National Library
of Canada

Acquisitions and
Bibliographic Services Branch

395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale
du Canada

Direction des acquisitions et
des services bibliographiques

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file Votre référence

Our file Notre référence

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

Canada

A PRIORI MESH SELECTION FOR SINGULARLY PERTURBED BOUNDARY VALUE PROBLEMS

by

Jiashun Liu

B.Sc., Qufu Normal University, 1983

M.Sc., Dalian University of Technology, 1986

THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

in the Department

of

Mathematics and Statistics

SIMON FRASER UNIVERSITY

April, 1991

ISBN 0-315-78305-2

APPROVAL

Name: Jiashun Liu

Degree: Master of Science

Title of Thesis: A Priori Mesh Selection for Singularly Perturbed Boundary Value Problems.

Examining Committee:

Chairman: Dr. A. H. Lachlan

Dr. R. D. Russell
Senior Supervisor

Dr. R. Lardner

Dr. M. Trummer

Dr. T. Tang
External Examiner
Department of Mathematics and Statistics
Simon Fraser University

Date Approved: April 4, 1991

PARTIAL COPYRIGHT LICENSE

I hereby grant to Simon Fraser University the right to lend my thesis, project or extended essay (the title of which is shown below) to users of the Simon Fraser University Library, and to make partial or single copies only for such users or in response to a request from the library of any other university, or other educational institution, on its own behalf or for one of its users. I further agree that permission for multiple copying of this work for scholarly purposes may be granted by me or the Dean of Graduate Studies. It is understood that copying or publication of this work for financial gain shall not be allowed without my written permission.

Title of Thesis/Project/Extended Essay

A Priori Mesh Selection for Singularly Perturbed Boundary Value Problems

Author:

(signature)

JIASHUN LIU

(name)

April 9, 1991

(date)

ABSTRACT

A brief description of boundary value problems and initial value problems for ODEs is given. Our particular interest is to solve large singularly perturbed boundary value problems, where both boundary layers and interior layers are expected. After we present the theoretical framework, we propose a method of obtaining the mesh for singularly perturbed boundary value problems by solving a differential Riccati equation. The solution is then computed by any standard numerical method (here, we use spline collocation). The numerical examples show that the mesh we obtained by this procedure is a good one.

DEDICATION

Dedicated to the memory of my father, Fengcheng Liu (1936-1990)

ACKNOWLEDGEMENT

I am very grateful to my senior supervisor, Dr. R. D. Russell, for his guidance and patient encouragement during the preparation of this thesis.

I also want to thank Dr. L. Dieci and Dr. M. Trummer for their help and cooperation. Thanks also due to the staffs of mathematics department especially Ms. Sylvia Holmes.

Last but not least, I would like to express my sincere gratitude to my wife, Guoying (Helen), for her understanding during this work.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. INTRODUCTION

Nowadays, there are a lot of Boundary Value Problems (BVPs) arising from application areas such as chemical kinetics, pollution modeling, fluid dynamics and biology, to name a few. It is important to study the theory of BVP as well as its numerical solution. This thesis is devoted to the numerical solution of BVPs.

Generally speaking, two kinds of methods are developed to solve BVPs. The first kind is so-called global methods or direct methods (e.g. collocation method, finite difference method and finite element method), which is characterized by solving global (linear) algebraic systems for the discrete solution. The second kind of methods is characterized by the association of the BVP with certain auxiliary Initial Value Problems (IVPs), they are called sequential methods or indirect methods (e.g. shooting methods, Riccati method, orthogonalization methods and invariant imbedding method).

The multi-shooting method which requires solving IVPs as well as algebraic systems, may be regarded as a hybrid method between global and sequential. The sophisticated package BOUNDPAC of Mattheij and Staarink [25] is based on this method. For the collocation method and finite difference method, there are the well known programs COLSYS [1] and PASVA [26] available. All of the programs can handle two-point BVPs with mild boundary layers.

For stiff two-point BVPs, both boundary layers and interior layers are expected. There are some difficulties in dealing with them by the Global method. The difficulties arise when trying to cope with the fast and slow modes of BVPs equally, or to select meshes without enough information about the layers. On the other hand,

most of the sequential methods can transform BVP into two classes of IVPs, which are solved with bi-directional strategies. Unfortunately, there is no sophisticated software based on these methods available. However, a solver of Differential Riccati Equation (DRE) has been developed by L. Dieci [11].

Both kinds of methods are well-represented in the literature. The relationship between these methods is explored in [2, 17]. The motivation of this thesis is to solve stiff two-point BVPs and to solve large BVPs efficiently. Our approach is to get a good a priori mesh for global methods, especially COLSYS, by solving DRE. In this thesis, we consider mostly the stiff linear two-point BVPs with separated boundary conditions (BCs).

In chapter 2, we review some analytical results for IVPs and BVPs. The standard forms of IVP and BVP are given to strengthen the understanding. The well-known existence and uniqueness theorem for the solution of IVPs is presented for completeness. Since it is very difficult to give an existence and uniqueness theorem for the solution of general BVPs, we only mentioned the existence and uniqueness theorem for the solution of linear BVPs. For some restricted results about the existence of solutions of nonlinear BVPs, one can refer to [8] and the references therein. The stability (well-poseness) and the stiffness of IVPs are also presented. The stability (well-conditioning) of the solution of BVPs depends on the dichotomy of the fundamental solution as well as the BCs, which is also briefly reviewed in this chapter.

From chapter 2 we know that the numerical solution of BVPs could be much more difficult than that of IVPs. Even for a linear BVP for which we can guarantee the existence and uniqueness of its solution, this solution could be ill-conditioned. Even

for well-conditioned BVPs, its solution can consist of boundary and/or interior layers if this BVP is stiff (or this BVP is a singularly perturbed problem). Chapter 3 provides one approach to solve stiff BVPs. Section 3.1 restates the idea of theoretical multiple shooting of [7, 8] which serves as a framework for analyzing the stability of numerical methods for stiff BVPs. The framework requires a segmentation of the interval where we want to find the solution of a stiff BVP. This segmentation identifies the layer regions and smooth regions. Section 3.2 describes the way in which [3] gets this segmentation. In section 3.3, after analyzing the Riccati method for solving stiff BVPs, we propose an idea of getting the segmentation for the interval of interest via solving the Differential Riccati Equations (DREs). Chapter 4 describes DRE and the numerical methods for solving DREs. The DRE solver DRESOL, which we used to get a DRE mesh, is also introduced here. We present the method of getting the simple DRE mesh, combined DRE mesh and trimmed DRE mesh. All the examples are given in chapter 5. From the numerical results we know that the trimmed DRE mesh is the mesh we desired.

2. Basic Theory

A general first order Ordinary Differential Equation(ODE) can be written as a first-order system:

$$y' = f(x,y), \qquad a < x < b \tag{2.1}$$

where $y(x)=(y_1(x), y_2(x),..., y_n(x))^T$ is the unknown function, $f(x,y) =(f_1(x,y), f_2(x,y),...,f_n(x,y))^T$ is a vector-valued function. If f is nonlinear in y, it refers to a nonlinear problem. Otherwise the ODE relates to a linear problem, which can be simplified to the following form:

$$y' = A(x)y + q(x), \qquad a < x < b \tag{2.2}$$

where A is an n×n matrix function of x, q is an n×1 vector function of x. In both linear and nonlinear cases, the interval ends a and b can be finite or infinite. As is well-known, high order ODEs can usually be converted to first order systems by a standard transformation or special ones. Without loss of generality, we consider only first order linear ODEs here. When $q(x) = 0$, the ODE is called homogeneous', otherwise it is nonhomogeneous.

A boundary value problem for an ordinary differential equations on a given interval includes two parts:

1. differential equations
2. explicit conditions that a solution of the ODEs must satisfy at one or several points, which are called Boundary Conditions(BCs).

If there are n BCs specified at two end points of the interval, these BCs are called Two-Point Boundary Conditions(TPBCs), which can be written as

$$g(y(a),y(b)) = 0, \tag{2.3}$$

where $g = (g_1, g_2, \ldots, g_n)^T$ is a vector function. It is generally nonlinear. If it is a linear TPBC, the general form is

$$B_a y(a) + B_b y(b) = \beta, \tag{2.4}$$

where $B_a, B_b \in R^{n\times n}$ and $\beta \in R^n$. Since the information of BCs given at two points is c ·upled together, these BCs are called non-separated BCs. The following are called separated BCs:

$$B_1 y(a) = \beta_1,$$

$$B_2 y(b) = \beta_2,$$

with $B_1 \in R^{k\times n}$, $B_2 \in R^{(n-k)\times n}$, $\beta_1 \in R^k$, $\beta_2 \in R^{n-k}$. We can derive the concept of separated BC for nonlinear BC similarly. If g can be reduced to the special form

$$y(a) = \alpha,$$

that is, the condition is specified at only one initial point, then we refer to this as an Initial Value Problem(IVP).

In section 2.1 we summarize some basic results about IVPs, including existence, uniqueness, stability and stiffness. In section 2.2 we list some limited results about BVPs, including existence, uniqueness, stability and dichotomy.

2.1 Initial Value Problem

Standard form of IVP

$$y' = f(x,y), \qquad a < x < b, \tag{2.5a}$$

$$y(a) = \alpha \tag{2.5b}$$

The theory and numerical techniques dealing with IVPs are matured comparing with those of BVPs. A unique solution is guaranteed to exist under very mild assumptions. We state the theorem

Existence and Uniqueness

Theorem 2.6 Suppose that f(x,y) is continuous on $D = \{(x,y) : a \le x \le b, \ | y - \alpha | \le \rho\}$ for some $\rho>0$, and suppose that f(x,y) is Lipschitz continuous with respect to y: i.e. there exists a constant $L>0$ such that for any (x,y) and (x,z) in D:

$$| f(x,y) - f(x,z) | < L | y - z |$$

If f(x,y) is bounded by $M > 0$ on D, and $c = \min\{ b-a , \rho/M \}$, the IVP has a unique solution for $a \le x \le a+c$. If the Lipschitz condition holds uniformly for all y and z, then the IVP has a unique solution for all $x > a$.

While we know the fundamental solution Y(x)=Y(x;a) of the corresponding homogeneous ODEs:

$$Y'(x;a) = A(x)Y(x;a), \qquad a < x < b, \tag{2.7a}$$

$$Y(a;a) = I \tag{2.7b}$$

it is easy to show that the solution of IVPs is

$$y(x)=Y(x)\left[\alpha+\int_a^x Y^{-1}(t)q(t)dt \right] \tag{2.8}$$

If $Y(a)\neq I$, we can get a more general form of solution:

$$y(x)=Y(x)Y^{-1}(a)\alpha + \int_a^x Y(x)Y^{-1}(t)q(t)dt,$$

or

$$y(x)=Y(x)Y^{-1}(a)\alpha + \int_a^x G(x,t)q(t)dt, \tag{2.9}$$

where the matrix function G(x,t) is defined as

$$G(x,t) = \begin{cases} Y(x)Y^{-1}(t) & \text{if } t \le x \\ 0 & \text{if } t > x \end{cases} \qquad (2.10)$$

Stability and stiffness

Definition 2.11 A solution $y(x)$ is said to be stable if given $\varepsilon > 0$, there is a $\delta > 0$ such that any other solution $\hat{y}(x)$ of the IVP satisfying

$$| y(a)-\hat{y}(a) | \le \delta,$$

also satisfies

$$| y(x)-\hat{y}(x) | \le \varepsilon \quad \text{for all } x > a,$$

$y(x)$ is asymptotically stable if it further satisfies

$$| y(x)-\hat{y}(x) | \to 0 \quad \text{as } x \to \infty,$$

$y(x)$ is uniform stable if given $\varepsilon > 0$, there is a $\delta > 0$ such that any other solution $\hat{y}(x)$ of IVP satisfying

$$| y(c)-\hat{y}(c) | \le \delta,$$

at some point $c \ge a$ also satisfies

$$| y(x)-\hat{y}(x) | \le \varepsilon \quad \text{for all } x > c,$$

The concept of asymptotic uniform stability can be defined in a similar way.

Let $y(x)$, $\hat{y}(x)$ be solutions of $y' = A(x)y + q(x)$, then the difference $z(x)=y(x) - \hat{y}(x)$ is a solution of $z' = A(x)z$. This means that only the homogeneous problem matters for stability. In order to state the stability properties of IVP, we introduce the following concept

Definition 2.12 The ODEs $y' = A(x)y$ and $w' = V(x)w$ are kinematically similar if there is a differentiable transformation $T(x) \in R^{n\times n}$, with cond(T;x,t) =

$\| T(x)\| \|T^{-1}(t)\|$ uniformly bounded for $x \geq t$, and if $w(x) = T^{-1}(x)y(x)$, then

$$w' = V(x)w \qquad x>a,$$

where $V(x)=T^{-1}(x)\left[A(x)T(x) - T'(x)\right]$

when V(x) is upper triangular form. Its diagonal elements are called the kinematic eigenvalues corresponding to T(x).

Theorem 2.13 Suppose that the homogeneous ODEs $y' = A(x)y$ and $w' = V(x)w$ are kinematically similar with V(x) upper triangular, and $\|A(x)\|$, $\|T'(x)\|$ are uniformly bounded in x, λ_i is kinematic eigenvalues corresponding to T(x). Then the solution of $y' = A(x)y$ is uniformly asymptotically stable iff there are positive constants c and λ such that

$$\mathrm{Re}\left(\int_t^x \lambda_i(s)ds\right) < -\lambda(x-t) \qquad \text{for } x-t > c \qquad (2.14)$$

In the special case where A(x) is a constant matrix, its solution is asymptotically stable iff the real parts of eigenvalues of A are negative.

Many applications involve initial value problems $y' = f(x,y)$ with fast and slow decay rates, especially in chemical kinetic problems and for the system of ODEs derived from PDEs discretized in space. This means that the solution contains different time scales, where one may change much faster than the others. This kind of problems which can cause difficulty in getting its numerical solution is called stiff. Stiffness can be expressed more accurately in terms of the Jacobian matrix $J(x^*,y^*)$.

Definition 2.15 An initial value problem $y' = f(x,y)$ is stiff at a point $x = x^*$, $y = y^*$, if the eigenvalues of the Jacobian matrix differ greatly in magnitude.

2.2 Boundary Value Problems(BVPs)

Standard form of BVP

$$y' = f(x,y), \qquad a < x < b,$$
$$g(y(a),y(b)) = 0 \tag{2.16}$$

This is generally a nonlinear BVP. For linear ODE with linear two point boundary conditions, we have the following linear BVP:

$$y' = A(x)y + q(x), \qquad a < x < b,$$
$$B_a y(a) + B_b y(b) = \beta \tag{2.17}$$

Existence and uniqueness

The existence and uniqueness determination of the solution of a BVP is much more difficult than that for IVPs. Generally speaking, there is no guarantee of the existence for a solution of a nonlinear BVP (2.16). Even if a solution of (2.16) exists, the uniqueness of it can only be guaranteed locally under certain assumption. However, if the BVP is a linear equation with linear BCs, we have the following theorem to guarantee the existence and uniqueness of its solution.

Theorem 2.18 Suppose that A(x) and q(x) in the linear differential equation (2.2) are continuous. The BVP (2.17) has a unique solution y(x) iff the matrix

$$Q = B_a Y(a) + B_b Y(b), \tag{2.19}$$

is nonsingular, and the solution is

$$y(x) = Y(x)Q^{-1}\left[\beta - B_b Y(b)\int_a^b Y^{-1}(t)q(t)dt\right] + Y(x)\int_a^x Y^{-1}(t)q(t)dt$$

where Y(x) is any fundamental solution of the corresponding homogeneous differential equation.

Let $\Phi(x) = Y(x)Q^{-1}$. The solution of (2.17) can be simplified to

$$y(x)=\Phi(x)\beta + \int_a^b G(x,t)q(t)dt \tag{2.20}$$

with G(x,t) being the n×n Green's matrix function, defined as

$$G(x,t) = \begin{cases} \Phi(x)B_a\Phi(a)\Phi^{-1}(t) & \text{if } t \le x \\ -\Phi(x)B_b\Phi(b)\Phi^{-1}(t) & \text{if } t > x \end{cases} \tag{2.21}$$

Stability and dichotomy

Stability which describes the asymptotic behaviour of the solution is an important concept for initial value problems. However the sensitivity of BVPs on finite intervals is more appropriately described in terms of conditioning. Since the solution of (2.17) is

$$y(x) = \Phi(x)\beta + \int_a^b G(x,t)q(t)dt,$$

if $$\kappa_1 = \| \Phi(x)\|_\infty = \| Y(x)Q^{-1}\|_\infty, \tag{2.22}$$

$$\kappa_2 = \sup \left\{ \left[\int_a^b \| G(x,t) \|^q dt \right]^{1/q} \right\}, \qquad \frac{1}{p}+\frac{1}{q}=1, \tag{2.23}$$

then we have $\|y\|_\infty \le \kappa_1\|\beta\| + \kappa_2\|q\|_p$

Choosing $p = \infty, 1$ respectively, we have

$$\|y\|_\infty \le \kappa_1\|\beta\| + \kappa_2\|q\|_\infty, \qquad \kappa_2 = \sup_x \{ \int_a^b \| G(x,t) \|dt \},$$

$$\|y\|_\infty \le \kappa_1\|\beta\| + \kappa_2\|q\|_1, \qquad \kappa_2 = \sup_{a\le x,t\le b} \| G(x,t) \|,$$

and $\kappa = \max\{ \kappa_1, \kappa_2\}$ may be called the conditioning constant. It gives a bound on how perturbations in data β and q may be amplified.

The stability of IVPs requires that all modes of the solution are decreasing. In the case of BVP, both decreasing and increasing modes can be involved. To make

sure the BVP is stable, it is natural to require that the increasing and decreasing modes be bounded. The splitting of the solution is called dichotomy.

Definition 2.24 Suppose Y(x) is a fundamental solution for the linear ODE

$$y' = A(x)y$$

where A(x) is a continuous matrix function. The ODE has an exponential dichotomy if there exists a constant orthogonal projection matrix $P \in R^{n\times n}$ of rank r, $0 \le r \le n$, and positive constants K, λ, μ with K of moderate size, such that

$$\|Y(x)PY^{-1}(t)\| \le Ke^{-\lambda(x-t)} \qquad \text{for } x > t \tag{2.25a}$$

$$\|Y(x)(I-P)Y^{-1}(t)\| \le Ke^{-\mu(t-x)} \qquad \text{for } x \le t \tag{2.25b}$$

for $a \le x, t \le b$. It is said to have an ordinary dichotomy if (2.25) holds with $\lambda = 0$ and/or $\mu = 0$.

Suppose that the ODE $y' = A(x)y$ has an exponential dichotomy. Let P be the projection such that (2.25) holds. Denote the solution space $S = \{Y(x)c;\ c \in R^n\}$, and let $S_2 = \{Y(x)Pc;\ c \in R^n\}$ and $S_1 = \{Y(x)(I-P)c;\ c \in R^n\}$, then $S = S_1 \oplus S_2$, and we have

Theorem 2.26 Any solutions $u(x) \in S_1$ and $w(x) \in S_2$ satisfy

$$\frac{\|u(x)\|}{\|u(t)\|} \le Ke^{-\lambda(x-t)} \qquad \text{for } x > t \tag{2.27a}$$

$$\frac{\|w(x)\|}{\|w(t)\|} \le Ke^{-\mu(t-x)} \qquad \text{for } x \le t \tag{2.27b}$$

This means that in a rough sense there are r increasing and n-r decreasing fundamental solution components.

3. Framework of numerical method for solving stiff BVPs

Stiff ODEs often have solutions with boundary and/or interior layers. In the layer regions, which are usually narrow, the solution varies rapidly compared to the other regions. When solving such a problem numerically, If we use a uniform mesh, we must have a dense mesh because of the fast modes, which are very expensive to calculate; if we do not use a uniform mesh, and try to use a dense mesh in the layer regions, we have to identify the layer regions. This is the purpose of this thesis.

We consider the ODE subject to well-scaled boundary conditions:

$$y' = A(x)y + q(x), \quad a < x < b, \tag{3.1a}$$

$$B_a y(a) + B_b y(b) = \beta \tag{3.1b}$$

where $B_a, B_b \in R^{n\times n}$. We assume that $[B_a, B_b]$ has orthonormal rows. It is convenient to assume that $A(x)$ and $q(x)$ depend on a small parameter ε, and as $\varepsilon \to 0$, $A(x)$, $q(x)$ may become unbounded. But the well-conditioning of the BVP is assumed to be independent of ε, i.e. the condition constant κ is of moderate size independent of ε, where

$$\kappa = \max(\kappa_1, \kappa_2),$$

$$\kappa_1 = \| \Phi \|_{[a,b]},$$

$$\kappa_2 = \| G \|_{[a,b]}.$$

$\Phi(x)$ is the fundamental solution of $y'=A(x)y$, satisfying $B_a\Phi(a) + B_b\Phi(b) = I$.

Suppose the solution of the stiff BVP has boundary layers and/or interior layers connecting longer subintervals where the solution varies slowly. We hope to find a segmentation of the interval $[a,b]$

$$a = t_1 < t_2 < \ldots < t_M < t_{M+1} = b \tag{3.2}$$

such that on each subinterval $[t_j, t_{j+1}]$, precisely one of the following occurs:

(i) The solution has a boundary layer, then j=1 for left boundary layer or j=M for right boundary layer, and $t_{j+1} - t_j \to 0$ as $\varepsilon \to 0$.

(ii) The solution has an interior layer, $1 < j < M$ and $t_{j+1} - t_j \to 0$ as $\varepsilon \to 0$

(iii) The solution is smooth on the subinterval, i.e. for some positive integer p: $\| y^{(v)} \|_{[tj,tj+1]} \leq \text{const}$ for $v = 0,1,2, \ldots, p$ where const is independent of ε.

The determination of this kind of segmentation can be identified from the sign and size of the eigenvalues λ of A(x). Basically, three types of solution modes can be identified: fast decreasing modes corresponding to $\text{Re}(\lambda) < 0, |\text{Re}(\lambda)| >> 0$; fast increasing modes corresponding to $\text{Re}(\lambda) > 0, |\text{Re}(\lambda)| >> 0$; and slow modes for which $|\lambda| << K$, K is a constant of moderate size. The fast modes must contribute very little to the solution in segments where it is smooth, so they need not necessarily be approximated well.

Once we find the segmentation (3.2), we can use a dense mesh in a subinterval of type (i) and (ii), while on an interval of type (iii), we can have a mesh with $h\|A(x)\| >> 1$. Now to solve the problems, one can use a collocation method, difference method etc. In section 3.1, the framework of [7,8] for numerical methods based on the segmentation(3.2) is provided. We discuss the segmentation obtained by KNB [3] in section 3.2. In 3.3, we propose a method for getting the segmentation by solving a differential Riccati equation.

3.1 Theoretical multiple shooting

In this section, we will describe the theoretical multiple shooting given by [7,8], which serves as a framework for analyzing numerical methods for stiff BVPs.

Theoretical multiple shooting

Suppose we know the segmentation (3.2). On each segment $[t_j,t_{j+1}]$, we have a sub BVP which is defined as:

$$y' = A(x)y + q(x), \qquad t_j \le x \le t_{j+1}, \tag{3.3a}$$

$$B_{1j}y(t_j) + B_{2j}y(t_{j+1}) = s_j \tag{3.3b}$$

where $B_{1j}, B_{2j} \in R^{n\times n}$ and the vector $s_j \in R^n$ is to be determined, and $[B_{1j}, B_{2j}]$ is assumed to have orthonormal rows.

Let $\Phi_j(x)$ be the fundamental solution of (3.3) and $v_j(x)$ be particular solution of the ODE, satisfying

$$B_{1j}\Phi_j(t_j) + B_{2j}\Phi_j(t_{j+1}) = I,$$

$$B_{1j}v_j(t_j) + B_{2j}v_j(t_{j+1}) = 0$$

Then the solution of (3.3) can be written as

$$y(x) = \Phi_j(x)s_j + v_j(x) \qquad t_j \le x \le t_{j+1}, \quad 1 \le j \le M$$

If we require y(x) to be a solution of (3.1), we can patch together the pieces through

$$y(t_j^-) = y(t_j^+) \qquad 2 \le j \le M$$

which can be written as:

$$\Phi_j(t_{j+1})s_j - \Phi_{j+1}(t_{j+1})s_{j+1} = \beta_j = v_{j+1}(t_{j+1}) - v_j(t_{j+1}) \qquad 1 \le j \le M-1$$

The BC is rewritten as:

$$B_a\Phi_1(t_1)s_1 + B_b\Phi_M(t_{M+1})s_M = \beta_M = \beta - B_av_1(t_1) - B_bv_M(t_{M+1})$$

Then we get a system of nM linear equations for $s^T = (s_1^T, \dots, s_M^T)$, which is

$$As = b, \tag{3.4}$$

with $b^T = (\beta_1^T, \dots, \beta_M^T)$

$$A = \begin{bmatrix} \Phi_1(t_2) & -\Phi_2(t_2) & & & \\ & \Phi_2(t_3) & -\Phi_3(t_3) & & \\ & & \cdots & \cdots & \\ & & & \Phi_{M-1}(t_M) & -\Phi_M(t_M) \\ B_a\Phi_1(t_1) & & & & B_b\Phi_M(t_M) \end{bmatrix}$$

This looks like the well-known multiple shooting method, which is why it is called the theoretical multiple shooting method. The difference between them is that one does not numerically integrate the sub BVP (generally it is not even an initial value problem).

Stability and error analysis

Let $\Phi(x)$, $G(x,t)$ be a fundamental solution and the Green's function of (3.1). We have the following relation:

$$\Phi_j(x) = \Phi(x)\left[B_{1j}\Phi(t_j) + B_{2j}\Phi(t_{j+1})\right]^{-1}. \qquad 1 \le j \le M \qquad (3.5a)$$

$$G_j(x,t) = G(x,t) - \Phi_j(x)\left[B_{1j}G(t_j,t) + B_{2j}G(t_{j+1},t)\right]. \qquad (3.5b)$$

$$v_j(x) = \int_{t_j}^{t_{j+1}} G_j(x,s)q(s)ds. \qquad (3.5c)$$

$$s_j = \left[B_{1j}\Phi(t_j) + B_{2j}\Phi(t_{j+1}) \right]\{ \sum_{k=1}^{M-1}\Phi^{-1}(t_j)G(t_j,t_k)\beta_k + \beta_M \}. \qquad (3.5d)$$

If we define the local condition constants

$$\kappa_{1j} = \| \Phi_j \|_{[tj,tj+1]},$$

$$\kappa_{2j} = \| G_j \|_{[tj ,tj+1]}$$

then we have

$$\kappa_{2j} \le \kappa_2(1 + 2\kappa_{1j}).$$

This means that if the original BVP (3.1) is well-conditioned (κ_2 is of moderate size), and if we choose local BCs for the sub BVPs properly to make κ_{1j} of moderate size, then the sub BVPs will be well-conditioned.

Suppose the BVP (3.1) has dichotomic structure (2.25) with

$$P = \begin{bmatrix} 0 & 0 \\ 0 & I_k \end{bmatrix}.$$

Then $\Phi(x) = (\Phi^1(x) \mid \Phi^2(x))$ with $\Phi^1(x) \in R^{n\times k}$ and $\Phi^2(x) \in R^{n\times(n-k)}$ denote the nondecreasing and nonincreasing modes respectively. Let $Q_{1j} \in R^{n\times k}$ and $Q_{2j} \in R^{n\times(n-k)}$ be two matrices with orthonormal columns such that

$$Q_{1j}^T \Phi^1(t_j) = 0, \qquad Q_{2j}^T \Phi^2(t_{j+1}) = 0. \tag{3.6}$$

Then defining

$$B_{1j} = \begin{bmatrix} 0 \\ Q_{1j}^T \end{bmatrix}, \qquad B_{2j} = \begin{bmatrix} Q_{2j}^T \\ 0 \end{bmatrix}, \tag{3.7}$$

we have

$$\kappa_{1j} \le 2K, \qquad \kappa_{2j} \le K.$$

See [7,8] for a proof. The result above can be summarized as a theorem:

Theorem 3.8 Suppose that the BVP (3.1) is well conditioned (κ_2 is of moderate size) and has dichotomic structure (2.25). If the local BC for sub BVP (3.3) is chosen as in (3.7), then the following hold:

(i) The sub BVP (3.3) are well-conditioned with $\kappa_{1j} \le 2K$ $\kappa_{2j} \le K$

(ii) The theoretical multiple shooting method is stable: there is a moderate size constant $K_1 = 4\kappa K$ such that $\text{cond}(A) \le K_1 M$

(iii) The vector s is bounded in terms of the original data by

$$|s| \le \kappa\left[\|\beta\| + 2\kappa(1+4\kappa)\|q\|_1\right]$$

Up to now, all the quantities in the theoretical multiple shooting method are exact. Suppose $\Phi_j^h(x)$, $v_j^h(x)$ are approximations to $\Phi_j(x)$ and $v_j(x)$ respectively, and s^h solves $A^h s = b^h$. Then the numerical solution $y^h(x)$ is given by

$$y^h(x) = \Phi_j^h(x)s_j^h + v_j^h(x), \qquad t_j \le x \le t_{j+1}, \quad 1 \le j \le M$$

This process of approximation (depending on the numerical method for (3.3) and mesh point t_j) may be called approximate theoretical multiple shooting. If we know the error of the approximations $\Phi_j^h(x)$ and $v_j^h(x)$, we can obtain a localized error estimate for $y^h(x)$ and s^h.

Theorem 3.9 In addition to the assumptions of theorem 3.8, suppose that there are constants $\delta_1, \delta_2 > 0$, such that

$$\| \Phi_j^h(t_j) - \Phi_j(t_j) \|, \quad \| \Phi_j^h(t_{j+1}) - \Phi_j(t_{j+1}) \| \le \delta_1 \tag{3.9a}$$

$$2\kappa M\delta_1 =: \gamma < 1 \tag{3.9b}$$

$$\| v_j^h(t_j) - v_j(t_j) \|, \quad \| v_j^h(t_{j+1}) - v_j(t_{j+1}) \| \le \delta_2 . \tag{3.9c}$$

Then the A^h, s^h, $y^h(x)$ is well defined and

$$\| [A^h]^{-1} \| \le \frac{KM}{1-\gamma}$$

$$| s^h - s | \le \frac{1}{1-\gamma}(2KM\delta_2 + \gamma|s|)$$

$$| y^h(x) - y(x) | \le \frac{2K}{1-\gamma}(2KM\delta_2 + \gamma|s|) + |(\Phi_j^h(x) - \Phi_j(x))s_j^h|$$

$$+ | v_j^h(t_j) - v_j(t_j) | \quad t_j \le x \le t_{j+1} \quad 1 \le j \le M$$

This theorem guarantees that approximate multiple shooting is well defined and stable. It encompasses different numerical schemes with different meshsizes in different segments. To get a uniformly accurate approximate solution on [a,b], we

must have a fine mesh in layer regions with $h_i = O(\varepsilon)$, while in the smooth regions $h_i >> \varepsilon$ is enough. If $\Phi_j(x)$ is not approximated well at mesh points of a segmentation $[t_j, t_{j+1}]$ with a smooth solution, then in general (3.9a,b) do not hold. In this case, one should consider a three-way splitting of modes into rapidly increasing, rapidly decreasing and slow ones. For details of the stability framework, one may refer to section 10.2.3 of [8].

3.2 Result of KNB

KNB[3] gave a practical procedure to construct the segmentation (3.2). Based on this segmentation, they derived a mesh for a difference method. On each mesh point they use either implicit Euler method or the trapezoidal rule. This combination method can deal with stiff problems with boundary layers and interior layers.

Mesh construction

Division of the eigenvalues of A(x) into subsets: Since the solution modes are related to the sign and size of the eigenvalues of A(x), [3] divided the eigenvalues of A(x) into different subsets $M^{(j)}$, where in each subset $M^{(j)}$, the eigenvalues are of the same magnitude. This can be done as follow: Let K, $\delta>0$ with $0 \le Kh << 1$ be constants. Then $\lambda \in M^{(0)}$ if either $|\lambda| \le K$ or there exists a $\tilde{\lambda} \in M^{(0)}$ such that

$$\big|\, |\tilde{\lambda}| - |\lambda| \,\big| \le \delta(|\lambda| + |\tilde{\lambda}|)\,. \tag{3.10}$$

By choosing δ sufficiently small, all $\lambda \in M^{(0)}$ can satisfy $|h\lambda| << 1$. If all eigenvalues $\lambda \in M^{(0)}$, then the construction of $M^{(j)}$ is done. Otherwise let $\lambda_1, ..., \lambda_m$ be the remaining eigenvalues, and let $|\lambda_j| = \min_{1\le v\le m} |\lambda_v|$. Then the set $M^{(1)}$ can be formed by taking $\lambda_j \in M^{(1)}$, $\lambda \in M^{(1)}$ if $Re(\lambda_j)Re(\lambda) \ge 0$ and there is a $\tilde{\lambda} \in M^{(1)}$ such that (3.10) holds. This can be done recursively until each eigenvalue of A(x) is in one

subset. The number of elements in $M^{(j)}$ depends on x. $M^{(j)}$ may have different numbers of elements at different points.

Since λ is continuous with respect to the elements of A(x), we can assume that λ is continuous with respect to x. Thus we can further divide the interval [a, b] into a finite number of subintervals: $c_i \le x \le c_{i+1}$ such that on each subinterval the number of elements of $M^{(j)}$ is constant. This process is referred to as blocking subintervals by KNB[3]. This segmentation is not fine enough to be the segmentation of (3.2). To refine it, KNB[3] transform A(x) into block diagonal form. In intervals where the solution is not smooth, it is refined by stretching the variable x.

Transform A(x) into block diagonal form: This step is to find a transformation S(x) such that

$$\tilde{A}(x) = S^{-1}(x)A(x)S(x) = \begin{bmatrix} A_r(x) & & & \\ & A_{r-1}(x) & & \\ & & \cdots & \\ & & & A_0(x) \end{bmatrix}$$

is in block diagonal form, and the eigenvalues of $A_j(x)$ are exactly the eigenvalues in $M^{(j)}$. The construction procedure of S(x) is as follows:

(i) Find a unitary matrix U(a) (by QR method) such that

$$U^H(a)A(a)U(a) = \begin{bmatrix} A_r & A_{r\,r-1} & \cdots & A_{r\,0} \\ & A_{r-1} & \cdots & A_{r-10} \\ & & \cdots & \cdots \\ & & & A_0 \end{bmatrix}$$

(ii) Find $\tilde{S}(a)$ such that

$$\tilde{S}(a) = \begin{bmatrix} I & S_{r\,r-1} & \cdots & S_{r\,0} \\ & I & \cdots & S_{r-1\,0} \\ & & \cdots & \cdots \\ & & & I \end{bmatrix}$$

$$\tilde{A}(a) = S^{-1}(a)A(a)S(a) = \begin{bmatrix} A_r & & & \\ & A_{r-1} & & \\ & & \cdots & \\ & & & A_0 \end{bmatrix}$$

where $S(a) = U(a)\tilde{S}(a)$.

(iii) $\tilde{A}(x) = \tilde{A}(a) + B(x)$, $B(a) = 0$,

$$B(x) = S^{-1}(a)[\ A(x) - A(a)\ ]S(a) = \begin{bmatrix} B_{r\,r} & B_{r\,r-1} & \cdots & B_{r\,0} \\ B_{r-1\,r} & B_{r-1r-1} & \cdots & B_{t-1\,0} \\ \cdots & \cdots & \cdots & \cdots \\ B_{0\,r} & B_{0\,r-1} & \cdots & B_{0\,0} \end{bmatrix}$$

(iv) By using an algebraic Riccati transformation (see KNB[3] for details), one can construct $\tilde{S}(x)$, such that

$$S^{-1}(x)A(x)S(x) = \begin{bmatrix} A_r & & & \\ & A_{r-1} & & \\ & & \cdots & \\ & & & A_0 \end{bmatrix}, \qquad \text{where } S(x) = S(a)\,\tilde{S}(x).$$

Up to now, S(x) has been constructed in a neighbourhood of x=a. One can continue the construction as long as the block structure does not change, say for $a \le x \le c_1$. Letting $S_-(c_1) = \lim S(x)$ at c_1, one can change S(x) from $S_-(c_1)$ to $S_+(c_1)$ in the following way:

(1) If two sets of $M^{(j)}$ merge, S does not change,

(2) If set $M^{(j)}$ splits into subsets, $S_+(c_1)$ can be computed in the same way as S(a).

One can construct S(x) for $[c_1, c_2]$, and so on. This completes the construction of S(x). Now, one gets a new system of ODEs on each subinterval $[c_i, c_{i+1}]$.

$$\frac{d\tilde{y}}{dx} = \hat{A}(x)\tilde{y} + H(x)\tilde{y} + G(x) \qquad (3.11)$$

where $$\hat{A}(x) = \begin{bmatrix} A_r(x) & & \\ & \cdots & \\ & & A_0(x) \end{bmatrix}$$

$$H(x) = - S^{-1} \frac{dS(x)}{dx}, \qquad G(x) = S^{-1}F(x), \qquad \tilde{y} = S^{-1}y.$$

Stretching variable: In one blocking subinterval, the smoothness property of the solution may still be different. One needs to refine the blocking subinterval $[c_i, c_{i+1}]$ further. Suppose $[c_i, c_{i+1}]$ is being divided into $s \ge 1$ stretching subintervals:

$$c_{ij} \le x \le c_{ij+1}, \quad j = 0, 1, \ldots, s-1, \text{ with } c_i = c_{i0} < c_{i1} < \ldots < c_{is} = c_{i+1}.$$

If $c_{i0}, c_{i1}, \ldots, c_{ij}$ have been determined, then c_{ij+1} is determined as follows: Let $\tilde{x}$ be a new stretching variable such that $x - c_{ij} = \alpha_{ij}\tilde{x}$, $0 \le \tilde{x} \le 1$, and the ODE (3.11) becomes:

$$\frac{d\tilde{y}}{d\tilde{x}} = \alpha\hat{A}(\alpha\tilde{x})\tilde{y} + \alpha H(\alpha\tilde{x})\tilde{y} + \alpha G(\alpha\tilde{x}) \qquad (3.12)$$

where $\alpha = \alpha_{ij}$ with $0 < \alpha \le c_{i+1} - c_{ij}$ (an approximation to) the largest value satisfying:

$$\max_{0\le x\le 1} \left(\frac{1}{\alpha|A_j| + 1} \left| \alpha \frac{d^v A_j}{d\tilde{x}^v} \right| \right) \le K , \tag{3.13a}$$

$$\max_{0\le x\le 1} \left| \alpha \frac{d^v H}{d\tilde{x}^v} \right| \le K , \tag{3.13b}$$

$$\max_{0\le x\le 1} \left\{ \left[\max(|\alpha G^{(j)}|, 1) \right]^{-1} \left| \alpha \frac{d^v G^{(j)}}{d\tilde{x}^v} \right| \right\} \le K \tag{3.13c}$$

$$j = 0, 1, 2, \ldots, r, \qquad v = 0, 1, 2, \ldots, p ,$$

$$\frac{|\mathrm{Im}\ \alpha\lambda(x)|}{|\mathrm{Re}\alpha\lambda(x)| + C_1/\rho} \le \rho \qquad \text{for all eigenvalues } \lambda(x) \text{ of } A(x) . \tag{3.13d}$$

Once α has been determined, we can set $c_{ij+1} = c_{ij} + \alpha$. Here (3.13d) guarantees that the problem is not highly oscillatory. The other conditions guarantee that the solution of (3.12) is smooth on the new stretching interval.

If $c_{ij+1} < c_{i+1}$, this procedure can be repeated until the endpoint c_{i+1} is reached. This procedure can be repeated until all blocking subintervals are divided into an appropriate number of stretching intervals. The new mesh of [a, b] can be denoted by (3.2). On every subinterval, the smoothness property is the same, so one can use a uniform meshsize h with $Kh << 1$ and employ the combination difference approximation (KNB[3]) using the Euler scheme or trapezoidal rule according to the size of $|ha_{ii}|$ at mesh point t_j, or one can employ collocation method or any other robust methods on each subintervals.

Practical considerations

In practice, the blocking and stretching subintervals are determined simultaneously. Stretching with the left end point x = a =t_1, and working to the right, [a,b] can be divided into stretching subintervals with end points t_1, t_2, ... The stretching parameter for $[t_j, t_{j+1}]$ is denoted by α_j. The block structure of the coefficient matrix is monitored as one proceeds, and appropriate points t_j are designated as blocking endpoints of subintervals when the structure changes. The mesh given here is a good mesh for difference schemes. Since at every mesh point, it is required to compute the eigevalues of matrix A(x), so getting this mesh as an initial mesh (for programs like the COLNEW) is expensive, especially when the BVP is large.

3.3 Riccati Method

Riccati method

The BVP(3.1) can be transformed to a BVP with separated boundary condition [8,18]

$$B_1y(a) = \beta_1, \qquad B_2y(b) = \beta_2$$

where $y \in R^n$, $\beta_1 \in R^k$, $\beta_2 \in R^{n-k}$. We consider the linear BVP

$$y' = A(x)y + q(x), \quad a \le x \le b, \tag{3.14a}$$

$$B_1y(a) = \beta_1, \qquad B_2y(b) = \beta_2. \tag{3.14b}$$

We assume that the BVP has a unique solution and is well-conditioned.

Let T(x) be a linear transformation of the form

$$T(x) = \begin{bmatrix} I & 0 \\ R(x) & I \end{bmatrix}$$

with R(x) being an (n-k)×k matrix to be determined later. Define

$$y(x) = T(x)w(x)\ . \tag{3.15}$$

Then $w' = U(x)w + g(x)$

where U(x) , T'(x) and g(x) satisfy

$$T' = AT - TU, \qquad g(x) = T^{-1}(x)q(x)$$

i.e. $w_1 = y_1, \qquad w_2 = -R(x)y_1 + y_2,$

$$g_1 = q_1, \qquad g_2 = -R(x)q_1 + q_2$$

where

$$y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}, \qquad w = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}, \qquad q = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}, \qquad g = \begin{bmatrix} g_1 \\ g_2 \end{bmatrix}$$

with $y_1, w_1, q_1, g_1 \in R^k$, $y_2, w_2, q_2, g_2 \in R^{(n-k)}$

If we require that U be " block upper triangular " corresponding to the dimension of R(x), i.e.

$$U(x) = \begin{bmatrix} U_{11}(x) & U_{12}(x) \\ U_{21}(x) & U_{22}(x) \end{bmatrix}$$

with $U_{11} \in R^{k\times k}$, $U_{22} \in R^{(n-k)\times(n-k)}$, $U_{12} \in R^{k\times(n-k)}$, $U_{21} = 0 \in R^{(n-k)\times k}$,

this gives ***the Riccati differential equation*** for R(x)

$$R' = A_{21} + A_{22}R - RA_{11} - RA_{12}R. \tag{3.16}$$

The block form of U(x) is given by

$$U = \begin{bmatrix} A_{11}+A_{12}R & A_{12} \\ 0 & A_{22}-RA_{12} \end{bmatrix}$$

The transformed ODE can be written in the decoupled form:

$$w_1' = U_{11}(x)w_1 + U_{12}(x)w_2 + g_1(x), \tag{3.17a}$$

$$w_2' = U_{22}(x)w_2 + g_2(x). \tag{3.17b}$$

Now if we know a proper initial condition for (3.16) we can solve it via an IVP solver. Further, if we know the initial value for (3.17b), we can get the solution of w_2, and then the solution of w_1 via (3.17a). Finally, we can get the solution of y(x). This gives an outline of the Riccati method. For BVP (3.14), if we assume that $B_1 = (B_{11} \mid B_{12})$ with B_{12} being a nonsingular k×k matrix then the boundary condition $B_1y(a) = \beta_1$ yields

$$B_{11}y_1(a) + B_{12}y_2(a) = \beta_1.$$

From (3.15) we know that $w_2(a) = -R(a)y_1(a) + y_2(a)$,

and if we choose $R(a) = -B_{12}^{-1}B_{11}$, (3.18)

then $w_2(a) = B_{12}^{-1}\beta_1$. (3.19)

Upon making this choice, we can find R(x) and $w_2(x)$. After finding R(b), $w_2(b)$, we can find $w_1(b)$ from the boundary condition (3.14b), and then solve (3.17a).

If we transform the ODE(3.14a) to a lower block triangular system, we can analogously define a linear transformation:

$$T(x) = \begin{bmatrix} I & S(x) \\ 0 & I \end{bmatrix}, \qquad y(x) = T(x)z(x)$$

with S(x) being a k×(n−k) matrix to be determined later. Then

$$z' = V(x)w + f(x)$$

where V(x) , T'(x) and g(x) satisfy

$$T' = AT - TV, \qquad f(x) = T^{-1}(x)q(x)$$

i.e. $z_1 = y_1 - Sy_2$, $z_2 = y_2$,

$f_1 = q_1 - Sq_2$, $f_2 = q_2$,

where $y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$, $z = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$, $q = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}$, $f = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}$

with $y_1, z_1, q_1, f_1 \in R^k$, $y_2, z_2, q_2, f_2 \in R^{(n-k)}$.

The requirement that V(x) be lower block triangular yields the differential Riccati equation

$$S' = A_{12} + A_{11}S - SA_{22} - SA_{21}S, \quad b \geq t \geq a, \tag{3.20}$$

$$S(b) = - B_{21}^{-1}B_{22}$$

and for particular integral z(t)

$$z_1' = (A_{11} - SA_{21})z_1 + f_1(x) \qquad b \geq t \geq a,$$

$$z_1(b) = B_{21}^{-1}\beta_2,$$

$$z_2' = (A_{22} + A_{21}S)z_2 + A_{21}z_1 + f_2(x),$$

$$z_2(a) = [B_{12} + B_{11}S(a)]^{-1}[\beta_1 - B_{11}z_1(a)].$$

Properties of Riccati method

Provided the BVP is well-conditioned, from theorem 3.107 in [8] we know the range of $\begin{bmatrix} I \\ R(a) \end{bmatrix}$ should induce initial values for nonincreasing modes only. Hence the choice of R(a) gives a satisfactory initial value for $w_2(a)$ as well as a stable algorithm. The kinematic eigenvalues of $U_{11}(x)$ can be expected to have positive real parts. To analyze the properties of the Riccati method, let us consider a fundamental solution for (3.14) with the first k columns being nonincreasing modes.

$$Y(x) = \begin{bmatrix} Y_{11}(x) & Y_{12}(x) \\ Y_{21}(x) & Y_{22}(x) \end{bmatrix} = [Y_+(x), Y_-(x)] \tag{3.21}$$

such that

$$\text{Range}\begin{bmatrix} Y_{11}(a) \\ \\ Y_{21}(a) \end{bmatrix} = \text{Range}\begin{bmatrix} I \\ \\ R(a) \end{bmatrix}$$

For stability of the Riccati method we need

$$\text{Range}\begin{bmatrix} Y_{11}(x) \\ \\ Y_{21}(x) \end{bmatrix} = \text{Range}\begin{bmatrix} I \\ \\ R(x) \end{bmatrix}.$$

This implies that $Y_{11}(x)$ must be nonsingular, and

$$R(x) = Y_{21}(x)[Y_{11}(x)]^{-1}$$

Analogously, if [$Y_{-}(x)$] are nondecreasing modes with $Y_{22}(x)$ nonsingular, then

$$S(x) = Y_{12}(x)[Y_{22}(x)]^{-1}$$

This links the stability question of the method with the feasibility of integrating the nonlinear Riccati equation (3.16) starting with (3.18). If $[Y_{11}(x)]^{-1}$ is bounded, the Riccati method is stable [8, chapter 10, section 10.4.2]. If difficulties arise, we can detect it when integrating for R(x). In general, difficulties in integrating the Riccati equation may certainly occur, i.e. $Y_{11}(x)^{-1}$ becomes unbounded or $Y_{11}(x)$ becomes singular. Since Y(x) is nonsingular over [a,b], $Y_{+}(x)$ has k linearly independent rows at any $x \in [a,b]$, and we can reorder Y(x) at a trouble point to make the new $Y_{11}(x)$ nonsingular. This idea sometimes is referred to as "reimbedding" [5,6,8,15], and the reordering is corresponding to permutation of the original BVP. This idea can be put into practical use because of the following result due to Taufer [28], and we restate it here in the form given by [5].

Theorem 3.22 If Y(x) is fundamental solution (3.21) with

$$Y(a) = \begin{bmatrix} I_k & Y_{12}(a) \\ R(a) & Y_{22}(a) \end{bmatrix}$$

then there exist a finite set of open intervals $\{ I_h \}_{h=1}^{N}$, where $\cup_{h=1}^{N} I_h$ cover [a,b] ($a \in I_1$, $b \in I_N$) and $I_h \cap I_{h+1} \neq \emptyset$, h=1, 2, . . . N-1 and such that on each subinterval I_h there is a permutation matrix P^h for which

$$Y^h(x) = P^h Y(x) = [\, Y_+^h(x),\ Y_-^h(x) \,] = \begin{bmatrix} Y_{11}^h(x) & Y_{12}^h(x) \\ Y_{21}^h(x) & Y_{22}^h(x) \end{bmatrix}, \qquad t \in I_h$$

where $Y_{11}^h(x)$ is nonsingular.

Mesh from DRE

From the discussion of the previous section, we know the solution of the differential Riccati equation is closely related to the fundamental solution of (3.14a). There is no doubt that the variation of the solution of (3.14) can be reflected somehow by the solutions of differential Riccati equations (3.15, 3.20), i.e. when the solutions of the differential Riccati equations R(x), S(x) are smooth, the solution of (3.14) is smooth, and when the solutions R(x), S(x) vary fast, the solution of (3.14) possibly has a fast variation. Since R(x), S(x) are solutions of initial value problems, they are easy to obtain using an IVP solver. While solving R(x), we get a mesh as described below

$$a = r_1 < r_2 < \ldots < r_M = b \tag{3.23}$$

generated by the IVP solver. The mesh for getting S(x) through the IVP solver is

$$a = s_1 < s_2 < \ldots < s_N = b \tag{3.24}$$

The union of (3.23) and (3.24) is denoted as

$$a = u_1 < u_2 < \ldots < u_L = b \tag{3.25}$$

We may refer these meshes as differential Riccati equation meshes or DRE meshes for short. For these DRE meshes, we have the following strategies:

(i) Meshes 3.23, 3.24, 3.25 can be used as segmentation (3.2)

(ii) We can feed the DRE meshes into programs for BVP based on a global method to get the solution of (3.14)

(iii) We can use the DRE mesh as an initial mesh for programs based on global methods.

The details of getting a DRE mesh is given in chapter 4.

4. DRE and DRE mesh

Differential Riccati Equations (DREs) are well-known matrix quadratic equations. They arise quite often in the mathematical and engineering literature [22,23,24], e.g when studying transmission line phenomena, theory of noise and random processes, variations theory, optimal control theory, diffusion problems and invariant imbedding. Regardless of the particular applications in which they arise, DREs are always the expression of a time dependent change of variables which decouples a linear system of ordinary differential equations. Given a linear system, through a proper transformation of variables there is a unique associated DRE; but there is no unique way to associate a given DRE to a given linear system. Because of this fact, we consider the DRE from the system viewpoint. In this chapter, we only consider the DRE as used in the two-point boundary value problems, although it can be used as a general decoupling tool for all ODEs. We do not emphasize the properties of the solution of DREs, neither do we emphasize the numerical methods for solving them. Rather, we focus on the mesh points which a DRE solver generates while solving the DRE. From this mesh we get a DRE mesh, and use it as an initial mesh for a global method for the corresponding BVPs.

4.1 Differential Riccati Equations

As we considered in section 3.3, for a given two-point boundary value problems with separated boundary conditions:

$$y' = \begin{bmatrix} y_1' \\ y_2' \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} + \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}, \qquad a \le x \le b \tag{4.1a}$$

$$(B_{11}, B_{12})y(a) = \beta_1, \qquad (B_{21}, B_{22})y(b) = \beta_2 \tag{4.1b}$$

where $A_{11} \in R^{k\times k}$, $A_{22} \in R^{(n-k)\times(n-k)}$, $A_{12} \in R^{k\times(n-k)}$, $A_{21} \in R^{(n-k)\times k}$, $y_1 \in R^{k}$, $y_2 \in R^{(n-k)}$, $\beta_2 \in R^{(n-k}$, $\beta_1 \in R^{k,}$, $B_{12} \in R^{(n-k)\times(n-k)}$, $B_{21} \in R^{k\times k}$, we can get a decoupled system $w' = \tilde{A}w + g(x)$ in a new variable $w=T^{-1}y$ with transformation:

$$T(t) = \begin{bmatrix} I_k & 0 \\ R(t) & I_{n-k} \end{bmatrix}, \quad R(t) \in R^{(n-k)},$$

where

$$\tilde{A} = T^{-1}(AT - T^{-1}T') = \begin{bmatrix} A_{11}+RA_{12} & A_{12} \\ & A_{22}-A_{12}R \end{bmatrix} = \begin{bmatrix} \tilde{A}_{11} & A_{12} \\ & \tilde{A}_{22} \end{bmatrix} \quad (4.2)$$

This is true if and only if R(t) satisfies the DRE:

$$R' = A_{21} + A_{22}R - RA_{11} - RA_{12}R =: F(t,R), \quad (4.3a)$$

$$R(a) = R_o \quad (4.3b)$$

From section 3.3 we know that a good choice for R_o is $R_o = -B_{12}^{-1}B_{11}$.

An important special case of (4.3) is the symmetric DRE:

$$R' = A_{21} - A_{11}^{T}R - RA_{11} - RA_{12}R, \qquad R(a) = R_o(= R_o^{T}) \quad (4.4)$$

which arises when the matrix A(t) is Hamiltonian:

$$A(t) = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & -A_{11}^{T} \end{bmatrix}, \qquad n=2k,\ A_{21}^{T} = A_{12},\ A_{21}^{T} = A_{21}^{T}$$

All solutions of (4.4) are symmetric , $R^T = R$. This special DRE is widely used in optimal control applications, and its solution has some special properties. But in this thesis, we will not give any special treatment for this DRE. We consider DRE (4.3) without assuming special structure for matrix A(t), i.e. A(t) is a continuous dense matrix.

4.2 Numerical methods for DREs and DRESOL

Numerical methods for DREs

Vector method. The typical way to integrate the DRE(4.3) numerically is to rewrite it as an k(n-k)-vector differential equation, and then to apply available initial value problem software to this differential equation. This can work well [6] as long as the dimensions of the problems are small and the DRE is not stiff. If k or n-k is large, and the DRE is stiff, and we have to use an implicit scheme, this approach becomes extremely expensive because of the frequent factorization of the Jacobian matrix which costs $O([k(n-k)]^3)$ flops at each step. This is not a promising approach.

Semi-implicit scheme. Babuska and Majer [18] proposed a semi-implicit scheme for solving DRE (4.3), which is:

$$R_{i+1/2} = \left[I_{n-k} - \frac{1}{2} h_i (A_{22}(t_m) - R_i A_{12}(t_m)) \right]^{-1} \left[R_i - \frac{1}{2} h_i (R_i A_{11}(t_m) - A_{21}(t_m)) \right]$$

$$R_{i+1} = \left[R_{i+1/2} + \frac{1}{2} h_i (A_{22}(t_m) R_{i+1/2} + A_{21}(t_m)) \right] \left[I_k + \frac{1}{2} h_i (A_{11}(t_m) + A_{12}(t_m) R_{i+1/2}) \right]^{-1}$$

where $t_m = t_{i+1/2}$ is the middle-point of interval $[t_i, t_{i+1}]$. The cost of this method is $O(k^3 + (n-k)^3)$ flops at each step. [18] reports that the scheme can handle stiff DREs successfully. However, this scheme is only of order 2. Although one can achieve high order accuracy through extrapolation, it requires an expensive stepsize selection procedure, and it does not seem suitable for exploiting the special structure of the symmetric DRE(4.4). As for the stability, it is prone to the same (superstability) problems of other implicit schemes [10].

Implicit scheme. Dieci [10] proposed an implicit scheme for solving DRE, which is to apply the backward difference formula (BDF) to the DRE (4.3):

$$R_{k+1} = \sum_{j=1}^{P-1} \alpha_j R_{k-j} + h\beta F(t_{k+1}, R_{k+1})$$

or

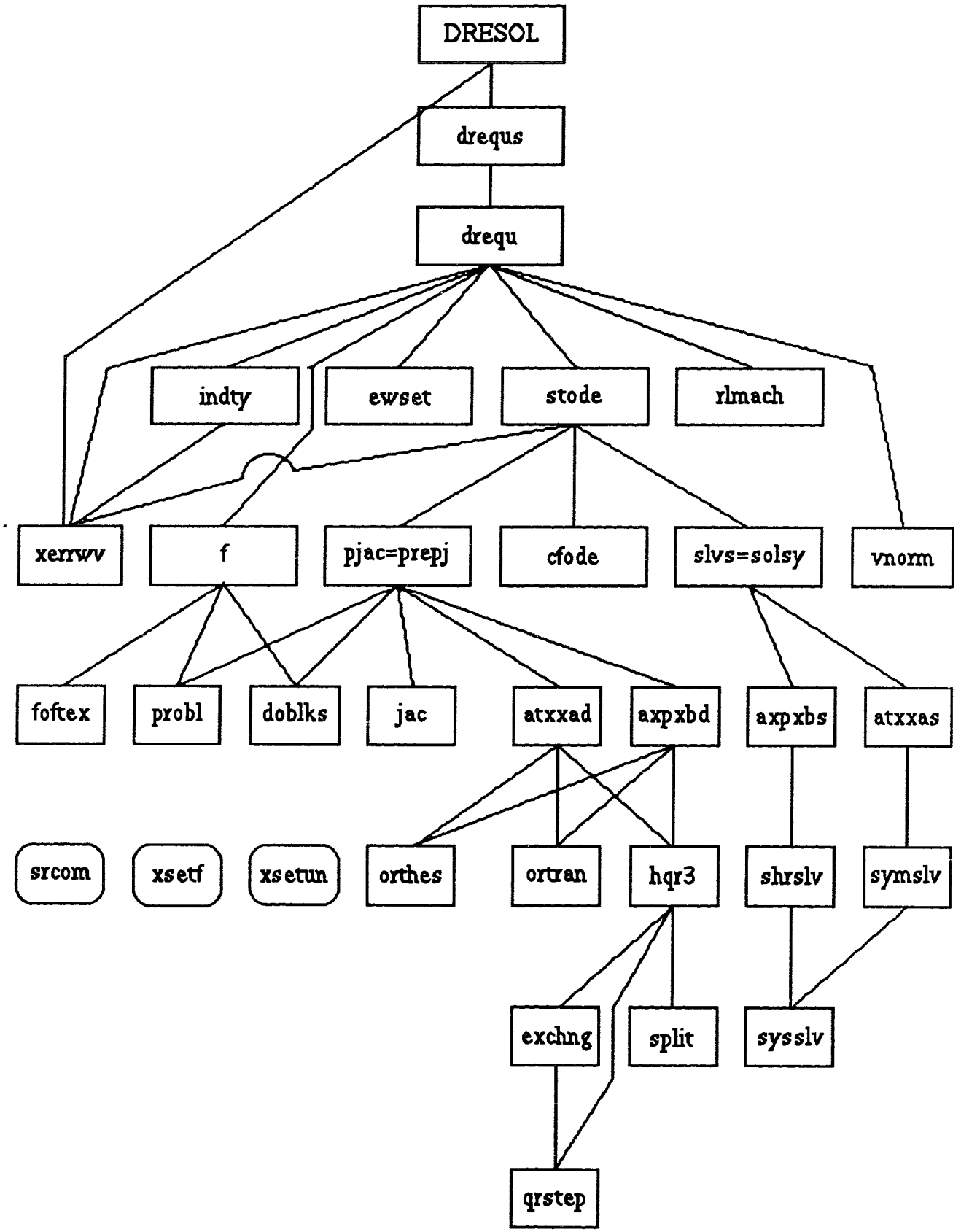
$$-R_{k+1} + h\beta(A_{21}+A_{22}R_{k+1}-R_{k+1}A_{11}-R_{k+1}A_{12}R_{k+1}) + \sum_{j=1}^{P-1} \alpha_j R_{k-j}$$

This is an algebraic Riccati equation (ARE) for R_{k+1}. This ARE can be solved by a Newton type iteration. The expense of this scheme implemented by [18] is also $O(k^3 + (n-k)^3)$, regardless of the order of the method. This idea can be adapted to the trapezoidal and implicit midpoint rules as well as other implicit integration schemes. For the symmetric DRE (4.4), this scheme can preserve the Hamiltonian structure [18].

DRESOL

DRESOL is a numerical integrator for the initial value problem of the 1st order DRE (4.3). It is an implementation of the implicit method proposed by [10]. It is written in FORTRAN77 and can integrate stiff or nonstiff DREs of symmetric and unsymmetric type. The basic IVP solver on which DRESOL based is the well-known integrator LSODE [27]. DRESOL keeps the original structure of LSODE and its criteria for order selection and local truncation error estimation (hence stepsize selection). However, DRESOL has a new linear algebra part, which keeps the problem in matrix form and solves it via efficient matrix algorithms.

The DRESOL package is a collection of subroutines for the direct numerical integration of DREs. It comprises 31 subroutines and two sets of block-data. The hierarchy chart of the subroutines is:


DRESOL
drequs
drequ
indty
ewset
stode
rlmach
xerrwv
f
pjac=prepj
cfode
slvs=solsy
vnorm
foftex
probl
doblks
jac
atxxad
axpxbd
axpxbs
atxxas
srcom
xsetf
xsetun
orthes
ortran
hqr3
shrslv
symslv
exchng
split
sysslv
qrstep

The call sequence for the solver DRESOL is:

CALL DRESOL(NEQ, NEQLEN, X, NX, Y, NY, MF, T, TOUT, RTOL, ATOL, ITASK, ISTATE, IOPT, RWORK, LREX, IWORK, LIW, PROBL, RARR, IARR)

The input parameters are: NEQ, NEQLEN, NX, NY, MF, TOUT, RTOL, ATOL, ITASK, IOPT, LREX, LIW, PROBL, RARR, IARR. The input/output parameters are: X, T, ISTATE. The working arrays Y, RWORK, IWORK, can be used for conditional input and output. To call DRESOL one has to

1. Provide a subroutine of the form SUBROUTINE PROBL (T, A, RARR, IARR) specifying the matrix A(t) in A.
2. Write a driver which calls subroutine DRESOL once for each point where a solution of R which is stored in X in DRESOL is required. Set the necessary parameters here.

For more explanation of DRESOL, see [10,12] and the documentation in the code.

4.3 Mesh from DRESOL

In this section, we discuss the DRE mesh (3.23), generated by DRESOL. We call it the simple DRE mesh. From now on, when we mention the simple DRE mesh we refer to the mesh (3.23) from DRESOL unless stated otherwise. Denote the simple DRE mesh as:

$$a = r_1 < r_2 < \ldots < r_m = b$$

We computed the simple DRE mesh for some examples in section 5.1. Generally speaking, the simple DRE is a good initial mesh for global methods for solving BVP (4.1), especially for stiff BVP with narrow layers. For stiff BVPs with narrow layers, the simple DRE mesh obtained from DRESOL with larger tolerance,

such as atol = rtol = 10^{-2}, is a better choice. Since this mesh is not well treated, it can have a few problems:

1. The DRESOL may miss the right boundary layers (see examples 8,9).
2. The simple DRE mesh may consist of too many mesh points.
3. The DRESOL may generate an artificial layer (see example 11).

Problem 1 is largely caused by the fact that the DRE involves only stable left to right integration. To recover the potential right boundary layer information, we can integrate (3.20) from right to left, i.e. from b to a. This requires setting a driver for DRE (3.20), which is tedious. An alternative way is to integrate (4.3) from right to left. Since we need only to recover the potential right boundary layer information, the right to left integration can be done for only a portion of the interval [a, b], say one tenth of the interval: [a+0.9*(b-a), b]. The union of the mesh for the right portion of the interval and the simple DRE mesh can serve as the DRE mesh.

Problem 2 is a computer dependent problem, since the maximum number of mesh points that a global BVP solver can handle depends on the machine to some degree. One choice is to pick some mesh points from the DRE mesh as a new DRE mesh which the global method can handle. The simplest choice is to pick a certain number of mesh points from the DRE mesh, say 50 mesh points.

Problem 3 is caused by improper choice of the fundamental solution of $y' = Ay$. If $Y_{11}(x)$ becomes almost singular where the BVP does not exhibit an layer, the DRE will give an artificial layer. This problem is not so important as long as the BVP is not too stiff, so the variation of R(x) in this region could not be too fast. However, if the BVP is too stiff, the DRE mesh from DRESOL may be totally misleading due to the artificial layer (see example 11). This problem can be solved by a reimbedding

strategy. Roughly speaking, the reimbedding strategy is to reorder the fundamental solution of $y' = Ay$ to make $Y_{11}(x)$ nonsingular when the magnitude of R becomes large.

4.4 More on DRE mesh

When we generate the simple DRE mesh, or the DRE mesh (with a right to left integration option), we extract the layers information for the BVP (4.1) from the DRE (4.3). While doing this we just ignore the nonhomogeneous term q(x), since the DRE (4.3) has nothing to do with q(x). However, ignoring q(x) may lose some information about the layers of the BVP (4.1). To take q(x) into account, let us recall the Riccati method discussed in section 3.3. The Riccati method solves the BVP via three initial value problems. Two of them involve integration from right to left:

DRE:

$$R' = A_{21} + A_{22}R - RA_{11} - RA_{12}R,$$

$$R(a) = - B_{12}^{-1}B_{11}$$

Particular integration:

$$v' = A_{22}v - RA_{12}v - Rq_1 + q_2,$$

$$v(a) = B_{12}^{-1}\beta_1$$

where $v = w_2$. These two initial value problems can be integrated together:

$$(R,v)' = (A_{21},g_1) + A_{22}(R,v) - (R,v)(A_{11}, g_1) - (R,v)\begin{bmatrix} A_{12} \\ 0 \end{bmatrix}(R,v), \quad (4.5a)$$

$$(R,v)_{x=a} = -B_{12}^{-1}(B_{11},\beta_1) \quad (4.5b)$$

This is a DRE corresponding to the system

$$z' = \bar{A}z, \qquad a \le x \le b \quad (4.6a)$$

$$(B_{11}, -\beta_1, B_{12})z(a) = 0, \quad (4.6b)$$

$$(B_{21}, 0 , B_{22})z(b) = \beta_2. \quad (4.6c)$$

with $z = (y_1^T, v, y_2^T)^T$, and

$$\tilde{A} = \begin{bmatrix} A_{11} & g_1 & A_{12} \\ 0 & 0 & 0 \\ A_{21} & g_2 & A_{22} \end{bmatrix}$$

We can integrate the DRE (4.5) to get a simple DRE mesh. We can also get a DRE mesh from (4.5) with the right to left integration option. Let us call this DRE mesh the Combined DRE mesh. In section 5.3, we give some numerical example for the combined DRE meshes.

Since the number of mesh points in a combined DRE mesh could be as many as 3000, it is more than sufficient. If we pick up some mesh points, say no more than 50, from the combined DRE mesh to form a sub DRE mesh, we call it a Trimmed DRE mesh. We give some numerical example of trimmed DRE meshes in section 5.3, which shows that for a proper number of mesh points, the trimmed DRE mesh is the mesh we desire. Here the points of the trimmed DRE mesh we obtained is equally distributed among the combined DRE mesh. One idea that has not been tried is that the mesh points of trimmed DRE mesh is distributed among the combined DRE mesh according to some density function.

5. Numerical Examples

This chapter consists of some numerical examples. Examples of the simple DRE mesh are given in section 5.1. Section 5.2 consists of examples of the combined mesh. The examples of the trimmed mesh are presented in section 5.3. The DRE meshes were obtained with the single precision FORTRAN77 code DRESOL. The solutions of BVPs were generated with the double precision FORTRAN IV code COLNEW. All computations were performed on SPARC STATIONS at Simon Fraser University.

5.1 Simple DRE mesh

The examples in this section can be divided into 4 groups. Examples 1 to 4 are BVPs with smooth solutions. For this kind of problems, most global methods work well, and there is basically no merit to getting the DRE mesh from DRESOL. Examples 5 to 7 are stiff BVPs. For ε not too small, COLNEW (and other global methods) can work well. For small ε (say $\varepsilon = 10^{-6}$), the layer region is narrow, the variation is fast, and COLNEW cannot work as desired with a uniform initial mesh. If we get an initial mesh for COLNEW from DRE (which is the simple DRE mesh in this section), then COLNEW works well. Examples 8 to 10 are stiff BVPs with right boundary layers. For these three examples, the simple DRE mesh missed the right boundary layer. Example 11 to 12 are stiff BVPs, the DRE mesh for these two examples consists of an artificial left boundary layer.

The computation for each example was summarized in the corresponding table. ε (or b) is the parameter in the BVP which is given in the first row of the table. The row labeled by ***atol*** = ***rtol*** is the tolerance used in computations (to get the simple

DRE mesh and to find the solution of the BVP). In the row of ***COLNEW***, the mesh sequence generated by COLNEW with a uniform initial mesh (10 subintervals) is given. The row of ***DREmesh + colnew*** gives the mesh sequence generated by COLNEW with the simple DRE mesh as initial mesh. ***DRE mesh double*** means that COLNEW computes the solution on the mesh points of the simple DRE mesh and the doubled DRE mesh. The row of ***cpu*** gives the cpu time for each computation and the estimated error in the solution ***y-err*** is provided by COLNEW for each run. The dot line in the summary tables means that there is no information available.

Example 1. Consider the BVP

$$u'' - (1+t^2)u = 0, \qquad 0 < t < b$$

$$u(0) = 1, \qquad u(b) = 0.$$

This is example 1 of [6]. This BVP has the exact solution

$$y(t) = \exp(t^2/2)(1-\mathrm{erf}(t)/\mathrm{erf}(b)).$$

It is smooth throughout the entire interval. The reduction $y = (u', u)^T$ gives DRE

$$R' = 1 - (1+t^2)R^2, \qquad R(0) = 0.$$

The simple DRE gives correct layer information of the BVP.

Table 1

b	5	5	5	10	10	10
atol=rtol	10^{-2}	10^{-4}	10^{-6}	10^{-2}	10^{-4}	10^{-6}
COLNEW	10,20	10,20	10,20	10,5,10	10,5,10	10,5,10
cpu	0.30	0.31	0.32	0.26	0.26	0.26
y-err	0.36d-8	0.36d-8	0.36d-8	0.15d-6	0.15d-6	0.15d-6
DREmesh +colnew	17,9,18	45,23,46	92,46,92	20,40	59,30,60	98,49,98
cpu	0.44	1.07	2.13	0.6	1.38	2.30
y-err	0.12d-8	0.40d-11	0.60d-13	0.16d-6	0.26d-11	0.89d-13
DREmesh double	17,34	45,90	92,184	20,40	59,118	98,196
cpu	0.5	1.26	2.53	0.6	1.65	2.73
y-err	0.78d-8	0.40d-11	0.15d-12	0.16d-8	0.12d-10	0.33d-12

Figure 1.1 Solution

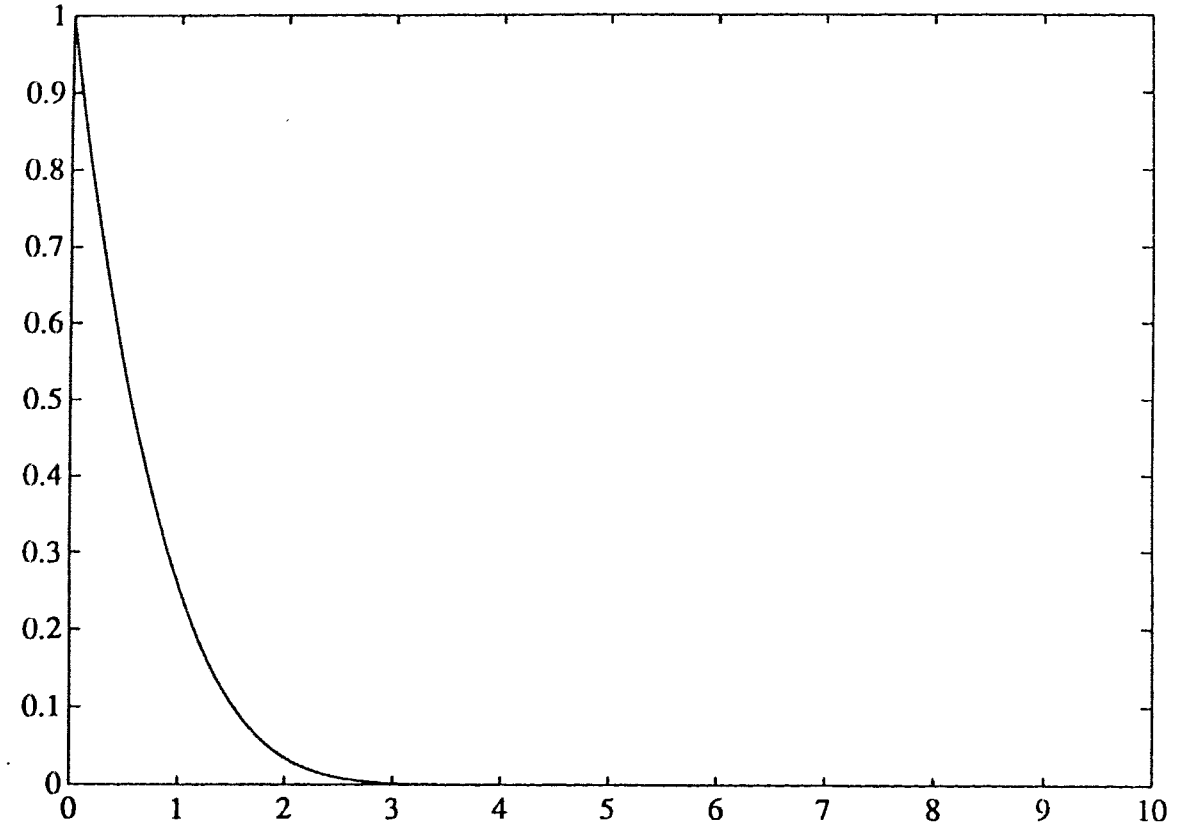
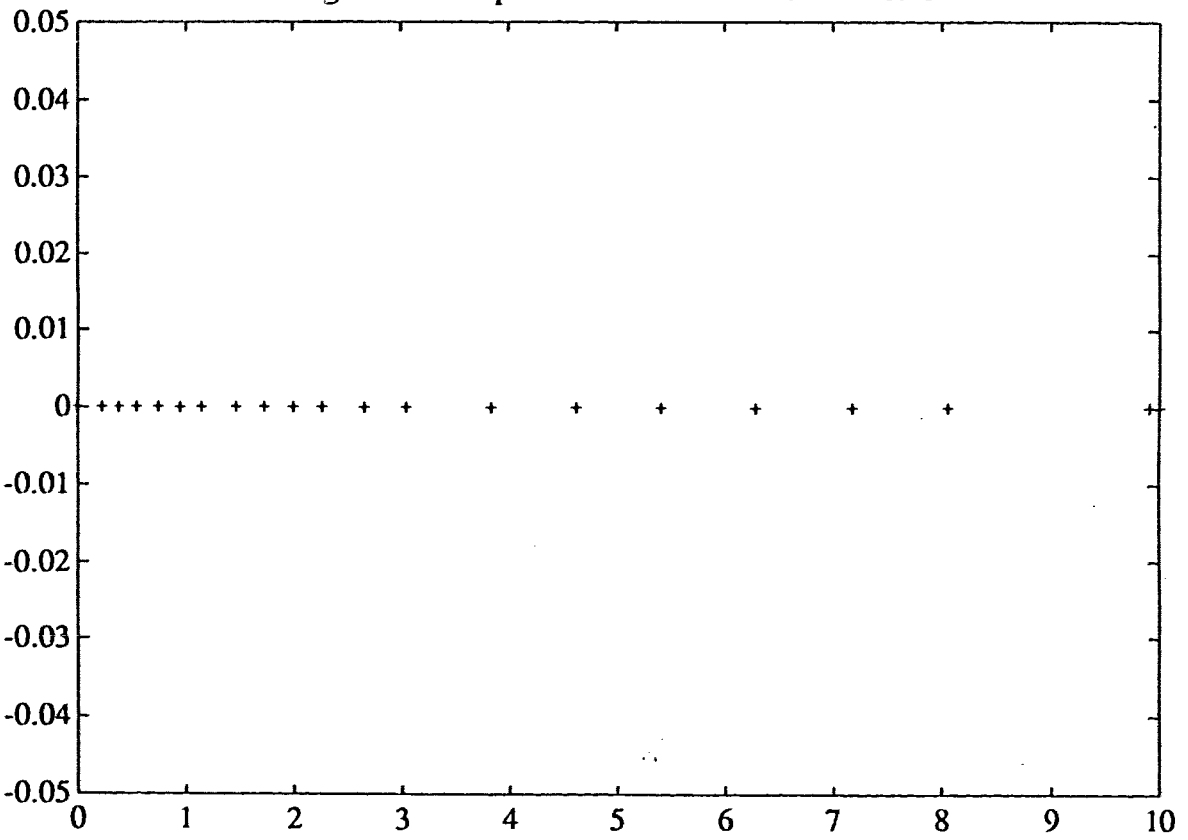


Figure 1.2 Simple DRE mesh: atol=1.e-2 T=0.20



Example 2. Consider the BVP

$$u''' + \frac{1}{t}u'' - \frac{1}{t^2}u' = \frac{1}{t}, \qquad 1 < t < 2$$

$$u''(1) + 0.3u'(1) = 0,$$

$$u''(2) + 0.15u'(2) = 0,$$

$$u(2) = 0.$$

This is example 1 of [15]. This BVP has the exact solution

$$u(t) = \frac{t^2}{4}\ln t - (\frac{1}{3}\ln 2 + \frac{33}{104})t^2 - \frac{26}{21}\ln 2 \ln t + \frac{33}{26} + \frac{1}{3}\ln 2 + \frac{26}{21}(\ln 2)^2$$

It is smooth throughout the whole interval. The DRE is given by the reduction $y = (u, u', u'')^T$. The simple DRE mesh gives correct layer information of the BVP.

Table 2

atol=rtol	10^{-2}	10^{-4}	10^{-6}
COLNEW	10,20	10,20	10,20
cpu	0.62	0.60	0.58
y-err	0.48d-10	0.48d-10	0.48d-10
DRE mesh +colnew	8,16	20,40	33,17,34
cpu	0.49	1.19	1.56
y-err	0.40d-9	0.14d-11	0.25d-12
DRE mesh double	8,16	20,40	33,66
cpu	0.51	1.14	1.88
y-err	0.40d-9	0.14d-11	0.26d-12

Figure 2.1 Solution

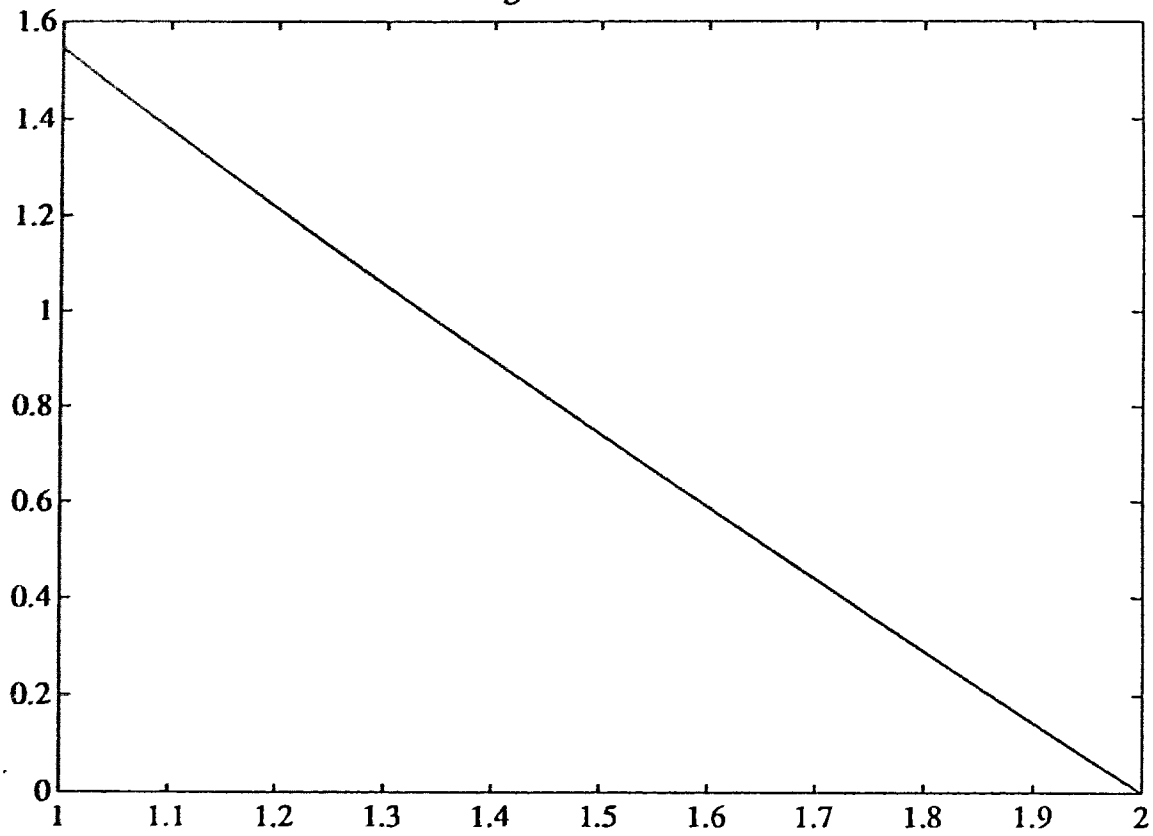
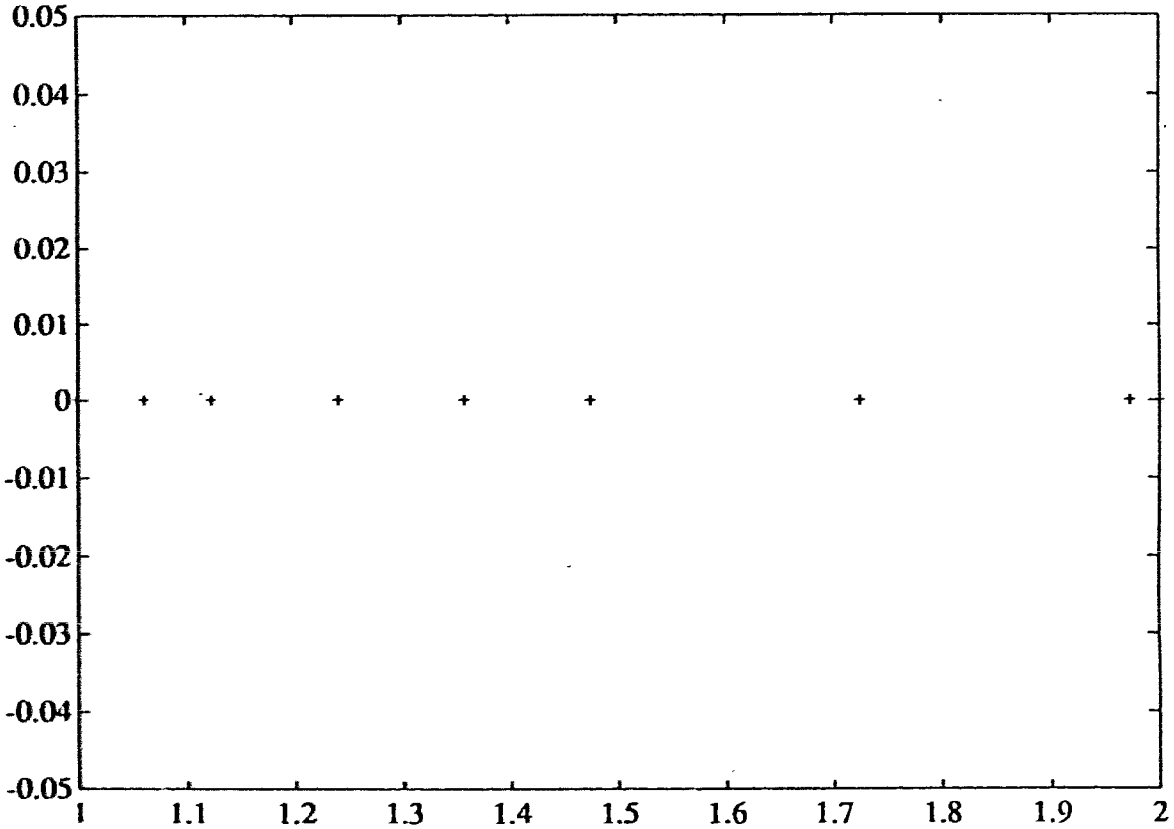


Figure 2.2 Simple DRE mesh: atol=1.e-2 T=0.13



Example 3. Consider the BVP

$$u^{(4)} = (t^4 + 14t^3 + 49t^2 + 32t - 12)\exp(t), \qquad 0 < t < 1$$

$$u(0) = u'(0) = 0,$$

$$u(1) = u'(1) = 0.$$

This is example 2 of [15]. This BVP has the exact solution:

$$u(t) = t^2(1-t)^2\exp(t)$$

It is smooth throughout the whole interval. The DRE is given by the reduction $y = (u'', u''', u, u')^T$. The simple DRE mesh gives correct layer information.

Table 3

atol=rtol	10^{-2}	10^{-4}	10^{-6}
COLNEW	10,20	10,20	10,20
cpu	0.84	0.85	0.84
y-err	0.17d-9	0.17d-9	0.17d-9
DRE mesh +colnew	7,4,8	14,7,14	21,11,22
cpu	0.54	0.96	1.47
y-err	0.27d-7	0.90d-9	0.73d-10
DRE mesh double	7,14	14,28	21,42
cpu	0.60	1.16	1.73
y-err	0.61d-7	0.24d-7	0.10d-7

Figure 3.1 Solution

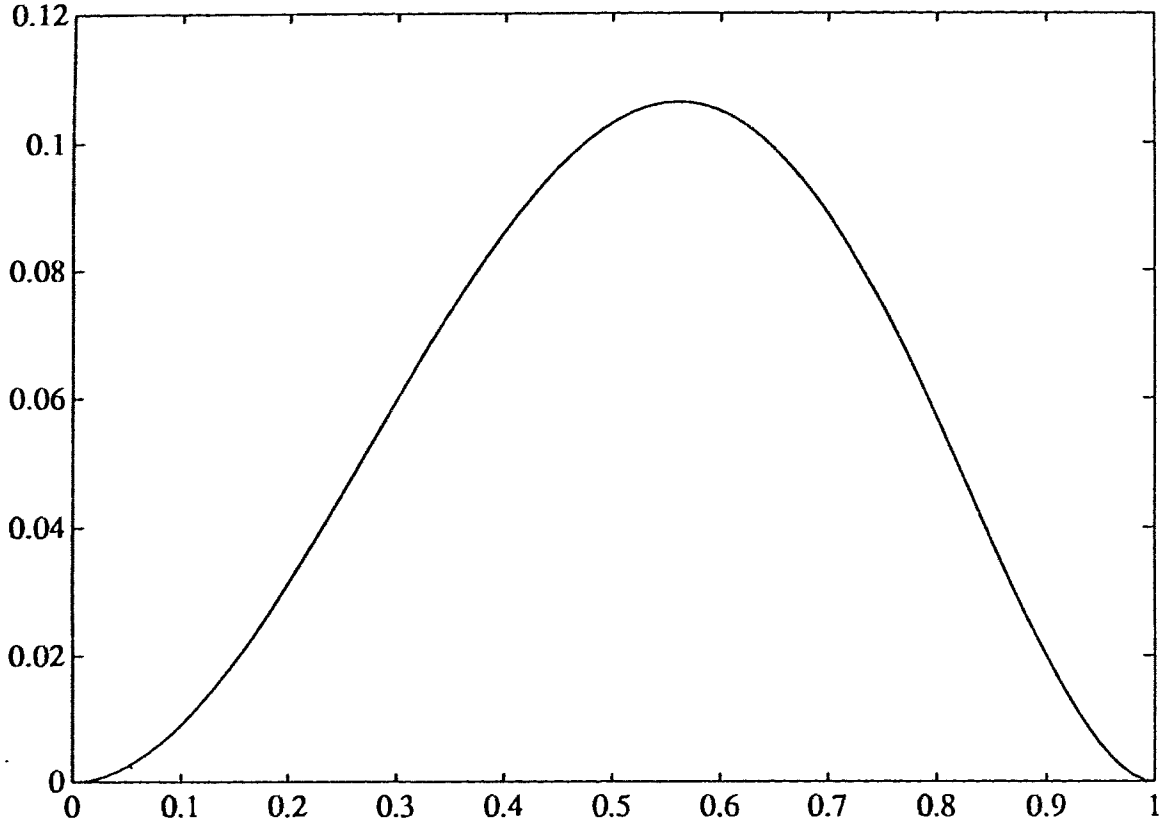
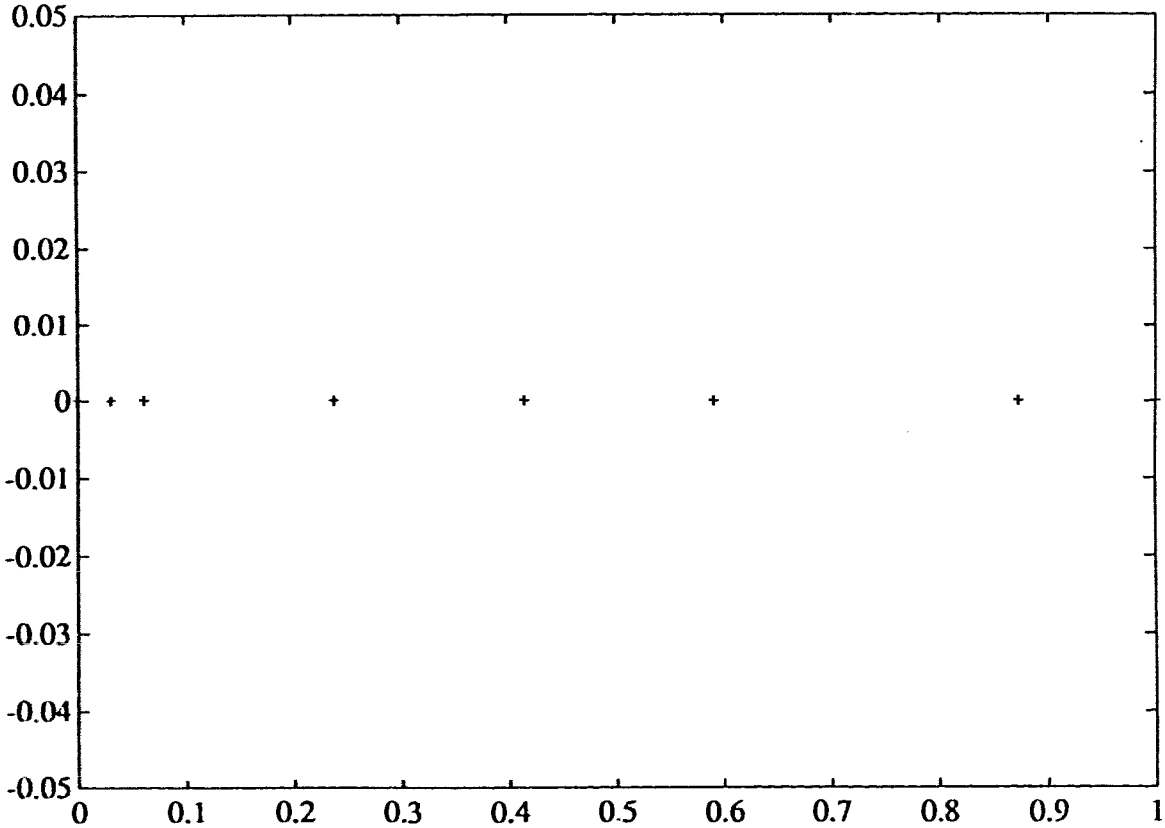


Figure 3.2 Simple DRE mesh: atol=1.e-2 T=0.13



Example 4. Consider the BVP:

$$u'' - 400u = 400\cos^2(\pi t) + 2\pi^2\cos(2\pi t), \quad 0 < t < 1,$$

$$u(0) = u(1) = 0.$$

This is example 6 of [15]. This BVP has the exact solution

$$u(t) = \frac{e^{-20}}{1+e^{-20}}e^{20t} + \frac{1}{1+e^{-20}}e^{-20t} - \cos^2(\pi t)$$

It is smooth throughout the whole interval. The reduction $y = (u', u)^T$ gives DRE

$$R' = 1 - 400R^2, \qquad R(0) = 0$$

The simple DRE mesh gives correct layer information.

Table 4

atol=rtol	10^{-2}	10^{-4}	10^{-6}
COLNEW	10,20	10,20	10,20,40
cpu	0.34	0.34	0.76
y-err	0.88d-5	0.88d-5	0.21d-6
DRE mesh+colnew	8,4,8	27,14,28	61,31,62
cpu	0.23	0.74	1.66
y-err	0.12d-2	0.11d-5	0.46d-8
DRE mesh double	8,16	27,54,108	61,122
cpu	0.28	1.96	1.89
y-err	0.13d-1	0.15d-3	0.26d-3

Figure 4.1 Solution

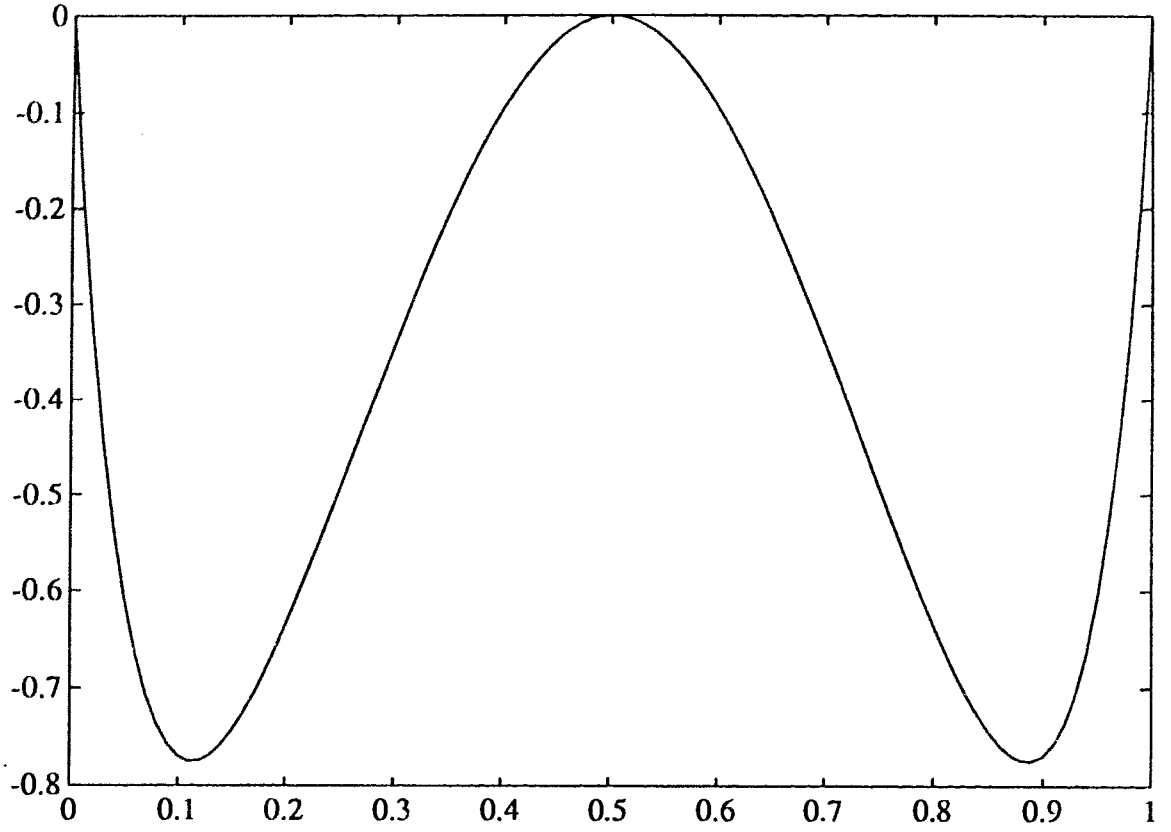
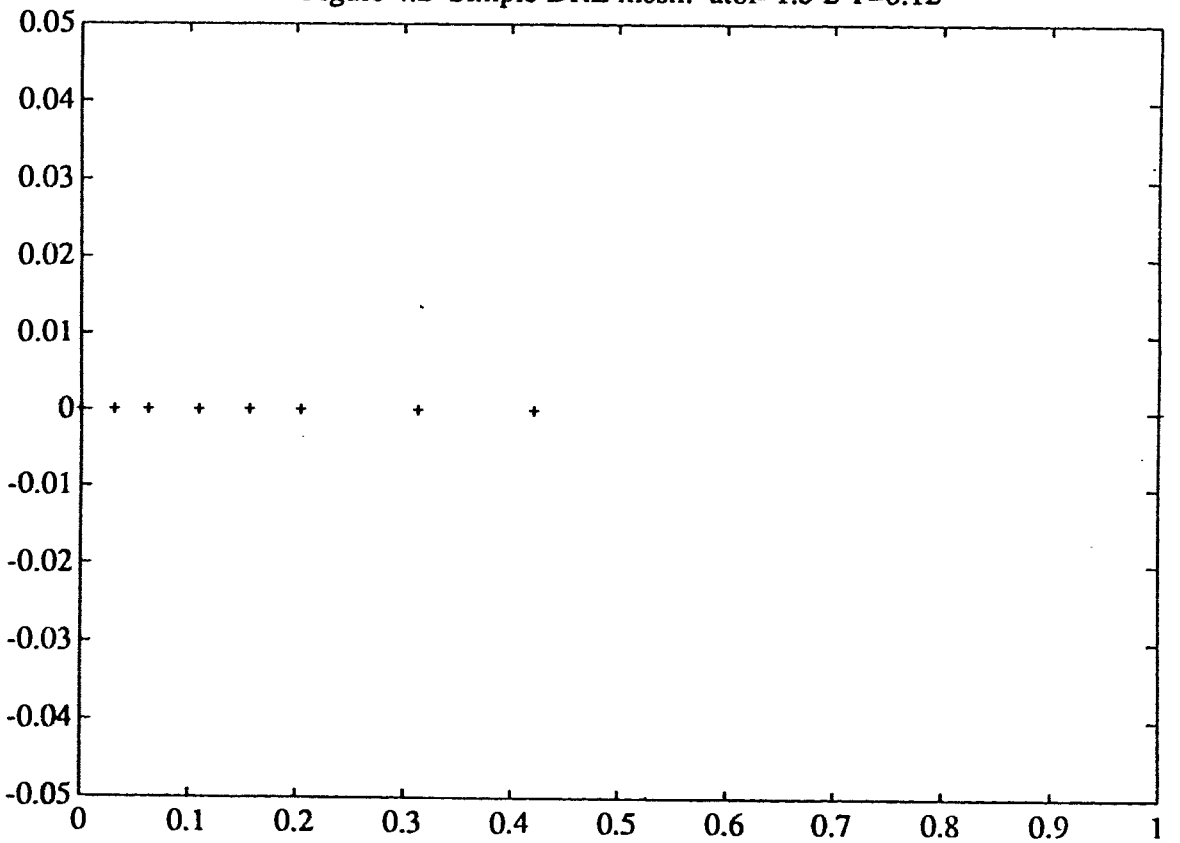


Figure 4.2 Simple DRE mesh: atol=1.e-2 T=0.12



Example 5. Consider the BVP:

$$\varepsilon u'' + u' = 0, \quad 0 < t < 1$$

$$u(0) = 0, \qquad u(1) = 1,$$

This is example 2 of [6]. This BVP has the exact solution

$$u(t) = (1-\exp(-t/\varepsilon)/(1-\exp(-1/\varepsilon)).$$

This example has a left boundary layer. The reduction $y = (\varepsilon u'+u, u)^T$ gives DRE:

$$R' = \frac{1}{\varepsilon} - \frac{1}{\varepsilon} R, \qquad R(0) = 0.$$

The simple DRE mesh gives correct layer information. The following is a summary table of the computation.

Table 5

ε	10^{-3}	10^{-3}	10^{-3}	10^{-6}	10^{-6}	10^{-6}
atol=rtol	10^{-2}	10^{-4}	10^{-6}	10^{-2}	10^{-4}	10^{-6}
COLNEW	10,20, 10,20	10,20,11, 22,11,22	10,20,10, 20,10,20, 10,20	10,20,40, 80,160, 99,198, 99,198, 99,198,	10,20,40, 80,160, 99,198, 99,198, 99,198	10,20,40, 80,160, 99,198, 99,198, 99,198
cpu	0.62	0.92	1.04	11.94	11.91	11.79
y-err	0.51d-2	0.52d-5	0.40d-8	0.20d-1	0.20d-1	0.20d-1
DREmesh +colnew	20,10,20	49,25,50	90,180	26,13,26	55,28,56	102,99, 198,
cpu	0.46	1.04	2.33	0.61	1.19	3.41
y-err	0.22d-7	0.52d-11	0.51d-13	0.51d-7	0.31d-11	0.41d-14
DREmesh double	20,40	49,98	90,180	26,52	55,110	102
cpu	0.54	1.25	2.30	0.72	1.42	--------
y-err	0.34d-8	0.49d-11	0.51d-13	0.33d-8	0.52d-11	--------

Figure 5.1 Solution: eps=1.e-6

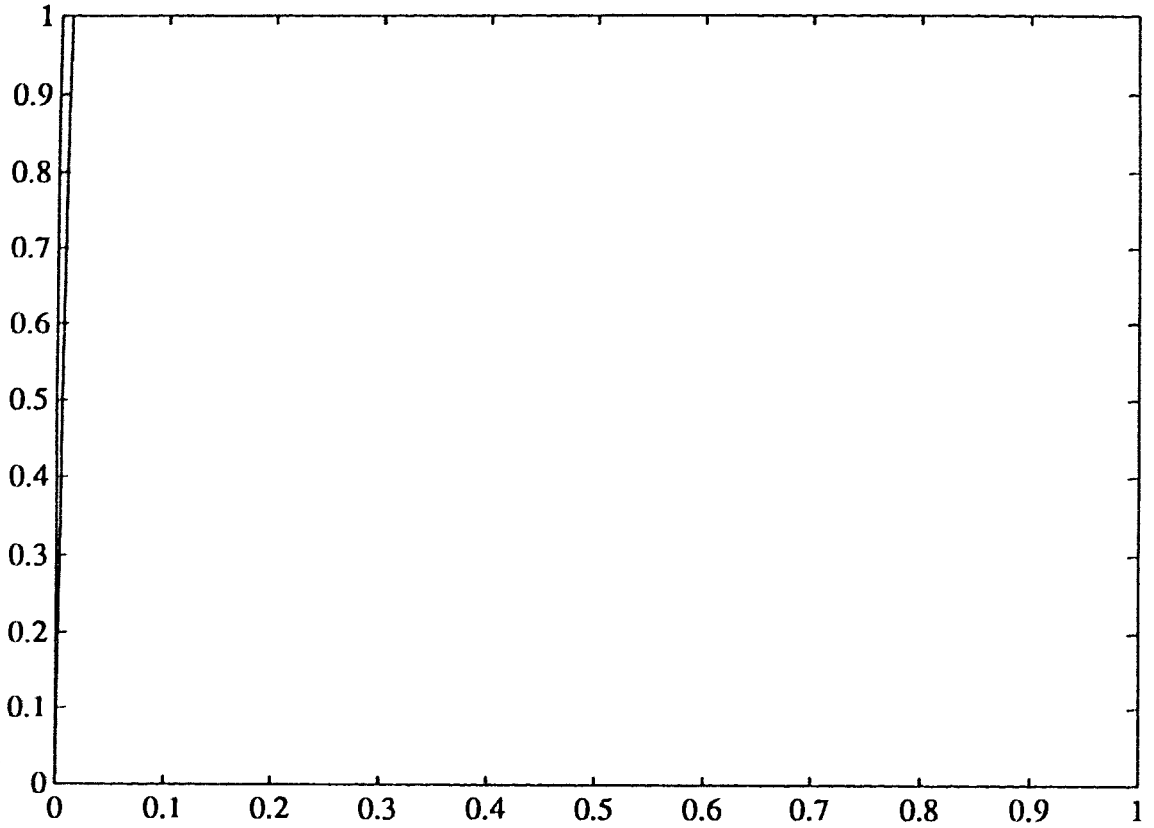
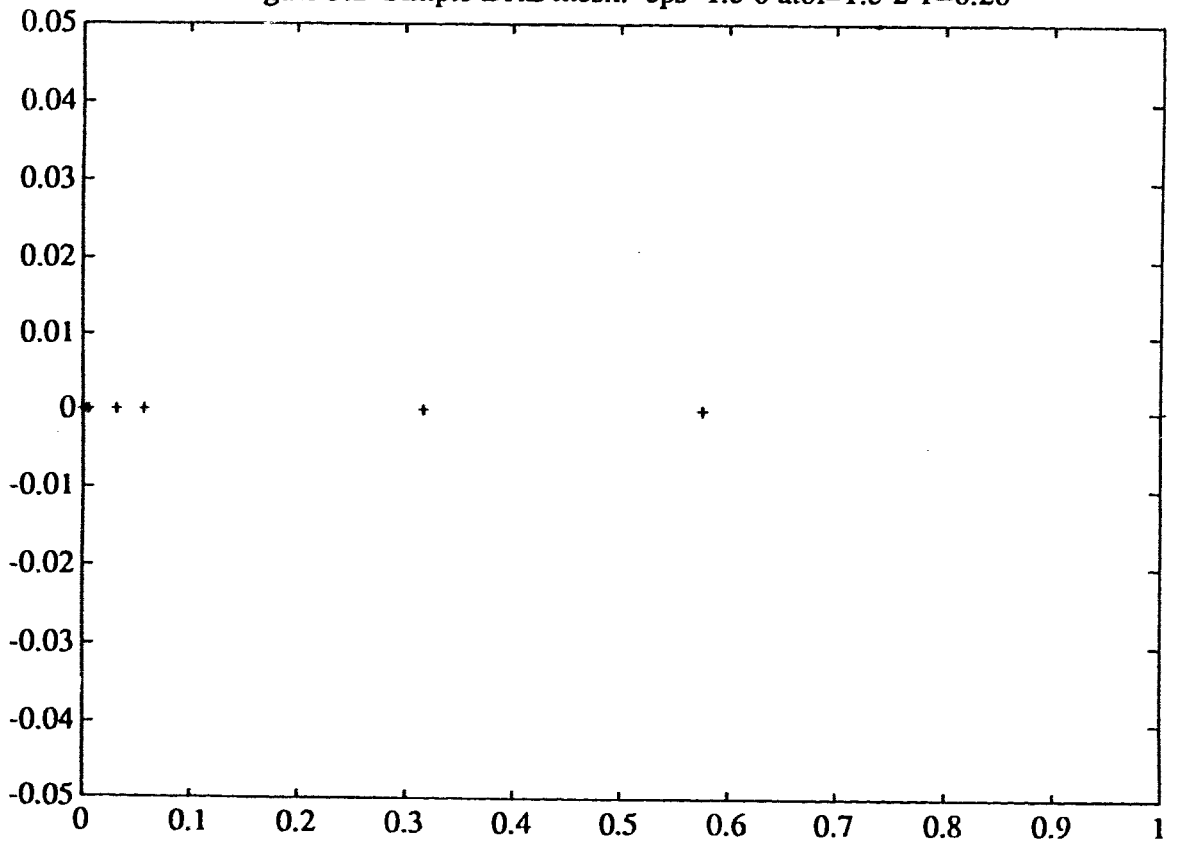


Figure 5.2 Simple DRE mesh: eps=1.e-6 atol=1.e-2 T=0.20



Example 6. Consider the BVP:

$$-\varepsilon u'' - \frac{t}{2}u' + \frac{t}{2}z' + z = \varepsilon\pi^2\cos(\pi t) + \frac{1}{2}(\pi t)\sin(\pi t)$$

$$\varepsilon z'' - z = 0$$

$$u(-1) = 1, \qquad z(-1) = 1, \qquad u(1) = z(1) = \exp(-\frac{2}{\sqrt{\varepsilon}}).$$

It is example 9 of [6]. This example has exact solution

$$u(t) = \mathrm{erf}(t/2\sqrt{\varepsilon})/\mathrm{erf}(1/2\sqrt{\varepsilon}) + z(t) + \cos(\pi t), \qquad z(t) = \exp(-(t+1)/\sqrt{\varepsilon}).$$

This example has a left boundary layer and interior layer at $t = 0$. The DRE is given by the reduction $y = (\varepsilon z', \varepsilon u' + \frac{t}{2}u, z, u)^T$. The simple DRE mesh gives correct layer information. We get a sub DRE mesh of 70 subintervals from the simple DRE mesh with atol=rtol $=10^{-2}$ and $\varepsilon=10^{-6}$. With this sub mesh, COLNEW spent 14" to achieve the accuracy 10^{-6} for u and 10^{-7} for z (we requested 10^{-6}).

Table 6

ε	10^{-3}	10^{-3}	10^{-3}	10^{-6}	10^{-6}	10^{-6}
atol=rtol	10^{-2}	10^{-4}	10^{-6}	10^{-2}	10^{-4}	10^{-6}
COLNEW	10,5,10	10,5,10,20	10,8,16,32	10,20,40, 20,40,20, 40	10,20,40 20,40,20, 40	10,20,40, 35,70,39, 78,39,78, 39,78
cpu	0.86	1.57	2.30	6.50	6.50	18.61
y-err	0.45d-2	0.25d-3	0.34d-4	0.28d-3	0.28d-3	0.40d-5
DREmesh +colnew	78,39,78	186	337	178	297	32199
cpu	6.41	--------	--------	---------	--------	--------
y-err	0.29d-7	--------	--------	---------	--------	--------
DREmesh double	78	186	337	178	297	32199
cpu	--------	--------	--------	--------	--------	--------
y-err	--------	--------	--------	--------	--------	--------

Figure 6.1 Solution: eps=1.e-6

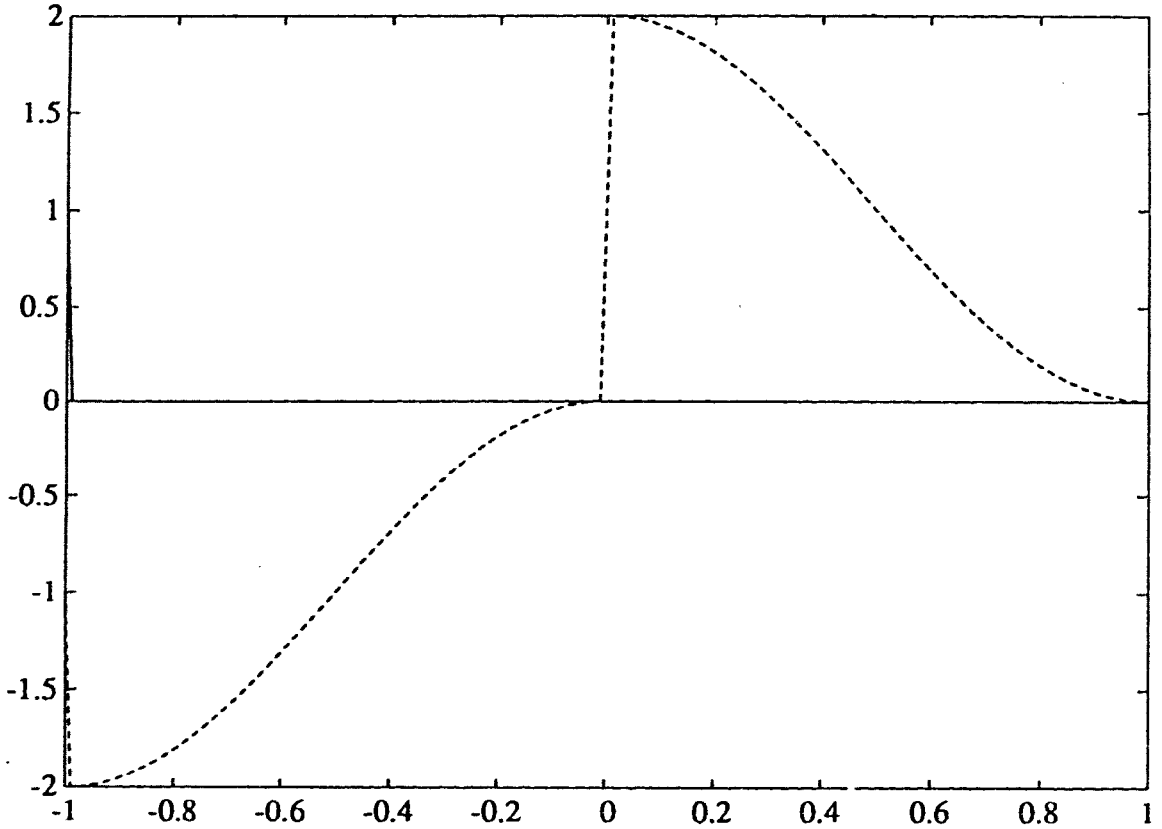


Figure 6.2 Simple DRE mesh: eps=1.e-6 atol=1.e-2 T=3.53

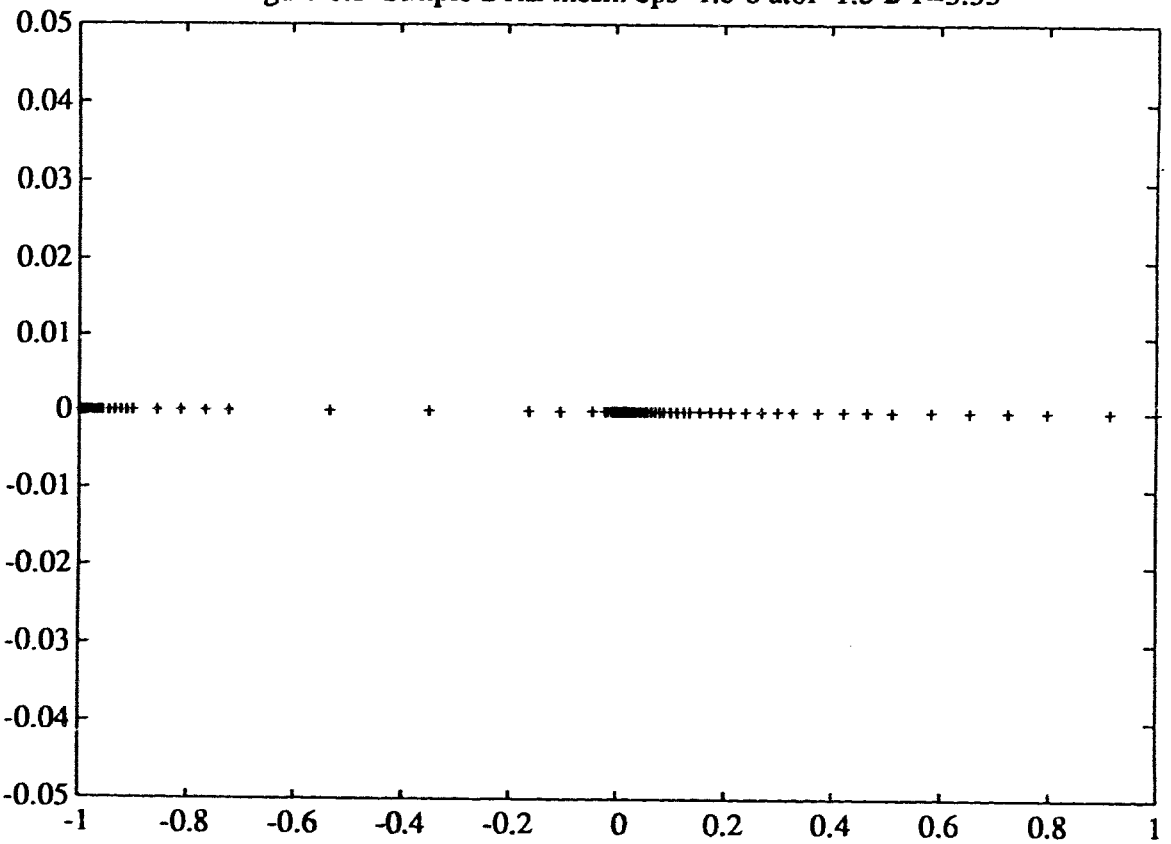


Figure 6.3 Combined DRE mesh

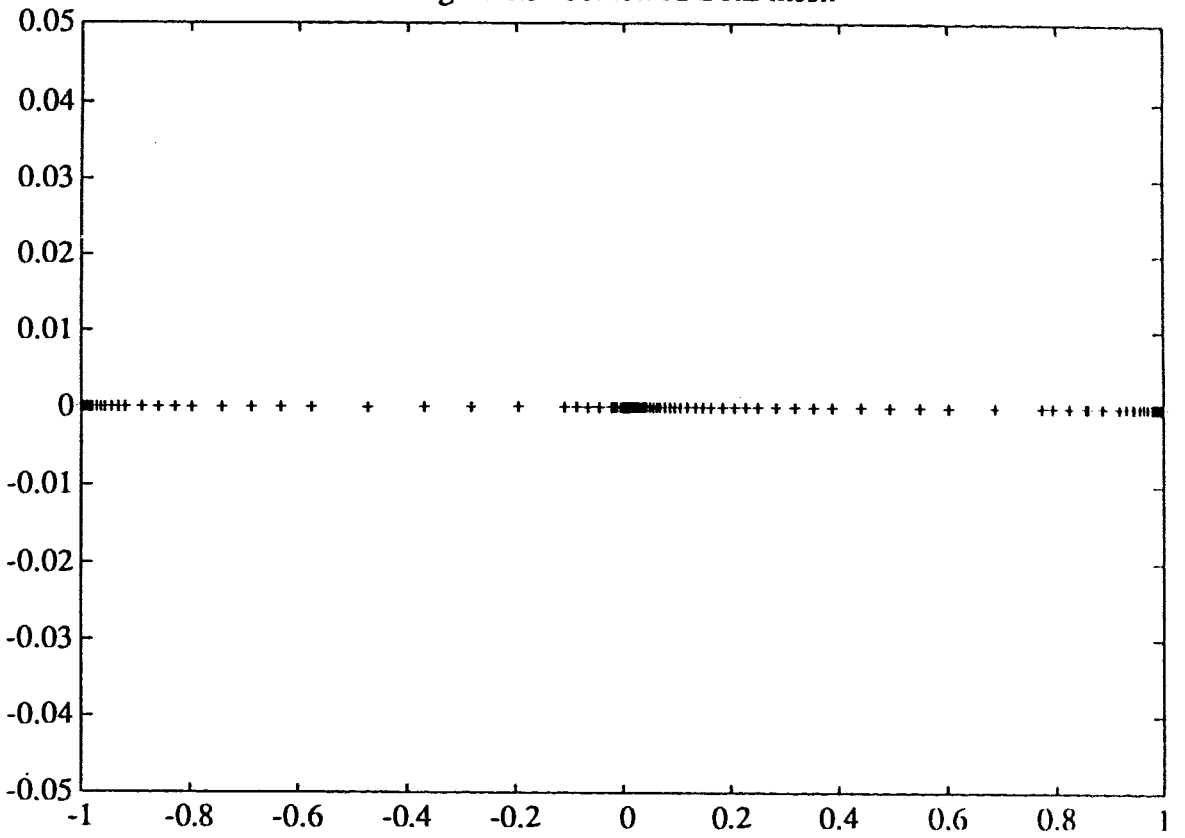
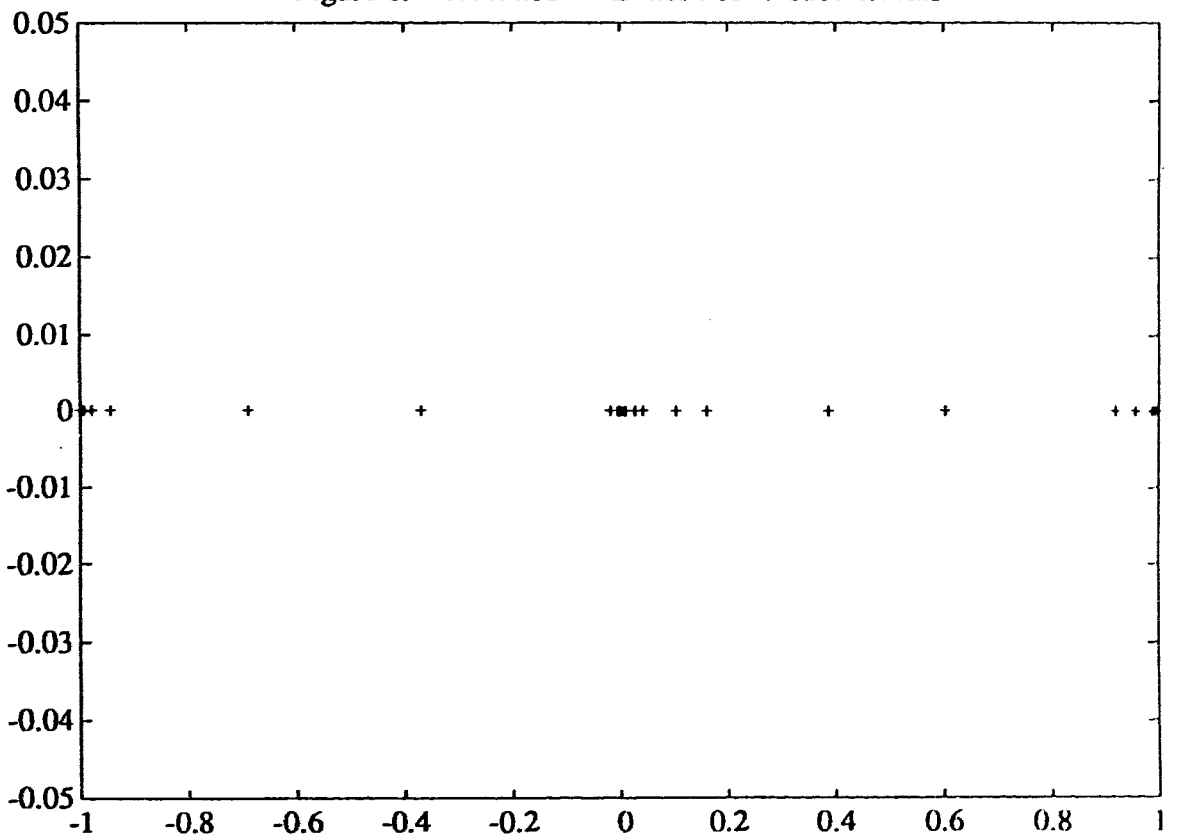


Figure 6.4 Trimmed DRE mesh of 49 subintervals



Example 7. Consider the BVP:

$$\varepsilon u'' + (t^3 - t/2)u' - u = 0, \qquad -1 < t < 1,$$

$$u(-1) = 1, \qquad u(1) = 2.$$

This is example 6 of [6]. This BVP has turning point behaviour at

$t = -\sqrt{2}/2,\ 0,\ \sqrt{2}/2$. The reduction $y_1 = \varepsilon u' + (t^3 - t/2)u$, $y = (y_1, u)^T$ gives the DRE:

$$R' = \frac{1}{\varepsilon} + \frac{1}{2\varepsilon}(t - 2t^3)R^2, \qquad R(-1) = 0$$

The simple DRE mesh gives correct layer information.

Table 7

ε	10^{-3}	10^{-3}	10^{-3}	10^{-6}	10^{-6}	10^{-6}
atol=rtol	10^{-2}	10^{-4}	10^{-6}	10^{-2}	10^{-4}	10^{-6}
COLNEW	10,20	10,20,10, 20	10,20,11, 22,44	10,20,40, 80	10,20,40, 80,160, 99,198	10,20,40, 80,160, 99,198, 99,198, 99,198
cpu	0.53	1.06	1.98	2.61	6.34	12.42
y-err	0.11d-3	0.35d-5	0.11d-6	0.94d-2	0.50d-4	0.13d-7
DREmesh +colnew	82,41,82	208	348	205,103, 206	348	1752
cpu	2.06	--------	--------	4.05	--------	--------
y-err	0.62d-9	--------	--------	0.31d-2	--------	--------
DREmesh double	82,164	208	348	205,410	348	1752
cpu	2.5	--------	-------	--------	-------	--------
y-err	0.38d-6	--------	-------	--------	-------	--------

Figure 7.1 Solution: eps=1.e-6

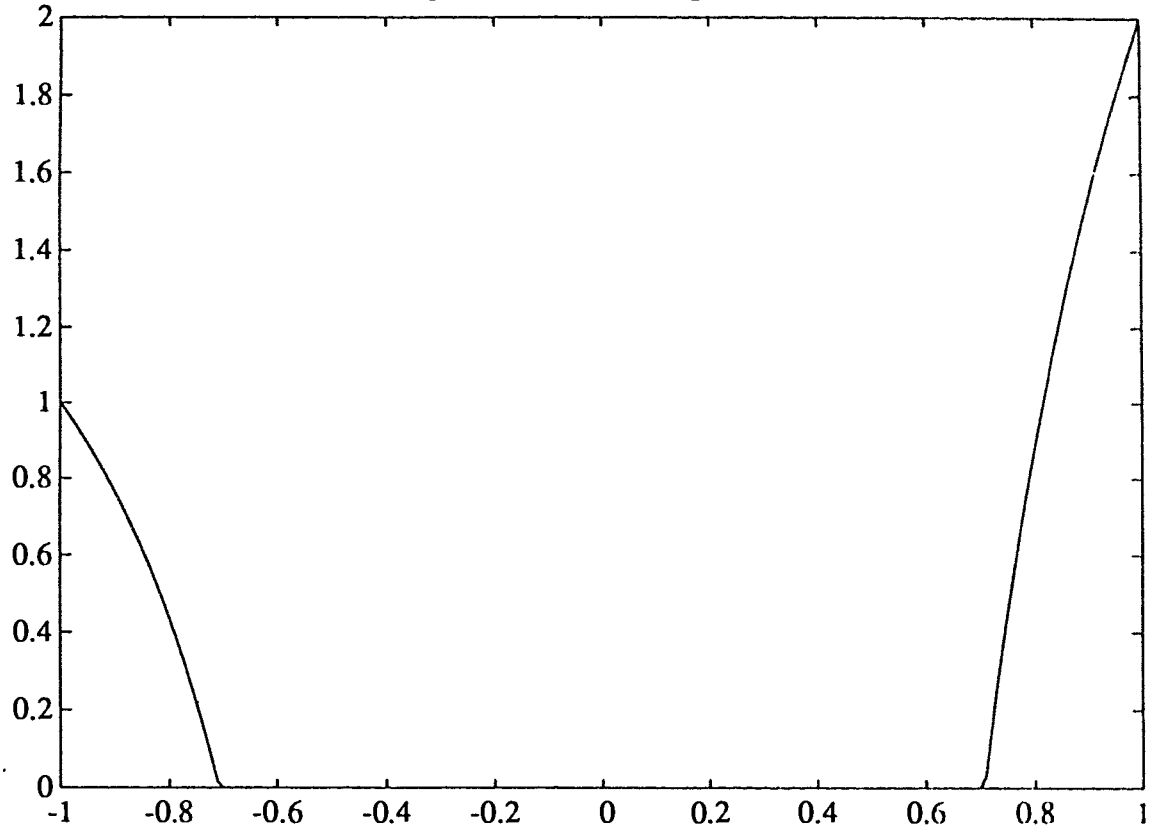
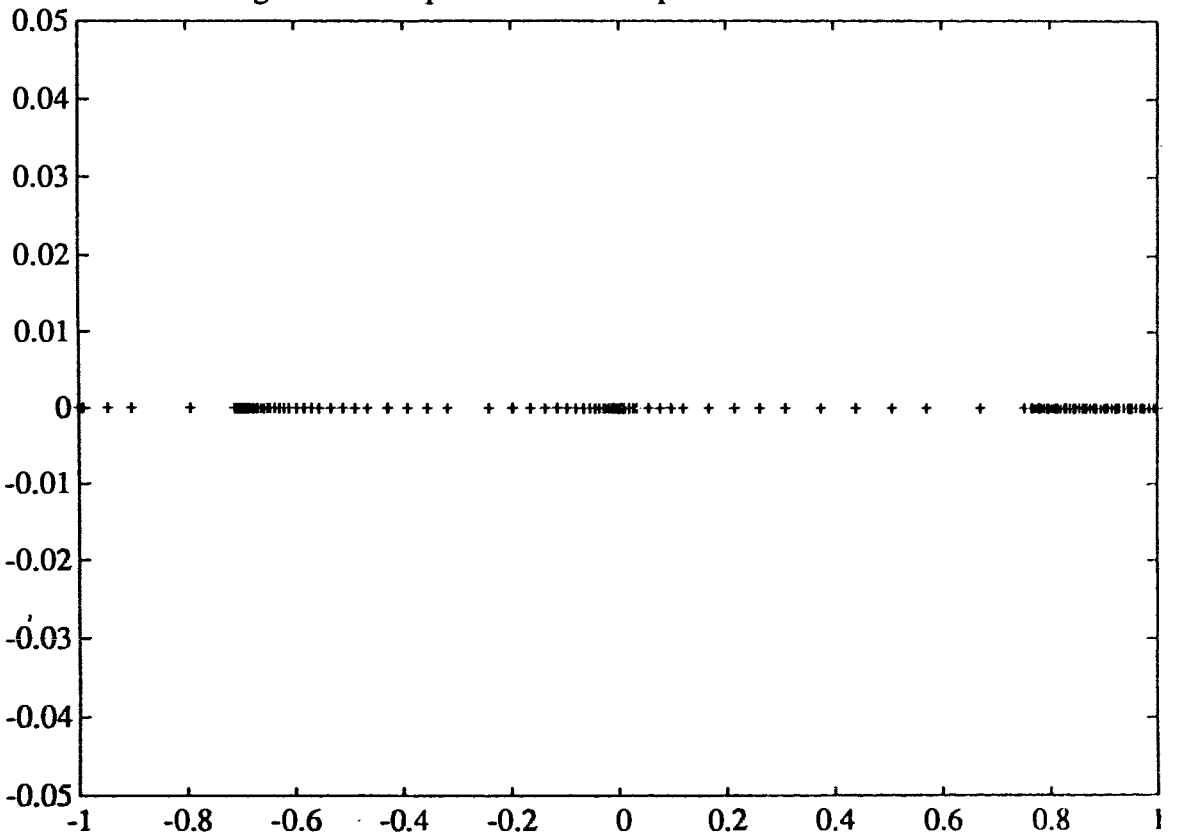


Figure 7.2 Simple DRE mesh: eps=1.e-6 atol=1.e-2 T=1.57



Example 8. Consider the BVP:

$$\varepsilon u'' + \varepsilon u' - u = 0, \qquad 0 < t < 1$$

$$u(0) = 1, \qquad u(1) = \frac{1}{2}$$

This is example 3 of [6]. This BVP has the exact solution

$$u(t) = c_1 \exp(r_1 t) + c_2 \exp(r_2 t),$$

$r_1 = -0.5 - \sqrt{0.25 + \frac{1}{\varepsilon}}$, $r_2 = -0.5 + \sqrt{0.25 + \frac{1}{\varepsilon}}$. This example has two boundary layers. The reduction $y = (\varepsilon u' + \varepsilon u, u)^T$ gives the DRE:

$$R' = \frac{1}{\varepsilon} - R - R^2, \qquad R(0) = 0$$

The simple DRE mesh missed the right boundary layer.

Table 8

ε	10^{-3}	10^{-3}	10^{-3}	10^{-6}	10^{-6}	10^{-6}
atol=rtol	10^{-2}	10^{-4}	10^{-6}	10^{-2}	10^{-4}	10^{-6}
COLNEW	10,20	10,20	10,20, 10,20	10,20	10,20,40, 20,40,20, 40	10,20,40, 23,46, 23,46,
cpu	0.34	0.35	0.64	0.32	1.93	2.18
y-err	0.45d-5	0.45d-5	0.92d-7	0.15d-1	0.22d-7	0.73d-8
DREmesh +colnew	32,16,32	55,28,56	102,51 102	47,24,48,	64,32,64, 32,64,128	111,56, 112,56, 112,56, 112,56,
cpu	0.80	1.34	2.5	1.17	3.77	6.22
y-err	0.21d-7	0.57d-9	0.15d-10	0.59d-2	0.10d-3	0.40d-10
DREmesh double	32,64	55,110	102	47,94	64,128,	111
cpu	0.94	1.56	--------	1.38	1.85	--------
y-err	0.11d-4	0.34d-5	--------	0.11d-1	0.11d-1	--------

Figure 8.1 Solution: eps=1.e-6

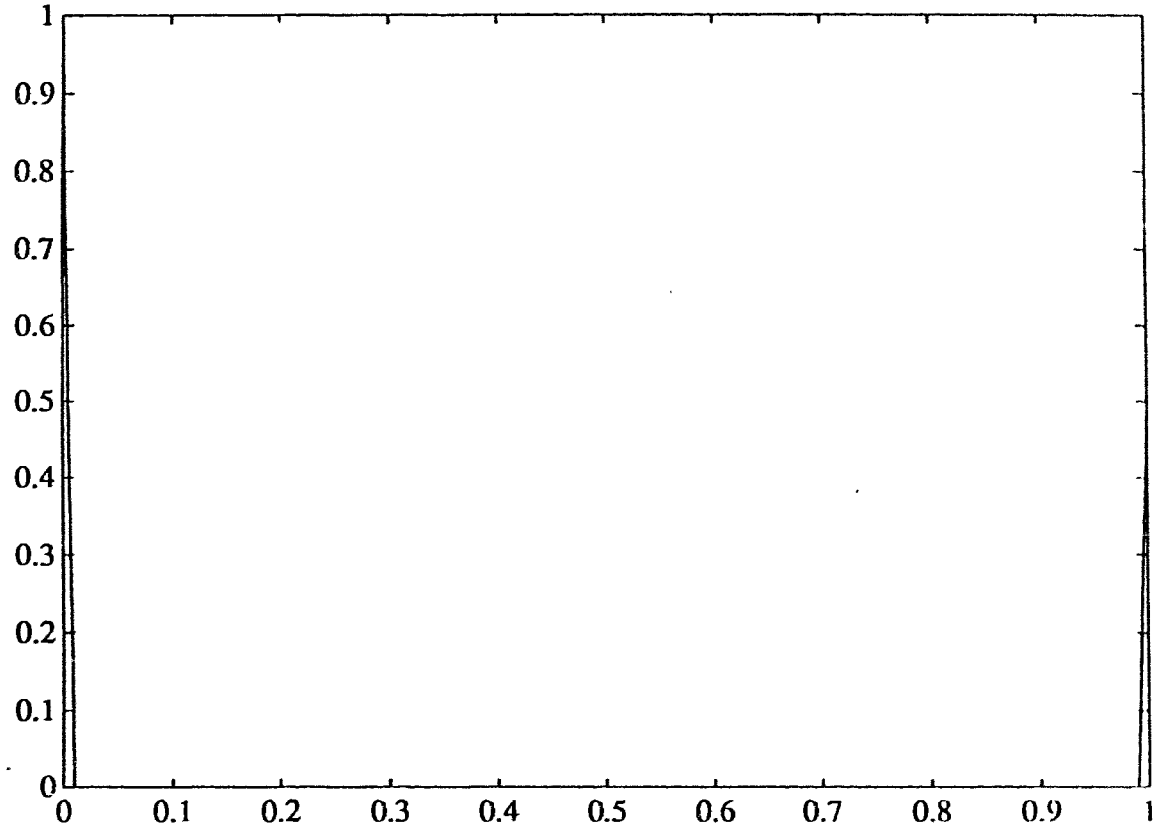


Figure 8.2 Simple DRE mesh: eps=1.e-6 atol=1.e-2 T=0.35

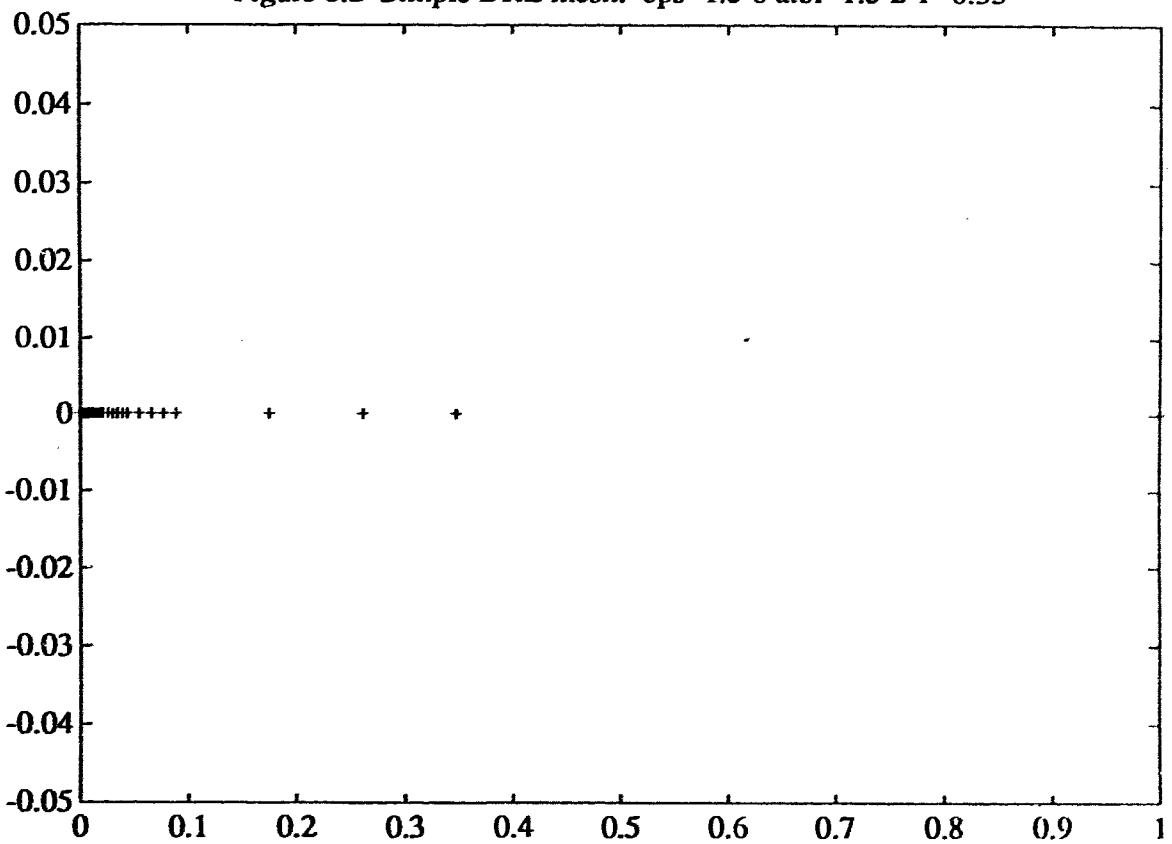


Figure 8.3 Combined DRE mesh

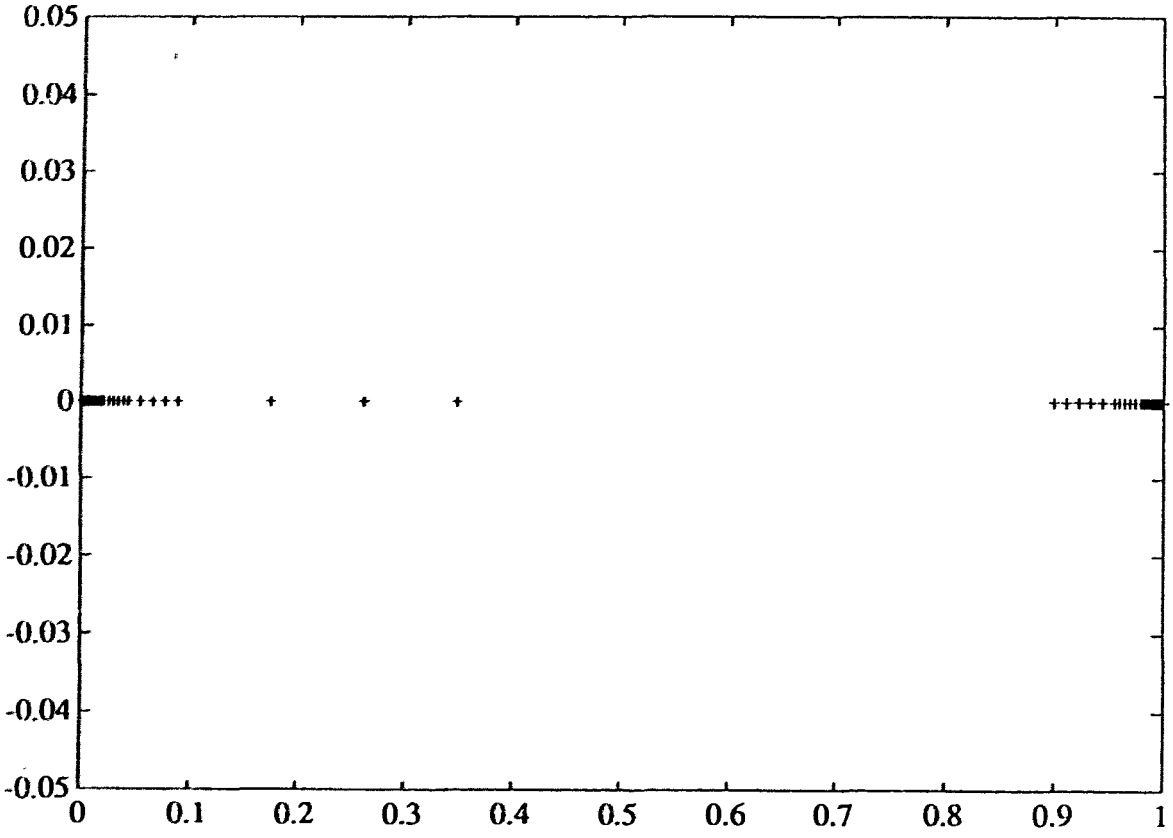
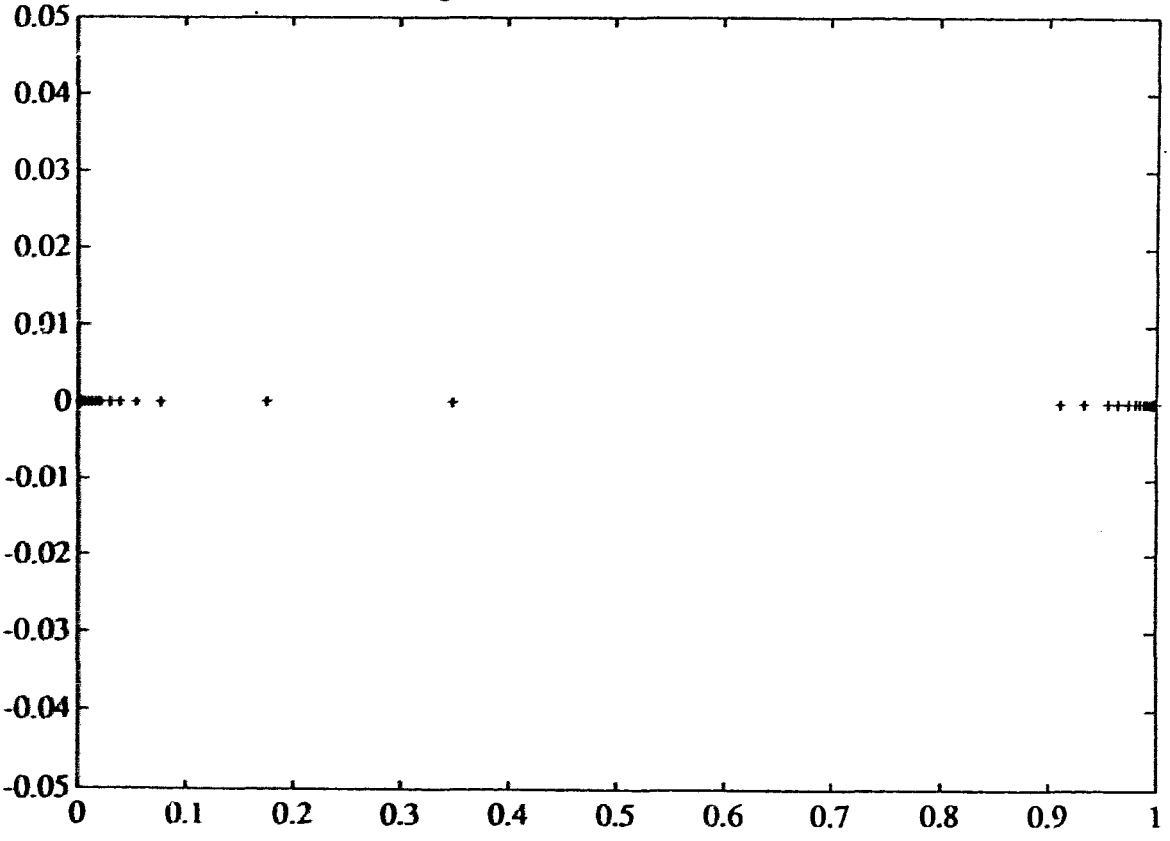


Figure 8.4 Trimmed DRE mesh



Example 9. Consider the BVP:

$$\varepsilon u'' - t^2 u + \varepsilon^{1/2} = 0, \quad -1 < t < 1,$$

$$u(-1) = a, \qquad u(1) = b.$$

This is example of [3, P333]. This example has two boundary layers and one interior layer around 0. The reduction $y = (\varepsilon u' + \varepsilon^{1/2} t , u)^T$ gives the DRE

$$R' = \frac{1}{\varepsilon} - t^2 R^2, \qquad R(-1) = 0.$$

The simple DRE mesh missed the right boundary layer. The computations for a=1.0 and b=0.5 are summarized in the following table.

Table 9

ε	10^{-3}	10^{-3}	10^{-3}	10^{-6}	10^{-6}	10^{-6}
atol=rtol	10^{-2}	10^{-4}	10^{-6}	10^{-2}	10^{-4}	10^{-6}
COLNEW	10,20	10,20	10,20,12, 24,48	10,20	10,20,40, 21,42,21 42,21,42	10,20,40, 40,31,62, 31,62
cpu	0.51	0.52	1.2	0.34	2.72	2.97
y-err	0.87d-4	0.87d-4	0.39d-7	0.20d-1	0.19d-5	0.25d-6
DREmesh +colnew	59,30,60	132,66, 132	233	103,52, 104	195,98, 196	343
cpu	2.91	5.28	--------	2.57	4.79	--------
y-err	0.94d-8	0.21d-10	--------	0.19d-3	0.11d-5	--------
DREmesh double	59,118	132	233	103	195	343
cpu	2.83	--------	--------	--------	--------	--------
y-err	0.35d-6	--------	--------	--------	--------	--------

Figure 9.1 Solution: eps=1.e-6

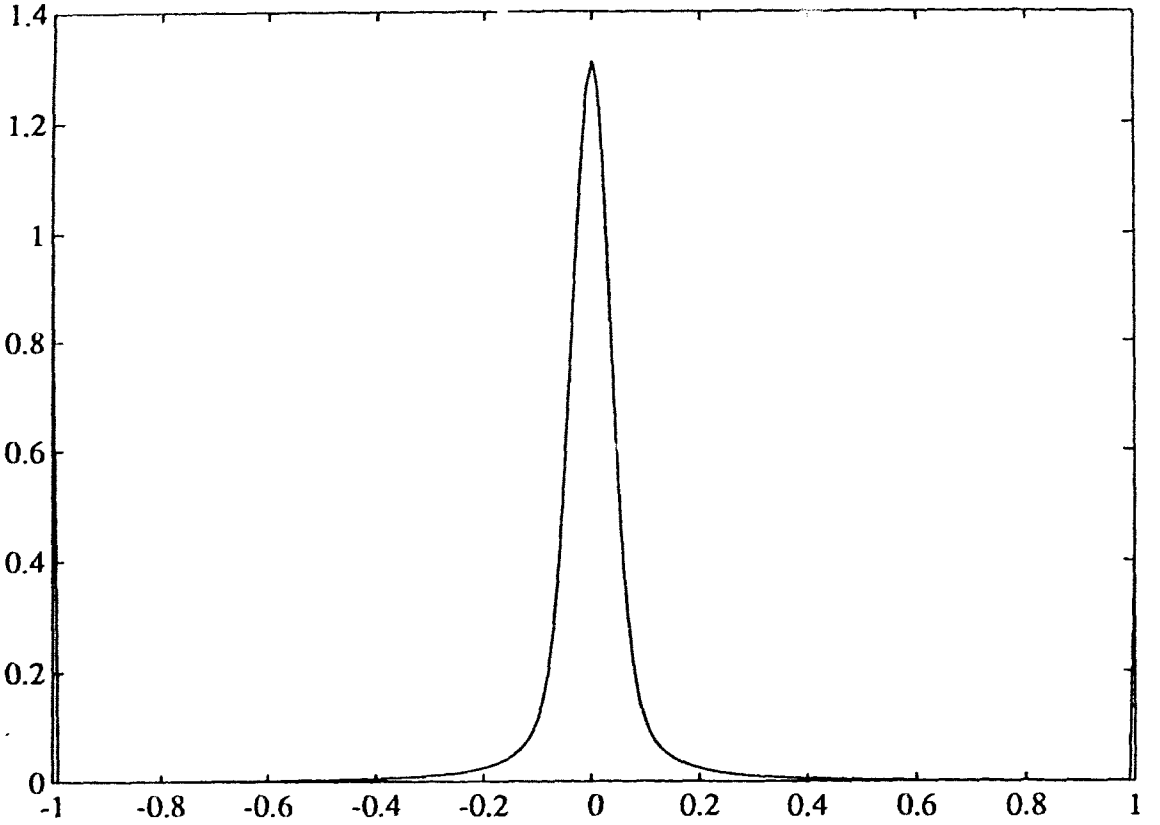


Figure 9.2 Simple DRE mesh: eps=1.e-6 atol=1.e-2 T=1.06

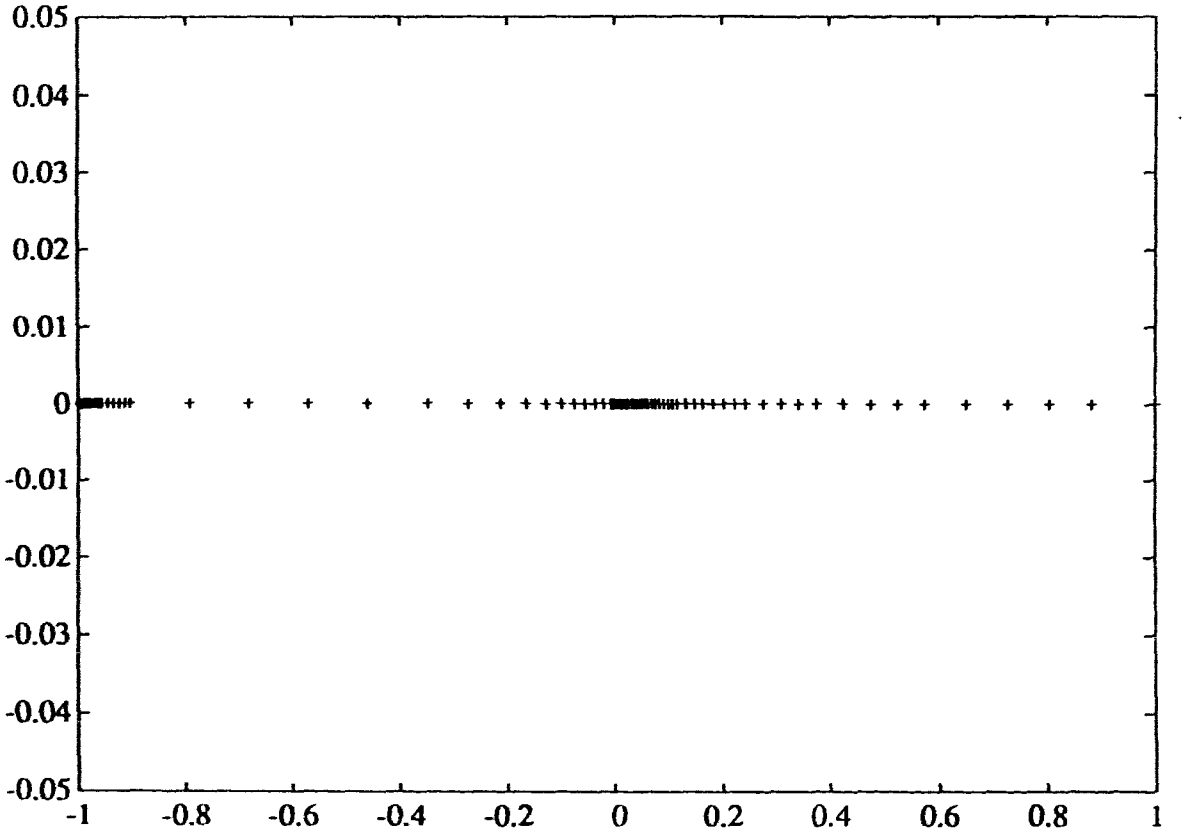


Figure 9.3 Combined DRE mesh

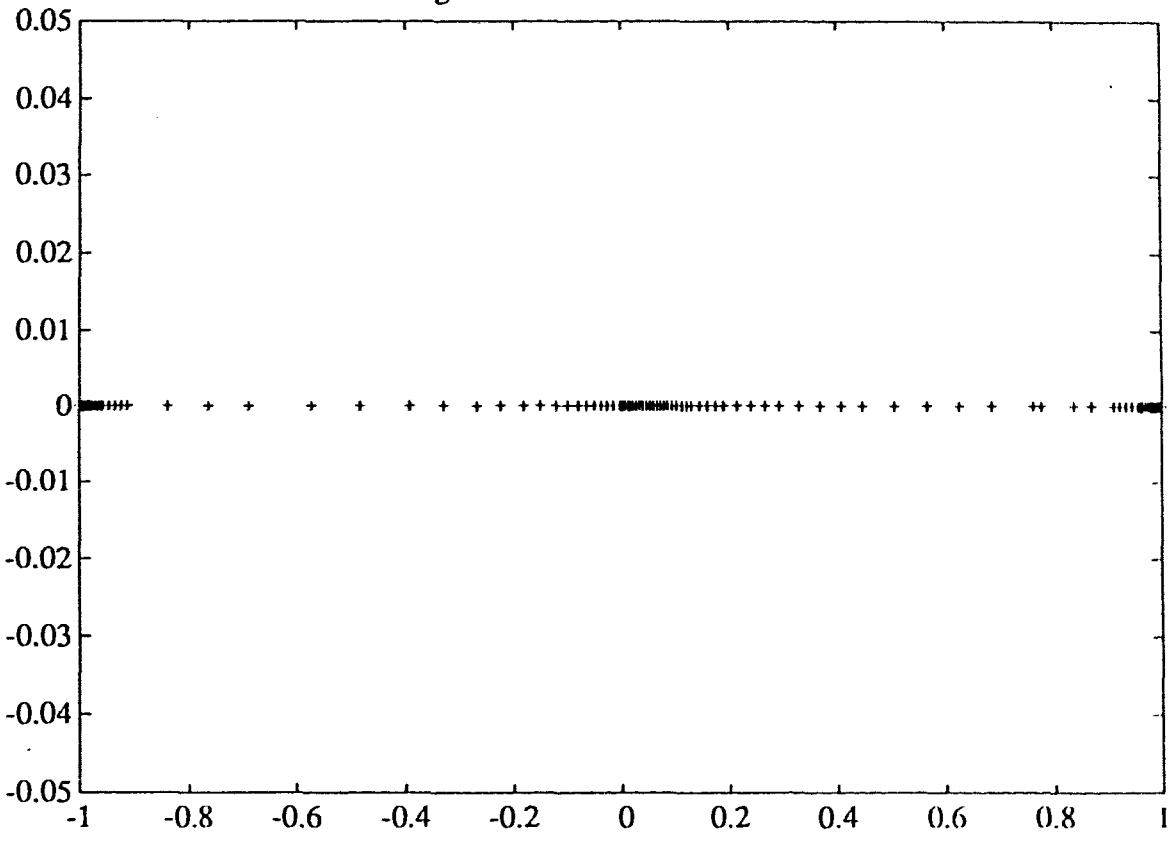
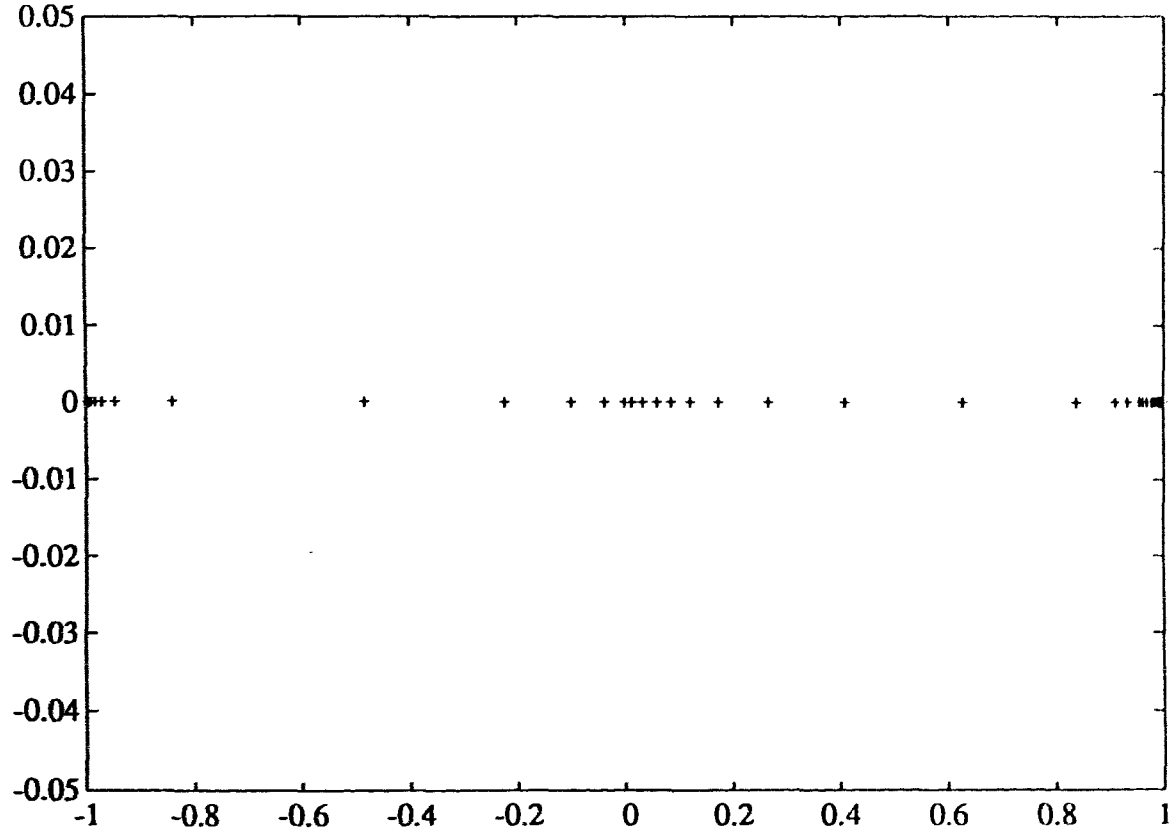


Figure 9.4 Trimmed DRE mesh



Example 10. Consider the BVP:

$$u'' = 100u - u' - 100z, \qquad 0 < t < 5,$$

$$z'' = -10000u + 10000z - z'$$

$$u(0) + u'(0) = 3 + \frac{101}{50}e,$$

$$z(0) = 1 - 2\exp(-505),$$

$$u(5) - z(5) = \frac{101}{50}(1 - \exp(-505)),$$

$$u'(5) + z(5) = 1 + \frac{201}{50}\exp(-505).$$

This is example 4 of [15]. This BVP has the exact solution

$$u(t) = 1 - 2\exp(-t) + (\exp(100t-500) - \exp(-101t))/50$$

$$z(t) = 1 - 2\exp(-t) - 2\exp(100t-500) + 2\exp(-101t)$$

This example has two boundary layers. The DRE is given by the reduction $y = (y, z', z, y')^T$. The simple DRE mesh missed the right boundary layer.

Table 10

atol=rtol	10^{-2}	10^{-4}	10^{-6}
COLNEW	10,20,10,2010,20,40	10,20,40,23,46, 23,46	10,20,40,40,80,80, 80,160,99,198,99, 198,99,198
cpu	4.43	5.08	16.75
y-err	0.80d-3	0.48d-4	0.59d-4
DRE mesh +colnew	36,18,36,18,36	84,42,84	181,99,198,99,198, 99,198
cpu	4.8	5.0	12.44
y-err	0.53d-5	0.29d-5	0.58d-4
DRE mesh double	36,72	84	181
cpu	4.68	--------	--------
y-err	0.17d-1	--------	--------

Figure 10.1 Solution

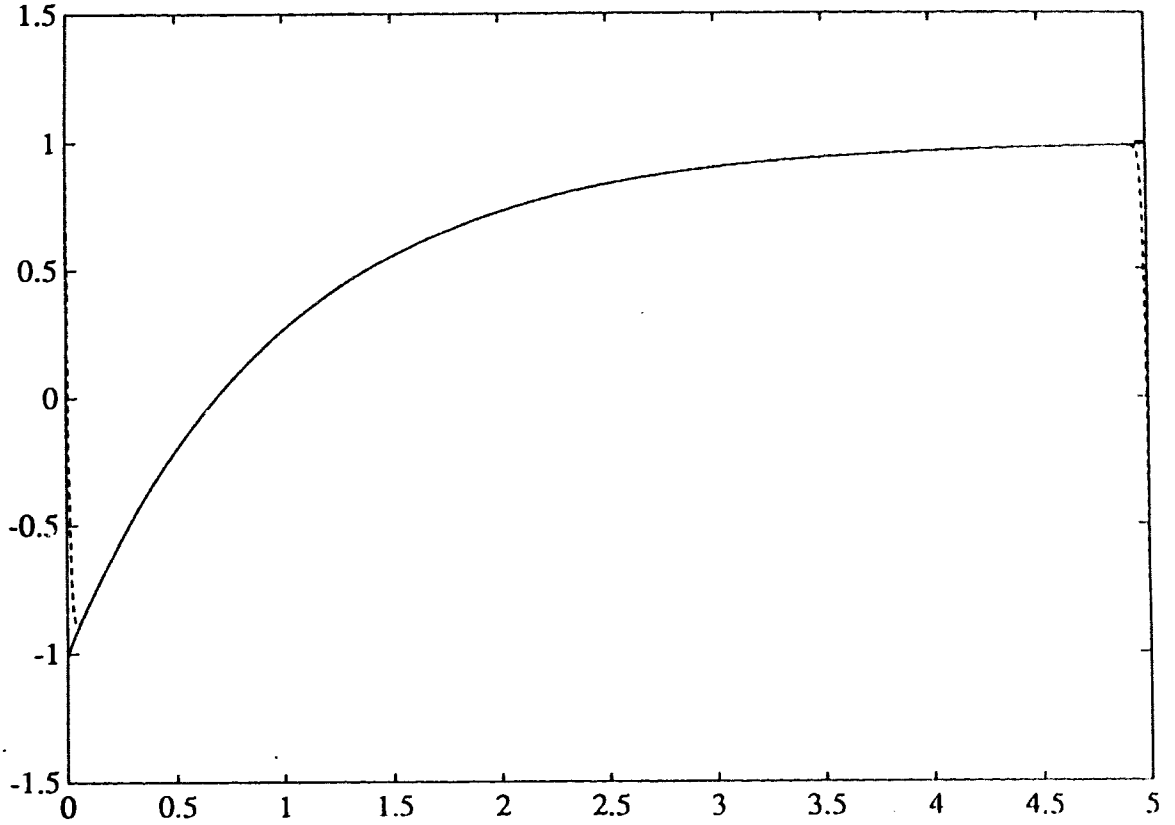
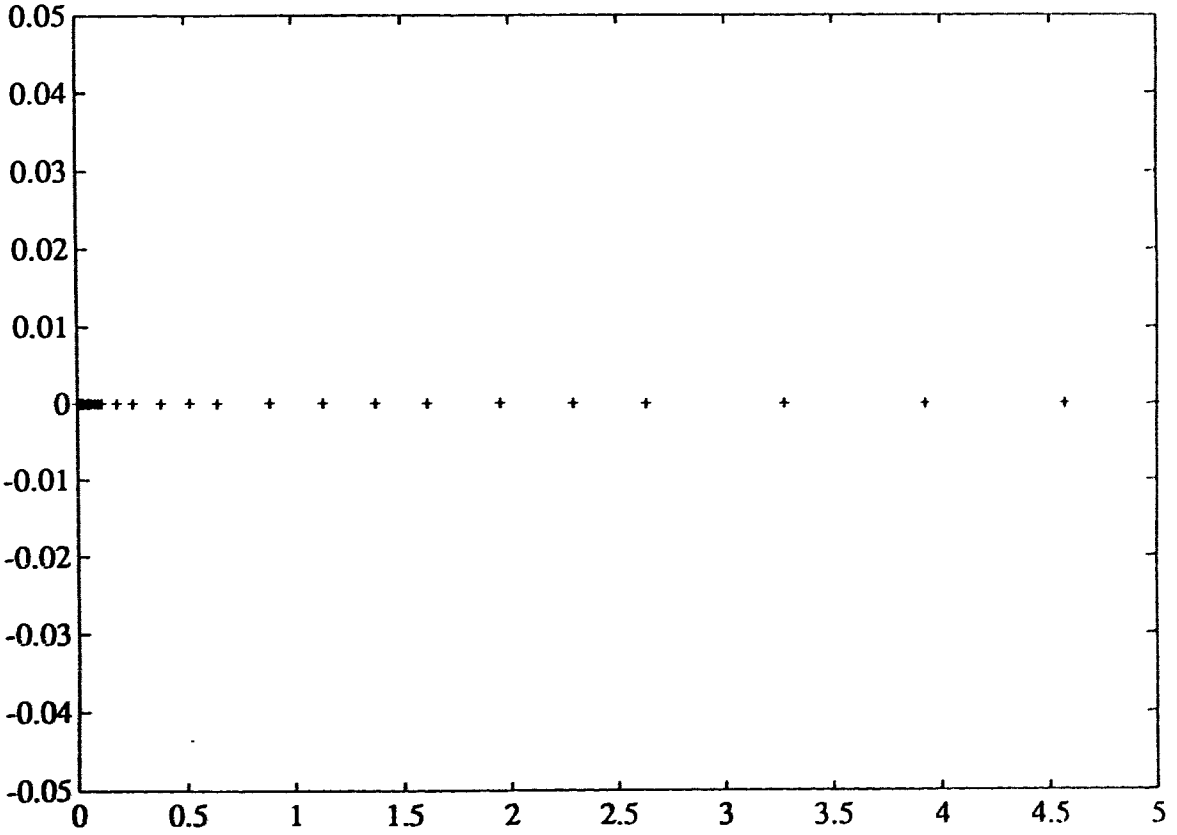


Figure 10.2 Simple DRE mesh: atol=1.e-2 T=0.56



Example 11. consider the BVP:

$$\varepsilon u'' + tu' = 0, \quad -1 < t < 1,$$

$$u(-1) = 0, \qquad u(1) = 1.$$

This is example 7 of [6]. This BVP has the exact solution

$$u(t) = 0.5 + \mathrm{erf}(t/\sqrt{2\varepsilon})/2\mathrm{erf}(1/\sqrt{2\varepsilon}).$$

It has an interior layer around 0. The reduction $y = (\varepsilon u' + u, u)^T$ gives the DRE

$$R' = \frac{1}{\varepsilon} - \frac{t}{\varepsilon} R - R^2, \qquad R(-1) = 0.$$

The simple DRE mesh contains an artificial left boundary layer. For a small ε, if the tolerance is large, the simple DRE mesh is still ok, but if the tolerance is small, the simple DRE mesh will be misled by the artificial left boundary layer. This problem is caused by the large magnitude of R. It can be fixed by the reimbedding strategy.

Table 11

ε	10^{-3}	10^{-3}	10^{-3}	10^{-6}	10^{-6}	10^{-6}
atol=rtol	10^{-2}	10^{-4}	10^{-6}	10^{-2}	10^{-4}	10^{-6}
COLNEW	10,20	10,20, 10,20	10,20, 10,20, 10,20	10,20,40, 80,160, 99,198, 99,198	10,20,40, 80,160,99, 198,99,198 99,198	10,20,40, 80,160,99 198,99,19 8,99,198
cpu	0.53	1.01	1.6	18.95	23.67	20.74
y-err	0.20d-3	0.14d-4	0.16d-6	0.62d-2	0.11d-3	0.11d-3
DREmesh +colnew	78,39,78	157,79, 158	259	128,64, 128	142,71,142 99,198,99, 198,99,198	523
cpu	3.11	3.72	--------	5.6	12.16	--------
y-err	0.29d-9	0.37d-11	--------	0.39d-9	0.37d-1	--------
DREmesh double	78,156	157	259	128	142	523
cpu	3.75	--------	--------	--------	--------	--------
y-err	0.19d-6	--------	--------	--------	--------	--------

Figure 11.1 Solution: eps=1.e-6

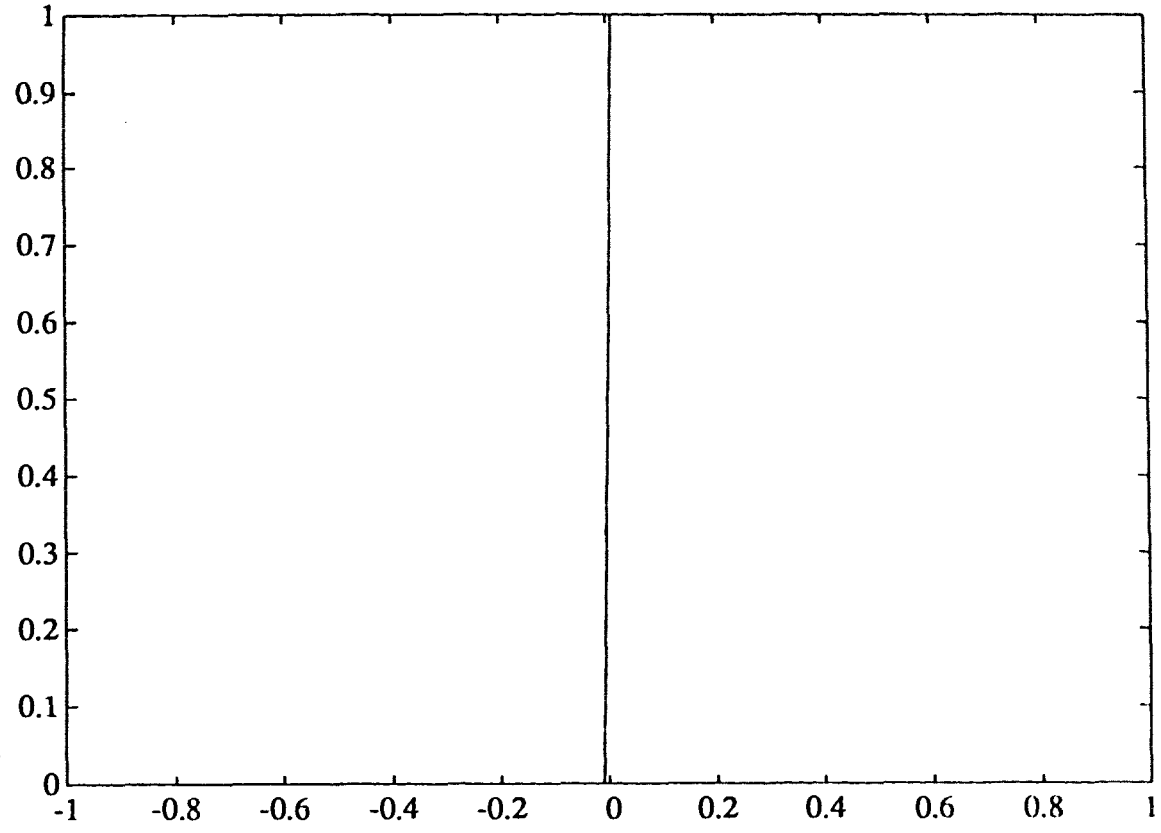


Figure 11.2 Simple DRE mesh: eps=1.e-6 atol=1.e-2 T=1.59

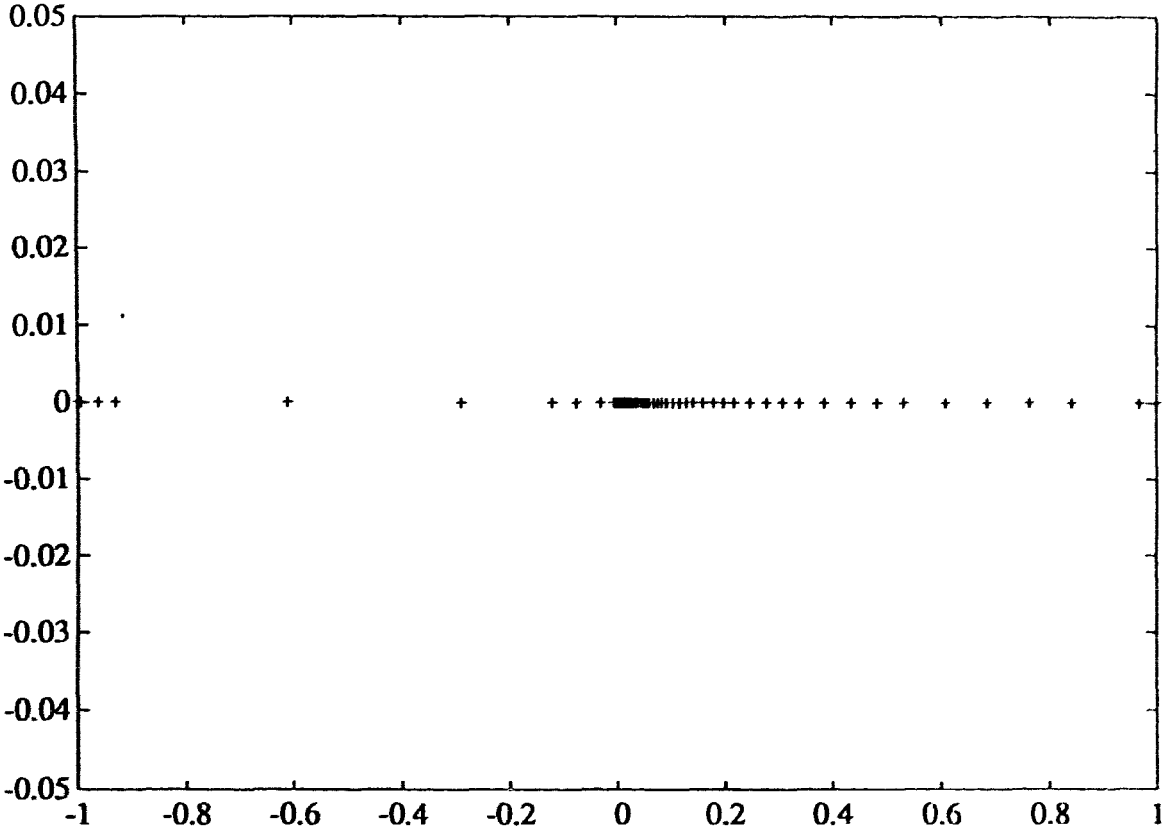


Figure 11.3 Combined DRE mesh

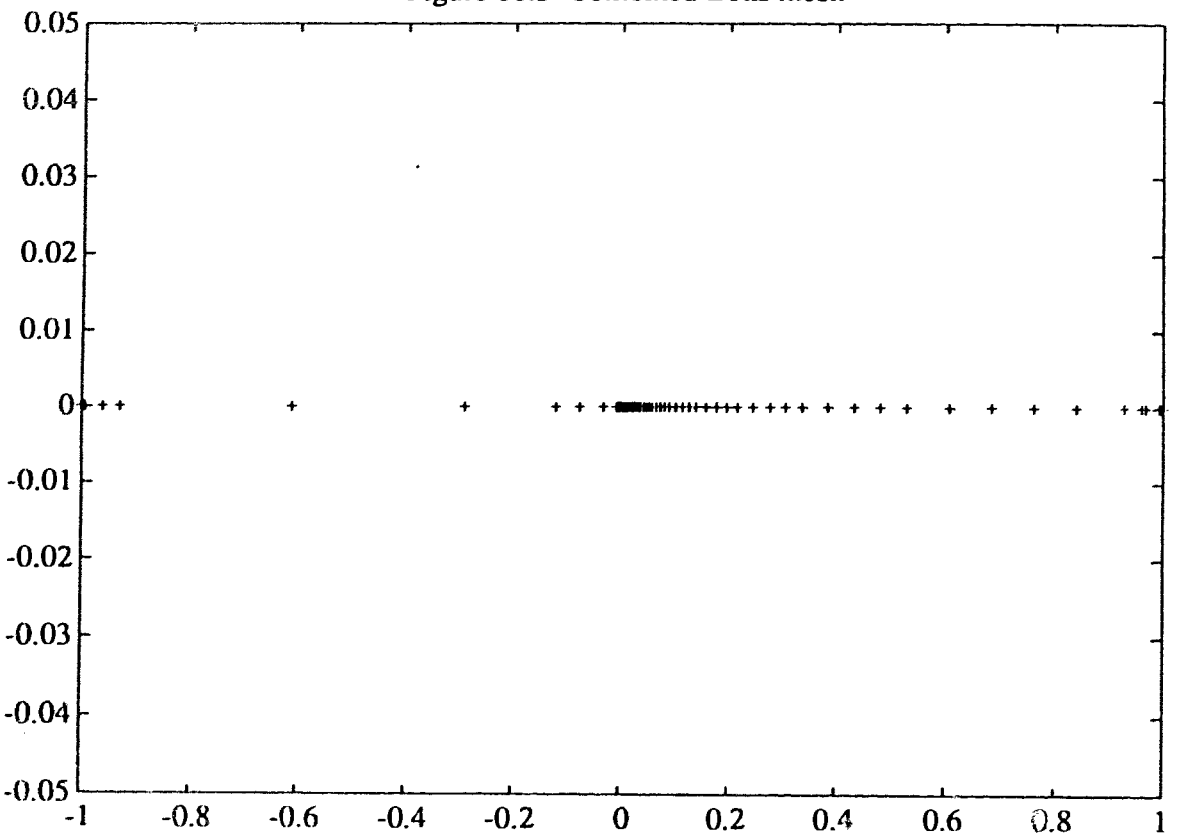
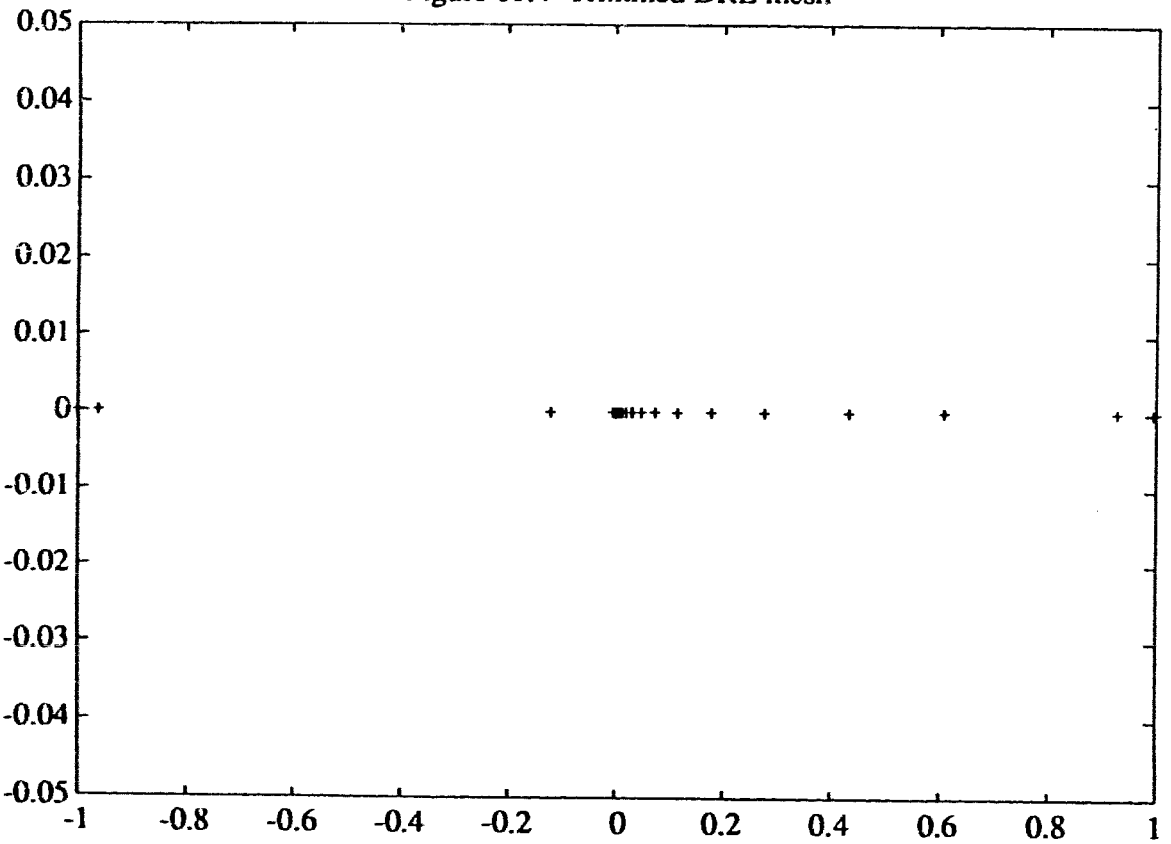


Figure 11.4 Trimmed DRE mesh



Example 12. Consider the BVP

$$\varepsilon u^{(4)} + u = 0, \quad 0 < t < 1,$$

$$u'(0) = 0, \qquad u'''(0) = 0,$$

$$u(1) = 1, \qquad u''(1) = 0.$$

This is example 15 with λ=0 of [15]. This example has a mild right boundary layer. The DRE is obtained from the reduction $y = (y, y'', y', y''')^T$. The simple DRE mesh missed the right boundary layer and consisted of an artificial left boundary layer.

Table 12

ε	10^{-3}	10^{-3}	10^{-3}	10^{-6}	10^{-6}	10^{-6}
atol=rtol	10^{-2}	10^{-4}	10^{-6}	10^{-2}	10^{-4}	10^{-6}
COLNEW	10,20	10,20	10,20	10,5,10	10,8,16,32	10,10,10, 20,40,80, 58,116
cpu	1.64	1.84	1.20	1.24	2.88	9.97
y-err	0.62d-7	0.62d-7	0.64d-5	0.44d+0	0.11d-1	0.19d-2
DREmesh +colnew	27,14,28	57,29,58	104,52, 104	38,19,38	89,45,90	150,75, 150
cpu	3.8	8.15	11.47	5.74	9.22	9.82
y-err	0.84d-8	0.11d-9	0.17d-8	0.15d-1	0.36d-4	0.81d-2
DREmesh double	27,54	57	104	38,76	89	150
cpu	4.65	--------	--------	6.79	--------	--------
y-err	0.74d-7	--------	--------	0.50d+2	--------	--------

Figure 12.1 Solution: eps=1.e-6

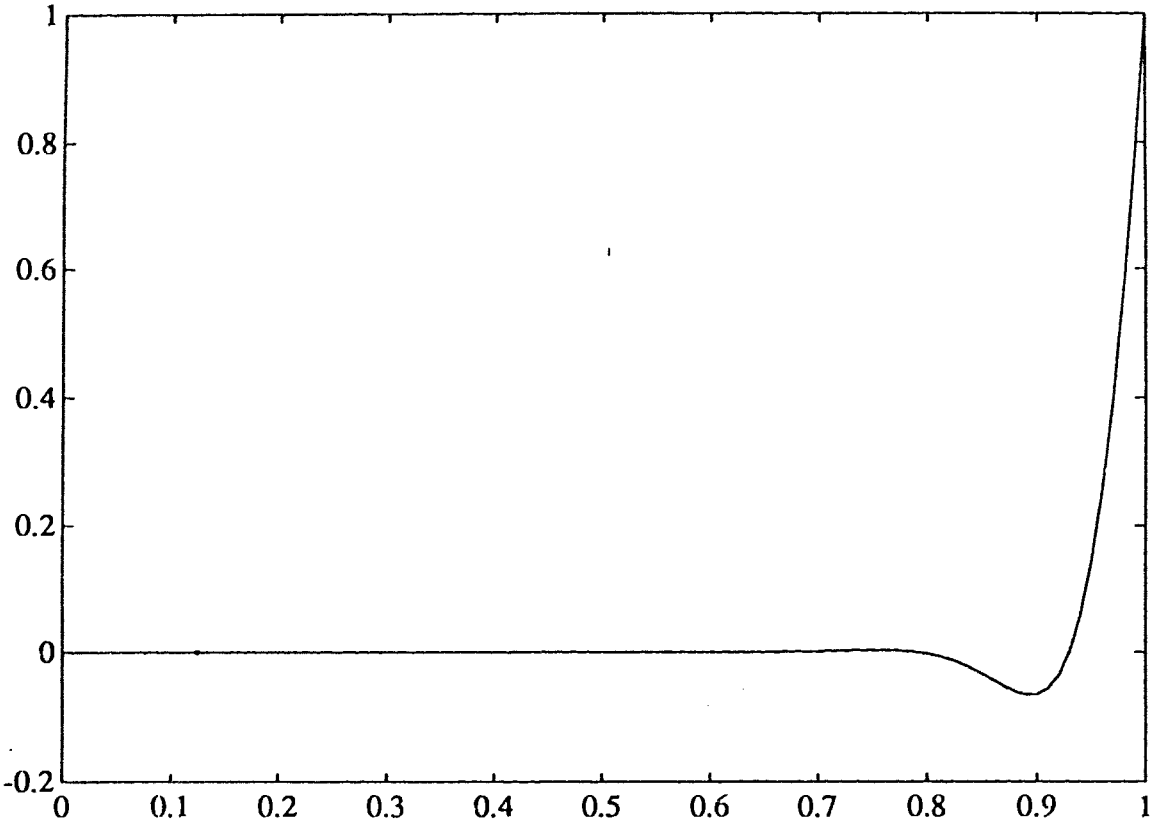


Figure 12.2 Simple DRE mesh: eps=1.e-6 atol=1.e-2 T=0.60

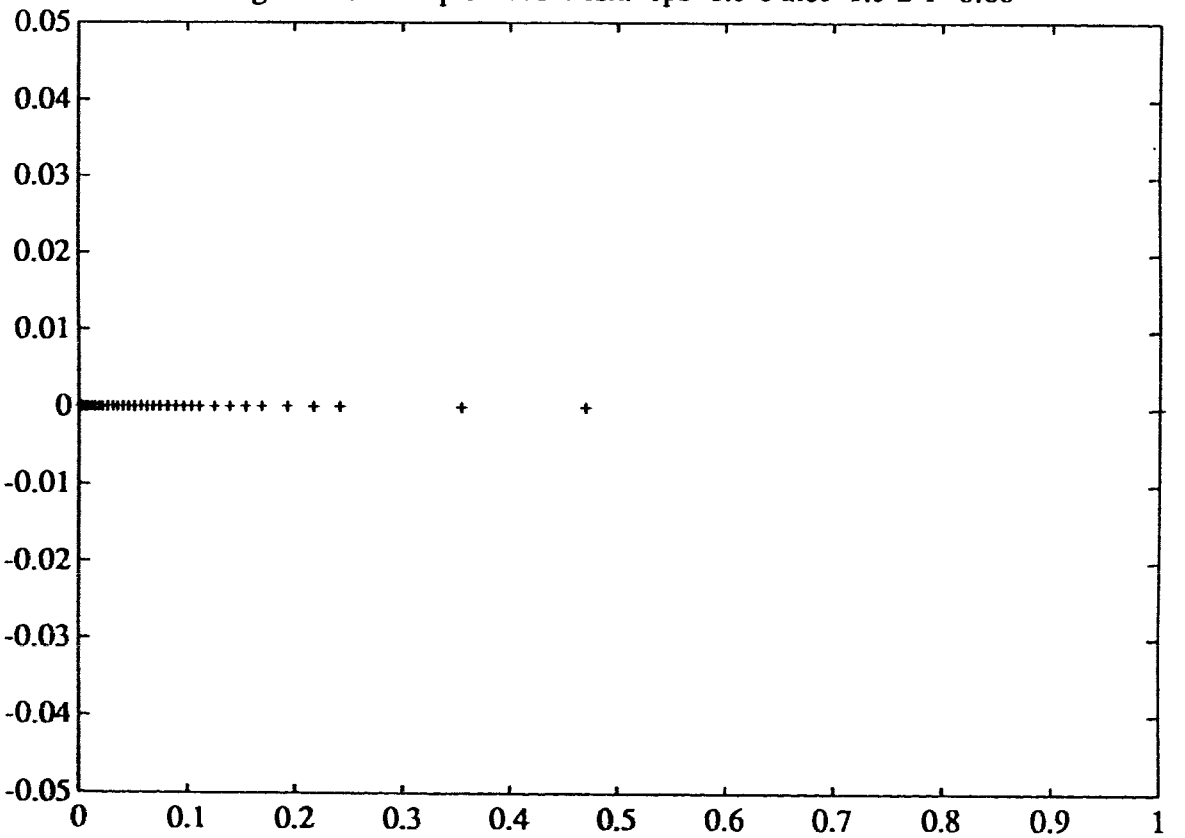


Figure 12.3 Combined DRE mesh

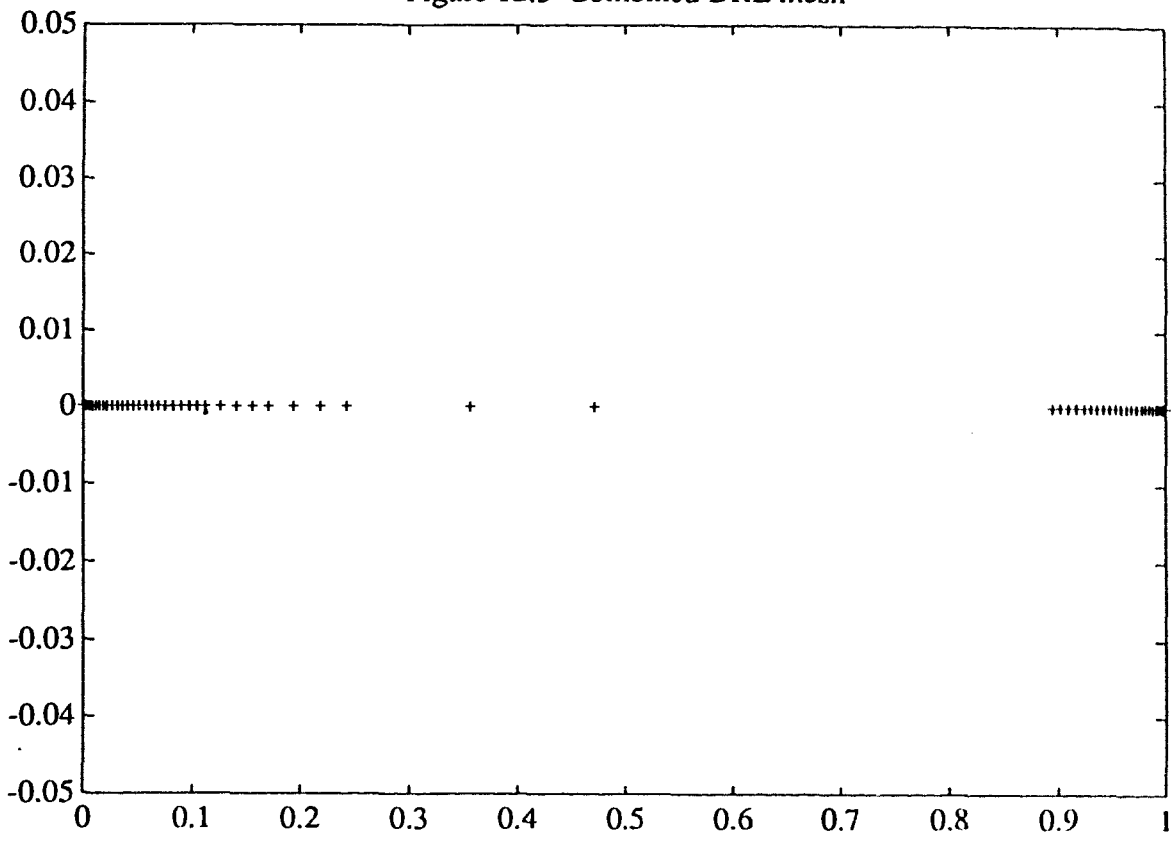
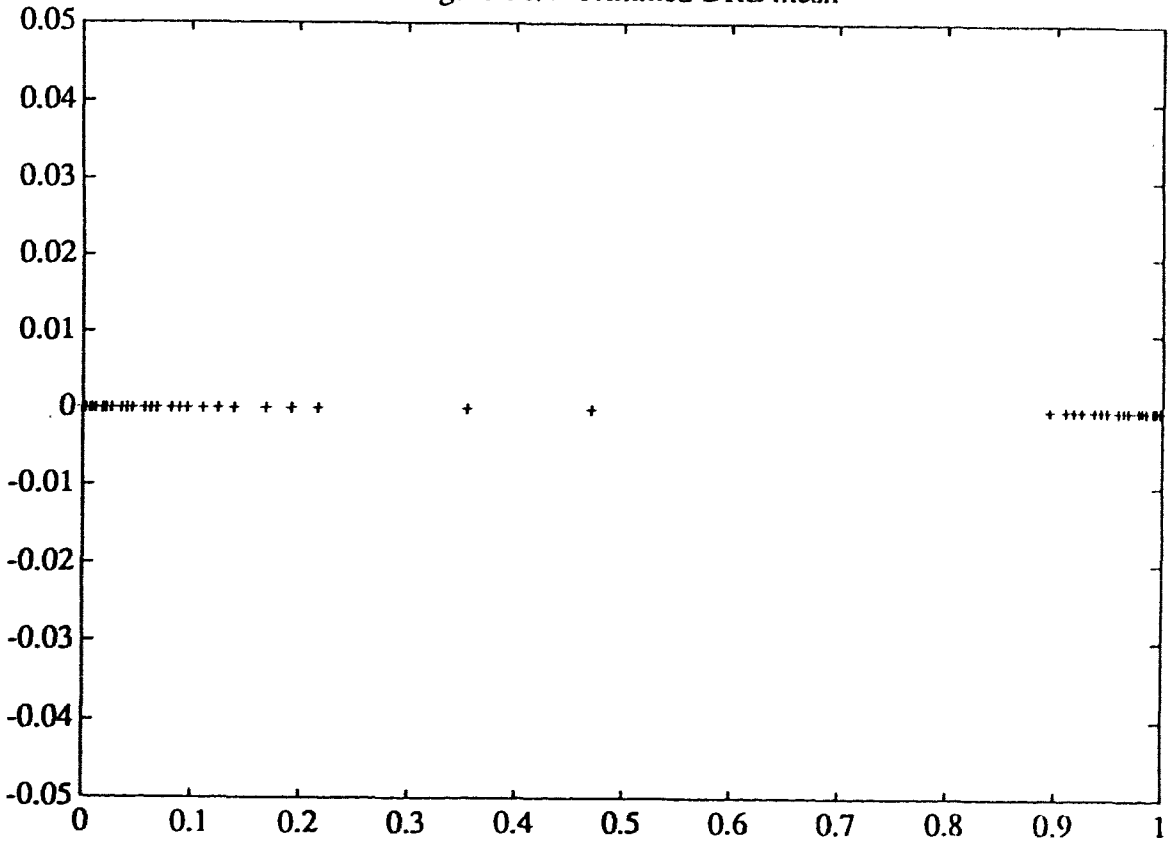


Figure 12.4 Trimmed DRE mesh



Example 13. Consider the BVP

$$\varepsilon Y' = A(t)Y + \varepsilon(A(t)g)' - g = 0, \qquad 0 < t < 1,$$

$$(\ v,\ 1\ )y(0) = 0,$$

$$(v\cos 1 - \sin 1,\ -v\sin 1 - \cos 1\ )y(1) = 0$$

$$\text{where } A(t) = \begin{bmatrix} -\theta\sin 2t & -(1+\theta\cos 2t) \\ 1-\theta\cos 2t & \theta\sin 2t \end{bmatrix}, \quad g(t) = \begin{bmatrix} \sin(\pi t) \\ \sin(\pi t) \end{bmatrix}$$

$$\theta = \mu/\sqrt{\mu^2-1}, \qquad v = \mu - \sqrt{\mu^2-1}, \qquad \mu = \pi/2$$

This is example 7 of [15]. This BVP has the exact solution $Y(t) = A(t)^{-1}g(t)$, which is smooth throughout the entire interval. The simple DRE mesh consisted of a mild artificial left boundary layer.

Table 13

ε	10^{-3}	10^{-3}	10^{-3}	10^{-6}	10^{-6}	10^{-6}
atol=rtol	10^{-2}	10^{-4}	10^{-6}	10^{-2}	10^{-4}	10^{-6}
COLNEW	10,20	10,20, 10,20	10,20,11, 22,11,22	10,20	10,20	10,20,40, 80,160,99 198,99, 198
cpu	0.63	1.22	2.28	0.60	0.60	17.59
y-err	0.37d-3	0.86d-4	0.54d-6	0.12d-5	0.12d-5	0.12d-5
DREmesh +colnew	27,14,28	69,35,70	133,67, 134	33,17,34	76,88,76	155,78, 156
cpu	1.39	3.29	5.8	1.73	3.62	6.73
y-err	0.16d-4	0.58d-9	0.13d-10	0.12d-5	0.12d-5	0.15d-5
DREmesh double	27,54	69,138	133	33,66	76,152	155
cpu	1.61	3.94	--------	2.18	4.57	--------
y-err	0.89d-4	0.33d-5	--------	0.11d-5	0.12d-5	--------

Figure 13.1 Solution

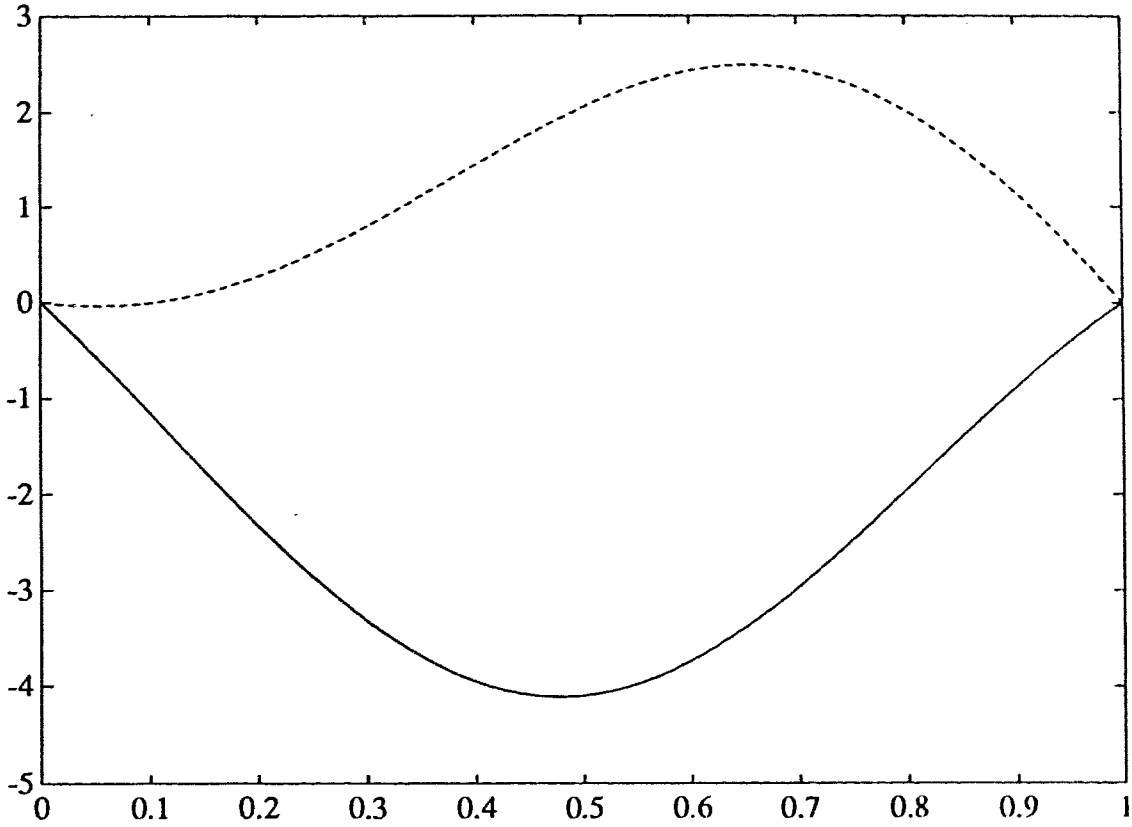
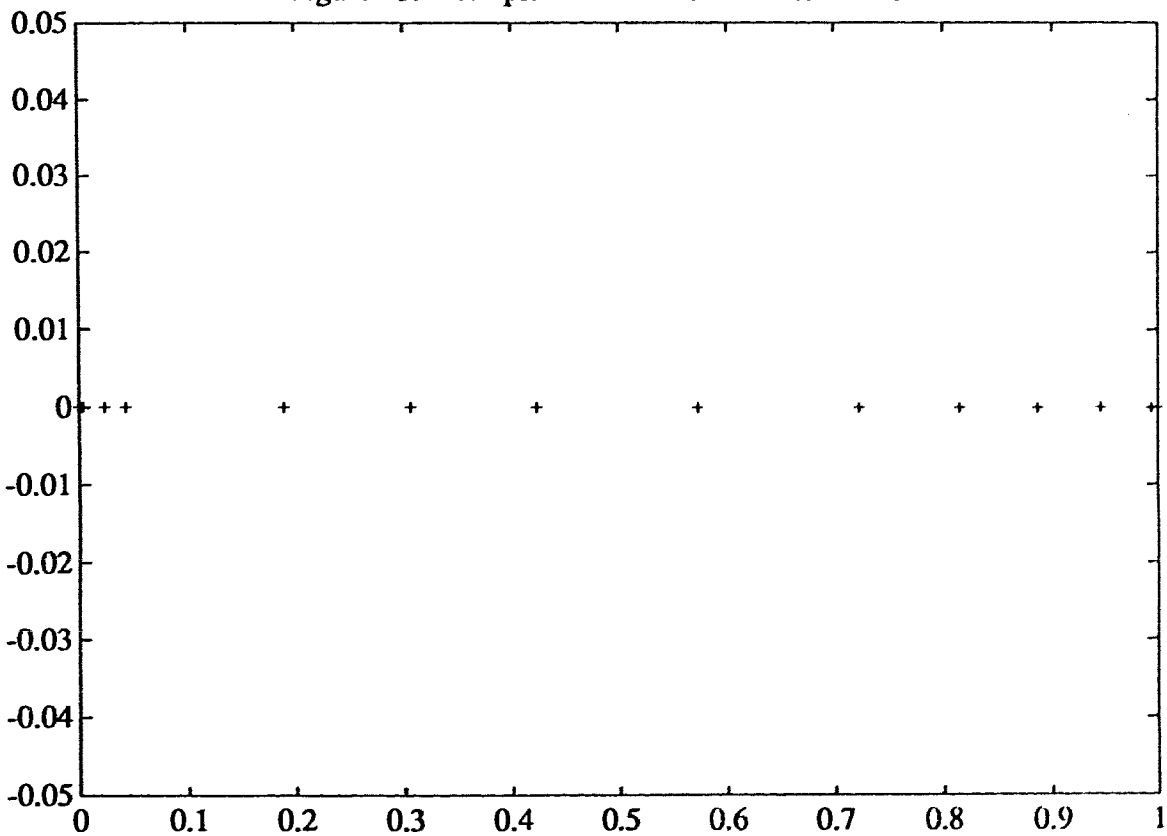


Figure 13.2 Simple DRE mesh: atol=1.e-2 T=0.28



Example 14. Consider the BVP

$$u^{(4)} - 3u''' - 63u'' - 85u' + 150u = -1500t + 15850, \qquad 0 < t < 1$$

$$u(0) = 100, \quad u''(0) = 0,$$

$$u(1) = 90, \quad u'''(1) = 0.$$

This is example 9 of [15]. This BVP has the exact solution $u(t) = 100-10t$, which is smooth throughout the whole interval. The DRE is given by the reduction $y = (u''', u', u'', u)^T$. The simple DRE mesh consisted of a mild left artificial boundary layer.

Table 14

atol=rtol	10^{-2}	10^{-4}	10^{-6}
COLNEW	10,20	10,20	10,20
cpu	1.08	1.08	1.08
y-err	0.16d-12	0.16d-12	0.16d-12
DRE mesh +colnew	27,14,28	85,43,86	146,73,146
cpu	2.4	5.19	6.07
y-err	0.24d-12	0.63d-12	0.14d-12
DRE mesh double	27,54	85	146
cpu	2.82	--------	--------
y-err	0.23d-12	--------	--------

Figure 14.1 Solution

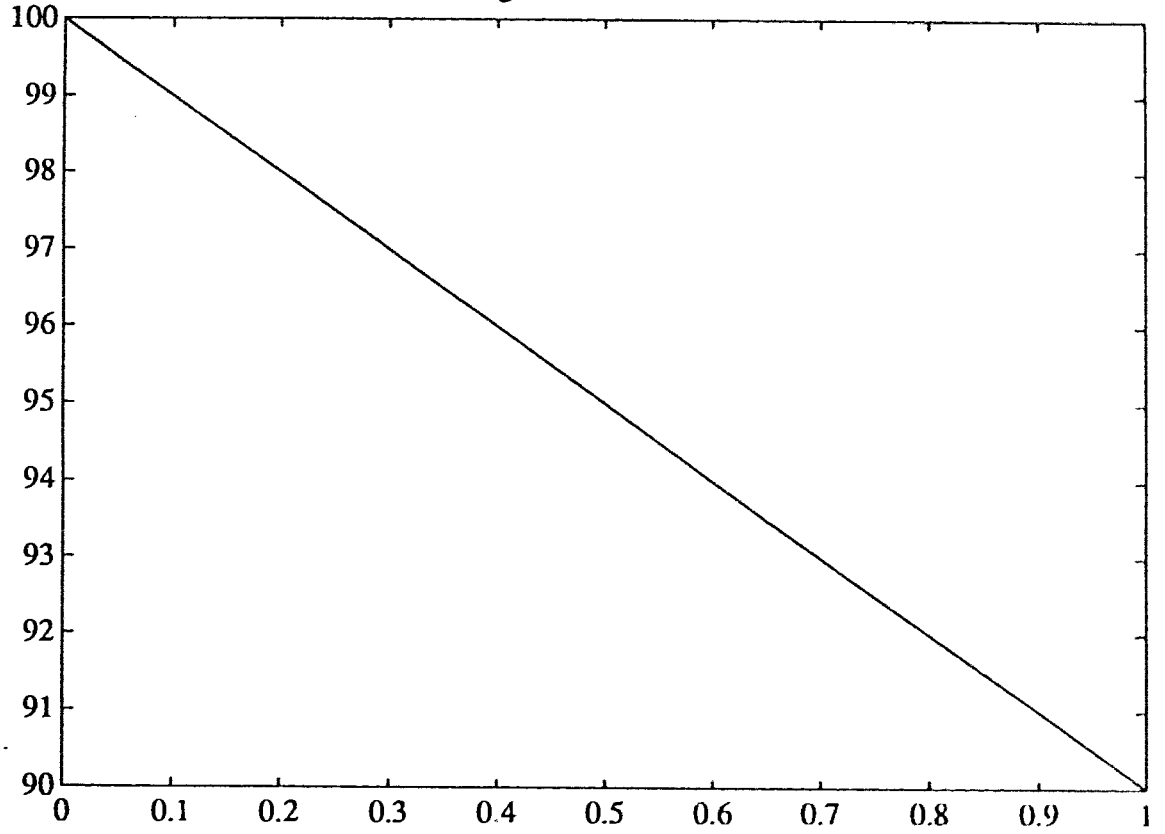
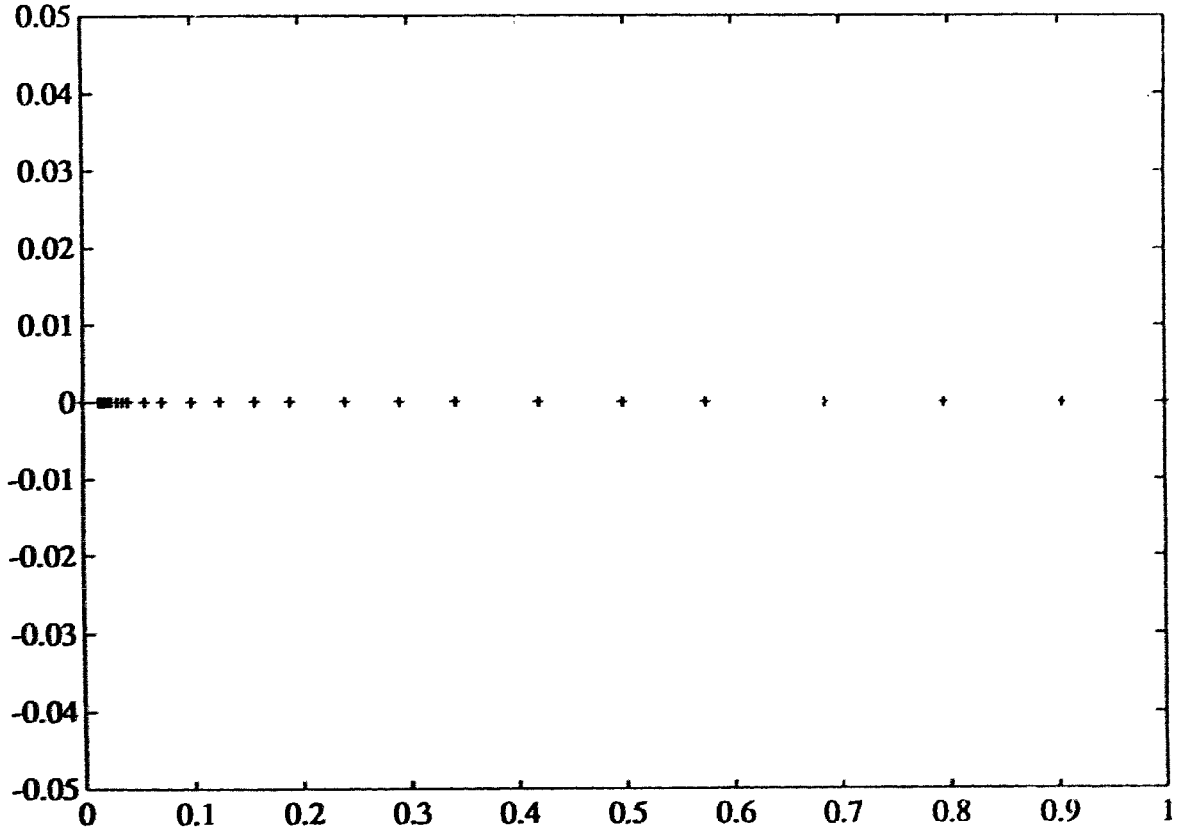


Figure 14.2 Simple DRE mesh: atol=1.e-2 T=0.44



5.2 Combined DRE mesh

From the examples in this section, we can see that the combined DRE mesh is a good mesh for a given stiff BVP, except for some cases where there may be too many mesh points. For example 6 with $\varepsilon=10^{-6}$, we get the combined DRE mesh with atol=rtol $=10^{-2}$. The combined DRE mesh detects the layer information and consists of 271 subintervals, which is a little expensive for COLNEW. The visual pattern of this combined DRE mesh is given in figure 6.3.

For example 8 with $\varepsilon=10^{-6}$, we get the combined DRE mesh with atol=rtol $=10^{-2}$. The combined DRE mesh detects the layer information of the BVP and consists of 91 subintervals. With this mesh, COLNEW spent 2.31" to achieve the accuracy 10^{-10} (we requested 10^{-6}). The visual pattern of this combined DRE mesh is given in figure 8.3.

For example 9 with $\varepsilon=10^{-6}$, we get the combined DRE mesh with atol=rtol $=10^{-2}$. The combined DRE mesh detects the layer information of the BVP and consists of 150 subintervals. With this mesh, COLNEW spent 3.78" to achieve the accuracy 10^{-9} (we requested 10^{-6}). The visual pattern of this combined DRE mesh is given in figure 9.3.

For example 11 with $\varepsilon=10^{-6}$, we get the combined DRE mesh with atol=rtol $=10^{-2}$. The combined DRE mesh detects the layer information of the BVP and consists of 186 subintervals. With this mesh, COLNEW spent 3.6" to achieve the accuracy 10^{-8} (we requested 10^{-6}). The visual pattern of this combined DRE mesh is given in figure 11.3.

For example 12 with $\varepsilon=10^{-6}$, we get the combined DRE mesh with atol=rtol $=10^{-2}$. The combined DRE mesh detects the layer information of the BVP and consists of 65 subintervals. With this mesh, COLNEW spent 5.14" to achieve the accuracy 10^{-9} (we requested 10^{-6}). The visual pattern of this combined DRE mesh is given in figure 12.3.

5.3 Trimmed DRE mesh

The examples in this section are numerical experiments on the sub mesh of the combined DRE mesh, which are called trimmed DRE meshes. For example 6 with $\varepsilon=10^{-6}$ we get a trimmed DRE mesh of 49 subintervals. With this mesh, COLNEW spent 13.26" to achieve the accuracy 10^{-4} for u and 10^{-7} for z(we requested 10^{-6}). We get another trimmed DRE mesh of 74 subintervals. With this mesh, COLNEW spent 14" to achieve accuracy 10^{-6} for u and 10^{-8} for z (we requested 10^{-6}). The visual patterns of these combined DRE meshes are given in figure 6.4, 6.5. The difference between these two trimmed meshes suggested that for BVPs with narrow layers, the mesh should not only detect right layer information, but also have enough mesh points in the layer regions.

For example 8 with $\varepsilon = 10^{-6}$, we get a trimmed DRE mesh of 49 subintervals. With this mesh, COLNEW spent 1.25" to achieve accuracy 10^{-7} (we requested 10^{-6}). The visual pattern of this trimmed DRE mesh is given in figure 8.4.

For example 9 with $\varepsilon = 10^{-6}$, we get a trimmed DRE mesh of 49 subintervals. With this mesh, COLNEW spent 2.29" to achieve accuracy 10^{-6} (we requested 10^{-6}). The visual pattern of this trimmed DRE mesh is given in figure 9.4.

For example 11 with $\varepsilon = 10^{-6}$, we get a trimmed DRE mesh of 49 subintervals. With this mesh, COLNEW spent 10.34" to achieve accuracy 10^{-10} (we requested 10^{-6}). The increase in time spent by COLNEW when compare with the combined DRE mesh means that this BVP has a very narrow layer (which is 10^{-3}). The visual pattern of this trimmed DRE mesh is given in figure 11.4.

For example 12 with $\varepsilon = 10^{-6}$, we get a trimmed DRE mesh of 49 subintervals. With this mesh, COLNEW spent 3.87" to achieve accuracy 10^{-8} (we requested 10^{-6}). The visual pattern of this trimmed DRE mesh is given in figure 12.4.

5.4 Future Work

There is still a lot of work to be done concerning the Riccati differential equation. One logical extension of this thesis is to implement the reimbedding strategy for DRESOL and perform some more numerical experiments.

We had only considered linear BVPs with separated BCs. If the BVP has non-separated BCs, the Riccati transformation still helps. However, it is not clear how to determine the dimension of it, i.e. find out the dimension of nonincreasing and nondecreasing subspaces.

The extension of the Riccati transformation to the nonlinear case is a natural idea, where a quasilinearization procedure has to be used.

6. REFERENCES

1. V. Ascher, M.R. Osborne and R.D. Russell
   A collocation solver for mixed order systems of boundary value problems
   Math. Computation, 33(1979), pp.659-674

2. M. Lentini, M.R. Osborne and R.D. Russell
   The close relationship between methods for solving TPBVPs
   SIAM J. Numer. Anal., 22(1985), pp.280-309

3. H.-O. Kreiss, N. Nichols and D.L. Brown
   Numerical method for stiff TPBVP
   SIAM J. Numer. Anal., 23(1986), pp.325-368

4. M.R. Osborne and R.D. Russell
   The Riccati transformation in the solution of BVPs
   SIAM J. Numer. Anal., 23(1986), pp.1023-1033

5. L. Dieci, M.R. Osborne and R.D. Russell
   A Riccati transformation method for solving BVPs. I Theoretical aspects
   SIAM J. Numer. Anal., 25(1988), pp.1055-1073

6. L. Dieci, M.R. Osborne and R.D. Russell
   A Riccati transformation method for solving BVPs. II Computational aspects
   SIAM J. Numer. Anal., 25(1988), pp.1074-1092

7. U.M. Ascher and R.M.M. Mattheij
   General framework, stability and error analysis for numerical stiff boundary value methods, manuscripts 1989

8. U.M. Ascher, R.M.M. Mattheij and R.D. Russell
   Numerical solution of boundary value problems for ordinary differential equations, Prentice-Hall, Englewood Cliffs, NJ, 1988

9. L. Dieci
   Some numerical considerations and Newton's method revisited for solving algebraic Riccati equations, manuscript, 1989

10. L. Dieci
    Numerical integration of the differential Riccati equation and some related issues, Submitted to SIAM J. Numer. Anal.(1990)

11. L. Dieci and D.Estep
    Some stability aspects of schemes for the adaptive integration of stiff initial value problems, manuscript, 1989

12. L.Dieci

DRESOL. A differential Riccati equations' solver: user's guide
manuscript, 1990

13. L.Dieci
On the numerical solution of differential and algebraic Riccati equations and related matters, manuscript, 1989

14. L.Dieci
Some aspects of using IV software to solve BVP via Riccati transformation
Appl. Math. & Comput., 31(1989), pp.463-472

15. L. Dieci
Theoretical and computational aspects of the Riccati transformation for solving differential equations,
Ph.D Thesis, Univ. of New Mexico, Albuquerque, NM(1986)

16. S. Bramley, L. Dieci and R.D. Russell
Numerical solution of eigenvalue problems for linear boundary value ODEs
manuscript, 1989

17. Y. Kuo
Equivalences between numerical methods for solving differential equations
M.sc. Thesis, Simon Fraser University, 1983

18. I. Babuska and V. Majer
The factorization method for numerical solution of two point boundary value problems for linear ODE's, SIAM J. Numer. Anal., 24(1987), pp1301-1334

19. G. Dahlquist, L. Edsberg, Gsköllermo and G. Söderlind
Are the numerical method and software satisfactory for chemical kinetics?
in ***Numerical Integration of Differential Equation and Large Linear Systems***
J. Hinze ed. Springer-Verlag, 1982

20. G.H. Meyer
Continuous orthogonalization for boundary value problems
J. Comp. Physics, 62(1986), pp.248-262

21. L.N. Trefethen
A course in finite difference and spectral method, manuscripts
Department of Mathematics, MIT, 1988

22. W.T. Reid
Riccati differential equation, Academic Pres, NY, 1972

23. G. H. Meyer
Initial value methods for boundary value problems, Academic Press, 1973

24. M. Scott
Invariant imbedding and its applications to ODEs, Addison-Wesley, 1973

25. R.M.M. Mattheij and G.W.M. Staarink
An efficient algorithm for solving general linear two point BVP
SIAM J. Sci. Statist. Comput., 5(1984), pp.745-763

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

27. A.C. Hindmarsh
LSODE and LSODL, two new initial value ODE solver
ACM Signum Newsletter 15(1980)

28. J. Taufer
On the factorization method, Apl. Mat., 11(1966), pp.427-450