close to zero. Of course this can be expected due to the weak singularity of k(s) at s = 0.

4.1.2 Let f(t)=1, φ(s)=0 and N=64.

Again we know that the true solution x has the representation $x(t)=(2/\pi)\, t^{1/2}$ for $0 \leq t \leq 1$. We see an improvement in the approximation close to zero as we increase the number of elements, N = 16 in Example 4.1.1, to N = 64 in this example. See Figure 6.1.2

4.1.3 Let $f(t)=3t^2$, φ(s)=0 and N=16

The true solution (red) is given by $[\Gamma(4)/\,\Gamma(.5)\,\Gamma(.5+3)]t^{(3-.5)}$. As can be seen in Figure 6.1.3 we have an excellent match between the approximate and true solutions.

4.1.4 Let $f(t)=10t^9$, φ(s)=0 and N=16

The true solution is given by $[\Gamma(11)/\,\Gamma(.5)\,\Gamma(.5+10)]t^{(10-.5)}$. Figure 6.1.4 shows an excellent match between the approximate (blue) and true (red) solutions.

4.1.5 Let f(t)=sin(t), $\varphi(s)=e^s$ and N=16

In this case we do not know the true solution. We note the numerical discontinuity in the graph, Figure 6.1.5, at t = 0.

4.1.6 Let $f(t)=\sin(t)$, $\varphi(s)=e^s$ and $N=64$

We increase the size of N from N = 16, Example 4.1.5, to N = 64. Observe that the discontinuity at t = 0, Figure 6.1.6, is less in magnitude when compared to Figure 6.1.5.

4.1.7 Let $f(t)=\sin(t)$, $\varphi(s)=e^s$ and $N=128$

Again we use the forcing function and initial function of Example 4.1.5. Here we let N = 128. Figure 6.1.7, compared to Figures 4.1.5 – 4.1.6, shows numerical convergence to the true solution at t = 0.

4.2 Special Case $\alpha=1/2, c(s)=(2-s)^{1/2}$: $k(s)=[(s-2)/s]^{-1/2}$

The kernel still has the same order of singularity as the kernel used in Chapter 4.1. Although the corresponding solution will not be the same we should see the same characteristics of the corresponding solution on [0,1].

4.2.1 Let $f(t)=1$, $\varphi(s)=0$ and $N=16$.

The approximate solution is given on [0,1], Figure 6.2.1. Compare this solution with the approximate solution in Figure 6.1.1.

4.2.2 Let $f(t)=1$, $\varphi(s)=0$ and $N=64$

The approximate solution is given on [0,1], Figure 6.2.2. Compare this solution with the approximate solution in Figure 6.1.2.

4.2.3 Let $f(t)=3t^2$, $\varphi(s)=0$ and $N=16$.

The approximate solution is given on [0,1], Figure 6.2.3. Compare this solution with the approximate solution in Figure 6.1.3.

4.2.4 Let $f(t)=10t^9$, $\varphi(s)=0$ and $N=16$.

The approximate solution is given on [0,1], Figure 6.2.4. Compare this solution with the approximate solution in Figure 6.1.4.

4.2.5 Let $f(t)=\sin(t)$, $\varphi(s)=e^s$ and $N=16$.

The approximate solution is given on [0,1], Figure 6.2.5. Compare this solution with the approximate solution in Figure 6.1.5.

4.2.6 Let $f(t)=\sin(t)$, $\varphi(s)=e^s$ and $N=16$.

The approximate solution is given on [0,5], Figure 6.2.6. We note the “corner” at t = 1.

4.3 Special Case $0<\alpha<1$, $c(s)=(2-s)^{-\alpha}$: $k(s)=[(s-2)/s]^{\alpha}$

4.3.1 Let $\alpha=.4$, $f(t)=3t^2$, $\varphi(s)=0$ and $N=16$.

The approximate solution is given on [0,5], Figure 6.3.1. Compare this solution on [0,1] with the approximate solution in Figure 6.2.3, $\alpha = .5$.

4.3.2 Let α=.6, f(t)=3t^2, φ(s)=0 and N=16.
The approximate solution is given on [0,5], Figure 6.3.2. Compare this solution on [0,1] with the approximate solution in Figure 6.2.3, α = .5.

4.3.3 Let α=.5, f(t)=cos(t), $\varphi(s)=(-s)^{.4}$ and N=32.
The approximate solution is given on [0,5], Figure 6.3.3.

4.3.4 Let α=.4, f(t)=cos(t), $\varphi(s)=(-s)^{.4}$ and N=32.
The approximate solution in given on [0,5], Figure 6.3.4.

4.3.5 Let α=.6, f(t)=cos(t), $\varphi(s)=(-s)^{.4}$ and N=32.
The approximate solution in given on [0,5], Figure 6.3.5.

REMARK: *For the last three examples, we let* $\varphi \in L_2(-1,0)$.. *The associated neutral integral equation (3.1) – (3.3) is well posed on* L_2 *only if* $\alpha < \frac{1}{2}$, *see [5]. However, our approximations for* $\alpha = .5$ *and* $\alpha = .6$ *appear to provide a good approximation.*

Chapter 5

5. MATLAB Programs

5.1 Single Integral Equation Program

```
%Run the numerical scheme for the singular integral equation
%d/dt[int over -1 to 0 of [(s-2)/(s)]^(alpha) *
x(t+s)ds]=f(t)=diefor(t)
%with initial function phi=dieif on [-1,0].
% DE: Ca'(t)=Da(t)+F(t):  with Integral Boundary Condition: gives a1,
....aN
% in a'(t)=Aa(t)+(-1/(2*J0)*inv(A)*F with F(1)=f(t)
%hat functions (linear splines) for the finite element approximation
for the finite
%deminsional approximation (ODE system)
% set up C D A, initial conditions TSPAN mesh for t interval: call
dsiecoefm.m
% dsiecoefm calls for number of elements N (really have N+1 elements
considering
%beta zero
% dsiecoefm calls for right hand endpoint T and the number of points M
(really M+1)
% that we ask matlab to evaluate the solution in [0, T]:  Tspan
global N M FT Y0 TSPAN
dsiecoefm

% solve the ODE : dcierhs.m  defines the right hand side

[T,Y]=ode45('dsierhs',TSPAN,Y0);

% need to create the vector JJ=[J(2) J(3) ... J(N+1)] to use scalar
products
% to obtain the solution via vector operations (as opposed to a for
loop)
for i=1:N
   JJ(i)=J(i+1);
end
% Construct the solution from the output of ODE45

 for j=1:1:M+1
    zt(j,1)=-1*(J(1)^(-1))*(Y(j,1:N)*JJ' - diefor(T(j)));
   end

%plot the approximate solution on [0,1]

plot(T,zt(:,1),'g')
hold on

%xlabel('Figure 3.1. Green: Approx. Red: True x(t)=1 ')
%xlabel('Figure 3.2. Green: Approx. Red: True x(t)=(2/pi)*t^1/2')
%xlabel('Figure 1. Green: Approx. Red: True
%x(t)=[Gamma(a+1)/Gamma(.5)Gamma(.5+a)]*t^(a-.5)
%xlabel('Figure 1. Green: Approx. Red: Initial Function ')
%ylabel('Values for x(t)')
```

```
%title('Elements: N+1=16  Mesh Points: M+1=101 ; phi(t)=1  f(t)=0')
% graph initial function on [-1,0]

IT(1)=-1;
Infun(1)=dieif(IT(1));
for i=1:100
IT(i+1)=-1+i/100;
Infun(i+1)=dieif(IT(i+1));
end
plot(IT,Infun,'r')
hold on

%plot exact solution
% for alpha =1/2 and f(t)=a*t^(a-1) the true solution is
% x(t)=Gamma(a+1)/(Gamma(1-.5)*Gamma(.5+a))*(t)^(.5+a-1);
%disp('Input Value for a in f(t)=a*t^(a-1)')
%a=input(' f(t)=a*t^(a-1)     a= ')
%for i=1:M+1
%Tx(i)=(Gamma(a+1)/(Gamma(1-.5)*Gamma(.5+a)))*(T(i))^(.5+a-1);
%end
%plot(T,Tx,'r')
%hold off
%for i=2:M+1
%Tx(i)=1;
%Tx(1)=0;
%Tx(i)=(2/pi)*(T(i))^(.5);
%Tx(i)=(1/pi)*T(i)^(-.5);
%TT(i)=T(i);
%end
%plot(TT,Tx,'r')

% to extend the interval we need M=N to get the initial conditions
%for the continuation - we could do an interpulation and keep N and m
% independent - but matlab has already done an interpulation - M is
%defined earlier only for graphing purposes change in coef file

for k=1:1:FT

   if k < FT
      for i=1:1:N
         Z0(i)=zt(N+1-i,k);
         Z0=Z0';
      end

   TEspan=k:1/M:k+1;
   [TE,Y]=ode45('dsierhs',TEspan,Z0);
    for j=1:M+1
       zt(j,k+1)=-1*(J(1)^(-1))*(Y(j,1:N)*JJ' - diefor(TE(j)));

    end
    plot(TE,zt(:,k+1),'b')
       hold on
    else
 end
 end
 hold off
```

5.2 Single Integral Equation – Right Hand Side

```
function zpr=dsierhs(t,z)
global C D N M Y0 A CINV J


zpr=zeros(N,1);

zpr(1)=A(1,1:N)*z(1:N,1)+(-1/2)*(J(1)^(-1))*CINV(1,1)*diefor(t);
for i=2:N

zpr(i)=A(i,1:N)*z(1:N,1)+(-1/2)*(J(1)^(-1))*CINV(i,1)*diefor(t);
end
```

5.3 Single Integral Equation – Coefficient Matrix

```
%dsiecoefm.m
% Build the C & D matrices for the Singular integral equation
% Ca'(t)=Da(t) corresponding to the partial differential equation
% d[x(t+s)]/dt=d[x(t+s)]/ds and integral boundary conditions
% Note this is for ai i=1, ... N; does not include a0(t)
% Input size of C and D: number of elements N

global A C D CINV N M TSPAN Y0 J FT

disp('Input Value of N')
N=input('Number of elements N=');

%Construct C and the inverse of C

c1=1/(6*N);
d1=1/2;
I=eye(N);
CC=4.*I+diag(ones(N-1,1),1)+diag(ones(N-1,1),-1);
CC(1,1)=-7;
CC(1,2)=-2;
CC(N,N)=2;
C=c1.*CC;
CINV=inv(C);

%Construct the matrix D

DC(1:N,1:N)=zeros(N)+diag(ones(N-1,1),-1)+diag(-1*ones(N-1,1),1);
DC(N,N-1)=1;
DC(N,N)=-1;

% construct the first row of D
% construct the J(i), i=1,...N+1

J(1)=-N*(((1+2*N)^.5)*(-1/N)-log(abs(-1-1/N+((1+2*N)^.5)*(-1/N))));
spq=-N*[((1-2*N/(-N+1))^.5)*((-N+1)/N)];
spq1=N*[log(abs(-1+((-N+1)/N)+((1-2*N/(-N+1))^.5)*((-N+1)/N)))];
spq2=-N*[(3^.5)*(-1)-log(abs(-2+((3)^.5)*(-1)))];
J(N+1)=spq+spq1-spq2;

p2=((1-2*N/(-2))^.5)*(-2/N);
q2=-1+(-2/N)+p2;
     w2=abs(q2);
      xx2=log(w2);
      yy2=p2-xx2;
      pp2=((1-2*N/(-2+1))^.5)*((-2+1)/N);
      qq2=-1+((-2+1)/N)+pp2;
      ww2=abs(qq2);
      xxx2=log(ww2);
      yyy2=pp2-xxx2;
      UJ=N*(yyy2-yy2);
      J(2)=UJ-J(1);
   for j=3:N
      p=((1-2*N/(-j))^.5)*(-j/N);
      q=-1+(-j/N)+p;
```

```
      w=abs(q);
      xx=log(w);
      yy=(p-xx);
      pp=((1-2*N/(-j+1))^.5)*((-j+1)/N);
      qq=-1+((-j+1)/N)+pp;
      ww=abs(qq);
      xxx=log(ww);
      yyy=(pp-xxx);
      UJ=N*(yyy-yy);

      pl=((1-2*N/(-j+2))^.5)*((-j+2)/N);
      ql=-1+((-j+2)/N)+pl;
      wl=abs(ql);
      xxl=log(wl);
      yyl=pl-xxl;
      LJ=-N*(yyl-yyy);

      J(j)=UJ+LJ;
   end

   for k=1:N
      jinv=J(1)^(-1);
      DC(1,k)=jinv*J(k+1);
   end
   DC(1,1)=DC(1,1)-1;

 DC(1,2)=DC(1,2)+2;

D=d1*DC;
A=CINV*D;

%Now we set the initial conditions for the approximating ODE
% Given initial function dieif we set  a1(0)=dieif(-1/N),
% ... aj(0)=dieif(-j/N),...,aN-1(0)=dieif(-1+1/N), aN(0)=dieif(-1).
%Note we have i=1, ... N; We do have a0(0)=dieif(0)
% dieif is a function file - initial function

for i=1:N
Y0(i)=dieif(-i/N);
end

% We give the mesh for time t on [0,1].
% M=number of mesh points in (0,1]; 0 is also a point thus
% we have M+1 points.

disp('Input Value of M')
M=input('number of mesh points in (0,1] M=')
FT=input('integer for right hand end point FT=')

% need M=N to set initial conditions for the extended interval problem

M=N;
MT=FT*M
TSPAN(1)=0;
for i=2:M+1
TSPAN(i)=(i-1)/M;
end
```

5.4 Single Integral Equation – Initial Function

```
function z=dieif(t);
% initial function for singular integral equation.
z=1;
%z=t;
%z=2;
%z=t^.5;
%z=t^2;
%z=t^3;
%z=0;
%z=3^.5+log(2+(3^.5));
%z=exp(t);
%z=(-t)^-.5;
```

5.5 Single Integral Equation – Forcing Function

```
function z=diefor(t)
% forcing function for singular integral equation f: iefor.m
%z=t;
%z=0;
%z=1;
%z=2*t;
%z=3*t^2;
%z=4*t^3;
%z=10*t^9;
%z=sin(t);
z=cos(t);
%z=3^.5+log(2+(3^.5));
% Create an approximation for the delta function

%if t >= .01
%z=0;
%else
%z=100;
%end
```

5.6 Single Integral Equation – Anti-derivitive General Kernel

```
function z=daalpha(s)
global alpha
% antiderivate for [(s-2)/s]^alpha
format long
alp=alpha;
a=1-alp;
b=-alp;
c=2-alp;
aa(1)=1;
aa(2)=(a*b)/c;
fi=1;
MM=100;
w(1)=1;
for k=3:MM+1
   aa(k)=aa(k-1)*((a+(k-2))*(b+(k-2)))/(c+(k-2));
end
for i=2:MM+1
   w(i)=((s/2)^(i-1))/fi;
   fi=(i)*fi;
end
zz=aa*w';
z=-[(((2-s)/2)^-alp)*(((s-2)/s)^alp)*s*zz]/(alp-1);
```

Chapter 6

6. MATLAB GRAPHS

6.1 Graphs for Special Case α=1/2, c(s)=1: k(s)=(-s) $^{-1/2}$

6.1.1 Sample 1

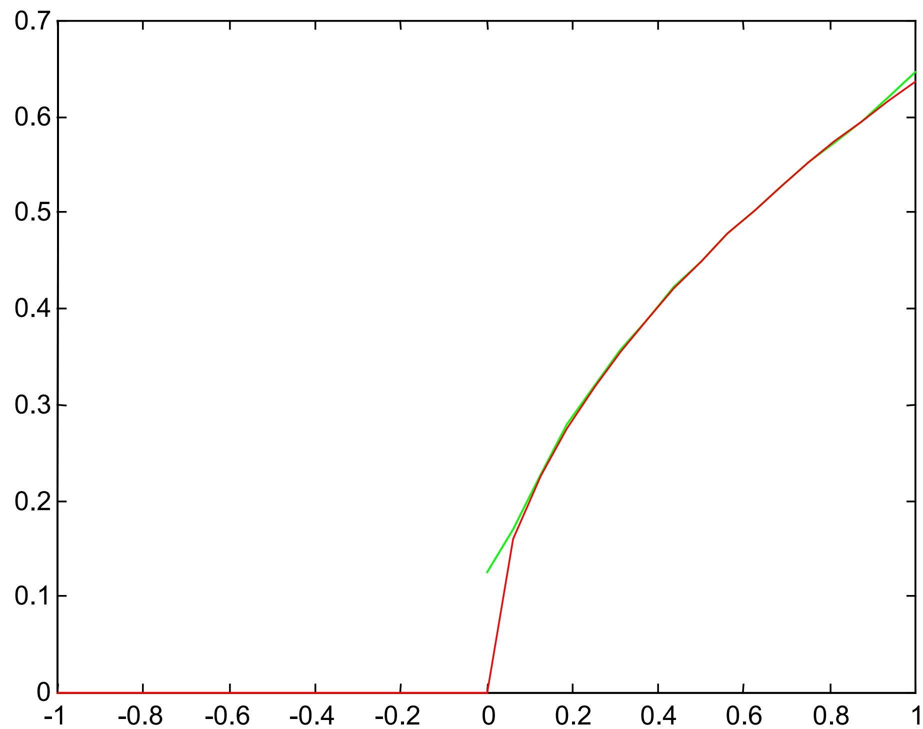

Figure 6.1.1. Initial Function = 0/Forcing Function = 1

Parameters:

Number of Elements N = 16
$\varphi(s)=0$

$f(t)=1$

Green represents the Approximate Solution

Red represents the True Solution: $x(t)=\left(2/\pi\right)t^{\frac{1}{2}}$

6.1.2 Sample 2

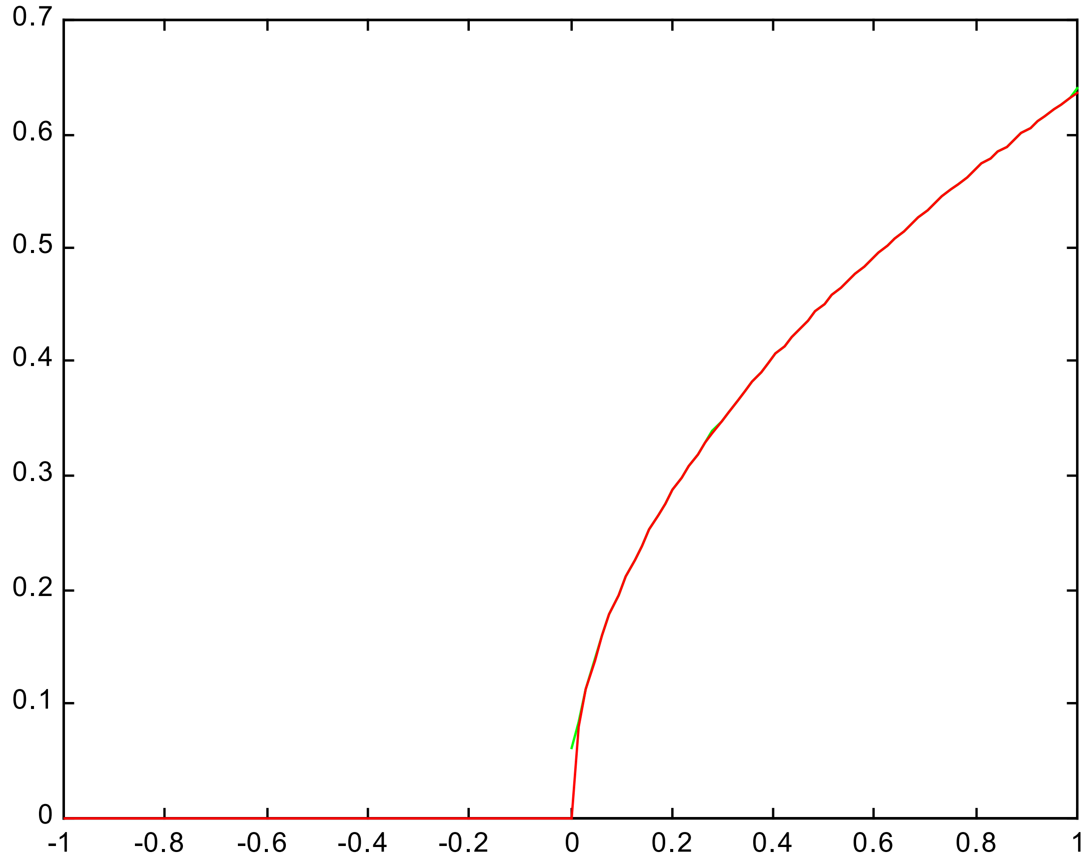

Figure 6.1.2. Initial Function = 0/Forcing Function = 1

Parameters:

Number of Elements N = 64
$\varphi(s) = 0$
$f(t) = 1$

Green represents the Approximate Solution

Red represents the True Solution: $x(t) = \left(2/\pi\right) t^{\frac{1}{2}}$

6.1.3 Sample 3

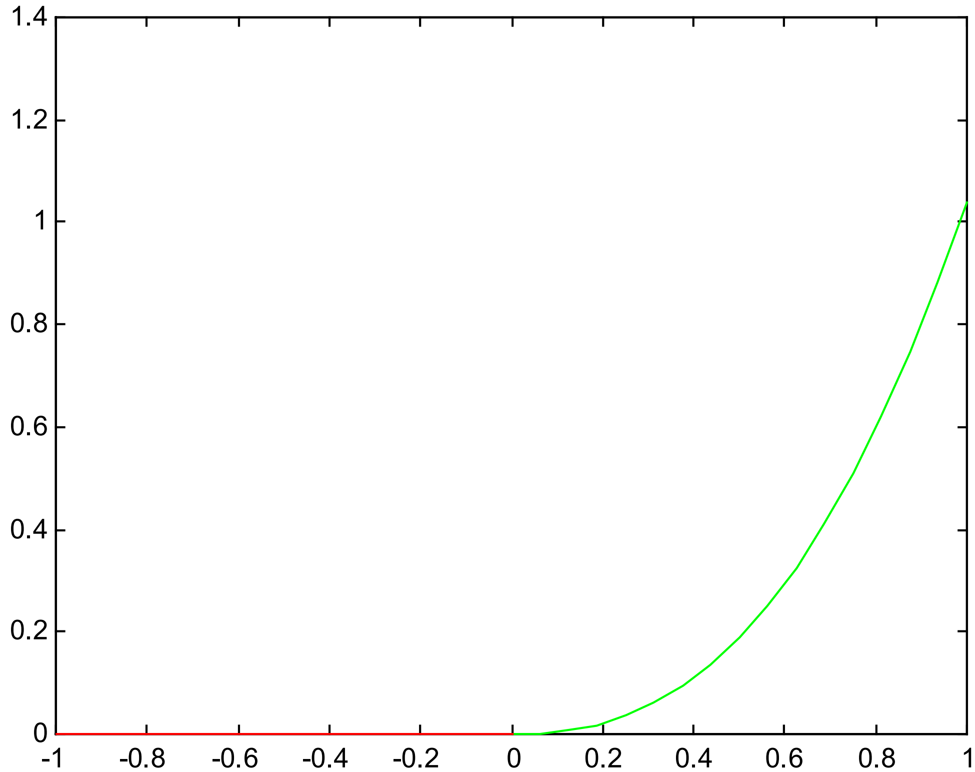

Figure 6.1.3. Initial Function = 0/Forcing Function = 3t²

Parameters:

Number of Elements N = 16
$\varphi(s) = 0$
$f(t) = 3t^2$

Green represents the Approximate Solution
Red represents the True Solution: $x(t) = [\alpha(3+1)/\alpha(.5)\alpha(.5+3)]t^{(3-.5)}$

6.1.4 Sample 4

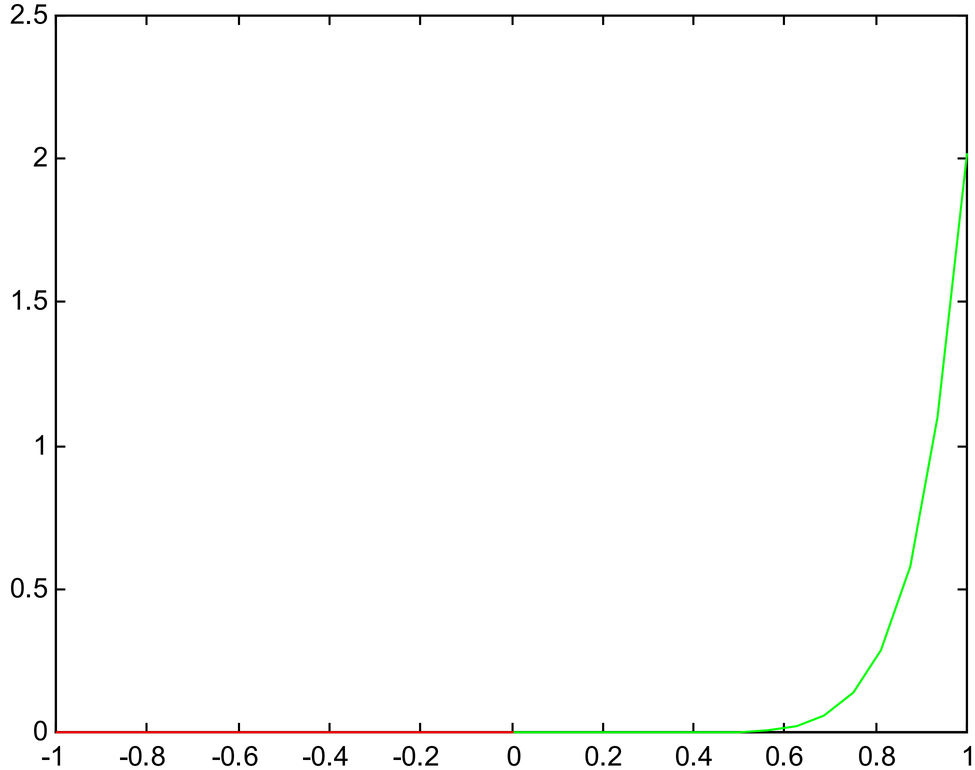

Figure 6.1.4. Initial Function = 0/Forcing Function = $10t^9$

Parameters:

Number of Elements N = 16
$\varphi(s) = 0$
$f(t) = 10t^9$

Green represents the Approximate Solution
Red represents the True Solution: $x(t) = [\alpha(10+1)/\alpha(.5)\alpha(.5+10)]t^{(10-.5)}$

6.1.5 Sample 5

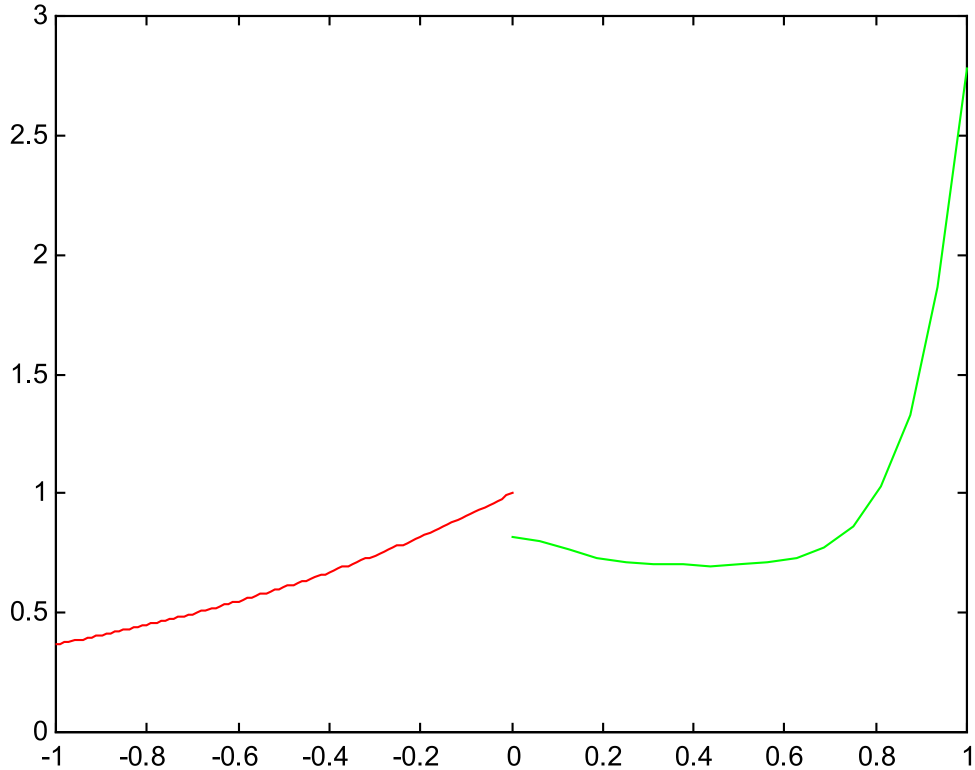

Figure 6.1.5. Initial Function = e^t/Forcing Function = sin(t)

Parameters:

Number of Elements N = 16

$\varphi(s) = e^s$

$f(t) = \sin(t)$

Green represents the Approximate Solution
Red represents the Initial Function

6.1.6 Sample 6

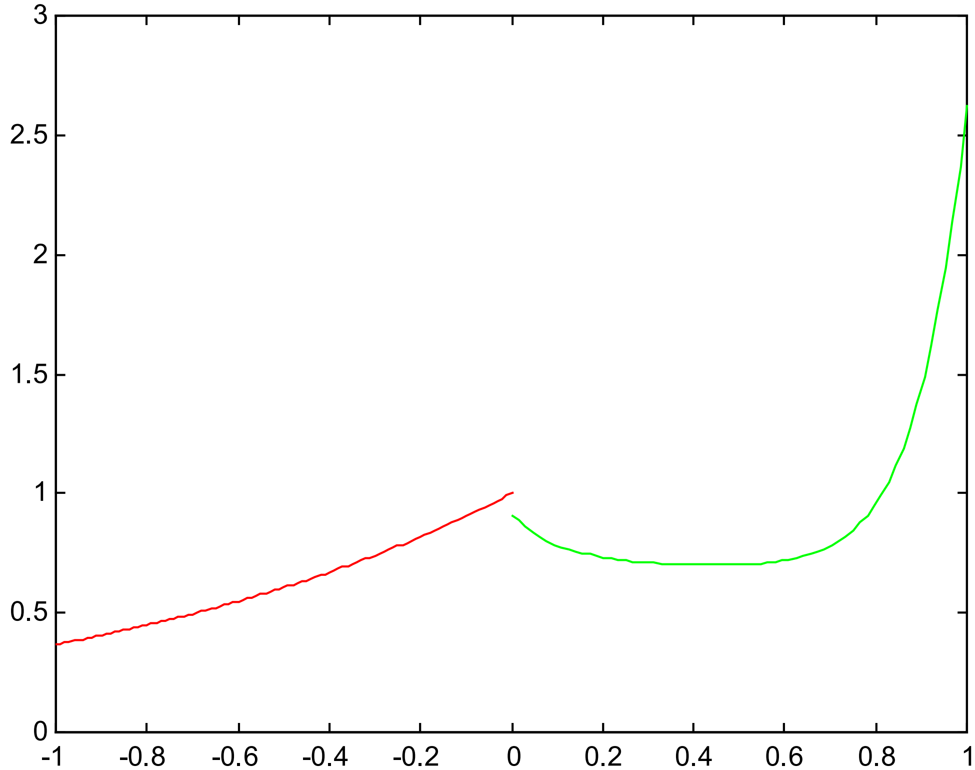

Figure 6.1.6. Initial Function = e^t/Forcing Function = sin(t)

Parameters:

Number of Elements N = 64

$\varphi(s) = e^s$

$f(t) = \sin(t)$

Green represents the Approximate Solution
Red represents the Initial Function

6.1.7 Sample 7

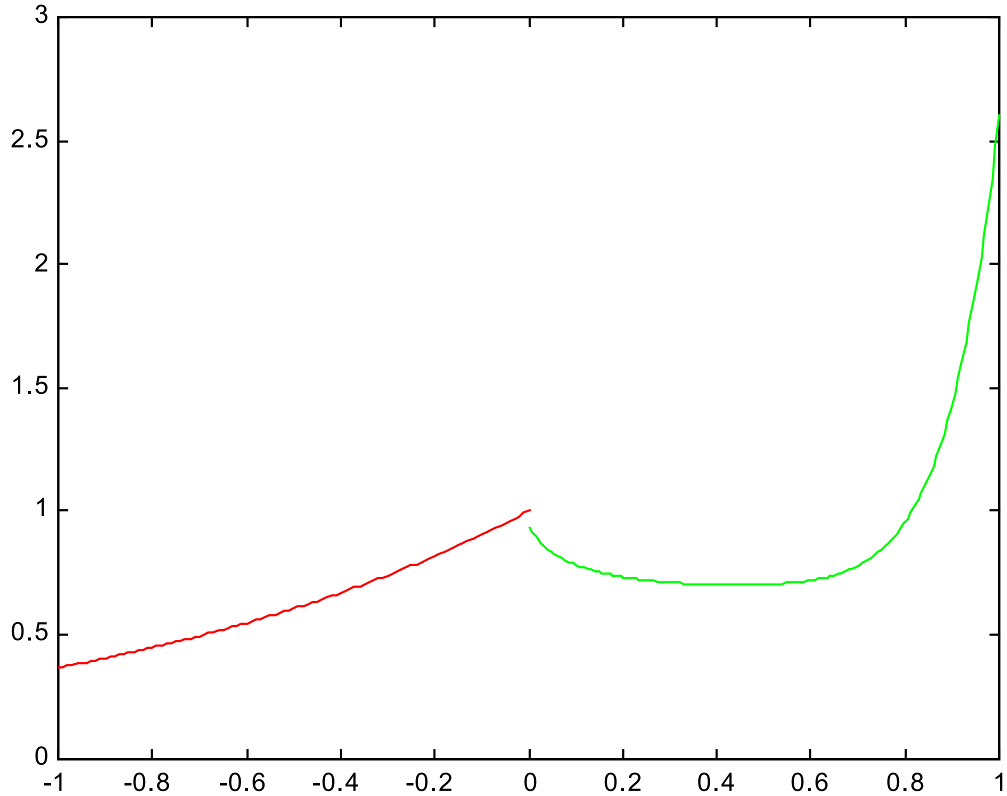

Figure 6.1.7. Initial Function = e^t/Forcing Function = sin(t)

Parameters:

Number of Elements N = 128
$\varphi(s) = e^s$
$f(t) = \sin(t)$

Red represents the Initial Function
Green represents the Approximate Solution on the interval [0,1]

6.2 Graphs for Special Case $\alpha=1/2, c(s)=(2-s)^{1/2}: k(s)=[(s-2)/s]^{-1/2}$

6.2.1 Sample 1

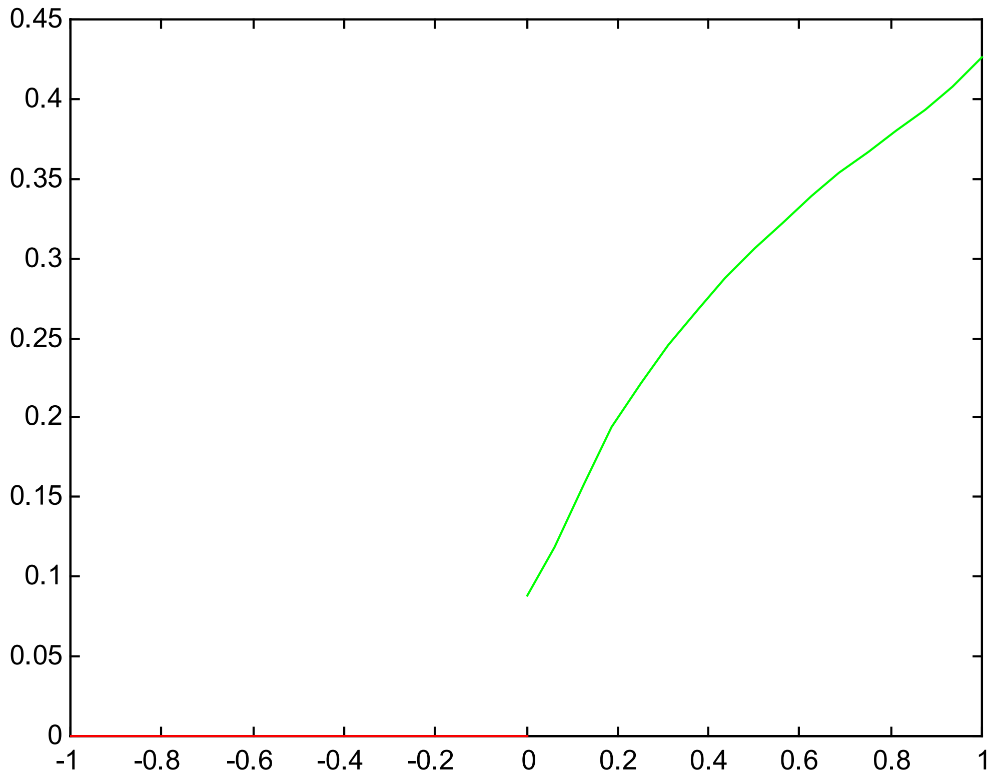

Figure 6.2.1. Initial Function = 0/Forcing Function = 1

Parameters:

Number of Elements N = 16
Interval of t = [0,1]
$\varphi(s) = 0$
$f(t) = 1$

Red represents the Initial Function
Green represents the Approximate Solution on the interval [0,1]

6.2.2 Sample 2

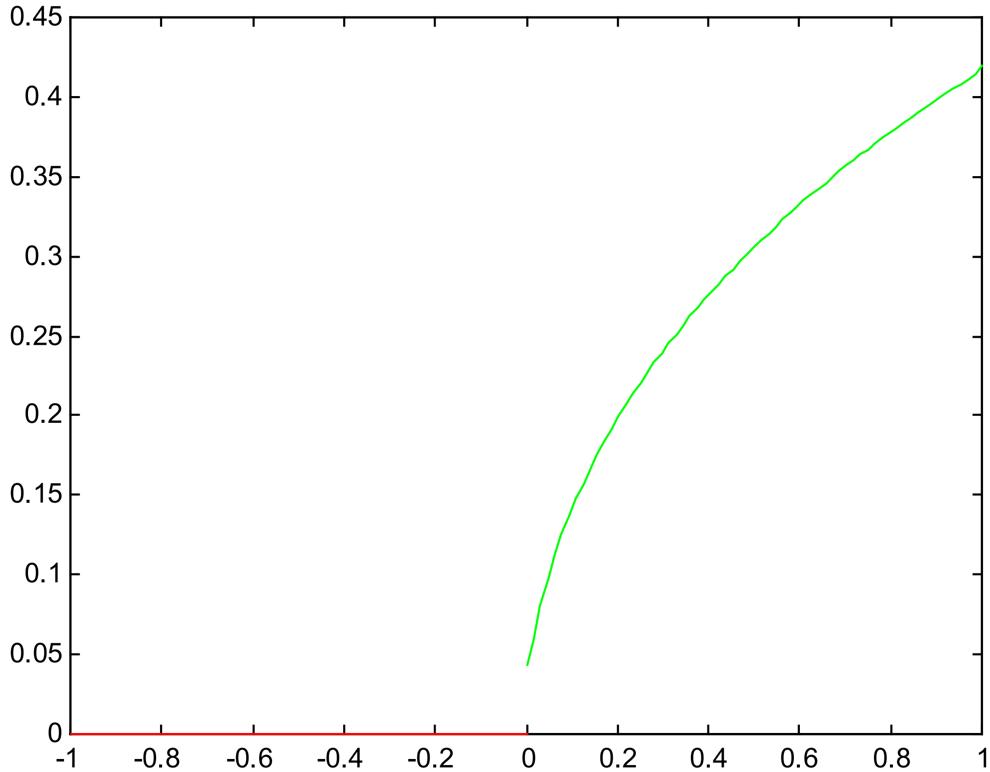

Figure 6.2.2. Initial Function = 0/Forcing Function = 1

Parameters:

Number of Elements N = 64
Interval of t = [0,1]
$\varphi(s) = 0$
$f(t) = 1$

Red represents the Initial Function
Green represents the Approximate Solution on the interval [0,1]

6.2.3 Sample 3

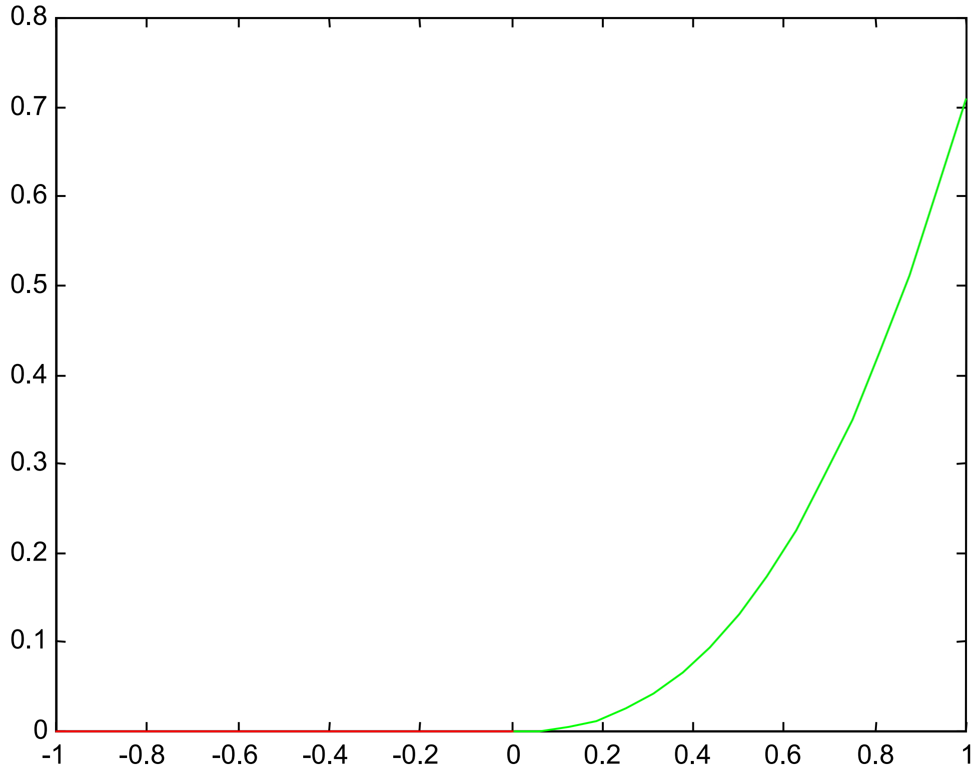

Figure 6.2.3. Initial Function = 0/Forcing Function = $3t^2$

Parameters:

Number of Elements N = 16
Interval of t = [0,1]
$\varphi(s) = 0$
$f(t) = 3t^2$

Red represents the Initial Function
Green represents the Approximate Solution on the interval [0,1]

6.2.4 Sample 4

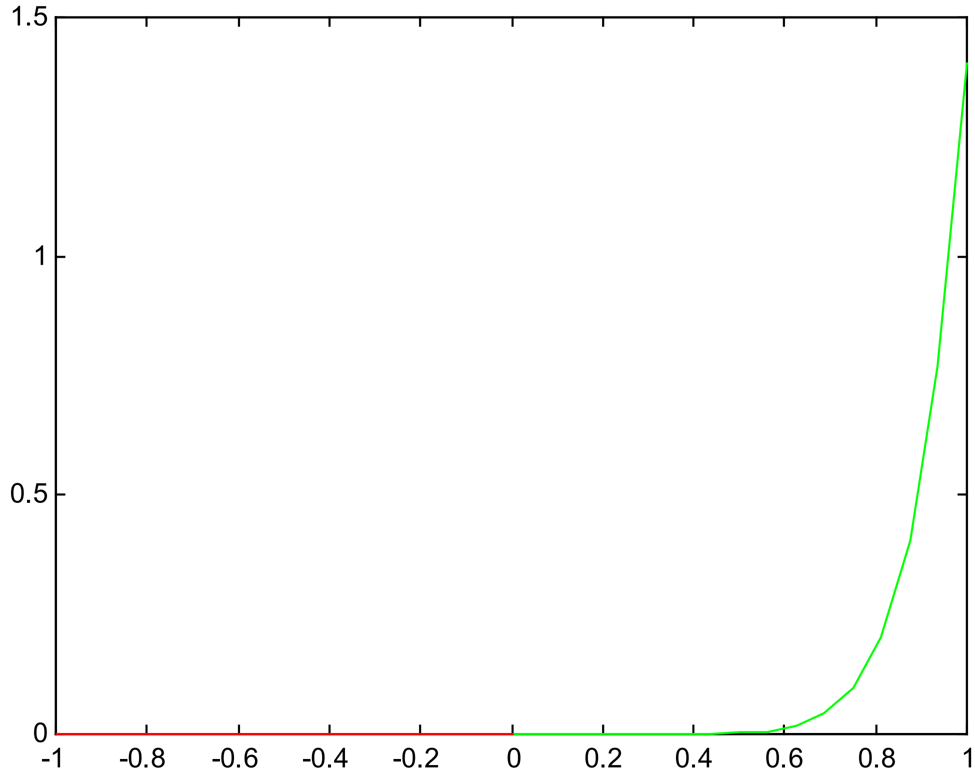

Figure 6.2.4. Initial Function = 0/Forcing Function = $10t^9$

Parameters:

Number of Elements N = 16
Interval of t = [0,1]
$\varphi(s) = 0$
$f(t) = 10t^9$

Red represents the Initial Function
Green represents the Approximate Solution on the interval [0,1]

6.2.5 Sample 5

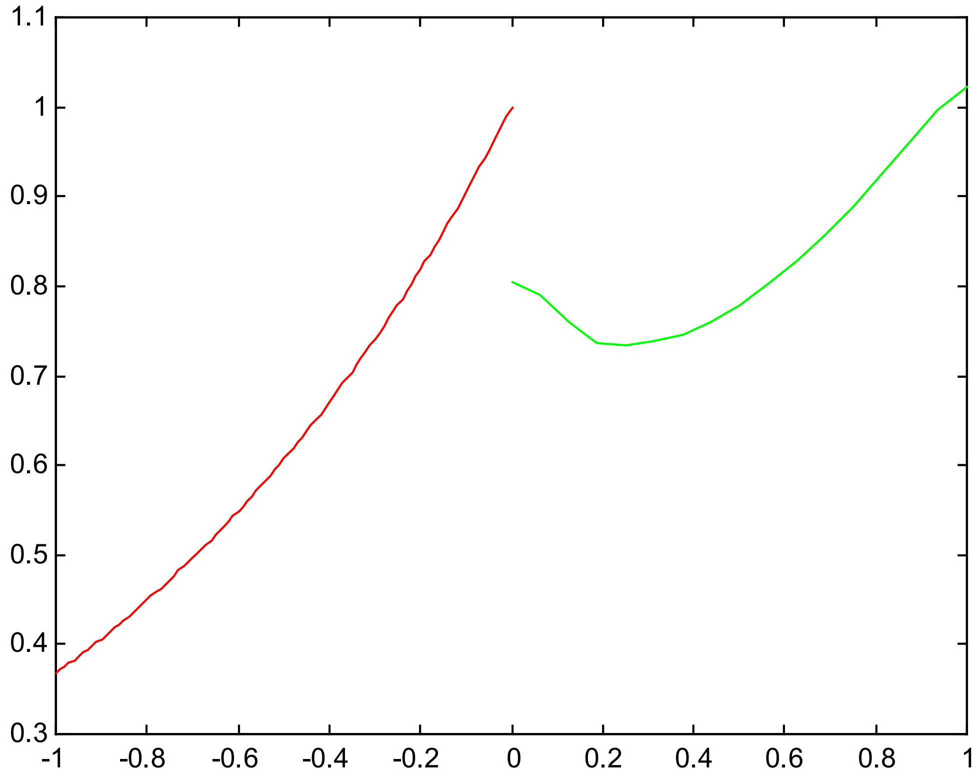

Figure 6.2.5. Initial Function = e^t/Forcing Function = sin(t)

Parameters:

Number of Elements N = 16
Interval of t = [0,1]
$\varphi(s) = e^s$
$f(t) = \sin(t)$

Red represents the Initial Function
Green represents the Approximate Solution on the interval [0,1]

6.2.6 Sample 6

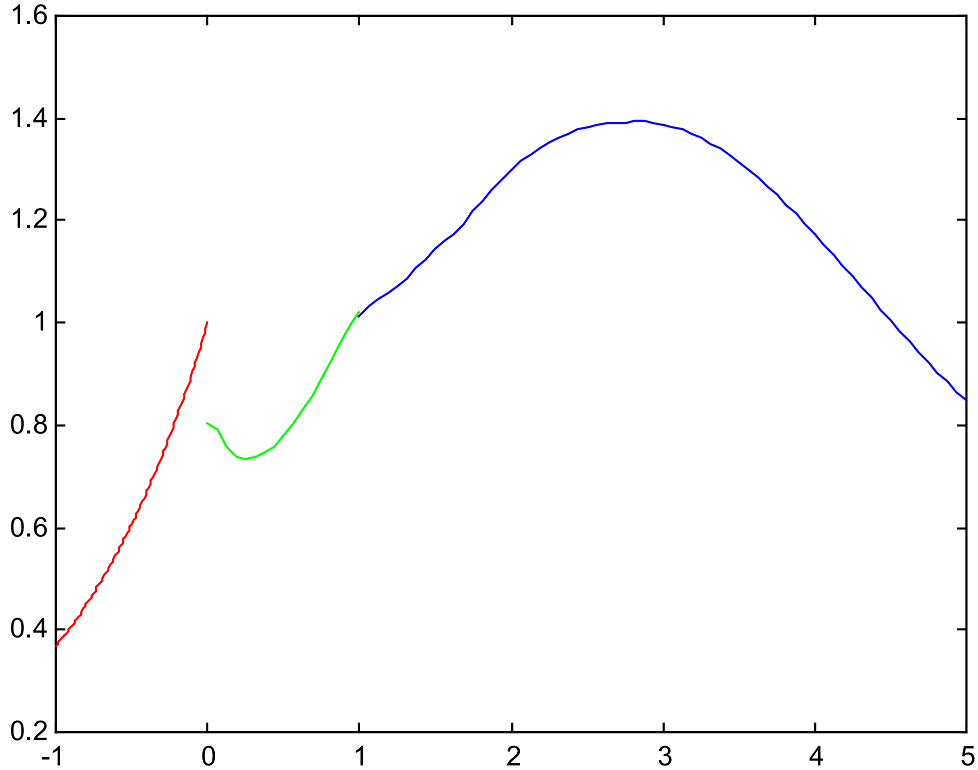

Figure 6.2.6. Initial Function = e^t/Forcing Function = sin(t)

Parameters:

Number of Elements N = 16
Interval of t = [0,5]
$\varphi(s) = e^s$
$f(t) = \sin(t)$

Red represents the Initial Function
Green represents the Approximate Solution on the interval [0,1]
Blue represents the Approximate Solution on the interval [1,5]

6.3 Graphs for Special Case $0<\alpha<1$, $c(s)=(2-s)^{-\alpha}$: $k(s)=[(s-2)/s]^{\alpha}$

6.3.1 Sample 1

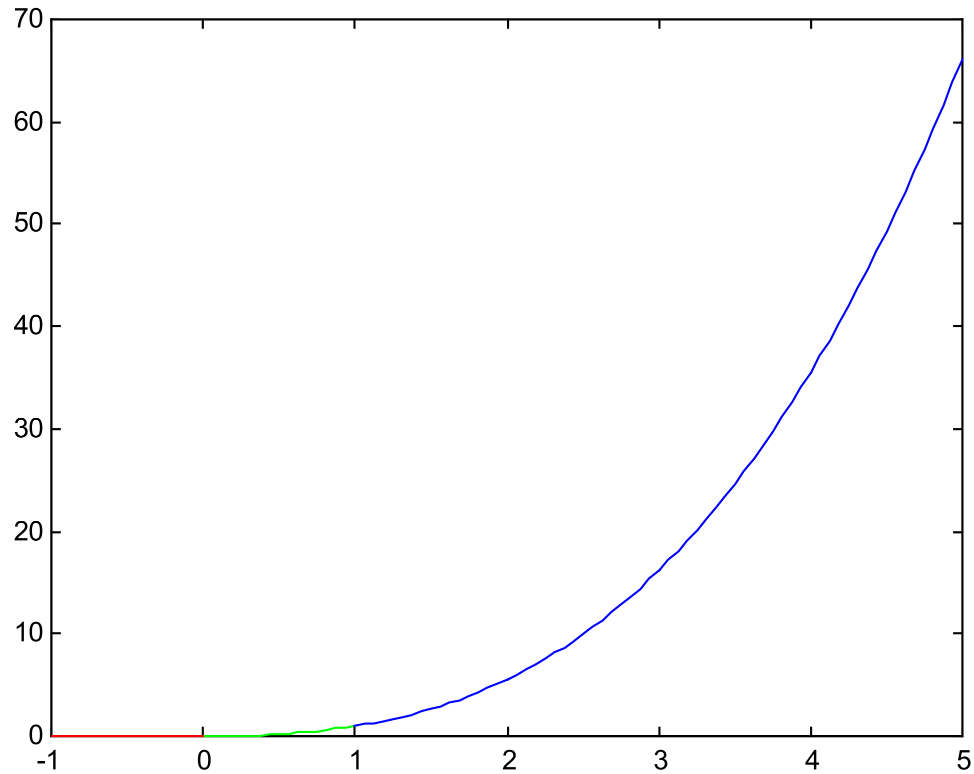

Figure 6.3.1. Initial Function = 0/Forcing Function = $3t^2$

Parameters:

Number of Elements N = 16
Interval of t = [0,5]
$\alpha = .4$
$\varphi(s) = 0$
$f(t) = 3t^2$

Red represents the Initial Function
Green represents the Approximate Solution on the interval [0,1]
Blue represents the Approximate Solution on the interval [1,5]

6.3.2 Sample 2

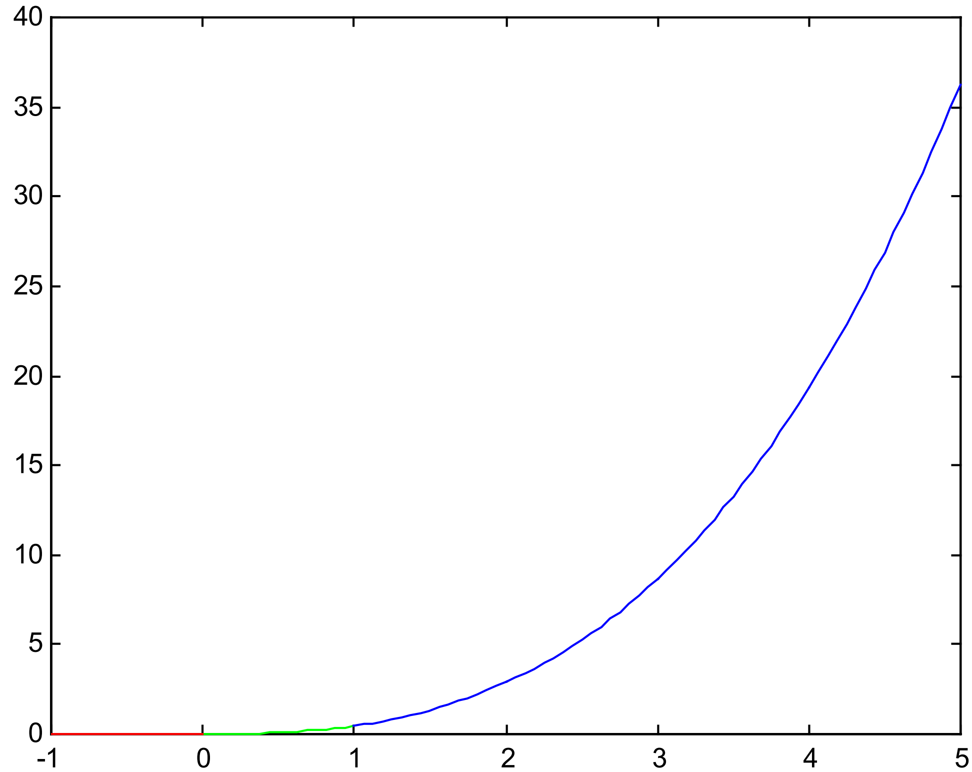

Figure 6.3.2. Initial Function = 0/Forcing Function = $3t^2$

Parameters:

Number of Elements N = 16
Interval of t = [0,5]
$\alpha = .6$

$\varphi(s) = 0$

$f(t) = 3t^2$

Red represents the Initial Function
Green represents the Approximate Solution on the interval [0,1]
Blue represents the Approximate Solution on the interval [1,5]

6.3.3 Sample 3

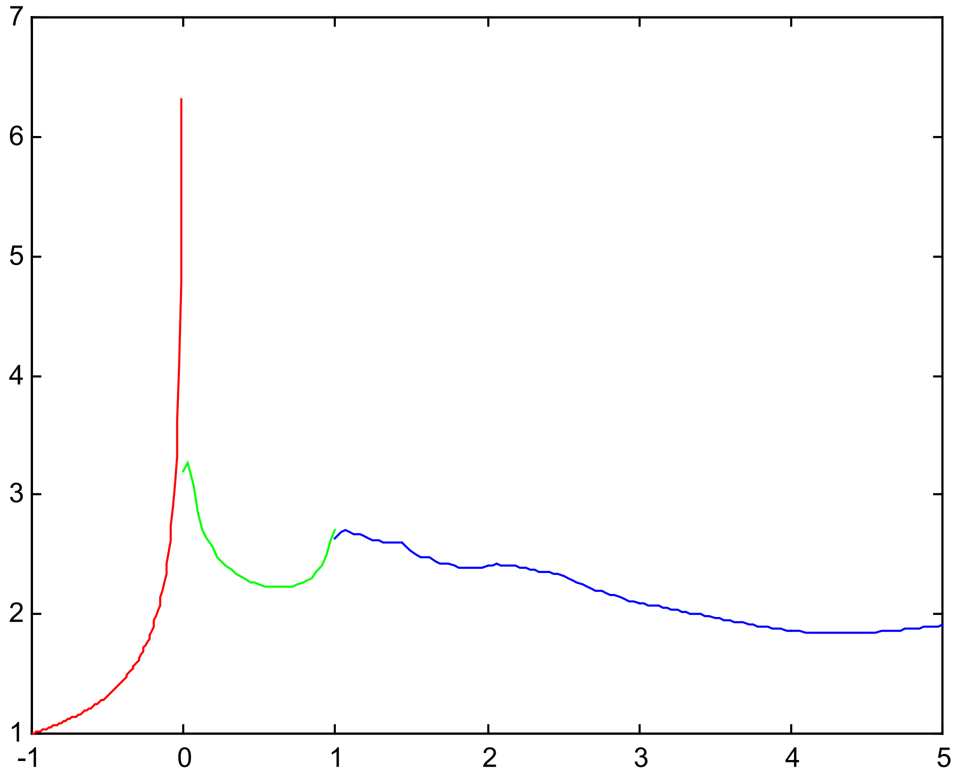

Figure 6.3.3. Initial Function = $(-s)^{.4}$/Forcing Function = cos(t)

Parameters:

Number of Elements N = 16
Interval of t = [0,5]
$\alpha = .5$

$\varphi(s) = (-s)^{.4}$

$f(t) = \cos(t)$

Red represents the Initial Function
Green represents the Approximate Solution on the interval [0,1]
Blue represents the Approximate Solution on the interval [1,5]

6.3.4 Sample 4

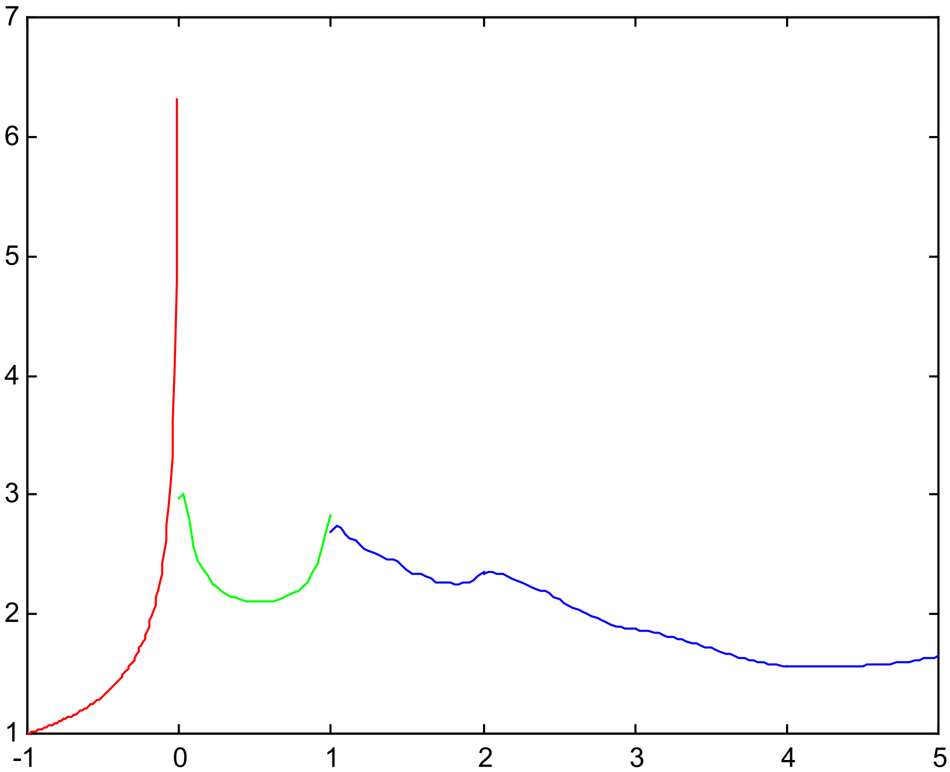

Figure 6.3.4. Initial Function = (-s)$^{.4}$/Forcing Function = cos(t)

Parameters:

Number of Elements N = 16
Interval of t = [0,5]
$\alpha = .4$
$\varphi(s) = (-s)^{.4}$
$f(t) = \cos(t)$

Red represents the Initial Function
Green represents the Approximate Solution on the interval [0,1]
Blue represents the Approximate Solution on the interval [1,5]

6.3.5 Sample 5

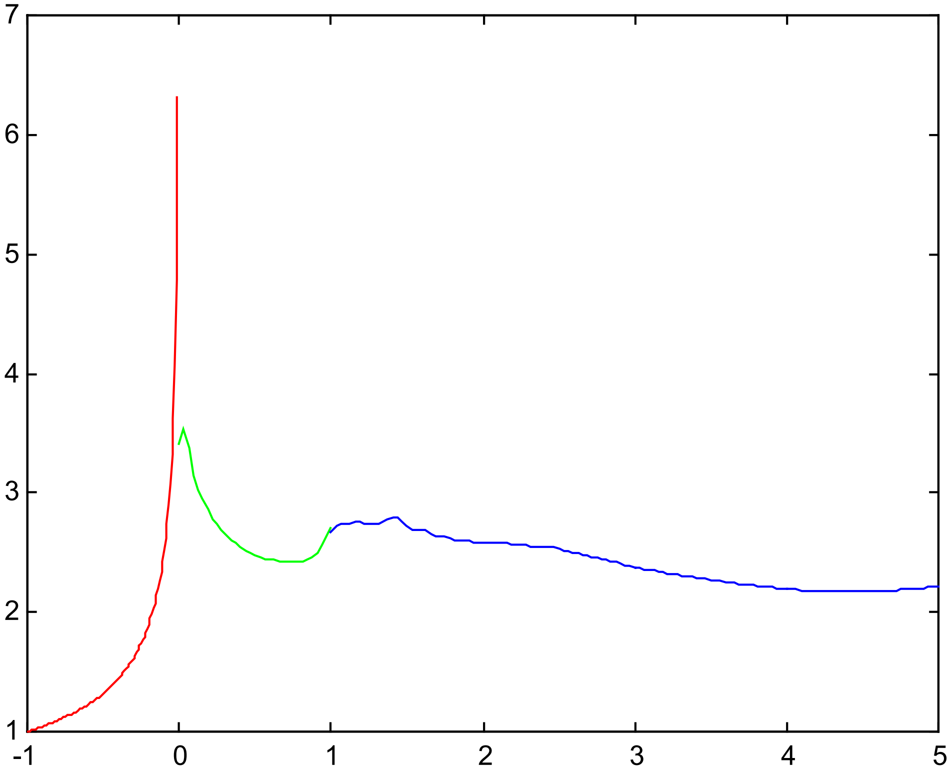

Figure 6.3.5. Initial Function = $(-s)^{.4}$/Forcing Function = cos(t)

Parameters:

Number of Elements N = 16
Interval of t = [0,5]
$\alpha = .6$

$\varphi(s) = (-s)^{.4}$

$f(t) = \cos(t)$

Red represents the Initial Function
Green represents the Approximate Solution on the interval [0,1]
Blue represents the Approximate Solution on the interval [1,5]

Chapter 7

7. CONCLUSIONS

The numerical results contained in this presentation provide excellent approximations for the class of weakly singular neutral equations that were investigated. The finite element scheme presented in Chapter 3 provided a finite dimensional approximation (ODE) for the infinite dimensional neutral functional differential equation (3.1)-(3.3). In Chapter 4 we presented various examples for which the true solution was known. In all such cases we observed that the numerical scheme provided excellent approximations to the true solution. Even in cases where the true solution was not known, the numerical scheme provided approximations that possessed the same (expected) characteristics of related equations for which the true solution was known. The method of steps coupled with the finite dimensional representation for the solutions provided an extension of the time interval from [0,1] to [0, T] for all T>0. In the cases where the numerical results were given on [0,T], T>1, we observed *lack of a high level of smoothness* at T=1, T=2. That is the numerical solution, in some cases, appeared to be non-differentiable at T=1, the second derivative failed to exist at T=2 and higher order derivatives fail to exist at T = n, n a non-negative integer greater then 2. This is an expected characteristic of solutions for the class of singular neutral equations studied in this presentation.

The numerical results for examples 4.3.3, 4.3.4 and 4.3.5 indicate that the numerical scheme presented here could be employed for parameter identification problems. Here the order of singularity α would play the role of the unknown parameter. It is well known that when one uses $L_p[-1,0]$ spaces, p>1, as the space of initial functions, see [5], then well-posedness for (3.1)-(3.3) depends on the order of singularity of the kernel k (value of α). In the case of $\alpha=1/2$ the system (3.1)-(3.3) is well posed on $L_p[-1,0]$ only if p<2. The identification problem gives rise to several theoretical questions. On the other hand the numerical technique presented here *appears* to provide a numerical framework for the identification problem. However, the parameter identification problem requires more investigation. This will be a topic for future work.

Chapter 8

8. Bibliography

[1] H. Ashley, *Flutter Suppression Within Reach*, Aerospace America, Aug 1988, 14-15.

[2] R.L. Bisplinghoff, H. Ashley and Halfman, *Aeroelasticity*, Addison-Wesley, Cambridge, Massachusetts, 1955, 527-532

[3] D.W. Brewer and T. L. Herdman, *A Parameter Dependence Problem In Neutral Functional Differential Equations*, Partial Differential Equations, L. Wiener and J. Hale Eds., Pitman Research Notes in Mathematics Series 273, 1992, 74-79.

[4] J. A. Burns, E. M. Cliff, and T. L. Herdman *A State Space Model for an Aerostatic System*, Proc. 22nd IEEE Conference on Decision and Control, San Antonio, Tx, 1983, 1074-1077

[5] J.A. Burns, T.L. Herdman and H.W. Stech, *Linear Functional Differential Equations as Semigroups on Product Spaces*, SIAM J. Math. Anal. Vol 14, No. 1, 1983, 98-116

[6] J. A. Burns, T. L. Herdman and J. Turi, Neutral Functional Integro-differential Equations with Weakly Singular Kernals, J. of Math. Anal. And Applications, Vol. 145. No. 2, 1990, 371-401

[7] J.A. Burns and K. Ito, *On Well-posedness of Solutions to Iintegro-differiential Equations of Neutral-type in a Weighted L_2-space*, Differential and Integral Equations, Vol. 8, No. 3, 1995, 627-646

[8] Y.Z. Cao, G.M. Cerezo, T.L. Herdman and J. Turi, *Solution Representations for a Class of Singular Neutral Equations*, preprint.

[9] G.M. Cerezo, Elena Fernandez Berdaguer, T. L. Herdman and J. Turi, *Co-location Techniques for Singular Neutral Equations*, Proc. 33rd Conference on Decision and Control, Lake Buena Vista, FL, 1994, 2534-2536.

[10] G.M. Cerezo, Elena Fernandez Berdaguer, T. L. Herdman and J. Turi, *Parameter Identification Techniques for Singular Neutral Equations*, Applied Mechanics in the Americas Vol. II, 1995, 322-327.

[11] J.W. Edwards, *Flight Test Results of an Active Flutter Suppression System Installed on a Remotely Piloted Research Vehicle*, NASA T.M. 83132, 1981.

[12] F. Hartung, T.L. Herdman and J. Turi, *Parameter Identification in a Class of Hereditary Systems of Neutral Type*, J. of Applied Mathematics and Computation, to appear.

[13] C.A. Harvey, G. Stein, and L.R. Felt, *Adaptive Control of Wing/Store Flutter – A Feasibility Study*.

[14] T.L. Herdman and J. Turi, *On the Solutions of a Class of Integral Equations Appearing in Unsteady Aerodynamics*, Differential Equations; Stability and Control Lecture Notes in Pure and Applied Mathematics, Vol. 127, S. Elaydi, Ed., Marcel Dekker, 1990, 241-148

[15] T.L. Herdman and J. Turi, *An Application of Finite Hilbert Transforms in the Derivation of a State Space Model for an Aeroelastic System*, J. of Integral Equations and Applications, Vol. 7, No. 4, 1991, 271-287.

[16] T.L. Herdman and J. Turi, *A Natural State-Space for an Aeroelastic Control System, J. of Integral Equations and Applications*, Vol. 7, No. 4, 1995, 413-424.

[17] K. Ito and J. Turi, Numerical Methods for Singular Integro-differential Equations Based on Semi-group Approximations, SIAM J. Numerical Analysis, Vol. 28, 1991, 1698-1722

[18] F. Kappel and Kang P. Zang, *On Neutral Functional Differential Equations with Nonatomic Difference Operator*, J. Math. Anal. 113, 1986, 311-343.

Vita

Darwin T. Herdman was born in Norman, Oklahoma in 1968. He graduated from Blacksburg High School in 1986 and received a full athletic scholarship to attend Virginia Polytechnic Institute and State University. Athletically, Mr. Herdman was awarded four varsity letters in football and received the Most Valuable Senior award. Academically, he studied both engineering and mathematics before receiving his BS degree in Mathematics the Spring of 1991. Throughout his academic career, he participated in summer intern programs at Veda Incorporated in Alexandria, Virginia. He joined Veda Incorporated full soon after graduation as a systems engineer focusing on aircraft systems integration and military command, control, communications, computers and intelligence (C4I). As the result of several acquisitions and mergers, Veda Incorporated became Veridian Corporation where Mr. Herdman currently serves as the Director, Information Technology for the Applied Technology Group (ATG). In August of 1996, he enrolled in the Interdisciplinary Applied Mathematics program at VPI&SU's Northern Virginia campus. He received his MS degree in the Spring of 1999.