Winter 1990

A Generalization of Linear Multistep Methods

Leon Arriola
Old Dominion University

A GENERALIZATION OF

LINEAR MULTISTEP METHODS

by

Leon Arriola

B.S., Idaho State University, 1981

A Dissertation Submitted to the Faculty of Old Dominion University
in Partial Fulfillment of the Requirements for the Degree of

DOCTOR OF PHILOSOPHY
in
COMPUTATIONAL AND APPLIED MATHEMATICS

OLD DOMINION UNIVERSITY
December 1990

Approved by:

John H. Heinbockel (Director)

Abstract

A Generalization of Linear Multistep Methods

Leon Michael Arriola

Old Dominion University, 1990

Director: Dr. John Heinbockel

A generalization of the methods that are currently available to solve systems of ordinary differential equations is made. This generalization is made by constructing linear multistep methods from an arbitrary set of monotone interpolating and approximating functions. Local truncation error estimates as well as stability analysis is given. Specifically, the class of linear multistep methods of the Adams and BDF type are discussed.

To my wife, Patty

and my parents, Leon and Ann

Acknowledgements

I would like to first thank my advisor John Heinbockel, who, without a doubt epitomizes the phrase: "gentleman and a scholar." His understanding, patience, encouragement and genuine care will not be soon forgotten. I would also like to thank Mark Dorrepaal and John Tweed for their excellent suggestions, comments and assistance.

It has been an honor to have been associated with NASA Langley Research Center. Specifically, I wish to thank Robert Costen, William Meador, John Wilson and Edward Conway for not only allowing me to pursue this particular direction of research, but encouraging it.

I would next like to thank Surendra Tiwari for the moral as well as financial support provided through the Mechanical Engineering department. Additionally, I wish to thank Barbara Jeffrey for her assistance in helping to prepare this manuscript.

Finally, I want to thank my wife Patty and my parents Leon and Ann for standing by me through these long and tedious years.

Contents

List of Tables

List of Figures

Chapter 1

Preliminaries

1.1 Introduction

One of the fastest growing fields in applied mathematics research today is that of nonlinear dynamical systems. The reason for this interest is the surprisingly rich and beautiful behavior exhibited by relatively simple systems. Phenomena such as chaos, fractals and strange attractors have created much excitement in fields as diverse as weather forecasting and cardiology [17, 29, 33]. As this interest increases, there will exist a greater need for developing a systematic and concise method for studying nonlinear dynamical systems.

The question immediately arises, how does one study a dynamic system? An answer to this question is given by the following procedure.

1. **Formulate the physical problem.** Decide what are the interesting and

important mechanisms of the physical problem and construct a mathematical model that encompasses them.

2. **Solve the mathematical problem.** Usually this requires that some sort of numerical method be employed in order to generate the solution. The numerical method is assumed to accurately and efficiently produce the desired solution. Furthermore, the method used should not contribute behavior that interferes with or obscures the true solution.

3. **Predict new phenomena.** Assuming the solution accurately fits the experimental data, the question becomes, can we extract or predict new phenomena from the numerical solution? If this new information can be verified by physical experiment, the whole analysis most likely will be considered a success.

4. **Reformulate the problem.** Suppose the mathematical solution is unrealistic, that is it does not agree with experimental data, or suppose the solution predicts physically impossible results. The obvious progression would be to go to the first stage and revaluate the assumptions made in the formulation of the mathematical model. Once the appropriate and necessary assumptions have been corrected, the entire analysis procedure is now repeated.

This dissertation concentrates on the second stage of the procedure, namely

the numerical methods used in solving nonlinear dynamical systems.

Since many dynamical systems are formulated in terms of differential equations, the numerical and computational aspects of solving nonlinear ordinary differential equations is of paramount importance. As such, a generalization of the methods that are currently available to solve systems of ordinary differential equations is made. This generalization is made by constructing linear multistep methods from an arbitrary set of monotone interpolating and approximating functions. Local truncation error estimates as well as stability analysis is given. Specifically, the class of linear multistep methods of the Adams and BDF type are discussed.

1.2 Organization

The second chapter answers the question, can we generalize the linear multistep methods, and if so, what are the resulting formulae? The reason for this generalization is *not* for some presumed abstract beauty resulting from page after page of mathematical derivation. Instead, the generalization of currently available methods allows a choice in applying more efficient methods to specific problems.

In order to motivate the reason for this generalization, consider a representative physical system. For this specific example, suppose we have a closed system in which chemical species react with one another. The physicist or chemist

needs to know how much of each individual species is present as the experiment progresses in time. In order to do this, a system of coupled nonlinear ordinary differential equations is formulated, whose solution represents the concentration of each species as a function of time.

Since the system is closed, that is, it is somehow isolated from unwanted external influences, qualitative aspects of the solution can be determined before any numerical or computational procedure is begun.

For instance, the system obviously contains finite amounts of each species. Intuition tells us that the sudden "magical" appearance of an infinite quantity of a particular species is nonsense. We may therefore conclude that the solutions must remain bounded for all time.

Furthermore, since several species are present, it is very likely there exists some particular species that are critical to the reactions occurring. If the necessary species are exhausted, obviously the reaction stops. This condition implies the solution may have limiting or asymptotic behavior.

Now suppose that a catalyst is present. Since catalysts create rapid changes as well as possibly creating new pathways in the reactions, we conclude that the solution may have some boundary layer behavior. Finally, different species may act in the roles of predator or prey. As one species increases another species decreases until come critical value is reached. Once this threshold is attained, the roles of predator and prey are now reversed. Here, the qualitative behavior of the solution would be of an oscillatory nature.

With all of this a priori knowledge of the behavior of the system, one would hope to be able to use this information in the numerical method. This however is not the usual case [1, 8]. Classical linear multistep methods are based on the idea of approximation by interpolation [18, 24]. A polynomial in time t is passed through various data points. Using this polynomial a linear multistep method is then constructed. The method is now used to approximate the future solution by a linear extrapolation.

Since polynomial based methods are well understood, they are always used in black box type differential equation solvers such as the GEAR and LSO type packages [14, 15, 19]. The generalization produced in Chapter Two allows functions other than polynomials to generate the numerical method. For instance, in the example above, boundary layer behavior suggests the use of exponential approximating functions rather than polynomial functions. Similarly, trigonometric functions instead of polynomials might better represent simple oscillatory behavior.

In Chapter Three, we discuss the accuracy of the methods constructed in Chapter Two. The concept of local truncation error is examined. Here, it is assumed that the actual solution is known to infinite accuracy for times up to but not including time $t = t_n$. After the approximation or prediction is performed, an estimate of the local error at time $t = t_n$ is made. A formula is given in terms of the arbitrary functions used in the construction of the methods.

The question of stability is addressed in Chapter Four. A numerical method

is considered to be stable if it does not excessively propagate errors. If in fact the method dampens errors as repeated application of the method is made, we would say the method is very stable. However, if errors are amplified, the method would justifiably be called unstable. These concepts are quantitatively defined in Chapter Four. Furthermore, a generalization of the classical stability analysis is given along with a comparison of stability regions for various approximating functions. Certain classes of nonpolynomial functions are shown to have larger regions of stability.

Chapter Five discusses the numerical application of some of the methods derived in Chapter Two. Several test problems are examined and solved using these various methods. In almost all cases it is shown that the test problems are better solved by nonpolynomial methods.

Chapter 2

Construction

We wish to solve the initial value problem IVP

$$y'(t) = f[y(t);t] \tag{2.0.1}$$

for $y(t)$ defined over a finite interval $[a,b]$, where $y(t)$ satisfies the initial condition

$$y(a) = y_0. \tag{2.0.2}$$

The function f is assumed to satisfy the conditions:

1. $f[y(t);t]$ is continuous for all $a \leq t \leq b$ and $-\infty < y(t) < \infty$.

2. f is Lipschitz continuous.

The second condition requires that there exist a constant L so that

$$\| f[y;t] - f[y^*;t] \| \leq L \| y - y^* \| \quad \forall t \in [a,b] \quad \text{and} \quad \forall y, y^* \in (-\infty,\infty).$$

With these assumptions, the existence and uniqueness of a differentiable solution satisfying the given initial condition is guaranteed [18].

In order to simplify the notation, it is also assumed that the IVP (2.0.1) is a scalar problem. The justification for this is that all proofs for the scalar case can easily be extended to the multidimensional case, with the exception that an unnecessarily cumbersome notation would be introduced.

The IVP (2.0.1) will be solved numerically by generating a sequence of approximates to $y(t)$ at a discrete set of points in the interval $[a, b]$. This is done by partitioning the interval $[a, b]$ and creating a set of mesh points

$$\{t_j \mid t_j < t_{j+1}, \quad j = 0, 1, \ldots, N-1; \quad t_0 = a,\, t_N = b\}.$$

The approximation to $y(t)$ and $f[y(t); t]$ at $t = t_{n-j}$ will be denoted by y_{n-j} and y'_{n-j} respectively.

The purpose of this chapter is to generate an approximation y_n to $y(t_n)$. The approximation is formed by constructing a linear difference equation

$$y_n = \sum_{j=1}^{k_1} \alpha_j y_{n-j} + \sum_{j=0}^{k_2} \beta_j y'_{n-j} \tag{2.0.3}$$

where the data values y_{n-j} and y'_{n-j} are assumed to be known, or can at least be estimated by some suitable means. The parameters α_j and β_j are as of yet unspecified, and the following sections will discuss how they are chosen. The difference equation (2.0.3) is calculated for three special cases:

1. If $\alpha_2 = \cdots = \alpha_{k_1} = \beta_0 = 0$, then (2.0.3) reduces to

$$y_n = \alpha_1 y_{n-1} + \sum_{j=1}^{k_2} \beta_j y'_{n-j}, \tag{2.0.4}$$

which is referred to as the Adams–Bashforth method.

2. If $\alpha_2 = \cdots = \alpha_{k_1} = 0$ and $\beta_0 \neq 0$, then (2.0.3) reduces to the implicit method

$$y_n = \alpha_1 y_{n-1} + \sum_{j=0}^{k_2} \beta_j y'_{n-j}, \tag{2.0.5}$$

which is referred to as the Adams–Moulton method.

3. If $\beta_1 = \cdots = \beta_{k_2} = 0$ and $\beta_0 \neq 0$, then (2.0.3) reduces to the implicit scheme

$$y_n = \sum_{j=1}^{k_1} \alpha_j y_{n-j} + \beta_0 y'_n, \tag{2.0.6}$$

which is referred to as the Backward Differentiation formula (BDF).

Notice that in all three cases the y_n is found by a linear combination of the known data values y_{n-j} and y'_{n-j}. In the case of the Adams methods, the information used to predict y_n is weighted toward derivative information, while in the case of the BDF methods more weight is given to the function information. This difference is of major importance when comparing methods that are applied to stiff problems [3, 15, 16, 24, 27, 28].

In the following sections we construct only the Adams–Moulton methods since the Adams–Bashforth methods are just a special case. Additionally, we

require that $k_1 = k_2 = k$, so for the Adams method we construct

$$y_n = \alpha_1 y_{n-1} + \sum_{j=0}^{k} \beta_j y'_{n-j},$$

and for the BDF methods we form

$$y_n = \sum_{j=1}^{k} \alpha_j y_{n-j} + \beta_0 y'_n.$$

2.1 Adams Methods

For the Adams methods, the difference scheme

$$y_n = \alpha_1 y_{n-1} + \sum_{j=0}^{k} \beta_j y'_{n-j} \tag{2.1.7}$$

is constructed by forming a function $\Phi_{k+1} \in C^2[t_{n-k}, t_n]$ which interpolates y_{n-1} at $t = t_{n-1}$ and whose derivative $\Phi'_{k+1}(t)$ interpolates $y'_n, y'_{n-1}, \ldots, y'_{n-k}$ at the points where $t = t_n, t_{n-1}, \ldots, t_{n-k}$ respectively. Once the function $\Phi_{k+1}(t)$ is completely specified, it is used to approximate $y(t_n)$.

We let $\phi_i \in C^2[t_{n-k}, t_n]$, for $i = 0, \ldots, (k+1)$, be $(k+2)$ given linearly independent and monotone functions. We then define the interpolating and approximating function by

$$\Phi_{k+1}(t) = \xi_0\phi_0(t) + \xi_1\phi_1(t) + \xi_2\phi_2(t) + \cdots + \xi_{k+1}\phi_{k+1}(t), \tag{2.1.8}$$

where

$$\phi_0(t) = 1,$$

and $\xi_0, \ldots, \xi_{k+1}$ are constants to be determined. We are therefore approximating $C^2[t_{n-k}, t_n]$ from the subspace generated by

$$Span\{1, \phi_1(t), \ldots, \phi_k(t)\}$$

and we will often refer to $\phi_i(t)$ as basis functions.

Since there are $(k+2)$ unknown constants ξ_i, for $i = 0, \ldots, (k+1)$, we require the $(k+2)$ interpolating conditions

$$\Phi_{k+1}(t_{n-1}) = y_{n-1} \tag{2.1.9}$$

and

$$\Phi'_{k+1}(t_{n-j}) = y'_{n-j}, \qquad j = 0, 1, \cdots, k. \tag{2.1.10}$$

Once the $\xi_0, \xi_1, \ldots, \xi_{k+1}$ are specified using these conditions, the approximating or predicting condition is given by

$$y_n = \Phi_{k+1}(t_n). \tag{2.1.11}$$

In matrix form the interpolating conditions are written as

$$\underset{\sim}{\Phi}\vec{\xi} = \vec{y} \tag{2.1.12}$$

where

$$\underset{\sim}{\Phi} = \begin{pmatrix} 1 & \phi_1(t_{n-1}) & \phi_2(t_{n-1}) & \cdots & \phi_{k+1}(t_{n-1}) \\ 0 & \phi'_1(t_n) & \phi'_2(t_n) & \cdots & \phi'_{k+1}(t_n) \\ 0 & \phi'_1(t_{n-1}) & \phi'_2(t_{n-1}) & \cdots & \phi'_{k+1}(t_{n-1}) \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & \phi'_1(t_{n-k}) & \phi'_2(t_{n-k}) & \cdots & \phi'_{k+1}(t_{n-k}) \end{pmatrix}, \tag{2.1.13}$$

$$\vec{\xi} = col(\xi_0, \xi_1, \ldots, \xi_{k+1})$$

and

$$\vec{y} = col(y_{n-1}, y'_n, y'_{n-1}, \ldots, y'_{n-k}).$$

2.1.1 Existence of the Adams Methods

In this section we will show that $det[\underset{\sim}{\Phi}]$ is nonzero and therefore a unique solution to (2.1.12) exists and is given by

$$\vec{\xi} = \underset{\sim}{\Phi}^{-1}\vec{y}. \tag{2.1.14}$$

Once the $\vec{\xi}$ is known, the approximating condition (2.1.11) is easily constructed. We first consider the special case

$$\phi_i(t) = \phi^i(t) \qquad i = 0, \ldots, (k+1), \tag{2.1.15}$$

where the superscript i denotes a power. Note that if we choose

$$\phi(t) = t, \tag{2.1.16}$$

then the resulting scheme will be the classical Adams method. At this point we assume only that $\phi(t)$ be strictly monotone on $[t_{n-k}, t_n]$. Substituting (2.1.15)

into (2.1.13) produces

$$\underset{\sim}{\Phi} = \begin{pmatrix} 1 & \phi(t_{n-1}) & \phi^2(t_{n-1}) & \cdots & \phi^{k+1}(t_{n-1}) \\ 0 & \phi'(t_n) & 2\phi(t_n)\phi'(t_n) & \cdots & (k+1)\phi^k(t_n)\phi'(t_n) \\ 0 & \phi'(t_{n-1}) & 2\phi(t_{n-1})\phi'(t_{n-1}) & \cdots & (k+1)\phi^k(t_{n-1})\phi'(t_{n-1}) \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & \phi'(t_{n-k}) & 2\phi(t_{n-k})\phi'(t_{n-k}) & \cdots & (k+1)\phi^k(t_{n-k})\phi'(t_{n-k}) \end{pmatrix}. \quad (2.1.17)$$

Expanding the $\det[\underset{\sim}{\Phi}]$ along the first column gives

$$\det[\underset{\sim}{\Phi}] = (k+1)!\det[\underset{\sim}{\mathbf{V}}_\phi]\prod_{j=0}^{k}\phi'(t_{n-j})$$

where

$$\underset{\sim}{\mathbf{V}}_\phi = \begin{pmatrix} 1 & \phi(t_n) & \phi^2(t_n) & \cdots & \phi^k(t_n) \\ 1 & \phi(t_{n-1}) & \phi^2(t_{n-1}) & \cdots & \phi^k(t_{n-1}) \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \phi(t_{n-k}) & \phi^2(t_{n-k}) & \cdots & \phi^k(t_{n-k}) \end{pmatrix}.$$

The matrix $\underset{\sim}{\mathbf{V}}_\phi$ is immediately recognized to be the Vandermonde matrix [13], and so we can write

$$\det\left[\underset{\sim}{\mathbf{V}}_\phi\right] = \prod_{\substack{j>i \\ i=0}}^{k}[\phi(t_{n-j}) - \phi(t_{n-i})].$$

Now $[\phi(t_{n-j}) - \phi(t_{n-i})]$ is nonzero for $i \neq j$, since ϕ is assumed to be strictly monotone on $[t_{n-k}, t_n]$. The monotonicity of ϕ also implies that $\phi'(t_{n-j})$ is nonzero for $j = 0, \ldots, k$, so that

$$\det\left[\underset{\sim}{\mathbf{V}}_\phi\right]\prod_{j=0}^{k}\phi'(t_{n-j}) \neq 0$$

implies

$$\det[\underset{\sim}{\Phi}] \neq 0.$$

Therefore there exists a unique set of coefficients ξ_i, for $i = 0, \cdots, (k+1)$, which satisfy the interpolation conditions (2.1.9) and (2.1.10). Let the inverse $\underset{\sim}{\Phi}^{-1}$ be denoted by

$$[\varphi_{i,j}] = \underset{\sim}{\Phi}^{-1}.$$

That is, $\varphi_{i,j}$ denotes the i, j th element of $\underset{\sim}{\Phi}^{-1}$. From (2.1.14) we have

$$\xi_i = \varphi_{i+1,1} y_{n-1} + \sum_{j=0}^{k} \varphi_{i+1,j+2} y'_{n-j}, \qquad i = 0, \ldots, (k+1).$$

Substituting ξ_i into (2.1.8) and using (2.1.11), the predicted value for $y(t_n)$ is found to be

$$y_n = \alpha_1 y_{n-1} + \sum_{j=0}^{k} \beta_j y'_{n-j} \tag{2.1.18}$$

where

$$\alpha_1 = \sum_{i=0}^{k+1} \varphi_{i+1,1} \phi^i(t_n) \tag{2.1.19}$$

and

$$\beta_j = \sum_{i=0}^{k+1} \varphi_{i+1,j+2} \phi^i(t_n), \qquad j = 0, \cdots, k, \tag{2.1.20}$$

which is exactly (2.1.7).

In the more general case of (2.1.13) we again expand the $\det[\underset{\sim}{\Phi}]$ along the

first column to get

$$\det[\underset{\sim}{\Phi}] = \det\begin{bmatrix} \phi_1'(t_n) & \phi_2'(t_n) & \cdots & \phi_{k+1}'(t_n) \\ \phi_1'(t_{n-1}) & \phi_2'(t_{n-1}) & \cdots & \phi_{k+1}'(t_{n-1}) \\ \vdots & \vdots & & \vdots \\ \phi_1'(t_{n-k}) & \phi_2'(t_{n-k}) & \cdots & \phi_{k+1}'(t_{n-k}) \end{bmatrix}.$$

In order to show that $\det[\underset{\sim}{\Phi}]$ is nonzero we prove the following theorem.

Theorem 2.1 *Let $\{\phi_1, \ldots, \phi_{k+1}\}$ be a set of linearly independent functions with $\phi_i \in C^2[t_{n-k}, t_n]$ and each ϕ_i nonconstant on the interval $[t_{n-k}, t_n]$, then*

$$\exists \quad \tau_0, \tau_1, \ldots, \tau_k \in [t_{n-k}, t_n],$$

with $\tau_i \neq \tau_j$ for $i \neq j$ so that

$$\det\begin{bmatrix} \phi_1'(\tau_0) & \phi_2'(\tau_0) & \cdots & \phi_{k+1}'(\tau_0) \\ \phi_1'(\tau_1) & \phi_2'(\tau_1) & \cdots & \phi_{k+1}'(\tau_1) \\ \vdots & \vdots & & \vdots \\ \phi_1'(\tau_k) & \phi_2'(\tau_k) & \cdots & \phi_{k+1}'(\tau_k) \end{bmatrix} \neq 0.$$

With this result we will then assume that the given mesh points $t_{n-k}, \ldots, t_n$ have been chosen to be some rearrangement of $\tau_0, \ldots, \tau_k$. The proof of this theorem is obtained by the following propositions.

Proposition 2.1 *Let $\mathcal{D} : C^2[t_{n-k}, t_n] \longmapsto C^1[t_{n-k}, t_n]$ be the linear differential operator defined by $\mathcal{D}[\phi] = \phi'(t)$, then*

$$\phi \sim \varphi \Longleftrightarrow \exists c \in \Re \quad \ni \quad \phi(t) = \varphi(t) + c \quad \forall t \in [t_{n-k}, t_n]$$

defines an equivalence relation, with $\mathcal{N}ull(\mathcal{D}) = \{0\}$ and therefore $\mathcal{D}^{-1}$ exists.

For ease of notation, the equivalence class $[[\phi]]$ will be denoted by ϕ.

Proposition 2.2 *If $\{\phi_1, \ldots, \phi_{k+1}\}$ are linearly independent nonconstant functions in $Dom(\mathcal{D})$ and if $\mathcal{D}^{-1}$ exists, then $\{\mathcal{D}[\phi_1], \ldots, \mathcal{D}[\phi_{k+1}]\} = \{\phi'_1, \ldots, \phi'_{k+1}\}$ are also linearly independent in $C^1[t_{n-k}, t_n]$.*

Proposition 2.3 *$\{\phi'_1, \ldots, \phi'_{k+1}\}$ is a set of linearly independent functions defined on $[t_{n-k}, t_n]$ if and only if*

$$\exists \quad \tau_0, \ldots, \tau_k \in [t_{n-k}, t_n],$$

with $\tau_i \neq \tau_j$ for $i \neq j$, so that

$$\det[\phi'_i(\tau_j)] \neq 0.$$

A generalization of Proposition 2.2 is found in Kreyszig [22], while a generalization of Proposition 2.3 is found in Cheney [10]. Therefore we are only required to prove Proposition 2.1.

Proof:

1. Let $c = 0$, then

$$\phi(t) = \phi(t) + c \quad \forall t \in [t_{n-k}, t_n] \quad \Longrightarrow \quad \phi \sim \phi.$$

2. If $\phi \sim \varphi$, this implies

$$\exists c \quad \ni \quad \phi(t) = \varphi(t) + c \qquad \forall t \in [t_{n-k}, t_n].$$

Now

$$\varphi(t) = \phi(t) + (-c) \qquad \forall t \in [t_{n-k}, t_n] \qquad \Longrightarrow \quad \varphi \sim \phi.$$

3. If $\phi \sim \varphi$, this implies

$$\exists c_1 \quad \ni \quad \phi(t) = \varphi(t) + c_1 \qquad \forall t \in [t_{n-k}, t_n]$$

and if $\varphi \sim \psi$, then this implies

$$\exists c_2 \quad \ni \quad \varphi(t) = \psi(t) + c_2 \qquad \forall t \in [t_{n-k}, t_n].$$

Therefore,

$$\exists c = c_1 + c_2 \quad \ni \quad \phi(t) = \psi(t) + c \qquad \forall t \in [t_{n-k}, t_n],$$

from which we conclude

$$\phi \sim \psi.$$

From (1) – (3), $\sim$ defines an equivalence relation.

4. Let

$$\phi \in \mathcal{N}ull(\mathcal{D}).$$

Then $\mathcal{D}[\phi] = 0$ implies

$$\phi'(t) = 0 \qquad \forall t \in [t_{n-k}, t_n].$$

Integrating we get

$$\int_{s=t_{n-k}}^{t} d\phi(s) = 0 \qquad \forall t \in [t_{n-k}, t_n],$$

from which

$$\phi(t) = 0 + \phi(t_{n-k}) \qquad \forall t \in [t_{n-k}, t_n].$$

So

$$\exists c = \phi(t_{n-k}) \quad \ni \quad \phi(t) = 0 + c \qquad \forall t \in [t_{n-k}, t_n].$$

Therefore

$$\phi \sim 0.$$

Hence the result.

Knowing that $\det[\underset{\sim}{\Phi}]$ is nonzero, we again let $[\varphi_{i,j}]$ denote the i, j th element of $\underset{\sim}{\Phi}^{-1}$. As above we obtain the results

$$y_n = \alpha_1 y_{n-1} + \sum_{j=0}^{k} \beta_j y'_{n-j}, \tag{2.1.21}$$

where

$$\alpha_1 = \sum_{i=0}^{k+1} \varphi_{i+1,1} \phi_i(t_n), \tag{2.1.22}$$

and

$$\beta_j = \sum_{i=0}^{k+1} \varphi_{i+1,j+2} \phi_i(t_n) \qquad j = 0, \ldots, k \tag{2.1.23}$$

which is of the desired form (2.1.7).

2.1.2 Construction by Collocation

We now consider the relationship of collocation with interpolation. In order to find the coefficients $\alpha_1, \beta_0, \beta_1, \ldots, \beta_k$, for the Adams method, form the linear functional

$$\mathcal{L}[y] = \alpha_1 y(t_{n-1}) + \sum_{j=0}^{k} \beta_j y'(t_{n-j}) - y(t_n), \tag{2.1.24}$$

which will play a major role in the determination of the local truncation error to be discussed later. By collocation we require that

$$\mathcal{L}\left[\phi_i\right] = 0, \qquad i = 0, \ldots, (k+1). \tag{2.1.25}$$

This collocation condition yields the matrix equation

$$\underset{\sim}{\Phi}^T \vec{\chi} = \vec{\phi}, \tag{2.1.26}$$

where

$$\vec{\chi} = col(\alpha_1, \beta_0, \beta_1, \ldots, \beta_k)$$

and

$$\vec{\phi} = col(1, \phi_1(t_n), \phi_2(t_n), \ldots, \phi_{k+1}(t_n)).$$

Using the result $\det[\underset{\sim}{\Phi}^T] = \det[\underset{\sim}{\Phi}]$ is nonzero, this implies that $[\underset{\sim}{\Phi}^T]^{-1}$ exists. Furthermore, the matrix operations of inverse and transpose are communative, that is

$$\left[\underset{\sim}{\Phi}^T\right]^{-1} = \left[\underset{\sim}{\Phi}^{-1}\right]^T.$$

Therefore the solution to the collocation problem can be expressed in the form

$$\vec{\chi} = \left[\underset{\sim}{\Phi}^{-1}\right]^T \vec{\phi}.$$

Expanding we get

$$\alpha_1 = \sum_{i=0}^{k+1} \varphi_{i+1,1} \phi_i(t_n),$$

and

$$\beta_j = \sum_{i=0}^{k+1} \varphi_{i+1,j+2} \phi_i(t_n), \qquad j = 0, \ldots, k,$$

which agrees with the previous results obtained in (2.1.22) and (2.1.23).

We now construct an explicit formula for the coefficients α_j and β_j in the special case

$$\phi_i(t) = \phi^i(t), \qquad i = 0, \ldots, (k+1), \tag{2.1.27}$$

where $\phi(t)$ is a monotone function. In order to solve (2.1.26) we use a modification of a manipulation first made by Jacobi [7]. Let $d_{i0}, d_{i1}, \ldots, d_{i\,k+1}$, $i = 0, 1, \ldots, k+1$ be $(k+2)^2$ unspecified parameters. Now multiply the r th row of the collocation problem by $d_{i\,r-1}$ and add the resulting equations. This produces the result

$$\alpha_1 \sum_{r=0}^{k+1} d_{ir}\phi^r(t_{n-1}) + \sum_{j=0}^{k} \beta_j \phi'(t_{n-j}) \sum_{r=1}^{k+1} r d_{ir}\phi^{r-1}(t_{n-j}) = \sum_{r=0}^{k+1} d_{ir}\phi^r(t_n) \tag{2.1.28}$$

where $i = 0, \ldots, (k+1)$. Now define

$$p_i(t) = \sum_{r=0}^{k+1} d_{ir}\phi^r(t), \qquad i = 0, \ldots, (k+1), \tag{2.1.29}$$

to be a $(k+1)$ st degree semigeneralized polynomial in $\phi(t)$ of degree $\leq (k+1)$ and differentiating $p_i(t)$ to obtain

$$p_i'(t) = \phi'(t) \sum_{r=1}^{k+1} r d_{ir}\phi^{r-1}(t), \qquad i = 0, \ldots, (k+1). \tag{2.1.30}$$

The equation (2.1.28) can then be written in the form

$$\alpha_1 p_i(t_{n-1}) + \sum_{j=0}^{k} \beta_j p_i'(t_{n-j}) = p_i(t_n), \qquad i = 0, \ldots, (k+1). \tag{2.1.31}$$

We now show how to select the coefficients d_{ir}. This is done for two cases.

Case 2.1 *For $i = 0$ let*

$$p_0(t_{n-1}) = 1 \tag{2.1.32}$$

and

$$p_0'(t_{n-j}) = 0, \qquad j = 0, \ldots, k. \tag{2.1.33}$$

Then (2.1.31) simplifies to

$$\alpha_1 = p_0(t_n).$$

Using (2.1.29) we have

$$\alpha_1 = \sum_{r=0}^{k+1} d_{0r}\phi^r(t_n).$$

These results imply that

$$d_{0r} = \begin{cases} 1 & r = 0 \\ 0 & r = 1, \ldots, (k+1) \end{cases}$$

as is now demonstrated. For $t = t_{n-j}$, (2.1.30) becomes

$$p_0'(t_{n-j}) = \phi'(t_{n-j}) \sum_{r=1}^{k+1} r d_{0r}\phi^{r-1}(t_{n-j}) = 0, \qquad j = 0, \ldots, k.$$

Recall $\phi(t)$ is assumed to be a monotone function on $[t_{n-k}, t_n]$. Therefore, $\phi'(t_{n-j})$ is nonzero for $j = 0, \ldots, k$ and consequently

$$\sum_{r=1}^{k+1} r d_{0r}\phi^{r-1}(t_{n-j}) = 0, \qquad j = 0, \ldots, k.$$

This implies that

$$\sum_{r=1}^{k+1} r d_{0r}\phi^{r-1}(t)$$

is a polynomial in $\phi(t)$ of degree $\leq k$ with $(k+1)$ zeros at $t = t_n, \ldots, t_{n-k}$, and therefore must be identically zero. That is,

$$d_{0r} = 0, \qquad r = 1, \ldots, (k+1).$$

Therefore, $p_0(t) \equiv d_{00}$, so $p_0(t_{n-1}) = 1$ implies

$$\alpha_1 = d_{00} = 1.$$

With this result the linear multistep method given in equation (2.1.7) reduces to

$$y_n = y_{n-1} + \sum_{j=0}^{k} \beta_j y'_{n-j}$$

Case 2.2 *For $i = 1, \ldots, (k+1)$ let*

$$p_i(t_{n-1}) = 0 \tag{2.1.34}$$

and

$$p'_i(t_{n-j}) = \delta_{i-1,j} \qquad j = 0, \ldots, k, \tag{2.1.35}$$

where $\delta_{i-1,j}$ denotes the usual Kronecker delta function.

With these conditions (2.1.31) reduces to

$$\beta_j = p_{j+1}(t_n) \qquad j = 0, \ldots, k. \tag{2.1.36}$$

For notational purposes we define

$$\pi_q(t) = \prod_{r=0}^{q} [\phi(t) - \phi(t_{n-r})], \qquad q = 0, \ldots, k. \tag{2.1.37}$$

Differentiating and letting $t = t_{n-j}$ we have

$$\pi_q'(t_{n-j}) = \phi'(t_{n-j}) \prod_{\substack{r=0 \\ r\neq j}}^{q} [\phi(t_{n-j}) - \phi(t_{n-r})]. \tag{2.1.38}$$

Now from (2.1.35) and (2.1.30) we find

$$\phi'(t_{n-j}) \sum_{r=1}^{k+1} r d_{ir} \phi^{r-1}(t_{n-j}) = \delta_{i-1,j}, \qquad j = 0, \ldots, k.$$

Observe that the monotonicity of $\phi(t)$ implies

$$\sum_{r=1}^{k+1} r d_{ir} \phi^{r-1}(t_{n-j}) = 0 \qquad j = 0, \ldots, i-2, i, \ldots, k.$$

Since

$$\sum_{r=1}^{k+1} r d_{ir} \phi^{r-1}(t)$$

is a polynomial in $\phi(t)$ of degree k, with k roots $\phi(t_{n-j})$, where $j \neq i-1$, then

$$\sum_{r=1}^{k+1} r d_{ir} \phi^{r-1}(t) = \kappa \prod_{\substack{r=0 \\ r\neq i-1}}^{k} [\phi(t) - \phi(t_{n-r})],$$

for some constant κ. Therefore $p_i'(t)$ can be written as

$$p_i'(t) = \kappa \phi'(t) \prod_{\substack{r=0 \\ r\neq i-1}}^{k} [\phi(t) - \phi(t_{n-r})].$$

We find κ by using the condition

$$p_i'(t_{n-i+1}) = 1$$

which gives

$$\kappa = \frac{1}{\pi_k'(t_{n-i+1})}.$$

Hence $p_i'(t)$ can be rewritten as

$$p_i'(t) = \frac{\phi'(t)}{\pi_k'(t_{n-i+1})} \prod_{\substack{r=0 \\ r\neq i-1}}^{k} [\phi(t) - \phi(t_{n-r})].$$

Equating this with (2.1.30) gives

$$\sum_{r=1}^{k+1} r d_{ir} \phi^{r-1}(t) = \frac{1}{\pi_k'(t_{n-i+1})} \prod_{\substack{r=0 \\ r\neq i-1}}^{k} [\phi(t) - \phi(t_{n-r})].$$

In order to find the coefficients d_{ir} we expand

$$\prod_{\substack{r=0 \\ r\neq q}}^{k} [\phi(t) - \phi(t_{n-r})] = \sum_{r=0}^{k} {}^{q}l_{k,r} \phi^r(t) \tag{2.1.39}$$

where

$${}^{q}l_{k,r} = {}^{q}S_{k,k-r}(-1)^{k-r} \tag{2.1.40}$$

and

$${}^{q}S_{k,s} = \begin{cases} \displaystyle\sum_{\substack{r_1=0 \\ r_1\neq q}}^{k} \sum_{\substack{r_2>r_1 \\ r_2\neq q}}^{k} \cdots \sum_{\substack{r_s>r_{s-1} \\ r_s\neq q}}^{k} \phi(t_{n-r_1})\phi(t_{n-r_2})\cdots\phi(t_{n-r_s}), & s = 1,\ldots,k \\ 1, & s = 0. \end{cases} \tag{2.1.41}$$

Solving for d_{ir} we get

$$d_{ir} = \frac{{}^{i-1}l_{k,r-1}}{r\pi_k'(t_{n-i+1})}, \qquad r = 1,\ldots,(k+1).$$

Since $p_i(t_{n-1}) = 0$ we also have

$$d_{i0} = -\sum_{r=1}^{k+1} \frac{{}^{i-1}l_{k,r-1}\phi^r(t_{n-1})}{r\pi_k'(t_{n-i+1})}.$$

This implies

$$p_i(t) = \sum_{r=1}^{k+1} \frac{{}^{i-1}l_{k,r-1}}{r\pi_k'(t_{n-i+1})}[\phi^r(t) - \phi^r(t_{n-1})], \qquad i = 1,\ldots,(k+1).$$

From equation (2.1.36) we obtain

$$\beta_j = \sum_{r=1}^{k+1} \frac{{}^j l_{k,r-1}}{r\pi_k'(t_{n-j})}[\phi^r(t_n) - \phi^r(t_{n-1})]. \tag{2.1.42}$$

Now the classical Adams methods results in the special case

$$\phi(t) = t.$$

Substituting this value into (2.1.38) gives

$$\pi_k'(t_{n-j}) = (-1)^j h^k j!(k-j)!$$

from which the β_j reduces to

$$\beta_j = \frac{(-1)^{j-1}}{h^k j!(k-j)!} \sum_{r=1}^{k+1} \frac{{}^j l_{k,r-1}}{r} \sum_{s=1}^{r} \binom{r}{s} t_n^{r-s}(-h)^s. \tag{2.1.43}$$

A compilation of the β_j are given in section 2.3 of this chapter.

We now consider the general case

$$\underset{\sim}{\Phi}^T \vec{\chi} = \vec{\phi}.$$

Using Cramer's rule we find

$$\alpha_1 = 1 \tag{2.1.44}$$

and

$$\beta_j = \frac{\det \begin{bmatrix} 1 & 0 & \cdots & 1 & \cdots & 0 \\ \phi_1(t_{n-1}) & \phi_1'(t_n) & \cdots & \phi_1(t_n) & \cdots & \phi_1'(t_{n-k}) \\ \phi_2(t_{n-1}) & \phi_2'(t_n) & \cdots & \phi_2(t_n) & \cdots & \phi_2'(t_{n-k}) \\ \vdots & \vdots & & \vdots & & \vdots \\ \phi_{k+1}(t_{n-1}) & \phi_{k+1}'(t_n) & \cdots & \phi_{k+1}(t_n) & \cdots & \phi_{k+1}'(t_{n-k}) \end{bmatrix}}{\det \left[\underset{\sim}{\Phi}^T\right]} \tag{2.1.45}$$

for $j = 0, \ldots, k$, where the $j+1$ st column

$$\begin{pmatrix} 0 \\ \phi_1'(t_{n-j}) \\ \phi_2'(t_{n-j}) \\ \vdots \\ \phi_{k+1}'(t_{n-j}) \end{pmatrix}$$

has been replaced by

$$\begin{pmatrix} 1 \\ \phi_1(t_{n-j}) \\ \phi_2(t_{n-j}) \\ \vdots \\ \phi_{k+1}(t_{n-j}) \end{pmatrix}.$$

Again, a compilation of the β_j for various basis functions $\phi_i(t)$ is given in section 2.3.

2.1.3 Newton and Lagrange Interpolation

We now generate equation (2.1.7) using another interpolating procedure. Integrating equation (2.0.1) we have

$$\int_{t=t_{n-1}}^{t_n} dy(t) = \int_{t=t_{n-1}}^{t_n} f[y(t);t]\,dt = \int_{\tau=t_{n-1}}^{t_n} f^*(\tau)\,d\tau$$

where f^* is treated as a known function of t. Solving for $y(t_n)$ gives

$$y(t_n) = y(t_{n-1}) + \int_{\tau=t_{n-1}}^{t_n} f^*(\tau)\,d\tau. \tag{2.1.46}$$

We now approximate f^* on the interval $[t_{n-1}, t_n]$ by the interpolating and approximating function $\Psi_k(t)$ which interpolates the data $y'_n, y'_{n-1}, \ldots, y'_{n-k}$ at the points where $t = t_n, \ldots, t_{n-k}$ respectively, that is

$$\Psi_k(t_{n-j}) = z_j \qquad j = 0, \ldots, k \tag{2.1.47}$$

where

$$z_j = y'_{n-j} \qquad j = 0, \ldots, k. \tag{2.1.48}$$

Notice that there is a subtle difference between this interpolation scheme and the previous one. Here interpolation is required only at $y'_n, \ldots, y'_{n-k}$, whereas in the previous scheme additional interpolation is required at y_{n-1}.

The following constructions will extend the classical Newton interpolating polynomials which are generated by the $Span\left\{1, t, \ldots, t^k\right\}$, to the generalized Newton interpolating functions generated by the $Span\{1, \phi_1(t), \ldots, \phi_k(t)\}$. This will be done by modifying the derivation used in the classical case [4].

For the special case where

$$\phi_i(t) = \phi^i(t), \tag{2.1.49}$$

define $\Psi_k(t)$ to be a semigeneralized polynomial in $\phi(t)$ of degree k. We wish to write

$$\Psi_k(t) = \Psi_{k-1}(t) + C_k(t)$$

where $C_k(t)$ is a correction polynomial in $\phi(t)$ of degree k. We require that

$$\Psi_k(t_{n-j}) = \begin{cases} \Psi_{k-1}(t_{n-j}) = z_j, & j = 0, \ldots, (k-1) \\ z_k, & j = k. \end{cases}$$

These conditions give

$$C_k(t_{n-j}) = \begin{cases} 0, & j = 0, \ldots, (k-1) \\ z_k - \Psi_{k-1}(t_{n-k}), & j = k. \end{cases}$$

Now $C_k(t_{n-j}) = 0$ for $j = 0, \ldots, (k-1)$ implies there exists some constant κ so that

$$C_k(t) = \kappa\pi_{k-1}(t).$$

Using the condition that $C_k(t_{n-k}) = z_k - \Psi_{k-1}(t_{n-k})$, we solve for κ and obtain

$$\Psi_k(t) = \Psi_{k-1}(t) + \frac{z_k - \Psi_{k-1}(t_{n-k})}{\pi_{k-1}(t_{n-k})}\pi_{k-1}(t).$$

Define the kth order Newton divided difference of z by

$$z[t_n, \ldots, t_{n-k}]_\phi = \frac{z_k - \Psi_{k-1}(t_{n-k})}{\pi_{k-1}(t_{n-k})}.$$

We then write $\Psi_k(t)$ in the form

$$\Psi_k(t) = \Psi_{k-1}(t) + z[t_n, \ldots, t_{n-k}]_\phi \pi_{k-1}(t).$$

We now construct a recurrence relation for the divided difference by comparing the Lagrange form with the Newton form. The Lagrange form of $\Psi_k(t)$ is given by

$$\Psi_k(t) = \sum_{j=0}^{k} z_j \prod_{\substack{i=0 \\ i \neq j}}^{k} \frac{\phi(t) - \phi(t_{n-i})}{\phi(t_{n-j}) - \phi(t_{n-i})},$$

or using the product notation of equations (2.1.37) and (2.1.38) we obtain

$$\Psi_k(t) = \sum_{j=0}^{k} z_j \frac{\phi'(t_{n-j})\pi_k(t)}{\pi'_k(t_{n-j})[\phi(t) - \phi(t_{n-j})]}.$$

Since $\Psi_{k-1}(t)$ is a polynomial in $\phi(t)$ of degree $\leq (k-1)$ this equation implies the k th divided difference is the coefficient of the $\phi^k(t)$ term. Therefore, the explicit formula for the divided difference is given by

$$z[t_n, \ldots, t_{n-k}]_\phi = \sum_{j=0}^{k} \frac{z_j \phi'(t_{n-j})}{\pi'_k(t_{n-j})}. \tag{2.1.50}$$

It is now shown that the divided difference has the same properties as the classical divided differences. The first property considered is that the divided difference is invariant under permutation of the mesh points. The reason for showing this property is that by Theorem 2.1, the mesh points $\{t_{n-k}, \ldots, t_n\}$ were chosen to be some rearrangement of $\{\tau_0, \ldots, \tau_k\}$. As such, the ordering of meshpoints must be shown to have no effect on the resulting formula.

Observe that for finite sums the order of summation is unimportant. Let $(j_0, j_1, \ldots, j_k)$ be any permutation of $(0, 1, \ldots, k)$. Then

$$\sum_{j=0}^{k} \frac{z_j \phi'(t_{n-j})}{\pi'_k(t_{n-j})} = \sum_{i=0}^{k} \frac{z_{j_i} \phi'(t_{n-j_i})}{\pi'_k(t_{n-j_i})}$$

and therefore

$$z[t_n, \ldots, t_{n-k}]_\phi = z[t_{n-j_0}, \ldots, t_{n-j_k}]_\phi$$

for any permutation $(j_0, \ldots, j_k)$ of $(0, \ldots, k)$.

A recurrence relation for the divided difference can be constructed as follows. Let $p_{k-1}(t)$ be the polynomial in $\phi(t)$ having a degree $\leq (k-1)$ and satisfying

the interpolating conditions

$$p_{k-1}(t_{n-j}) = z_j, \qquad j = 1, \ldots, k. \tag{2.1.51}$$

Also let $q_{k-1}(t)$ be the polynomial in $\phi(t)$ of degree $\leq (k-1)$ that satisfies the interpolating conditions

$$q_{k-1}(t_{n-j}) = z_j, \qquad j = 0, \ldots, (k-1). \tag{2.1.52}$$

Define the function

$$r_k(t) = \frac{[\phi(t) - \phi(t_n)]_\phi p_{k-1}(t) - [\phi(t) - \phi(t_{n-k})]_\phi q_{k-1}(t)}{\phi(t_{n-k}) - \phi(t_n)} \tag{2.1.53}$$

to be a polynomial in $\phi(t)$ of degree $\leq k$. Using the interpolating conditions given by equations (2.1.51) and (2.1.52) we find that this polynomial satisfies the conditions

$$r_k(t_{n-j}) = z_j, \qquad j = 0, \ldots, k.$$

Therefore, $r_k(t)$ is a polynomial in $\phi(t)$ of degree $\leq k$ which interpolates $z_0, \ldots, z_k$. Since interpolating semigeneralized polynomials are unique, this implies

$$\Psi_k(t) \equiv r_k(t)$$

and therefore the k th divided difference of $\Psi_k(t)$ must be equal to the coefficient of $\phi^k(t)$ in (2.1.53). Since $z[t_{n-1}, \ldots, t_{n-k}]_\phi$ is the divided difference of $p_{k-1}(t)$ and $z[t_n, \ldots, t_{n-(k-1)}]_\phi$ is the divided difference of $q_{k-1}(t)$, then the k th coefficient of $r_k(t)$ must be equal to the k th divided difference of $\Psi_k(t)$, and we write

$$z[t_n, \ldots, t_{n-k}]_\phi = \frac{z[t_{n-1}, \ldots, t_{n-k}]_\phi - z[t_n, \ldots, t_{n-(k-1)}]_\phi}{\phi(t_{n-k}) - \phi(t_n)}.$$

This of course agrees with the classical algebraic polynomial result

$$z[t_n,\dots,t_{n-k}] = \frac{z[t_{n-1},\dots,t_{n-k}] - z[t_n,\dots,t_{n-(k-1)}]}{t_{n-k} - t_n}$$

when $\phi(t) = t$. We may now write the interpolating function in the Newton form

$$\Psi_k(t) = \sum_{j=0}^{k} z[t_n,\dots,t_{n-j}]_\phi \pi_{j-1}(t),$$

where $\pi_{-1}(t) = 1$. Substituting into (2.1.46) gives

$$y_n = y_{n-1} + \sum_{j=0}^{k} z[t_n,\dots,t_{n-j}]_\phi \int_{\tau=t_{n-1}}^{t_n} \pi_{j-1}(\tau)\,d\tau. \tag{2.1.54}$$

Recall the expansion of $\pi_{j-1}(t)$ here written in the unrestricted form

$$\pi_{j-1}(t) = \prod_{r=0}^{j-1}[\phi(t) - \phi(t_{n-r})] = \sum_{r=0}^{j} l_{j,r}\phi^r(t), \tag{2.1.55}$$

where we redefine the $l_{j,r}$ to be

$$l_{j,r} = S_{j,j-r}(-1)^{j-r}, \tag{2.1.56}$$

with

$$S_{j,s} = \begin{cases} \displaystyle\sum_{r_1=0}^{j-1}\sum_{r_2>r_1}^{j-1}\cdots\sum_{r_s>r_{s-1}}^{j-1} \phi_{r_1}\phi_{r_2}\cdots\phi_{r_s}, & s = 1,\dots,j \\ 1, & s = 0. \end{cases} \tag{2.1.57}$$

We then have

$$\int_{\tau=t_{n-1}}^{t_n} \pi_{j-1}(\tau)\,d\tau = \sum_{r=0}^{j} l_{j,r}\Delta\phi_r, \tag{2.1.58}$$

where

$$\Delta\phi_r = \begin{cases} \displaystyle\int_{\tau=t_{n-1}}^{t_n} \phi^r(t)\,d\tau, & r = 1,\dots,(k-1) \\ h, & r = 0. \end{cases} \tag{2.1.59}$$

Substituting equation (2.1.50) into (2.1.54) and using equations (2.1.55) thru (2.1.59) we obtain

$$y_n = y_{n-1} + \sum_{j=0}^{k} \sum_{s=0}^{j} \frac{z_s \phi'(t_{n-s})}{\pi_j'(t_{n-s})} \sum_{r=0}^{j} l_{j,r} \Delta\phi_r.$$

Rearranging terms we get

$$y_n = y_{n-1} + \sum_{j=0}^{k} \beta_j y'_{n-j}, \tag{2.1.60}$$

where

$$\beta_j = \phi'(t_{n-j}) \sum_{r=j}^{k} \frac{\sum_{s=0}^{r} l_{r,s} \Delta\phi_s}{\pi_r'(t_{n-j})}. \tag{2.1.61}$$

For the classical Adams method we again choose the basis function

$$\phi(t) = t$$

from which β_j becomes

$$\beta_j = \frac{(-1)^{j-1}}{j!} \sum_{r=j}^{k} \sum_{s=0}^{r} \frac{l_{r,s}}{(s+1)h^r(r-j)!} \sum_{q=0}^{s} \binom{s+1}{q} t_n^q (-h)^{s+1-q}. \tag{2.1.62}$$

In the more general case of arbitrary basis functions, the interpolating and approximating function

$$\Phi_k(t) = \xi_0 + \xi_1\phi_1(t) + \ldots + \xi_k\phi_k(t),$$

defines a generalized Newton form by

$$\Phi_k(t) = \Phi_{k-1}(t) + C_k(t), \tag{2.1.63}$$

where $C_k(t)$ is a correction function. We require that

$$\Phi_k(t_{n-j}) = \begin{cases} \Phi_{k-1}(t_{n-j}) = z_j, & j = 0, \ldots, (k-1) \\ z_k, & j = k. \end{cases} \tag{2.1.64}$$

This implies that

$$C_k(t_{n-j}) = \begin{cases} 0, & j = 0, \ldots, (k-1) \\ z_k - \Phi_{k-1}(t_{n-k}), & j = k. \end{cases} \tag{2.1.65}$$

Define $\pi_k^*(t)$ to be the generalized shifted product function

$$\pi_k^*(t) = \prod_{i=1}^{k} \left[\phi_i(t) - \phi_i(t_{n-i+1})\right], \tag{2.1.66}$$

where $\pi_0^*(t) = 1$. The first condition of equation (2.1.65) implies there exists some constant κ so that

$$C_k(t) = \kappa \pi_k^*(t).$$

Using the second condition and solving for κ we find

$$\Phi_k(t) = \Phi_{k-1}(t) + \frac{z_k - \Phi_{k-1}(t_{n-k})}{\pi_k^*(t_{n-k})} \pi_k^*(t).$$

We therefore define the k th order generalized divided difference with respect to the functions $\phi_i(t)$ by

$$z\left[t_n, \ldots, t_{n-k}\right]_{\phi_i} = \frac{z_k - \Phi_{k-1}(t_{n-k})}{\pi_k^*(t_{n-k})}, \tag{2.1.67}$$

from which $\Phi_k(t)$ becomes

$$\Phi_k(t) = \Phi_{k-1}(t) + z\left[t_n, \ldots, t_{n-k}\right]_{\phi_i} \pi_k^*(t). \tag{2.1.68}$$

Using this recurrence relation we find

$$\Phi_k(t) = \sum_{j=0}^{k} z\left[t_n, \ldots, t_{n-j}\right]_{\phi_i} \pi_j^*(t). \tag{2.1.69}$$

We now construct a recurrence relation for the generalized divided differences by expanding (2.1.69) and letting $t = t_{n-k}$ to obtain

$$\frac{z\left[t_{n-k}\right]_{\phi_i} - z\left[t_n\right]_{\phi_i}}{\phi_1(t_{n-k}) - \phi_1(t_n)} - z\left[t_n, t_{n-1}\right]_{\phi_i} = \sum_{j=2}^{k} z\left[t_n, \ldots, t_{n-j}\right]_{\phi_i} \frac{\pi_j^*(t_{n-k})}{\pi_1^*(t_{n-k})}, \tag{2.1.70}$$

where we have defined

$$z\left[t_n, t_{n-1}\right]_{\phi_i} = \frac{z\left[t_{n-1}\right]_{\phi_i} - z\left[t_n\right]_{\phi_i}}{\phi_1(t_{n-1}) - \phi_1(t_n)}. \tag{2.1.71}$$

Using the notation

$$z\left[t_n, t_{n-k}\right]_{\phi_i} = \frac{z\left[t_{n-k}\right]_{\phi_i} - z\left[t_n\right]_{\phi_i}}{\phi_1(t_{n-k}) - \phi_1(t_n)}, \tag{2.1.72}$$

we substitute into (2.1.70), and rearrange terms to find

$$\frac{z\left[t_n, t_{n-k}\right]_{\phi_i} - z\left[t_n, t_{n-1}\right]_{\phi_i}}{\phi_2(t_{n-k}) - \phi_2(t_{n-1})} - z\left[t_n, t_{n-1}, t_{n-2}\right]_{\phi_i} = \sum_{j=3}^{k} z\left[t_n, \ldots, t_{n-j}\right]_{\phi_i} \frac{\pi_j^*(t_{n-k})}{\pi_2^*(t_{n-k})}. \tag{2.1.73}$$

Similarly, by defining

$$z\left[t_n, t_{n-1}, t_{n-k}\right]_{\phi_i} = \frac{z\left[t_n, t_{n-k}\right]_{\phi_i} - z\left[t_n, t_{n-1}\right]_{\phi_i}}{\phi_2(t_{n-k}) - \phi_2(t_{n-1})} \tag{2.1.74}$$

and

$$z\left[t_n, \ldots, t_{n-k}\right]_{\phi_i} = \frac{z\left[t_n, \ldots, t_{n-k+2}, t_{n-k}\right]_{\phi_i} - z\left[t_n, \ldots, t_{n-k+1}\right]_{\phi_i}}{\phi_k(t_{n-k}) - \phi_k(t_{n-k+1})}, \tag{2.1.75}$$

we substitute equation (2.1.69) into (2.1.46) to get

$$y_n = y_{n-1} + \sum_{j=0}^{k} z\left[t_n, \ldots, t_{n-j}\right]_{\phi_i} \int_{\tau=t_{n-1}}^{t_n} \pi_j^*(\tau)d\tau. \tag{2.1.76}$$

From a practical standpoint this particular form is tedious to work with, and is given only for the sake of completeness. We will therefore use the more easily implemented Lagrange form. From Cheney [10] we write

$$\Phi_k(t) = \sum_{j=0}^{k} z_j \varphi_{j,k}(t),$$

where

$$\varphi_{j,k}(t) = \frac{\mathbf{V}\begin{pmatrix} 1 & \phi_1 & \cdots & \phi_{j-1} & \phi_j & \phi_{j+1} & \cdots & \phi_k \\ t_n & t_{n-1} & \cdots & t_{n-j+1} & t & t_{n-j-1} & \ldots & t_{n-k} \end{pmatrix}}{\mathbf{V}\begin{pmatrix} 1 & \phi_1 & \cdots & \phi_k \\ t_n & t_{n-1} & \ldots & t_{n-k} \end{pmatrix}}$$

and $\mathbf{V}$ is defined to be the determinant

$$\mathbf{V}\begin{pmatrix} 1 & \phi_1 & \cdots & \phi_k \\ t_n & t_{n-1} & \cdots & t_{n-k} \end{pmatrix} = \det \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \phi_1(t_n) & \phi_1(t_{n-1}) & \cdots & \phi_1(t_{n-k}) \\ \vdots & \vdots & & \vdots \\ \phi_k(t_n) & \phi_k(t_{n-1}) & \cdots & \phi_k(t_{n-k}) \end{bmatrix}.$$

Here $\varphi_{j,k}(t)$ has the desired property

$$\varphi_{j,k}(t_{n-i}) = \delta_{i,j},$$

where $\delta_{i,j}$ is the Kronecker delta. Upon substituting into equation (2.1.46) we

obtain

$$y_n = y_{n-1} + \sum_{j=0}^{k} \beta_j y'_{n-j},$$

where

$$\beta_j = \frac{\displaystyle\int_{\tau=t_{n-1}}^{t_n} \mathbf{V}\begin{pmatrix} 1 & \cdots & \phi_{j-1} & \phi_j & \phi_{j+1} & \cdots & \phi_k \\ t_n & \cdots & t_{n-j+1} & \tau & t_{n-j-1} & \cdots & t_{n-k} \end{pmatrix} d\tau}{\mathbf{V}\begin{pmatrix} 1 & \phi_1 & \cdots & \phi_k \\ t_n & t_{n-1} & \cdots & t_{n-k} \end{pmatrix}}. \tag{2.1.77}$$

2.2 Backward Differentiation Formula

For the BDF methods, the difference scheme

$$y_n = \sum_{j=1}^{k} \alpha_j y_{n-j} + \beta_0 y'_n \tag{2.2.78}$$

is constructed by forming a function $\Phi_k \in C^2[t_{n-k}, t_n]$ which interpolates

$$y_{n-1}, \ldots, y_{n-k}$$

at the points $t = t_{n-1}, \ldots, t_{n-k}$ respectively and whose derivative $\Phi'_k(t)$ interpolates y'_n. We define an interpolating and approximating function to be

$$\Phi_k(t) = \xi_0\phi_0(t) + \xi_1\phi_1(t) + \xi_2\phi_2(t) + \cdots + \xi_k\phi_k(t), \tag{2.2.79}$$

where $\phi_0(t) = 1$. Since there are $(k+1)$ unknown constants ξ_i for $i = 0, \ldots, k$, we require the $(k+1)$ interpolating conditions

$$\Phi_k(t_{n-j}) = y_{n-j} \qquad j = 1, \ldots, k \tag{2.2.80}$$

and

$$\Phi'_k(t_n) = y'_n. \tag{2.2.81}$$

Observe that once the $\xi_0, \xi_1, \ldots, \xi_k$ are specified using these conditions, the approximating or predicting condition is given by

$$y_n = \Phi_k(t_n). \tag{2.2.82}$$

These interpolating conditions can be expressed using the matrix equation

$$\underset{\sim}{\Phi}\vec{\xi} = \vec{y}, \tag{2.2.83}$$

where now

$$\underset{\sim}{\Phi} = \begin{pmatrix} 1 & \phi_1(t_{n-1}) & \phi_2(t_{n-1}) & \cdots & \phi_k(t_{n-1}) \\ 1 & \phi_1(t_{n-2}) & \phi_2(t_{n-2}) & \cdots & \phi_k(t_{n-2}) \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \phi_1(t_{n-k}) & \phi_2(t_{n-k}) & \cdots & \phi_k(t_{n-k}) \\ 0 & \phi'_1(t_n) & \phi'_2(t_n) & \cdots & \phi'_k(t_n) \end{pmatrix} \tag{2.2.84}$$

$$\vec{\xi} = col(\xi_0, \xi_1, \ldots, \xi_k)$$

and

$$\vec{y} = col(y_{n-1}, y_{n-2}, \ldots, y_{n-k}, y'_n).$$

2.2.1 Existence of the BDF Methods

In this section we will show that $det[\underset{\sim}{\Phi}]$ is nonzero and therefore the unique solution to (2.2.83) is given by

$$\vec{\xi} = \underset{\sim}{\Phi}^{-1}\vec{y}. \tag{2.2.85}$$

With the $\vec{\xi}$ being known, the approximating condition (2.2.82) is easily determined.

Consider the special case

$$\phi_i(t) = \phi^i(t), \qquad i = 0, \ldots, k.$$

Substituting into (2.2.84) gives

$$\underset{\sim}{\Phi} = \begin{pmatrix} 1 & \phi(t_{n-1}) & \phi^2(t_{n-1}) & \cdots & \phi^k(t_{n-1}) \\ 1 & \phi(t_{n-2}) & \phi^2(t_{n-2}) & \cdots & \phi^k(t_{n-2}) \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \phi(t_{n-k}) & \phi^2(t_{n-k}) & \cdots & \phi^k(t_{n-k}) \\ 0 & \phi'(t_n) & 2\phi(t_n)\phi'(t_n) & \cdots & k\phi^{k-1}(t_n)\phi'(t_n) \end{pmatrix}.$$

Now if $\phi(t_{n-i}) = \phi(t_{n-j})$ for $1 \leq i < j \leq k$, then row i is equal to row j, in which case $[\phi(t_{n-i}) - \phi(t_{n-j})]$ is a factor of $\det[\underset{\sim}{\Phi}]$. Furthermore, by expanding $\det[\underset{\sim}{\Phi}]$ along the last row we find

$$|\det[\underset{\sim}{\Phi}]| = \prod_{\substack{i<j \\ i=1}}^{k} |\phi(t_{n-i}) - \phi(t_{n-j})||p_{k-1}(\phi(t_n))|,$$

where $p_{k-1}(\phi(t_n))$ is a polynomial in $\phi(t_n)$ of degree $\leq (k-1)$. Consider the derivative of the product function

$$\pi_k(t) = \prod_{r=1}^{k} \left[\phi(t) - \phi(t_{n-r})\right].$$

Upon differentiating and letting $t = t_n$ we find

$$p_k(\phi(t_n)) = \pi_k'(t_n),$$

where

$$\pi_k'(t_n) = \phi'(t_n)\pi_k(t_n)\sum_{r=1}^{k}\frac{1}{\phi(t_n) - \phi(t_{n-r})}.$$

By the monotonicity of ϕ on $[t_{n-k}, t_n]$ we have

$$\det[\underset{\sim}{\Phi}] = 0 \quad \Longleftrightarrow \quad \sum_{r=1}^{k}\frac{1}{\phi(t_n) - \phi(t_{n-r})} = 0.$$

If $\phi(t)$ is increasing on $[t_{n-k}, t_n]$, then

$$\phi(t_{n-k}) < \phi(t_{n-k+1}) < \cdots < \phi(t_{n-1}) < \phi(t_n)$$

and so

$$\sum_{r=1}^{k}\frac{1}{\phi(t_n) - \phi(t_{n-r})} > 0.$$

Similarly if $\phi(t)$ is decreasing on $[t_{n-k}, t_n]$, then

$$\sum_{r=1}^{k}\frac{1}{\phi(t_n) - \phi(t_{n-r})} < 0$$

from which we conclude

$$\det[\underset{\sim}{\Phi}] \neq 0.$$

Therefore there exists a unique set of coefficients ξ_i, for $i = 0, \ldots, k$ which satisfy the interpolation conditions (2.2.80) and (2.2.81). Let $\underset{\sim}{\Phi}^{-1}$ be denoted by

$$[\varphi_{i,j}] = \underset{\sim}{\Phi}^{-1}.$$

From (2.2.85) we write

$$\xi_i = \sum_{j=1}^{k} \varphi_{i+1,j} y_{n-j} + \varphi_{i+1,k+1} y_n', \qquad i = 0, \ldots, k$$

and substitute ξ_i into (2.2.79). Using (2.2.82), the predicted value for $y(t_n)$ is found to be

$$y_n = \sum_{j=1}^{k} \alpha_j y_{n-j} + \beta_0 y_n',$$

where

$$\alpha_j = \sum_{i=0}^{k} \varphi_{i+1,j} \phi^i(t_n)$$

and

$$\beta_0 = \sum_{i=0}^{k} \varphi_{i+1,k+1} \phi^{i}(t_n),$$

which is exactly (2.2.78).

In the more general case we will additionally require that

$$\vec{\phi}'(t_n) \notin Span\left\{\vec{\phi}(t_{n-1}), \ldots, \vec{\phi}(t_{n-k})\right\}, \tag{2.2.86}$$

where

$$\vec{\phi}(t) = row(1, \phi_1(t), \ldots, \phi_k(t)).$$

The reason for this added constraint is that although $\{\vec{\phi}(t_{n-1}), \ldots, \vec{\phi}(t_{n-k})\}$ are linearly independent, this does not imply that the larger set

$$\{\vec{\phi}'(t_n), \vec{\phi}(t_{n-1}), \ldots, \vec{\phi}(t_{n-k})\}$$

is also linearly independent. This is easily verified with the specific example given by

$$\phi_1(t) = \frac{1}{3}t^2 + \frac{2}{3}t$$

and

$$\phi_2(t) = -\frac{2}{11}t^3 - \frac{20}{11}t,$$

where $t \in [0,2]$ and $t_n = 2, t_{n-1} = 1, t_{n-2} = 0$.

If (2.2.86) is true, then the rows of $\underset{\sim}{\Phi}$ are linearly independent, and hence

$$\det[\underset{\sim}{\Phi}] \neq 0.$$

Assuming this to be true, we again let $[\varphi_{i,j}]$ denote the i,j th element of $[\underset{\sim}{\Phi}]^{-1}$. As above we obtain

$$y_n = \sum_{j=1}^{k} \alpha_j y_{n-j} + \beta_0 y_n'$$

where now

$$\alpha_j = \sum_{i=0}^{k} \varphi_{i+1,j}\phi_i(t_n)$$

and

$$\beta_0 = \sum_{i=0}^{k} \varphi_{i+1,k+1}\phi_i(t_n).$$

A compilation of the α_j and β_0 is given in section 2.3.

2.2.2 Construction by Collocation

We now consider the relationship of collocation with interpolation. Form the linear functional

$$\mathcal{L}[y] = \sum_{j=1}^{k} \alpha_j y(t_{n-j}) + \beta_0 y'(t_n) - y(t_n), \tag{2.2.87}$$

and using collocation we require that

$$\mathcal{L}[\phi_i] = 0 \qquad i = 0, \ldots, k. \tag{2.2.88}$$

This collocation condition yields the matrix equation

$$\underset{\sim}{\Phi}^T \vec{\chi} = \vec{\phi}, \tag{2.2.89}$$

where

$$\vec{\chi} = col(\alpha_1, \ldots, \alpha_k, \beta_0)$$

and

$$\vec{\phi} = col(1, \phi_1(t_n), \ldots, \phi_k(t_n)).$$

Assuming that the ϕ_i satisfy the necessary condition for $\det[\underset{\sim}{\Phi}]$ to be nonzero, then the solution to the collocation problem can be expressed in the form

$$\vec{\chi} = \left[\underset{\sim}{\Phi}^{-1}\right]^T \vec{\phi}. \tag{2.2.90}$$

Expanding we get

$$\alpha_j = \sum_{i=0}^{k} \varphi_{i+1,j} \phi_i(t_n) \tag{2.2.91}$$

and

$$\beta_0 = \sum_{i=0}^{k} \varphi_{i+1,k+1}\phi_i(t_n) \tag{2.2.92}$$

which agrees with the previous results.

We now construct an explicit formula for the α_j and β_0 in the special case

$$\phi_i(t) = \phi^i(t) \qquad i = 0, \ldots, k,$$

where $\phi(t)$ is the previously used monotone function. In addition, if we choose

$$\phi(t) = t,$$

then the resulting formula will be the widely used BDF scheme of Gear [15]. We construct the solution of (2.2.89) by again using a modification of the manipulation of Jacobi [7]. Let $d_{i0}, d_{i1}, \ldots, d_{ik}$, for $i = 0, \ldots, k$ be $(k+1)$ unspecified parameters. Multiply the r th row of the collocation problem (2.2.89) by $d_{i,r-1}$ and adding equations we, get

$$\sum_{j=1}^{k} \alpha_j \sum_{r=0}^{k} d_{ir}\phi^r(t_{n-j}) + \beta_0\phi'(t_n)\sum_{r=1}^{k} r d_{ir}\phi^{r-1}(t_n) = \sum_{r=0}^{k} d_{ir}\phi^r(t_n). \tag{2.2.93}$$

Now define

$$p_i(t) = \sum_{r=0}^{k} d_{ir}\phi^r(t) \qquad i = 0, \ldots, k \tag{2.2.94}$$

to be a k th degree semigeneralized polynomial in $\phi(t)$ of degree $\leq k$. Differentiating $p_i(t)$ gives

$$p_i'(t) = \phi'(t)\sum_{r=1}^{k} r d_{ir}\phi^{r-1}(t) \qquad i = 0, \ldots, k, \tag{2.2.95}$$

which implies

$$\sum_{j=1}^{k} \alpha_j p_i(t_{n-j}) + \beta_0 p_i'(t_n) = p_i(t_n) \qquad i = 0, \ldots, k. \tag{2.2.96}$$

We now choose the d_{ir} for the following cases:

Case 2.3 *For $i = k$ let*

$$p_k'(t_n) = 1 \tag{2.2.97}$$

and

$$p_k(t_{n-j}) = 0 \qquad j = 1, \ldots, k. \tag{2.2.98}$$

Then (2.2.96) simplifies to

$$\beta_0 = p_k(t_n). \tag{2.2.99}$$

Now for $t = t_{n-j}$ (2.2.94) becomes

$$\sum_{r=0}^{k} d_{kr} \phi^r(t_{n-j}) = 0 \qquad j = 1, \ldots, k.$$

This implies that

$$\sum_{r=0}^{k} d_{kr} \phi^r(t)$$

is a polynomial in $\phi(t)$ of degree $\leq k$ with k zeros at $t = t_{n-1}, \ldots, t_{n-k}$ and therefore can be written as

$$\sum_{r=0}^{k} d_{kr} \phi^r(t) = \kappa \prod_{r=1}^{k} [\phi(t) - \phi(t_{n-r})]$$

for some constant κ. Using the notation

$$\pi_q(t) = \prod_{r=1}^{q} [\phi(t) - \phi(t_{n-r})],$$

we differentiate and set $t = t_n$ to obtain

$$\pi_k'(t_n) = \phi'(t_n)\pi_k(t_n)\sum_{r=1}^{k}\frac{1}{\phi(t_n) - \phi(t_{n-r})}.$$

The condition $p_k'(t_n) = 1$ implies that

$$\kappa = \left[\phi'(t_n)\pi_k(t_n)\sum_{r=1}^{k}\frac{1}{\phi(t_n) - \phi(t_{n-r})}\right]^{-1},$$

and consequently

$$p_k(t) = \frac{\pi_k(t)}{\phi'(t_n)\pi_k(t_n)\sum_{r=1}^{k}\frac{1}{\phi(t_n) - \phi(t_{n-r})}},$$

and from (2.2.99) we have

$$\beta_0 = \left[\phi'(t_n)\sum_{r=1}^{k}\frac{1}{\phi(t_n) - \phi(t_{n-r})}\right]^{-1}.$$

Case 2.4 *For $i = 0, \ldots, (k-1)$ let*

$$p_i(t_{n-j}) = \delta_{i+1,j} \qquad j = 1, \ldots, k \tag{2.2.100}$$

and

$$p_i'(t_n) = 0. \tag{2.2.101}$$

where $\delta_{i,j}$ is the Kronecker delta.

Then (2.2.96) reduces to

$$\alpha_j = p_{j-1}(t_n) \qquad j = 1, \ldots, k. \tag{2.2.102}$$

Now from (2.2.100) we have that

$$\sum_{r=0}^{k} d_{ir}\phi^r(t)$$

is a polynomial in $\phi(t)$ of degree $\leq k$ with zeros at $t = t_{n-1}, \ldots, t_{n-i}, t_{n-i-2}, \ldots, t_{n-k}$ and therefore can be written in the form

$$\sum_{r=0}^{k} d_{ir}\phi^r(t) = \kappa_i(t) \prod_{\substack{r=1 \\ r\neq i+1}}^{k} [\phi(t) - \phi(t_{n-r})], \tag{2.2.103}$$

where

$$\kappa_i(t) = a_i\phi(t) + b_i \tag{2.2.104}$$

and

$$\kappa_i(t_{n-r}) \neq 0 \qquad r = 1, \ldots, k. \tag{2.2.105}$$

Now from (2.2.101) we have

$$\kappa_i(t_n) \sum_{\substack{r=1 \\ r\neq i+1}}^{k} \frac{1}{\phi(t_n) - \phi(t_{n-r})} + a_i = 0, \tag{2.2.106}$$

and for $t = t_{n-i-1}$ we have from (2.2.100) that

$$\kappa_i(t_{n-i-1}) = \frac{\phi'(t_{n-i-1})}{\pi'_k(t_{n-i-1})}. \tag{2.2.107}$$

These results enable us to solve for a_i, using (2.2.104) thru (2.2.107). We find that

$$\alpha_j = \frac{\phi'(t_{n-j}) \prod\limits_{\substack{r=1 \\ r\neq j}}^{k} [\phi(t_n) - \phi(t_{n-r})]}{\pi'_k(t_{n-j}) \left[\phi(t_n) - \phi(t_{n-j})\right] \sum\limits_{r=1}^{k} \frac{1}{\phi(t_n) - \phi(t_{n-r})}}, \qquad j = 1, \ldots, k. \tag{2.2.108}$$

Now for the BDF methods developed by Gear [15] we choose

$$\phi(t) = t.$$

We then have

$$\alpha_j = \binom{k}{j} \frac{(-1)^{j-1}}{j\sum_{r=1}^{k} \frac{1}{r}} \tag{2.2.109}$$

and

$$\beta_0 = \frac{h}{\sum_{r=1}^{k} \frac{1}{r}}. \tag{2.2.110}$$

In the more general case of

$$\underset{\sim}{\Phi}^T \vec{\chi} = \vec{\phi},$$

we find using Cramer's rule that

$$\alpha_j = \frac{\det \begin{bmatrix} 1 & \cdots & 1 & 1 & 1 & \cdots & 1 & 0 \\ \phi_1(t_{n-1}) & \cdots & \phi_1(t_{n-j+1}) & \phi_1(t_n) & \phi_1(t_{n-j-1}) & \cdots & \phi_1(t_{n-k}) & \phi_1'(t_n) \\ \vdots & & \vdots & \vdots & \vdots & & \vdots & \vdots \\ \phi_k(t_{n-1}) & \cdots & \phi_k(t_{n-j+1}) & \phi_k(t_n) & \phi_k(t_{n-j-1}) & \cdots & \phi_k(t_{n-k}) & \phi_k'(t_n) \end{bmatrix}}{\det[\underset{\sim}{\Phi}]} \tag{2.2.111}$$

and

$$\beta_0 = \frac{(-1)^k \det[K(k)]}{\det[\underset{\sim}{\Phi}]}, \tag{2.2.112}$$

where $K(k)$ is the Casorati matrix [23]

$$K(k) = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \phi_1(t_n) & \phi_1(t_{n-1}) & \cdots & \phi_i(t_{n-k}) \\ \vdots & \vdots & & \vdots \\ \phi_k(t_n) & \phi_k(t_{n-1}) & \cdots & \phi_k(t_{n-k}) \end{pmatrix}.$$

2.2.3 Newton and Lagrange Interpolation

We now generate equation (2.2.78) by forming a function $\Phi_k(t)$ which interpolates $y(t)$ at the points $t = t_n, \ldots, t_{n-k}$. In the differential equation

$$y'(t) = f[y(t); t]$$

we replace $y'(t)$ by $\Phi'(t)$ and require that $f[y(t_n); t_n] = y'_n$. Since we have already constructed the generalizations of the Newton and Lagrange forms in the development of the Adams methods, we only highlight the construction of the BDF methods.

For the special case

$$\phi_i(t) = \phi^i(t) \qquad i = 1, \ldots, k,$$

the semigeneralized Newton polynomial that interpolates $y(t)$ at $t = t_n, \ldots, t_{n-k}$ is given by

$$\Phi_k(t) = \sum_{j=0}^{k} y[t_n, \ldots, t_{n-j}]_\phi \pi_{j-1}(t),$$

where

$$\pi_q(t) = \prod_{r=0}^{q} [\phi(t) - \phi(t_{n-r})]$$

and $\pi_{-1}(t) = 1$. The Lagrange form for $\Phi_k(t)$ is given by

$$\Phi_k(t) = \sum_{j=0}^{k} y_{n-j} \frac{\phi'(t_{n-j})\pi_k(t)}{\pi'_k(t_{n-j})\,[\phi(t) - \phi(t_{n-j})]}.$$

Since $y[t_n, \ldots, t_{n-k}]_\phi$ is the coefficient of $\phi^k(t)$ in the Newton form, we equate

the analogous coefficient in the Lagrange form to get

$$y[t_n, \ldots, t_{n-k}]_\phi = \sum_{j=0}^{k} y_{n-j} \frac{\phi'(t_{n-j})}{\pi_k'(t_{n-j})}.$$

By reduction of indices we find

$$y[t_n, \ldots, t_{n-j}]_\phi = \sum_{r=0}^{j} y_{n-r} \frac{\phi'(t_{n-r})}{\pi_j'(t_{n-r})}.$$

Using the condition $\Phi_k'(t_n) = y_n'$ we get

$$y_n' = \sum_{j=1}^{k} \sum_{r=0}^{j} y_{n-r} \frac{\phi'(t_{n-r})\pi_{j-1}'(t_n)}{\pi_j'(t_{n-r})}.$$

Rearranging and simplifying we find that

$$y_n = \sum_{j=1}^{k} \alpha_j y_{n-j} + \beta_0 y_n',$$

where

$$\beta_0 = \left[\phi'(t_n) \sum_{i=1}^{k} \frac{1}{\phi(t_n) - \phi(t_{n-i})}\right]^{-1}$$

and

$$\alpha_j = -\beta_0 \phi'(t_{n-j}) \sum_{i=j}^{k} \frac{\pi_{i-1}'(t_n)}{\pi_i'(t_{n-j})},$$

which agrees with the results given previously.

In the more general case of

$$\Phi_k(t) = \xi_0 + \xi_1 \phi_1(t) + \cdots + \xi_k(t) \phi_k(t),$$

the Newton form is given by

$$\Phi_k(t) = \sum_{j=0}^{k} y[t_n, \ldots, t_{n-j}]_{\phi_i} \pi_{j-1}^*(t), \tag{2.2.113}$$

where $\pi_q^*(t)$ is the generalized shifted product function

$$\pi_q^*(t) = \prod_{r=1}^{q} [\phi_r(t) - \phi_r(t_{n-r+1})]$$

and $\pi_0^*(t) = 1$. Now using the condition $\Phi_k'(t_n) = y_n'$ gives

$$y_n' = \sum_{j=1}^{k} y[t_n, \dots, t_{n-j}]_{\phi_i} \phi_1'(t_n) \prod_{r=2}^{j} [\phi_r(t_n) - \phi_r(t_{n-r+1})] \, .$$

Although a recurrence relation exists for $y[t_n, \dots, t_{n-j}]_{\phi_i}$, this form is tedious to work with, so we again use the Lagrange form

$$\Phi_k(t) = \sum_{j=0}^{k} y_{n-j} \varphi_{j,k}(t),$$

where $\varphi_{j,k}(t)$ previously defined. Differentiating and letting $t = t_n$ we get

$$y_n' = \sum_{j=0}^{k} y_{n-j} \varphi_{j,k}'(t_n),$$

where

$$\varphi_{j,k}'(t_n) = \frac{\det[\vec{\phi}(t_n), \dots, \vec{\phi}(t_{n-j+1}), \vec{\phi}'(t_n), \vec{\phi}(t_{n-j-1}), \dots, \vec{\phi}(t_{n-k})]}{\mathbf{V}\begin{pmatrix} 1 & \phi_1 & \cdots & \phi_k \\ t_n & t_{n-1} & \cdots & t_{n-k} \end{pmatrix}}$$

and

$$\vec{\phi}(t) = col(1, \phi_1(t), \dots, \phi_k(t)).$$

Rearranging terms we write

$$y_n = \sum_{j=1}^{k} \alpha_j y_{n-j} + \beta_0 y_n' \tag{2.2.114}$$

where

$$\beta_0 = \frac{\mathbf{V}\begin{pmatrix} 1 & \phi_1 & \cdots & \phi_k \\ t_n & t_{n-1} & \cdots & t_{n-k} \end{pmatrix}}{\det[\vec{\phi}'(\mathbf{t_n}), \vec{\phi}(\mathbf{t_{n-1}}), \ldots, \vec{\phi}(\mathbf{t_{n-k}})]} \tag{2.2.115}$$

and

$$\alpha_j = -\frac{\det[\vec{\phi}(t_n), \ldots, \vec{\phi}(t_{n-j+1}), \vec{\phi}'(t_n), \vec{\phi}(t_{n-j-1}), \ldots, \vec{\phi}(t_{n-k})]}{\det[\vec{\phi}'(t_n), \vec{\phi}(t_{n-1}), \ldots, \vec{\phi}(t_{n-k})]}. \tag{2.2.116}$$

2.3 Compilation of Adams and BDF Methods

The coefficients for the Adams–Moulton and BDF methods with various basis functions are now given.

Coefficients for $\left\{1, t, t^2, \ldots, t^{k+1}\right\}$							
k	β_0	β_1	β_2	β_3	β_4	β_5	β_6
1	$\frac{1}{2}h$	$\frac{1}{2}h$					
2	$\frac{5}{12}h$	$\frac{2}{3}h$	$-\frac{1}{12}h$				
3	$\frac{3}{8}h$	$\frac{19}{24}h$	$-\frac{5}{24}h$	$\frac{1}{24}h$			
4	$\frac{251}{720}h$	$\frac{323}{360}h$	$-\frac{11}{30}h$	$\frac{53}{360}h$	$-\frac{19}{720}h$		
5	$\frac{95}{288}h$	$\frac{1427}{1440}h$	$-\frac{133}{240}h$	$\frac{241}{720}h$	$\frac{173}{1440}h$	$\frac{3}{160}h$	
6	$\frac{19087}{60480}h$	$\frac{2713}{2520}h$	$-\frac{15487}{20160}h$	$\frac{586}{945}h$	$-\frac{6737}{20160}h$	$\frac{263}{2520}h$	$-\frac{863}{60480}h$

Table 2.1:
Coefficients for the Adams–Moulton methods using the classical monomial basis.

Coefficients for $\left\{1, e^{t}, e^{2t}, \ldots, e^{(k+1)t}\right\}$ †				
k	β_0	β_1	β_2	β_3
1	$\frac{z-1}{2z}$	$\frac{z-1}{2}$		
2	$\frac{(z-1)(2z+3)}{6z(z+1)}$	$\frac{(z-1)(z+3)}{6}$	$-\frac{z^3(z-1)}{6(z+1)}$	
3	$\frac{(z-1)(3z^3+8z^2+10z+6)}{12z(z+1)(z^2+z+1)}$	$\frac{(z-1)(z^3+4z^2+8z+6)}{12(z+1)}$	$-\frac{z^3(z-1)(z^2+2z+2)}{12(z+1)}$	$\frac{z^7(z-1)(z+2)}{12(z+1)(z^2+z+1)}$

† $z = e^h$

Table 2.2:
Coefficients for the Adams–Moulton methods using the positive exponential basis.

Coefficients for $\left\{1, e^{-t}, e^{-2t}, \ldots, e^{-(k+1)t}\right\}$ †				
k	β_0	β_1	β_2	β_3
1	$\frac{z-1}{2}$	$\frac{z-1}{2z}$		
2	$\frac{(z-1)(3z+2)}{6(z+1)}$	$\frac{(z-1)(3z+1)}{6z^2}$	$\frac{-(z-1)^2}{6z^3(z+1)}$	
3	$\frac{(z-1)(6z^3+10z^2+8z+3)}{12(z+1)(z^2+z+1)}$	$\frac{(z-1)(6z^3+8z^2+4z+1)}{12z^3(z+1)}$	$\frac{-(z-1)(2z^2+2z+1)}{12z^5(z+1)}$	$\frac{(z-1)(2z+1)}{12z^6(z+1)(z^2+z+1)}$

† $z = e^h$

Table 2.3:
Coefficients for the Adams–Moulton methods using the negative exponential basis.

Coefficients for $\left\{1, t, t^2, \ldots, t^k\right\}$							
k	β_0	α_1	α_2	α_3	α_4	α_5	α_6
1	h	1					
2	$\frac{2}{3}h$	$\frac{4}{3}$	$-\frac{1}{3}$				
3	$\frac{6}{11}h$	$\frac{18}{11}$	$-\frac{9}{11}$	$\frac{2}{11}$			
4	$\frac{12}{25}h$	$\frac{48}{25}$	$-\frac{36}{25}$	$\frac{16}{25}$	$-\frac{3}{25}$		
5	$\frac{60}{137}h$	$\frac{300}{137}$	$-\frac{300}{137}$	$\frac{200}{137}$	$-\frac{75}{137}$	$\frac{12}{137}$	
6	$\frac{20}{49}h$	$\frac{120}{49}$	$-\frac{150}{49}$	$\frac{400}{147}$	$-\frac{75}{49}$	$\frac{24}{49}$	$-\frac{10}{147}$

Table 2.4:
Coefficients for the BDF methods using the classical monomial basis.

Coefficients for $\left\{1, e^{t}, e^{2t}, \ldots, e^{kt}\right\}$ †				
k	β_0	α_1	α_2	α_3
1	$\frac{z-1}{z}$	1		
2	$\frac{(z-1)(z+1)}{z(2z+1)}$	$\frac{(z+1)^2}{2z+1}$	$\frac{-z^2}{(2z+1)}$	
3	$\frac{(z-1)(z+1)(z^2+z+1)}{z(3z^3+4z^2+3z+1)}$	$\frac{(z+1)(z^2+z+1)^2}{3z^3+4z^2+3z+1}$	$\frac{-z^2(z^2+z+1)^2}{3z^3+4z^2+3z+1}$	$\frac{z^5(z+1)}{3z^3+4z^2+3z+1}$

† $z = e^h$

Table 2.5:
Coefficients for the BDF methods using the positive exponential basis.

Coefficients for $\left\{1, e^{-t}, e^{-2t}, \ldots, e^{-kt}\right\}$ †				
k	β_0	α_1	α_2	α_3
1	$z-1$	1		
2	$\frac{(z-1)(z+1)}{z+2}$	$\frac{(z+1)^2}{z(z+2)}$	$\frac{-1}{z(z+2)}$	
3	$\frac{(z-1)(z+1)(z^2+z+1)}{z^3+3z^2+4z+3}$	$\frac{(z+1)(z^2+z+1)^2}{z^2(z^3+3z^2+4z+3)}$	$\frac{-(z^2+z+1)^2}{z^3(z^3+3z^2+4z+3)}$	$\frac{z+1}{z^3(z^3+3z^2+4z+3)}$

† $z = e^h$

Table 2.6:
Coefficients for the BDF methods using the negative exponential basis.

Chapter 3

Error Analysis

In this chapter we examine how well the Adams method

$$y_n = \alpha_1 y_{n-1} + \sum_{j=0}^{k} \beta_j y'_{n-j}, \tag{3.0.1}$$

and the BDF method

$$y_n = \sum_{j=1}^{k} \alpha_j y_{n-j} + \beta_0 y'_n \tag{3.0.2}$$

approximate the exact solution $y(t_n)$. In order to do so, we consider the errors that can occur:

- **Errors in Starting Values**

  For the Adams method (3.0.1), the first approximation will require the values $y'_0, \ldots, y'_k$. Since only y_0 is known exactly, we must therefore approximate $y'(t_1), \ldots, y'(t_k)$ by $y'_1, \ldots, y'_k$ with respective errors

$$e'_i = y'(t_i) - y'_i \quad i = 1, \ldots, k.$$

Obviously the analogous problem is incurred in the BDF method (3.0.2) as well.

- **Local Truncation Error**

Suppose we have proceeded to the $(n-1)$ st step and have at our disposal $y_{n-k}, \ldots, y_{n-1}$ which we will assume to be exactly $y(t_{n-k}), \ldots, y(t_{n-1})$. Application of the BDF method produces y_n which in general does not agree with $y(t_n)$. Hence we define the local truncation error

$$e(t_n) = y_n - y(t_n)$$

to be the error in approximating $y(t_n)$ by y_n, when using a linear multistep method. The analogous problem is incurred in the Adams method as well.

- **Round Off Error**

Since the Adams and BDF methods given by equations (3.0.1) and (3.0.2) are implicit methods, there will always be some roundoff error due to approximating the solution of a nonlinear difference equation. Furthermore, since computers can only represent real numbers by rational numbers of finite accuracy, additional round off errors are introduced.

- **Accumulated or Propagated Error**

Upon first application of a linear multistep method, start up, local truncation and roundoff errors are incurred. When the linear multistep method

is applied again, these errors are added to the additional errors introduced at the second application. Subsequent error is propagated with each application of the method.

Assuming that the start up and round off error are insignificant, we expect a good numerical method to keep both the local truncation and accumulated error from becoming excessive. In this chapter we give an estimate of the local truncation error for the generalized Adams and BDF methods.

In the remaining chapters, we assume that the set of mesh points partitions the interval $[a, b]$ uniformly. In other words, we let

$$h = t_{j+1} - t_j, \qquad j = 0, \ldots, (N-1),$$

where h denotes a uniform stepsize. Although this restriction was not needed in the construction of the Adams and BDF methods, it is assumed in the analysis of the local truncation error and in the analysis of the stability. This restriction ensures a tractable analysis. Hence, the more general case of variable stepsizes will not be considered at this time.

3.1 Error of the Adams Methods

Recall the linear functional (2.1.24)

$$\mathcal{L}[y] = \alpha_1 y(t_{n-1}) + \sum_{j=0}^{k} \beta_j y'(t_{n-j}) - y(t_n). \tag{3.1.3}$$

If we assume that

$$y_{n-1} = y(t_{n-1}) \tag{3.1.4}$$

and

$$y'_{n-j} = y'(t_{n-j}), \qquad j = 0, \ldots, k, \tag{3.1.5}$$

then

$$\mathcal{L}[y] = y_n - y(t_n). \tag{3.1.6}$$

The linear functional $\mathcal{L}[y]$ is therefore the error made in approximating $y(t_n)$ by y_n. That is, $\mathcal{L}[y]$ is the local truncation error.

For notational purposes, define

$$e(t) = \Phi_{k+1}(t) - y(t) \tag{3.1.7}$$

as the error in approximating $y(t)$ by the interpolating and approximating function $\Phi_{k+1}(t)$ defined in (2.1.8). From (3.1.6) the local truncation error is given by

$$e(t_n) = \mathcal{L}[y].$$

Assuming that the basis functions $\{1, \phi_1(t), \ldots, \phi_{k+1}(t)\}$ and the exact solution $y(t)$ are sufficiently differentiable, we may expand the interpolating condition

$$\Phi_{k+1}(t_{n-1}) = y(t_{n-1})$$

in a Taylor series about $t = t_n$. Using the condition

$$\Phi^{(1)}_{k+1}(t_n) = y^{(1)}(t_n)$$

implies

$$e(t_n) = -\sum_{j=2}^{k+2} \frac{(-h)^j}{j!} e^{(j)}(t_n) + O(h^{k+3}). \tag{3.1.8}$$

Similarly, expanding the interpolating conditions

$$\Phi^{(1)}(t_{n-j}) = y^{(1)}(t_{n-j}) \qquad j = 1, \ldots, k$$

we construct the matrix equation

$$\underset{\sim}{\mathbf{H}}\vec{d} = \vec{e} + \vec{O}(h^{k+2}), \tag{3.1.9}$$

where

$$\underset{\sim}{\mathbf{H}} = \begin{pmatrix} \frac{(-h)^1}{1!} & \frac{(-h)^2}{2!} & \cdots & \frac{(-h)^k}{k!} \\ \frac{(-2h)^1}{1!} & \frac{(-2h)^2}{2!} & \cdots & \frac{(-2h)^k}{k!} \\ \vdots & \vdots & & \vdots \\ \frac{(-kh)^1}{1!} & \frac{(-kh)^2}{2!} & \cdots & \frac{(-kh)^k}{k!} \end{pmatrix},$$

$$\vec{d} = col\left(e^{(2)}(t_n), \ldots, e^{(k+1)}(t_n)\right),$$

and

$$\vec{e} = -\frac{e^{(k+2)}(t_n)}{(k+1)!} col\left((-h)^{k+1}, \ldots, (-kh)^{k+1}\right).$$

We solve for $e^{(j)}(t_n)$, where $j = 2, \ldots, (k+1)$, by truncating this matrix equation and forming an associated problem

$$\underset{\sim}{\mathbf{H}}^T \vec{x} = \vec{b}, \tag{3.1.10}$$

where

$$\vec{x} = col\,(x_1, \ldots, x_k)$$

is unknown, and

$$\vec{b} = col\,(b_1, \ldots, b_k)$$

is assumed to be known. We explicitly construct $\left[\underset{\sim}{\mathbf{H}}^T\right]^{-1}$ using the procedure developed in section 2.1.2. Let $d_{i1}, \ldots, d_{ik}$, for $i = 1, \ldots, k$ be k^2 unspecified parameters. Multiply the r th row of (3.1.10) by d_{ir} and add the resulting equations. This produces the result

$$\sum_{j=1}^{k} x_j \sum_{r=1}^{k} d_{ir} \frac{(-jh)^r}{r!} = \sum_{r=1}^{k} d_{ir} b_r,$$

where $i = 1, \ldots, k$. Substituting

$$p_i(\tau) = \sum_{r=1}^{k} d_{ir} \frac{(-\tau)^r}{r!}, \qquad i = 1, \ldots, k, \tag{3.1.11}$$

implies

$$\sum_{j=1}^{k} p_i(jh) x_j = \sum_{r=1}^{k} d_{ir} b_r, \qquad i = 1, \ldots, k.$$

We now show how to select the coefficients d_{ir}. If we require for each $i = 1, \ldots, k$ that

$$p_i(jh) = \delta_{ij}, \qquad j = 1, \ldots, k, \tag{3.1.12}$$

then

$$x_j = \sum_{r=1}^{k} d_{jr} b_r, \qquad j = 1, \ldots, k.$$

This implies $\left[\underset{\sim}{\mathbf{H}}^T\right]^{-1} = [d_{jr}]$, from which

$$\underset{\sim}{\mathbf{H}}^{-1} = [d_{rj}].$$

Using (3.1.12) we find

$$p_i(\tau) = \left(\frac{\tau}{ih}\right) \prod_{\substack{s=1 \\ s\neq i}}^{k} \frac{\tau - sh}{h(i-s)}. \tag{3.1.13}$$

Substitute

$$\tau = h\mu,$$

into equations (3.1.11) and (3.1.13) and rearranging to produce

$$\sum_{r=1}^{k} d_{ir} \frac{(-h)^r}{r!} \mu^{r-1} = \frac{(-1)^{k-i}}{k!} \binom{k}{i} \prod_{\substack{s=1 \\ s\neq i}}^{k} [\mu - s].$$

Define

$$\prod_{\substack{s=1 \\ s\neq i}}^{k} [\mu - s] = \sum_{r=0}^{k-1} {}^{i}l_{k-1,r}\mu^r,$$

where

$${}^{i}l_{k-1,r} = {}^{i}S_{k-1,k-1-r}(-1)^{k-1-r},$$

and

$${}^{i}S_{k-1,s} = \begin{cases} \displaystyle\sum_{\substack{r_1=1 \\ r_1\neq i}}^{k} \sum_{\substack{r_2>r_1 \\ r_2\neq i}}^{k} \cdots \sum_{\substack{r_s>r_{s-1} \\ r_s\neq i}}^{k} r_1 r_2 \cdots r_s, & s = 0, 1, \ldots, (k-1) \\ 1, & s = 0. \end{cases}$$

We now solve for d_{ij} to get

$$d_{ij} = \frac{(-1)^{k-i-j} j!}{h^j k!} \binom{k}{i} {}^{i}l_{k-1,j-1}, \qquad i, j = 1, \ldots, k.$$

Using the truncated version of (3.1.9) we now solve for $e^{(q+1)}(t_n)$ to get

$$e^{(q+1)}(t_n) = h^{k-q+1} c_{q+1} e^{(k+2)}(t_n) \qquad q = 1, \ldots, k, \tag{3.1.14}$$

where

$$c_{q+1} = \frac{q!}{(k+1)!k!} \sum_{r=1}^{k} (r)^{k+1} (-1)^{1-r-q} \binom{k}{r} {}^{r}l_{k-1,q-1} \qquad q = 1, \ldots, k. \tag{3.1.15}$$

Substituting into the local truncation error equation (3.1.8) we get

$$e(t_n) = h^{k+2} e^{(k+2)}(t_n) \sum_{q=1}^{k+1} c_{q+1} \frac{(-1)^q}{(q+1)!} + O(h^{k+3}). \tag{3.1.16}$$

We now find $e^{(k+2)}(t_n)$. Differentiating

$$e(t) = \Phi_{k+1}(t) - y(t),$$

$(k+2)$ times gives a system of equations

$$\begin{aligned}
\xi_1 \phi_1^{(1)}(t_n) + \xi_2 \phi_2^{(1)}(t_n) + \cdots + \xi_{k+1} \phi_{k+1}^{(1)}(t_n) &= y^{(1)}(t_n) \\
\xi_1 \phi_1^{(2)}(t_n) + \xi_2 \phi_2^{(2)}(t_n) + \cdots + \xi_{k+1} \phi_{k+1}^{(2)}(t_n) - e^{(2)}(t_n) &= y^{(2)}(t_n) \\
&\;\;\vdots \\
\xi_1 \phi_1^{(k+2)}(t_n) + \xi_2 \phi_2^{(k+2)}(t_n) + \cdots + \xi_{k+1} \phi_{k+1}^{(k+2)}(t_n) - e^{(k+2)}(t_n) &= y^{(k+2)}(t_n)
\end{aligned}$$

Using equation (3.1.14) we solve for $e^{(k+2)}(t_n)$ to get

$$e^{(k+2)}(t_n) = \frac{\Delta_{top}}{\Delta_{bot}^*}, \tag{3.1.17}$$

where

$$\Delta_{top} = \det \begin{bmatrix} \phi_1^{(1)}(t_n) & \phi_2^{(1)}(t_n) & \cdots & \phi_{k+1}^{(1)}(t_n) & y^{(1)}(t_n) \\ \phi_1^{(2)}(t_n) & \phi_2^{(2)}(t_n) & \cdots & \phi_{k+1}^{(2)}(t_n) & y^{(2)}(t_n) \\ \vdots & \vdots & & \vdots & \vdots \\ \phi_1^{(k+2)}(t_n) & \phi_2^{(k+2)}(t_n) & \cdots & \phi_{k+1}^{(k+2)}(t_n) & y^{(k+2)}(t_n) \end{bmatrix}$$

and

$$\Delta^*_{bot} = \det \begin{bmatrix} \phi_1^{(1)}(t_n) & \phi_2^{(1)}(t_n) & \cdots & \phi_{k+1}^{(1)}(t_n) & 0 \\ \phi_1^{(2)}(t_n) & \phi_2^{(2)}(t_n) & \cdots & \phi_{k+1}^{(2)}(t_n) & -h^k c_2 \\ \phi_1^{(3)}(t_n) & \phi_2^{(3)}(t_n) & \cdots & \phi_{k+1}^{(3)}(t_n) & -h^{k-1} c_3 \\ \vdots & \vdots & & \vdots & \vdots \\ \phi_1^{(k+2)}(t_n) & \phi_2^{(k+2)}(t_n) & \cdots & \phi_{k+1}^{(k+2)}(t_n) & -c_{k+2} \end{bmatrix}.$$

Expanding Δ^*_{bot} along the last column gives

$$\frac{1}{\Delta^*_{bot}} = \frac{1}{h^k c_2(-1)^{k+1}\mathbf{D_2} + h^{k-1}c_3(-1)^{k+2}\mathbf{D_3} + \cdots + c_{k+2}(-1)^{2k+1}\mathbf{D_{k+2}}},$$

where $\mathbf{D_j}$ are the appropriate sub determinants of Δ^*_{bot}. Taking a Taylor series about $h = 0$ gives

$$\frac{1}{\Delta^*_{bot}} = \frac{1}{c_{k+2}(-1)^{2k+1}\mathbf{D_{k+2}}} + O(h).$$

Substituting equation (3.1.17) into equation (3.1.16) gives an estimate of the local truncation error as

$$e(t_n) = \frac{h^{k+2}\Delta_{top}}{c_{k+2}\Delta_{bot}} \sum_{q=1}^{k+1} c_{q+1}\frac{(-1)^{q+1}}{(q+1)!} + O(h^{k+3}), \tag{3.1.18}$$

where

$$\Delta_{bot} = \det \begin{bmatrix} \phi_1^{(1)}(t_n) & \phi_2^{(1)}(t_n) & \cdots & \phi_{k+1}^{(1)}(t_n) \\ \phi_1^{(2)}(t_n) & \phi_2^{(2)}(t_n) & \cdots & \phi_{k+1}^{(2)}(t_n) \\ \vdots & \vdots & & \vdots \\ \phi_1^{(k+1)}(t_n) & \phi_2^{(k+1)}(t_n) & \cdots & \phi_{k+1}^{(k+1)}(t_n) \end{bmatrix}.$$

Observe that the order of the local truncation error is independent of the

interpolating and approximating basis chosen. In fact,

$$Order\left[e(t_n)\right] = Card\left\{1, \phi_1(t), \ldots, \phi_{k+1}(t)\right\}.$$

The magnitude of $\Delta_{top}, \Delta_{bot}$ and c_{q+1} do however depend on the basis and hence will vary according to how the basis is chosen, along with the various derivatives $y^{(j)}(t_n)$. This implies that when comparing two methods, both methods *must* be constructed from the same number of basis functions in order for a fair comparison to be made.

Furthermore, since $y(t)$ is in fact not generally known, an estimate of Δ_{top} cannot be made without some a priori assumptions on $y(t)$. As such, we will assume that two methods of the same error order $(k+2)$ also have coefficients of similar magnitude. Therefore, in order to decide which of two methods is better, stability considerations will rule.

For an alternative error estimate recall (2.1.46)

$$y(t_n) = y(t_{n-1}) + \int_{\tau=t_{n-1}}^{t_n} f^*(\tau)\, d\tau$$

which upon replacement of $f^*(\tau)$ with the interpolating and approximating generalized function $\Psi_k(\tau)$, gives

$$y_n = y(t_{n-1}) + \int_{\tau=t_{n-1}}^{t_n} \Psi_k(\tau)\, d\tau. \tag{3.1.19}$$

Consider

$$\Psi_{k+1}(t) = \Psi_k(t) + z[t_n, \ldots, t_{n-k}, \tau]_{\phi_i} \pi^*_{k+1}(t),$$

where the generalized shifted product function π^*_{k+1} is defined by

$$\pi^*_{k+1}(t) = [\phi_{k+1}(t) - \phi_{k+1}(\tau)] \prod_{i=1}^{k} [\phi_i(t) - \phi_i(t_{n-i+1})],$$

with $z(t) = y'(t)$ and $\tau \in (t_{n-k}, t_n)$ but distinct from $t_{n-k}, \ldots, t_n$. Clearly $\Psi_{k+1}(t)$ interpolates $z(t)$ at $t = t_{n-k}, \ldots, t_n, \tau$. So

$$\Psi_{k+1}(\tau) = z(\tau)$$

implies

$$\Psi_k(\tau) = z(\tau) - z[t_n, \ldots, t_{n-k}, \tau]_{\phi_i} \pi^*_{k+1}(\tau)$$

which when substituted into (3.1.19) gives a bound on the local truncation error

$$|\mathcal{L}[y]| = |y_n - y(t_n)| \le \int_{\tau=t_{n-1}}^{t_n} \left|z\left[t_n, \ldots, t_{n-k}, \tau\right]_{\phi_i}\right| |\pi^*_{k+1}(\tau)|\, d\tau.$$

Assuming $\tau \in (t_{n-j+1}, t_{n-j})$ for some $0 \le j \le k$ and using the monotonicity of ϕ_i, a bound on $\pi^*_{k+1}(\tau)$ is given by

$$|\pi^*_{k+1}(\tau)| \le \prod_{i=1}^{j+1} |\phi_i(t_{n-j-1}) - \phi_i(t_{n-i+1})| \prod_{i=j+2}^{k+1} |\phi_i(t_{n-i+1}) - \phi_i(t_{n-j})|.$$

Since $z\left[t_n, \ldots, t_{n-k}, \tau\right]_{\phi_i}$ is of such a complex form, a bound for this divided difference is not easily found. Hence the bound on the local truncation error $|\mathcal{L}[y]|$ given by

$$\prod_{i=1}^{j+1} |\phi_i(t_{n-j-1}) - \phi_i(t_{n-i+1})| \prod_{i=j+2}^{k+1} |\phi_i(t_{n-i+1}) - \phi_i(t_{n-j})| \int_{\tau=t_{n-1}}^{t_n} z\left[t_n, \ldots, t_{n-k}, \tau\right]_{\phi_i} d\tau$$

is stated only for completeness and not for its practical value.

3.2 Error of the BDF Methods

Using the linear functional (2.2.87)

$$\mathcal{L}[y] = \sum_{j=1}^{k} \alpha_j y(t_{n-j}) + \beta_0 y'(t_n) - y(t_n),$$

and assuming that

$$y_{n-j} = y(t_{n-j}) \qquad j = 1, \ldots, k$$

and

$$y'_n = y'(t_n),$$

then, once again

$$\mathcal{L}[y] = y_n - y(t_n)$$

is the local truncation error. For notational purposes, let

$$e(t) = \Phi_k(t) - y(t)$$

denote the error of approximating $y(t)$ by the interpolating and approximating function $\Phi_k(t)$ defined in equation (2.2.79). From above, the local truncation error is given by

$$e(t_n) = \mathcal{L}\left[y\right].$$

Expanding in a Taylor series the interpolating conditions

$$\Phi_k(t_{n-j}) = y(t_{n-j}) \quad j = 1, \ldots, k$$

and using the condition

$$\Phi_k^{(1)}(t_n) = y^{(1)}(t_n)$$

gives

$$\sum_{r=2}^{k+1} \frac{(-jh)^r}{r!} e^{(r)}(t_n) = -e(t_n) - \frac{(-jh)^{k+2}}{(k+2)!} e^{(k+2)}(t_n) + O(h^{k+3}) \qquad j = 1, \ldots, k.$$

In matrix form, this equation is given by

$$\underset{\sim}{\mathbf{H}}\vec{d} = \vec{e} + \vec{O}(h^{k+2}), \tag{3.2.20}$$

where

$$\underset{\sim}{\mathbf{H}} = \begin{pmatrix} \frac{(-h)^2}{2!} & \cdots & \frac{(-h)^{k+1}}{(k+1)!} \\ \vdots & & \vdots \\ \frac{(-kh)^2}{2!} & \cdots & \frac{(-kh)^{k+1}}{(k+1)!} \end{pmatrix},$$

$$\vec{d} = col\left(e^{(2)}(t_n), \ldots, e^{(k+1)}(t_n)\right),$$

and

$$\vec{e} = -e(t_n) \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}.$$

Consider the associated truncated and transposed problem

$$\underset{\sim}{\mathbf{H}}^T \vec{x} = \vec{b} \tag{3.2.21}$$

where

$$\vec{x} = col\,(x_1, \ldots, x_k)$$

is unknown, and

$$\vec{b} = col\,(b_1, \ldots, b_k)$$

is assumed to be known. Again, we explicitly construct $\left[\underset{\sim}{\mathbf{H}}^T\right]^{-1}$ using the procedure developed in section 2.1.2. Let $d_{i1},\ldots,d_{ik}$, for $i = 1,\ldots,k$ be k^2 unspecified parameters. Multiply the r th row of (3.2.21) by d_{ir} and add the resulting equations. This produces the result

$$\sum_{j=1}^{k} x_j \sum_{r=1}^{k} d_{ir} \frac{(-jh)^{r+1}}{(r+1)!} = \sum_{r=1}^{k} d_{ir} b_r,$$

where $i = 1,\ldots,k$. Substituting

$$p_i(\tau) = \tau^2 \sum_{r=1}^{k} d_{ir} \frac{(-\tau)^{r-1}}{(r+1)!}, \qquad i = 1,\ldots,k, \tag{3.2.22}$$

implies

$$\sum_{j=1}^{k} p_i(jh) x_j = \sum_{r=1}^{k} d_{ir} b_r, \qquad i = 1,\ldots,k.$$

We now show how to select the coefficients d_{ir}. If we require for each $i = 1,\ldots,k$ that

$$p_i(jh) = \delta_{ij}, \qquad j = 1,\ldots,k, \tag{3.2.23}$$

then

$$x_j = \sum_{r=1}^{k} d_{jr} b_r, \qquad j = 1,\ldots,k.$$

This implies $\left[\underset{\sim}{\mathbf{H}}^T\right]^{-1} = [d_{jr}]$, from which

$$\underset{\sim}{\mathbf{H}}^{-1} = [d_{rj}].$$

Using (3.2.23) we find

$$p_i(\tau) = \left(\frac{\tau}{ih}\right)^2 \prod_{\substack{s=1 \\ s\neq i}}^{k} \frac{\tau - sh}{h(i-s)}. \tag{3.2.24}$$

Substitute

$$\tau = h\mu,$$

into equations (3.2.22) and (3.2.24) and rearrange terms to produce

$$h^2 \sum_{r=1}^{k} d_{ir} \frac{(-h\mu)^{r-1}}{(r+1)!} = \frac{(-1)^{k-i}}{i(k!)} \binom{k}{i} \prod_{\substack{s=1 \\ s\neq i}}^{k} [\mu - s] .$$

Define

$$\prod_{\substack{s=1 \\ s\neq i}}^{k} [\mu - s] = \sum_{r=0}^{k-1} {}^{i}l_{k-1,r}\mu^{r},$$

where

$${}^{i}l_{k-1,r} = {}^{i}S_{k-1,k-1-r}(-1)^{k-1-r},$$

and

$${}^{i}S_{k-1,s} = \begin{cases} \displaystyle\sum_{\substack{r_1=1 \\ r_1\neq i}}^{k} \sum_{\substack{r_2>r_1 \\ r_2\neq i}}^{k} \cdots \sum_{\substack{r_s>r_{s-1} \\ r_s\neq i}}^{k} r_1 r_2 \cdots r_s, & \text{for } s = 0, 1, \ldots, (k-1) \\ 1, & \text{for } s = 0. \end{cases}$$

We now solve for d_{ij} to get

$$d_{jr} = \frac{(-1)^{k+r-j+1}(r+1)!}{j(k!)h^{r+1}} \binom{k}{j} {}^{j}l_{k-1,r-1} \qquad j, r = 1, \ldots, k.$$

Using the truncated version of (3.2.20) we may solve for $e^{(q+1)}(t_n)$ to get

$$e^{(q+1)}(t_n) = -e(t_n)c^{*}_{q+1} \qquad q = 1, \ldots, k,$$

where

$$c^{*}_{q+1} = \sum_{r=1}^{k} \frac{(-1)^{k+q-r+1}(q+1)!}{r(k!)h^{q+1}} \binom{k}{r} {}^{r}l_{k-1,q-1} \qquad q = 1, \ldots, k.$$

We now find $e(t_n)$. Differentiating

$$e(t) = \Phi_{k+1}(t) - y(t),$$

$(k+1)$ times gives a system of equations

$$\begin{aligned}
\xi_0 + \xi_1\phi_1(t_n) + \xi_2\phi_2(t_n) + \cdots + \xi_k\phi_k(t_n) - e(t_n) &= y(t_n) \\
\xi_1\phi_1^{(1)}(t_n) + \xi_2\phi_2^{(1)}(t_n) + \cdots + \xi_k\phi_k^{(1)}(t_n) &= y^{(1)}(t_n) \\
\xi_1\phi_1^{(2)}(t_n) + \xi_2\phi_2^{(2)}(t_n) + \cdots + \xi_k\phi_k^{(2)}(t_n) - e^{(2)}(t_n) &= y^{(2)}(t_n) \\
&\vdots \\
\xi_1\phi_1^{(k+1)}(t_n) + \xi_2\phi_2^{(k+1)}(t_n) + \cdots + \xi_k\phi_k^{(k+1)}(t_n) - e^{(k+1)}(t_n) &= y^{(k+1)}(t_n).
\end{aligned}$$

Using Cramer's rule, we find

$$e(t_n) = \frac{\Delta_{top}}{\Delta_{bot}}$$

where

$$\Delta_{top} = \det \begin{bmatrix} \phi_1^{(1)}(t_n) & \cdots & \phi_k^{(1)}(t_n) & y^{(1)}(t_n) \\ \phi_1^{(2)}(t_n) & \cdots & \phi_k^{(2)}(t_n) & y^{(2)}(t_n) \\ \phi_1^{(3)}(t_n) & \cdots & \phi_k^{(3)}(t_n) & y^{(3)}(t_n) \\ \vdots & & \vdots & \vdots \\ \phi_1^{(k+1)}(t_n) & \cdots & \phi_k^{(k+1)}(t_n) & y^{(k+1)}(t_n) \end{bmatrix}$$

and

$$\Delta_{bot} = \det \begin{bmatrix} \phi_1^{(1)}(t_n) & \cdots & \phi_k^{(1)}(t_n) & 0 \\ \phi_1^{(2)}(t_n) & \cdots & \phi_k^{(2)}(t_n) & -c_2^* \\ \phi_1^{(3)}(t_n) & \cdots & \phi_k^{(3)}(t_n) & -c_3^* \\ \vdots & & \vdots & \vdots \\ \phi_1^{(k+1)}(t_n) & \cdots & \phi_k^{(k+1)}(t_n) & -c_{k+1}^* \end{bmatrix}$$

A compilation of the local truncation error estimates is given for various interpolating and approximating functions in Section 3.3.

As in the Adams methods an alternative error estimate may be constructed. Recall (2.2.113), then we define

$$\Phi_{k+1}(t) = \Phi_k(t) + y\left[t_n, \ldots, t_{n-k}, \tau\right]_{\phi_i} \pi_k^*(t),$$

where the generalized shifted product function $\pi_k^*(t)$ is defined by

$$\pi_q^*(t) = \prod_{r=0}^{q} \left[\phi_r(t) - \phi_i(t_{n-r+1})\right]$$

and $\tau \in (t_{n-k}, t_n)$ but τ distinct from $t_{n-k}, \ldots, t_n$. Clearly $\Phi_{k+1}(t)$ interpolates $y(t)$ at $t = t_{n-k}, \ldots, t_n, \tau$, which implies

$$\Phi_k(\tau) = y(\tau) - y\left[t_n, \ldots, t_{n-k}, \tau\right]_{\phi_i} \pi_k^*(\tau).$$

Differentiating gives

$$y'(\tau) - \Phi_k'(\tau) = \frac{d}{dt} y\left[t_n, \ldots, t_{n-k}, t\right]_{\phi_i} \pi_k^*(t)\big|_{t=\tau}.$$

Again, due to the complex nature of the divided difference $y\left[t_n, \ldots, t_{n-k}, \tau\right]_{\phi_i}$, little practical value is obtained by pursuing this further.

3.3 Compilation of Adams and BDF Methods

k	Local Truncation Error
1	$y^{(3)}(t_n)h^3/12 + O(h^4)$
2	$y^{(4)}(t_n)h^4/24 + O(h^5)$
3	$y^{(5)}(t_n)19h^5/720 + O(h^6)$
4	$y^{(6)}(t_n)3h^6/160 + O(h^7)$
5	$y^{(7)}(t_n)863h^7/60480 + O(h^8)$
6	$y^{(8)}(t_n)275h^8/24192 + O(h^9)$

Table 3.1:
Local truncation error for the Adams–Moulton methods using the classical monomial basis $\{1, t, t^2, \ldots, t^{k+1}\}$.

k	Local Truncation Error
1	$(2y'(t_n) - 3y^{(2)}(t_n) + y^{(3)}(t_n))h^3/12 + O(h^4)$
2	$(-6y'(t_n) + 11y^{(2)}(t_n) - 6y^{(3)}(t_n) + y^{(4)}(t_n))h^4/24 + O(h^5)$
3	$19(24y'(t_n) - 50y^{(2)}(t_n) + 35y^{(3)}(t_n) - 10y^{(4)}(t_n) + y^{(5)})h^5/720 + O(h^6)$

Table 3.2:
Local truncation error for the Adams–Moulton methods using the positive exponential basis $\{1, e^t, e^{2t}, \ldots, e^{(k+1)t}\}$.

k	Local Truncation Error
1	$(2y'(t_n) + 3y^{(2)}(t_n) + y^{(3)}(t_n))h^3/12 + O(h^4)$
2	$(6y'(t_n) + 11y^{(2)}(t_n) + 6y^{(3)}(t_n) + y^{(4)}(t_n))h^4/24 + O(h^5)$
3	$19(24y'(t_n) + 50y^{(2)}(t_n) + 35y^{(3)}(t_n) + 10y^{(4)}(t_n) + y^{(5)})h^5/720 + O(h^6)$

Table 3.3:
Local truncation error for the Adams–Moulton methods using the negative exponential basis $\{1, e^{-t}, e^{-2t}, \ldots, e^{-(k+1)t}\}$.

k	Local Truncation Error
1	$y^{(2)}(t_n)h^2/2 + O(h^3)$
2	$y^{(3)}(t_n)2h^3/9 + O(h^4)$
3	$y^{(4)}(t_n)3h^4/22 + O(h^5)$
4	$y^{(5)}(t_n)12h^5/125 + O(h^6)$
5	$y^{(6)}(t_n)10h^6/137 + O(h^7)$
6	$y^{(7)}(t_n)20h^7/343 + O(h^8)$

Table 3.4:
Local truncation error for the BDF methods using the classical monomial basis $\{1, t, t^2, \ldots, t^k\}$.

k	Local Truncation Error
1	$(-y'(t_n) + y^{(2)}(t_n))h^2/2 + O(h^3)$
2	$2(2y'(t_n) - 3y^{(2)}(t_n) + y^{(3)}(t_n))h^3/9 + O(h^4)$
3	$3(-6y'(t_n) + 11y^{(2)}(t_n) - 6y^{(3)}(t_n) + y^{(4)}(t_n))h^4/22 + O(h^5)$

Table 3.5:
Local truncation error for the BDF methods using the positive exponential basis $\{1, e^t, e^{2t}, \ldots, e^{kt}\}$.

k	Local Truncation Error
1	$(y'(t_n) + y^{(2)}(t_n))h^2/2 + O(h^3)$
2	$2(2y'(t_n) + 3y^{(2)}(t_n) + y^{(3)}(t_n))h^3/9 + O(h^4)$
3	$3(6y'(t_n) + 11y^{(2)}(t_n) + 6y^{(3)}(t_n) + y^{(4)}(t_n))h^4/22 + O(h^5)$

Table 3.6:
Local truncation error for the BDF methods using the negative exponential basis $\{1, e^{-t}, e^{-2t}, \ldots, e^{-kt}\}$.

Chapter 4

Stability Analysis

The purpose of this chapter is to examine how sensitive the generalized Adams and BDF methods are to small perturbations. The method of analysis presented below differs from the more usual methods for the following reasons:

1. In general, the interpolating and approximating basis $\{1, \phi_1(t), \ldots, \phi_k(t)\}$ produces coefficients α_j and β_j that are *dependent* on t_n as well as the step-size h. This implies the Adams and BDF methods are nonconstant linear difference equations. It is well known that exact solutions to the general nonconstant linear difference problem cannot be produced in terms of elementary functions [31]. Hence, the usual approach to stability analysis fails because of this dependency of knowing the exact or explicit solutions.

2. The concept of stability of a numerical method means that the method does not propagate or amplify errors as repeated application of the method

is made. This underlying idea is reinforced in the following analysis, whereas in the classical approach the equally valid but less intuitive concept of importance is that of extraneous solutions [4, 6, 9, 11, 16, 18, 24, 27, 28]. The following analysis when applied to the classical Adams and BDF methods reduces to the usual stability analysis. Hence the analysis provided below is a generalization of the usual analysis.

3. The Lyapunov stability [12] of a system of differential equations may be determined by an application of the ideas developed in this alternative analysis. This is not true however of the classical approach to stability analysis.

4. The following method is also applicable to the study of nonlinear dynamical systems and the location of chaotic regimes [29, 33].

We now discuss the idea behind the analysis. It will be shown in the following sections that when the k step Adams or BDF methods are subjected to small linear perturbations, then the equation describing the resulting behavior is given by

$$\vec{\delta}_n = \underset{\sim}{\Delta}_n \vec{\delta}_{n-1} \qquad n \geq 1, \tag{4.0.1}$$

where $\underset{\sim}{\Delta}_n$ is a $(k \times k)$ matrix whose elements are dependent on n. The vector $\vec{\delta}_n$ and $\vec{\delta}_{n-1}$ represent the perturbations at the n th and $(n-1)$ st stage respectively.

Assume that a small perturbation with norm

$$\| \vec{\delta}_0 \| = \epsilon$$

is introduced, where $\epsilon > 0$. For ease of notation, the matrix and vector norms will be denoted by $\| \, . \, \|$ without any distinguishing marks. From (4.0.1) we have

$$\vec{\delta}_n = \underset{\sim}{\Delta}_n \underset{\sim}{\Delta}_{n-1} \cdots \underset{\sim}{\Delta}_2 \underset{\sim}{\Delta}_1 \vec{\delta}_0, \qquad n \geq 0.$$

Let $\underset{\sim}{\mathbf{T}}_{\mathbf{n}}$ be defined to be the matrix product

$$\underset{\sim}{\mathbf{T}}_{\mathbf{n}} = \prod_{j=1}^{n} \underset{\sim}{\Delta}_{n-j+1}.$$

The solution of (4.0.1) in terms of the matrix product $\underset{\sim}{\mathbf{T}}_{\mathbf{n}}$ is given by

$$\vec{\delta}_n = \underset{\sim}{\mathbf{T}}_{\mathbf{n}} \vec{\delta}_0,$$

and therefore the behavior of the sequence $\left\{\| \vec{\delta}_n \|\right\}_{n=0}^{\infty}$ depends on $\left\{\underset{\sim}{\mathbf{T}}_{\mathbf{n}}\right\}_{n=0}^{\infty}$. We now determine what conditions are needed on $\underset{\sim}{\mathbf{T}}_{\mathbf{n}}$ in order for stable methods to result.

Since any sequence of positive numbers is either bounded or unbounded, there exists only two choices for $\left\{\| \vec{\delta}_n \|\right\}_{n=0}^{\infty}$. If the sequence $\left\{\| \vec{\delta}_n \|\right\}_{n=0}^{\infty}$ is unbounded we would unequivocally say that the method is unstable. Now suppose that the sequence $\left\{\| \vec{\delta}_n \|\right\}_{n=0}^{\infty}$ is bounded by M, that is

$$\exists M > 0 \qquad \ni \qquad \| \vec{\delta}_n \| \leq M \qquad \forall n \geq 0.$$

This means that for a small initial disturbance $\vec{\delta}_0$, the propagation thru repeated application of the method is no larger than M. In some problems a large M

would be tolerable, while in other problems this same M would be unacceptable. Since the value of M is highly problem dependent, its practical value is at best questionable.

If we require that $\left\{\| \vec{\delta}_n \|\right\}_{n=0}^{\infty}$ not only be bounded by M, but additionally

$$M = \epsilon,$$

then the propagation of the disturbance is no larger than the initial disturbance. Since this condition is more desirable than the previous, we are led to make the following definition.

Definition 4.1 (Conditional Stability) *Let*

$$\vec{\delta}_n = \mathbf{T_n}\vec{\delta}_0 \quad \forall n \geq 0,$$

with

$$\| \vec{\delta}_0 \| = \epsilon.$$

The sequence $\left\{\vec{\delta}_n\right\}_{n=0}^{\infty}$ *is said to be conditionally stable with bound* ϵ *provided*

$$\| \vec{\delta}_n \| \leq \epsilon \qquad \forall n \geq 0.$$

The reason the term conditional is used instead of unconditional is that although the sequence $\left\{\| \vec{\delta}_n \|\right\}_{n=0}^{\infty}$ remains small, this condition does not imply

$$\lim_{n\to\infty} \| \vec{\delta}_n \| = 0.$$

This stronger condition implies the initial disturbance is damped out upon repeated application of the method. Hence we make the following definition.

Definition 4.2 (Unconditional Stability) *Let*

$$\vec{\delta}_n = \underset{\sim}{\mathbf{T}}_{\mathbf{n}}\, \vec{\delta}_0 \quad \forall n \geq 0,$$

with

$$\| \vec{\delta}_0 \| = \epsilon.$$

The sequence $\left\{\vec{\delta}_n\right\}_{n=0}^{\infty}$ *is said to be unconditionally stable with bound* ϵ *provided*

1. *The sequence* $\left\{\vec{\delta}_n\right\}_{n=0}^{\infty}$ *is conditionally stable with bound* ϵ.
2. $\lim_{n\to\infty} \| \vec{\delta}_n \| = 0$.

We now give criteria for determining conditional and unconditional stability.

Theorem 4.1 (Conditional Stability) *If*

$$\| \underset{\sim}{\Delta}_n \| \leq 1 \quad \forall n \geq 1,$$

then the sequence $\left\{\vec{\delta}_n\right\}_{n=0}^{\infty}$ *is conditionally stable.*

Proof: Since the matrix product satisfies the inequality

$$\| \underset{\sim}{\mathbf{T}}_{\mathbf{n}} \| = \| \prod_{j=1}^{n} \underset{\sim}{\Delta}_{n-j+1} \| \leq \prod_{j=1}^{n} \| \underset{\sim}{\Delta}_{n-j+1} \| \leq 1 \qquad \forall n \geq 1,$$

this implies

$$\| \underset{\sim}{\mathbf{T}}_{\mathbf{n}} \| \leq 1 \qquad \forall n \geq 1.$$

By definition of operator norm

$$\frac{\| \vec{\delta}_n \|}{\epsilon} = \frac{\| \underset{\sim}{\mathbf{T}}_{\mathbf{n}} \vec{\delta}_0 \|}{\| \vec{\delta}_0 \|} \leq \sup_{\|\vec{\delta}_0\| \neq 0} \frac{\| \underset{\sim}{\mathbf{T}}_{\mathbf{n}} \vec{\delta}_0 \|}{\| \vec{\delta}_0 \|} = \| \underset{\sim}{\mathbf{T}}_{\mathbf{n}} \| \leq 1,$$

from which

$$\| \vec{\delta}_n \| \leq \epsilon \qquad \forall n \geq 1.$$

Hence the result.

Theorem 4.2 (Unconditional Stability) *If the sequence* $\left\{\vec{\delta}_n\right\}_{n=0}^{\infty}$ *is conditionally stable and if*

$$\lim_{n\to\infty} \| \underset{\sim}{\Delta}_n \| = 0,$$

then the sequence $\left\{\vec{\delta}_n\right\}_{n=0}^{\infty}$ *is unconditionally stable.*

Proof: Given

$$\exists N > 0 \qquad \ni \qquad \| \underset{\sim}{\Delta}_m \| < \frac{1}{2} \qquad \forall m \geq N.$$

Let m be the smallest such integer, then for $n > m$ we have

$$\| \underset{\sim}{\Delta}_n \cdots \underset{\sim}{\Delta}_m \underset{\sim}{\Delta}_{m-1} \cdots \underset{\sim}{\Delta}_1 \| \leq \| \underset{\sim}{\mathbf{T}}_{\mathbf{m-1}} \| \prod_{j=0}^{n-m} \| \underset{\sim}{\Delta}_{n-j} \| < \| \underset{\sim}{\mathbf{T}}_{\mathbf{m-1}} \| \left(\frac{1}{2}\right)^{n-m+1}.$$

Let

$$b(m) = \max\left\{\| \underset{\sim}{\mathbf{T}}_{\mathbf{m-1}} \|, 1\right\},$$

then

$$\| \underset{\sim}{\mathbf{T}}_{\mathbf{n}} \| < b(m) \left(\frac{1}{2}\right)^{n-m+1} \qquad \forall n, m \geq N.$$

Taking the limit $\lim_{n\to\infty}$ of both sides, while holding m fixed implies

$$\| \underset{\sim}{\mathbf{T}}_{\mathbf{n}} \| \to 0 \qquad n \to 0.$$

Now $\| \underset{\sim}{\mathbf{T}}_{\mathbf{n}} \| \to 0$ implies

$$\forall \gamma^* > 0, \qquad \exists N > 0 \qquad \ni \qquad \| \underset{\sim}{\mathbf{T}}_{\mathbf{n}} \| < \gamma^* \qquad \forall n \geq N.$$

The definition of operator norm

$$\| \underset{\sim}{\mathbf{T}}_{\mathbf{n}} \| = \sup_{\|\vec{\delta}_0\| \neq 0} \frac{\| \underset{\sim}{\mathbf{T}}_{\mathbf{n}} \vec{\delta}_0 \|}{\| \vec{\delta}_0 \|}$$

implies

$$\| \underset{\sim}{\mathbf{T}}_{\mathbf{n}} \vec{\delta}_0 \| < \gamma^* \epsilon \qquad \forall n \geq N.$$

Let

$$\gamma = \gamma^* \epsilon,$$

we have

$$\forall \gamma > 0, \qquad \exists N > 0 \qquad \ni \qquad \| \vec{\delta}_n \| < \gamma \qquad \forall n \geq N,$$

therefore

$$\lim_{n \to \infty} \| \vec{\delta}_n \| = 0.$$

Hence the result.

Consider the special case where $\underset{\sim}{\Delta}_n$ is independent of n, that is

$$\underset{\sim}{\Delta} = \underset{\sim}{\Delta}_n \qquad \forall n \geq 1.$$

If

$$\rho\left(\underset{\sim}{\Delta}\right) < 1,$$

where $\rho\left(\underset{\sim}{\Delta}\right)$ denotes the spectral radius of $\underset{\sim}{\Delta}$, then

$$\lim_{n \to \infty} \| \underset{\sim}{\Delta}^n \| = 0$$

for some natural norm $\| \,.\, \|$. This is a restatement of the convergence of matrices [9]. Now

$$\lim_{n\to\infty} \| \underset{\sim}{\Delta}^n \| = 0 \quad \Rightarrow \quad \lim_{n\to\infty} \| \underset{\sim}{\mathbf{T}}_{\mathbf{n}} \| = 0.$$

From the proof of Theorem 4.2 we have,

$$\lim_{n\to\infty} \| \vec{\delta}_n \| = 0$$

and hence the sequence $\left\{\vec{\delta}_n\right\}_{n=0}^{\infty}$ is unconditionally stable. It will be shown in the next sections that for the classical Adams and BDF methods, the condition $\rho(\underset{\sim}{\Delta}) < 1$ reduces to the more usual stability criteria requiring the extraneous roots of the linear multistep method be bounded by one.

4.1 Stability of the Adams Methods

Recall the Adams method

$$y_n = \alpha_1 (h, n)\, y_{n-1} + \sum_{j=0}^{k} \beta_j (h, n)\, y'_{n-j},$$

where the coefficients α_1 and β_j are written for emphasis as depending on the stepsize h and the step number n. For the interpolating and approximating basis $\{1, \phi_1(t), \ldots, \phi_{k+1}(t)\}$ it was found previously that $\alpha_1 (h, n) = 1$, from which the Adams method reduces to

$$y_n = y_{n-1} + \sum_{j=0}^{k} \beta_j (h, n)\, y'_{n-j}.$$

Suppose a small perturbation $\delta y_n, \delta y_{n-1}$ and $\delta y'_{n-j}$ are introduced into this method. The resulting difference equation is given by

$$y_n + \delta y_n = y_{n-1} + \delta y_{n-1} + \sum_{j=0}^{k} \beta_j(h,n)\left[y'_{n-j} + \delta y'_{n-j}\right].$$

Subtracting the unperturbed equation from the perturbed equation gives the difference equation

$$\delta y_n = \delta y_{n-1} + \sum_{j=0}^{k} \beta_j(h,n)\,\delta y'_{n-j},$$

whose solution δy_n describes how the perturbation is propagated.

We now invoke the standard assumption that the rate of growth of the perturbation is at worst linear [4, 6, 9, 11, 15, 16, 18, 23, 24, 27, 28], that is

$$\delta y' = \lambda \delta y$$

where λ is a constant. For

$$1 - \lambda\beta_0(h,n) \neq 0$$

we have

$$\delta y_n = \sum_{j=1}^{k} \beta_j^*(h,n,\lambda)\,\delta y_{n-j}$$

where

$$\beta_1^*(h,n,\lambda) = \frac{1 + \lambda\beta_1(h,n)}{1 - \lambda\beta_0(h,n)}$$

and

$$\beta_j^*(h,n,\lambda) = \frac{\lambda\beta_j(h,n)}{1 - \lambda\beta_0(h,n)} \qquad j = 2, \ldots, k.$$

Let n be replaced by $n+k$, and make the change of variables

$$^{(j)}\delta_n = \delta y_{n+j} \qquad j = 1, \ldots, k; \qquad ^{(1)}\delta_{n-1} = \delta y_n,$$

and use the property

$$^{(j)}\delta_n = {}^{(j+1)}\delta_{n-1} \qquad j = 1, \ldots, (k-1),$$

we obtain

$$\vec{\delta}_n = \underset{\sim}{\Delta}_n \vec{\delta}_{n-1} \qquad n \geq 1,$$

where

$$\vec{\delta}_n = col\left({}^{(1)}\delta_n, \ldots, {}^{(k)}\delta_n\right)$$

and $\underset{\sim}{\Delta}_n$ is the companion matrix

$$\underset{\sim}{\Delta}_n = \begin{pmatrix} 0 & 1 & & 0 \\ 0 & 0 & \ddots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 1 \\ \beta^*_k(h,n,\lambda) & \beta^*_{k-1}(h,n,\lambda) & \cdots & \beta^*_1(h,n,\lambda) \end{pmatrix}.$$

This is of course the form discussed in (4.0.1). The criteria for conditional stability is therefore

$$\| \underset{\sim}{\Delta}_n \| \leq 1 \qquad \forall n \geq 1,$$

and for unconditional stability the criteria is

$$\lim_{n\to\infty} \| \underset{\sim}{\Delta}_n \| = 0.$$

Recall that the basis set $\left\{1, t, \ldots, t^{k+1}\right\}$ produced the classical Adams method, for which the coefficients β_j are independent of t_n, that is,

$$\beta_j^* = \beta_j^*(n).$$

This implies the companion matrix $\underset{\sim n}{\Delta}$ is also independent of t_n and can therefore be written as

$$\underset{\sim}{\Delta} = \underset{\sim n}{\Delta}.$$

If

$$\rho\left(\underset{\sim}{\Delta}\right) < 1,$$

then from the characteristic equation

$$\mu^k - \beta_1^* \mu^{k-1} - \beta_2^* \mu^{k-2} - \cdots - \beta_k^* = 0, \tag{4.1.2}$$

we have

$$|\mu| < 1$$

for each eigenvalue μ. The usual stability analysis requires that the difference equation

$$y_n - \beta_1^* y_{n-1} - \beta_2^* y_{n-2} - \cdots - \beta_k^* y_{n-k} = 0$$

has solutions $r(n)$ so that

$$|r(n)| \leq 1 \qquad \forall n \geq 0.$$

If we let $y_n = e^{in\theta}$, then the difference equation reduces to

$$e^{ik\theta} - \beta_1^* e^{i(k-1)\theta} - \beta_2^* e^{i(k-2)\theta} - \cdots - \beta_k^* = 0.$$

This equation is clearly (4.1.2), from which we conclude that the usual stability analysis is a special case of the analysis presented here. A compilation of the stability regions for various interpolating and approximating basis is given in section 4.3.

4.2 Stability of the BDF Methods

Recall the BDF method

$$y_n = \sum_{j=1}^{k} \alpha_j (h,n)\, y_{n-j} + \beta_0 (h,n)\, y_n'.$$

Again the coefficients α_j and β_0 are written for emphasis as depending on the stepsize h and the step number n. Suppose a small perturbation $\delta y_n, \delta y_{n-j}$ and $\delta y_n'$ are introduced into this method. The resulting difference equation is given by

$$y_n + \delta y_n = \sum_{j=1}^{k} \alpha_j (h,n) \left[y_{n-j} + \delta y_{n-j}\right] + \beta_0 (h,n) \left[y_n' + \delta y_n'\right].$$

Subtracting the unperturbed equation from the perturbed equation gives the difference equation

$$\delta y_n = \sum_{j=1}^{k} \alpha_j (h,n)\, \delta y_{n-j} + \beta_0 (h,n)\, \delta y_n',$$

whose solution δy_n again describes how the perturbation propagates.

We again invoke the standard assumption that the rate of growth of the perturbation is at worst linear, that is

$$\delta y' = \lambda \delta y$$

where λ is a constant. For

$$1 - \lambda\beta_0(h,n) \neq 0$$

we have

$$\delta y_n = \sum_{j=1}^{k} \alpha_j^*(h,n,\lambda)\,\delta y_{n-j}$$

where

$$\alpha_j^*(h,n,\lambda) = \frac{\alpha_j(h,n)}{1-\lambda\beta_0(h,n)}, \qquad j = 1,\ldots,k.$$

Let n be replaced by $n+k$, make the change of variables

$${}^{(j)}\delta_n = \delta y_{n+j} \qquad j = 1,\ldots,k; \qquad {}^{(1)}\delta_{n-1} = \delta y_n,$$

and use the property

$${}^{(j)}\delta_n = {}^{(j+1)}\delta_{n-1}, \qquad j = 1,\ldots,(k-1),$$

we get

$$\vec{\delta}_n = \underset{\sim n}{\Delta}\,\vec{\delta}_{n-1} \qquad n \geq 1,$$

where

$$\vec{\delta}_n = col\left({}^{(1)}\delta_n,\ldots,{}^{(k)}\delta_n\right)$$

and $\underset{\sim n}{\Delta}$ is the companion matrix

$$\underset{\sim n}{\Delta} = \begin{pmatrix} 0 & 1 & & 0 \\ 0 & 0 & \ddots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 1 \\ \alpha_k^*(h,n,\lambda) & \alpha_{k-1}^*(h,n,\lambda) & \cdots & \alpha_1^*(h,n,\lambda) \end{pmatrix}.$$

This is of course the form discussed in (4.0.1). The criteria for conditional stability is therefore

$$\| \underset{\sim n}{\Delta} \| \leq 1 \qquad \forall \quad n \geq 1,$$

and for unconditional stability the criteria is

$$\lim_{n \to \infty} \| \underset{\sim n}{\Delta} \| = 0.$$

Recall that the basis set $\left\{1, t, \ldots, t^k\right\}$ produced the classical BDF method where the coefficients α_j are independent of t_n, that is

$$\alpha_j^* = \alpha_j^* \left(n\right).$$

This implies the companion matrix $\underset{\sim n}{\Delta}$ is also independent of t_n, and therefore we write

$$\underset{\sim}{\Delta} = \underset{\sim n}{\Delta}.$$

If

$$\rho \left(\underset{\sim}{\Delta}\right) < 1,$$

then from the characteristic equation

$$\mu^k - \alpha_1^* \mu^{k-1} - \alpha_2^* \mu^{k-2} - \cdots - \alpha_k^* = 0 \tag{4.2.3}$$

we have

$$| \mu | < 1$$

for each eigenvalue μ. The usual stability analysis requires that the difference equation

$$y_n - \alpha_1^* y_{n-1} - \alpha_2^* y_{n-2} - \cdots - \alpha_k^* y_{n-k} = 0$$

has solutions $r(n)$ so that

$$|\, r(n) \,| \leq 1 \qquad \forall \quad n.$$

If we let $y_n = e^{i\,n\,\theta}$, then the difference equation reduces to

$$e^{i\,k\,\theta} - \alpha_1^* e^{i\,(k-1)\theta} - \alpha_2^* e^{i\,(k-2)\theta} - \cdots - \alpha_k^* = 0.$$

This equation is clearly (4.2.3), from which we conclude that the usual stability analysis is a special case of the analysis presented here. A compilation of the stability regions for various interpolating and approximating functions is given in the next section.

4.3 Compilation of Adams and BDF Methods

Recall that in the Adams and BDF methods the stability is determined by the behavior of

$$\vec{\delta}_n = \underset{\sim}{\Delta}_n \vec{\delta}_{n-1} \qquad \forall n \geq 1,$$

where $\| \vec{\delta}_0 \| = \epsilon$. Specifically, $\| \underset{\sim}{\Delta}_n \|$ determines whether the method is conditionally or unconditionally stable. Since the basis sets

$$\{1, t, t^2, \ldots, t^k\}, \quad \{1, e^t, e^{2t}, \ldots, e^{kt}\}, \quad \{1, e^{-t}, e^{-2t}, \ldots, e^{-kt}\}$$

produce coefficients α_j and β_j that are independent of t_n, we use the usual criteria for stability. In the following figures, each method graphed depicts the

region $\mathcal{R}$ in the complex plane

$$\mathcal{R} = \left\{ q \in C | \quad q = h\lambda, \quad \rho(\underset{\sim}{\Delta}) \leq 1 \right\}.$$

In other words, the region $\mathcal{R}$ is where the method is unconditionally stable.

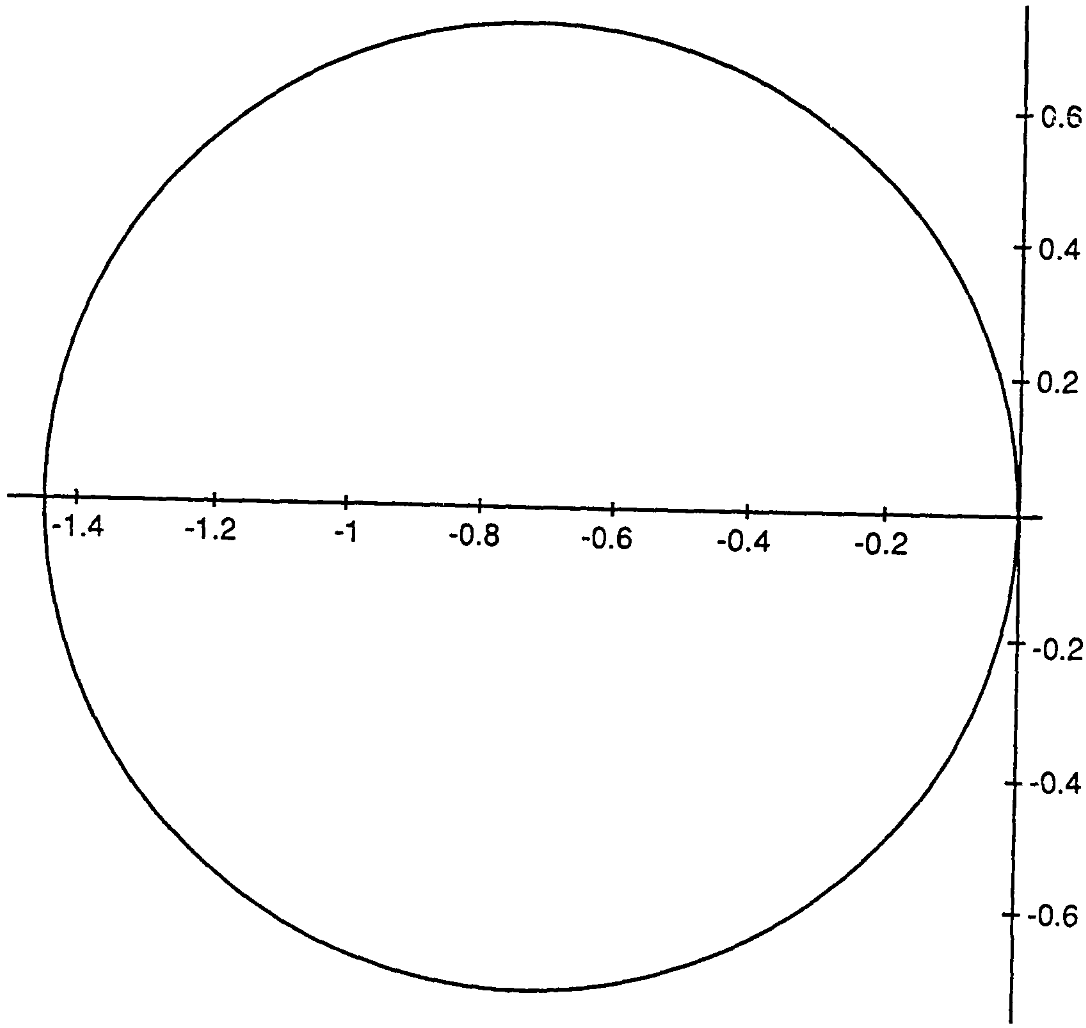

Figure 4.1:
Stability Regions for Adams–Moulton methods, where $h = 2$. Monomial basis $\{1, t, t^2\}$ is stable in the entire left–hand plane. Positive exponential basis $\{1, e^t, e^{2t}\}$ is stable inside the indicated region.

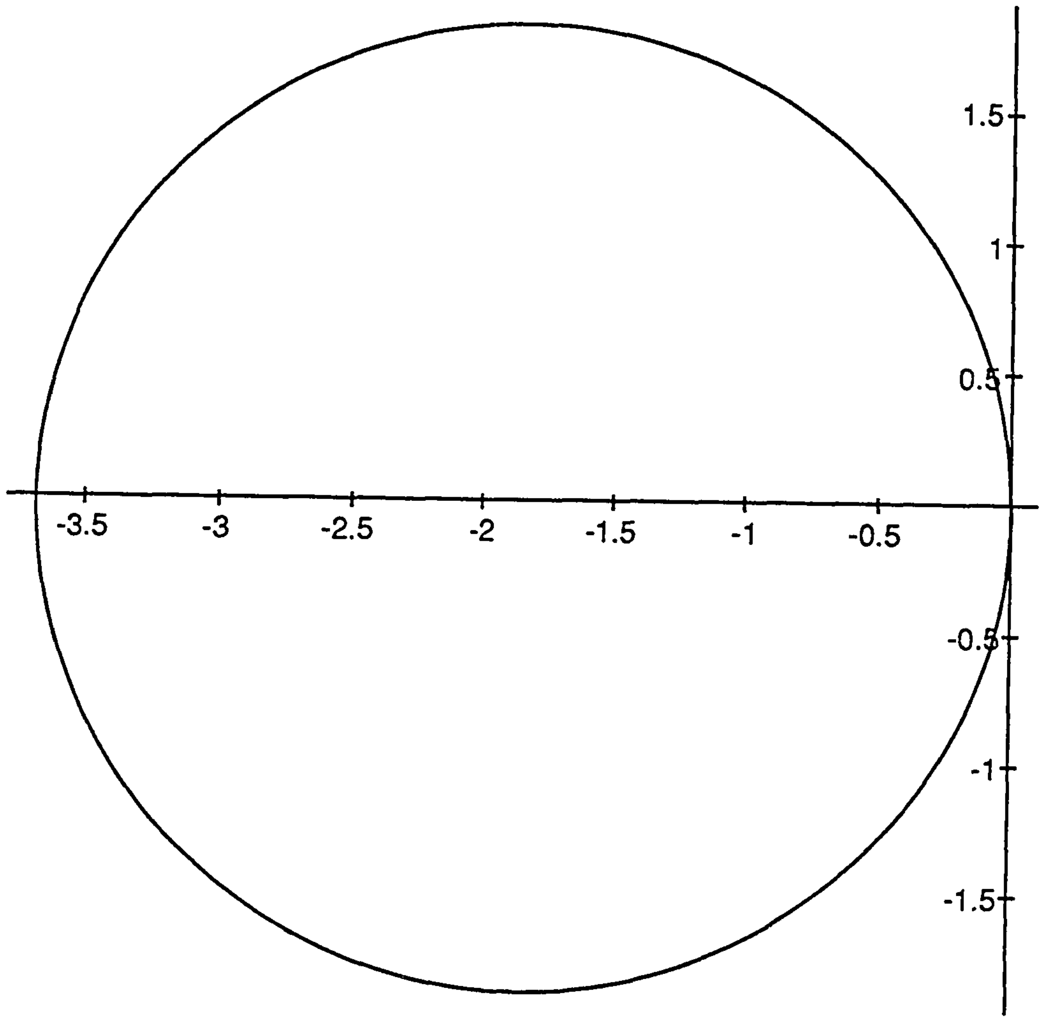

Figure 4.2:
Stability Regions for Adams–Moulton methods, where $h = 1$. Monomial basis $\{1, t, t^2\}$ is stable in the entire left-hand plane. Positive exponential basis $\{1, e^t, e^{2t}\}$ is stable inside the indicated region.

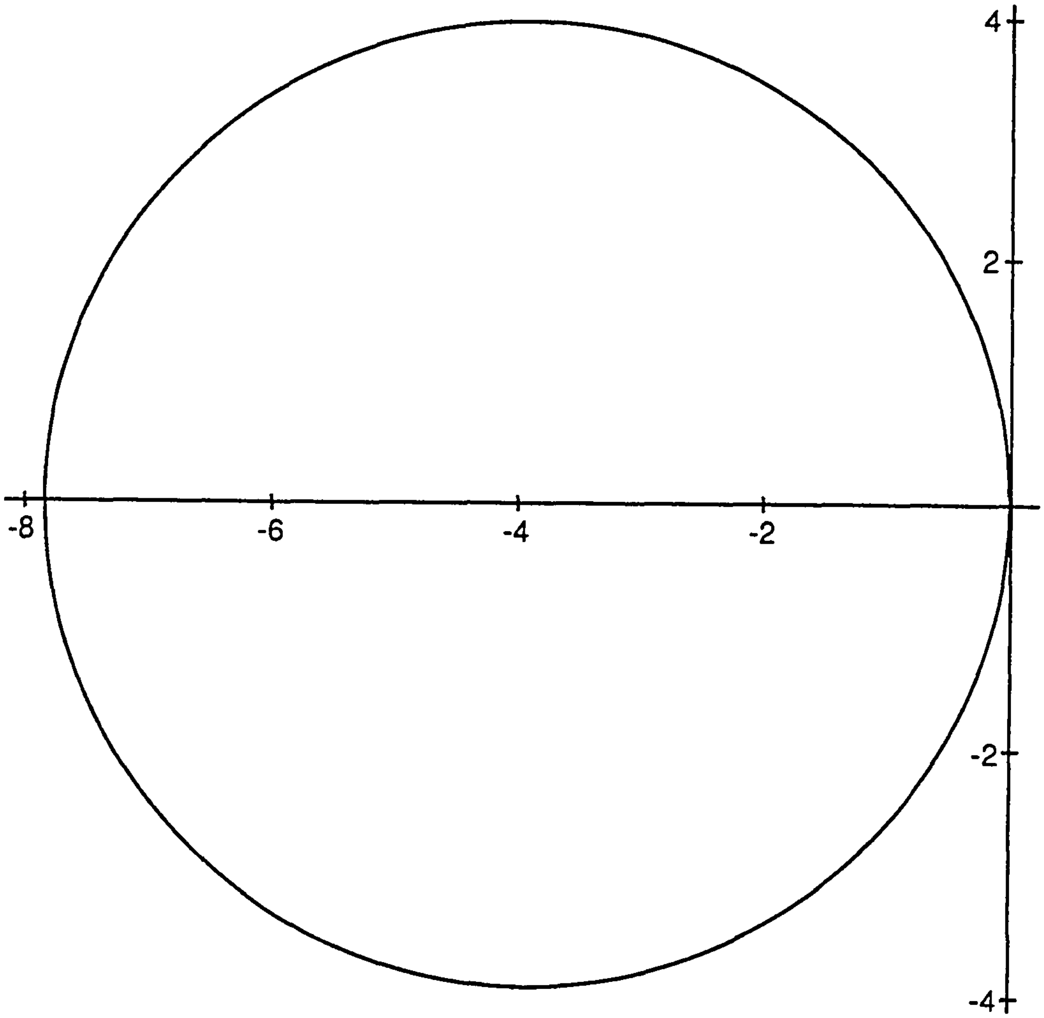

Figure 4.3:
Stability Regions for Adams–Moulton methods, where $h = .5$. Monomial basis $\{1, t, t^2\}$ is stable in the entire left–hand plane. Positive exponential basis $\{1, e^t, e^{2t}\}$ is stable inside the indicated region.

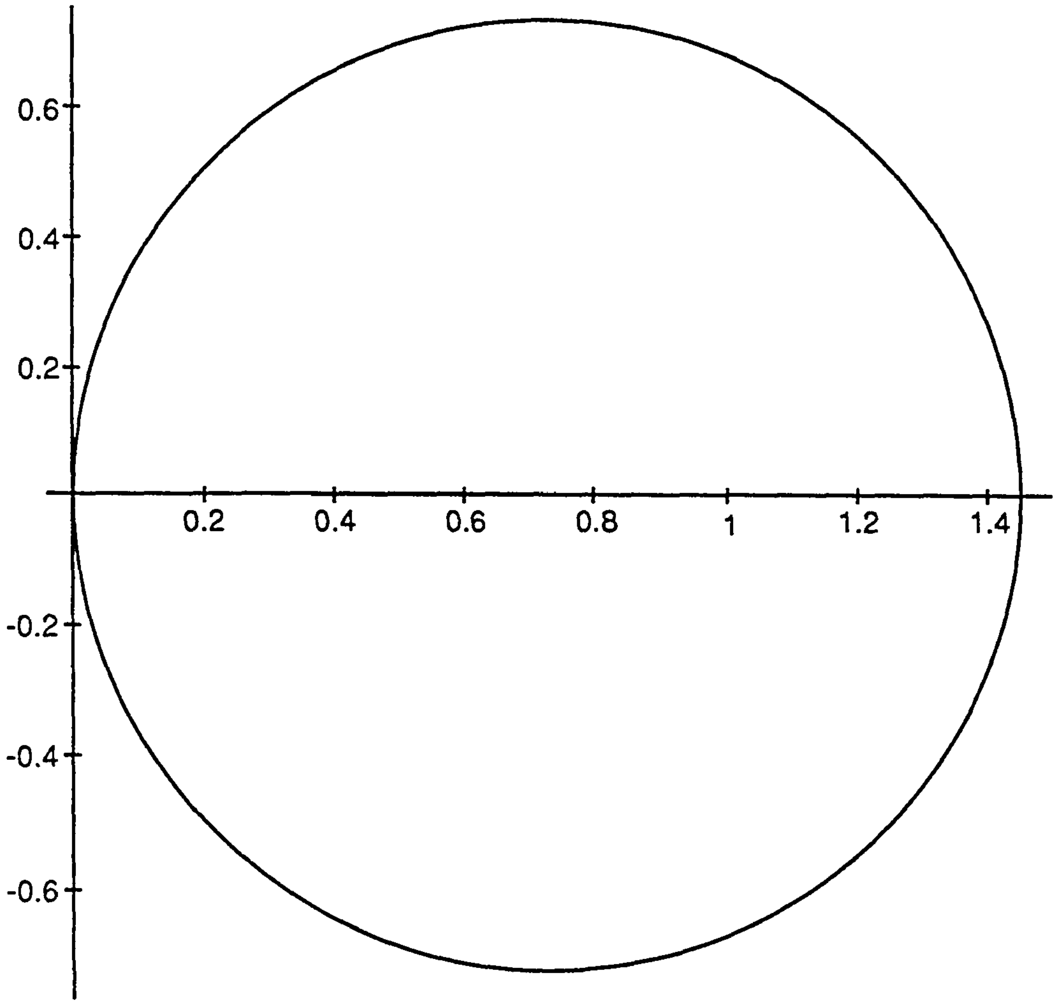

Figure 4.4:
Stability Regions for Adams–Moulton methods, where $h = 2$. Monomial basis $\{1, t, t^2\}$ is stable in the entire left–hand plane. Negative exponential basis $\{1, e^{-t}, e^{-2t}\}$ is stable inside the indicated region.

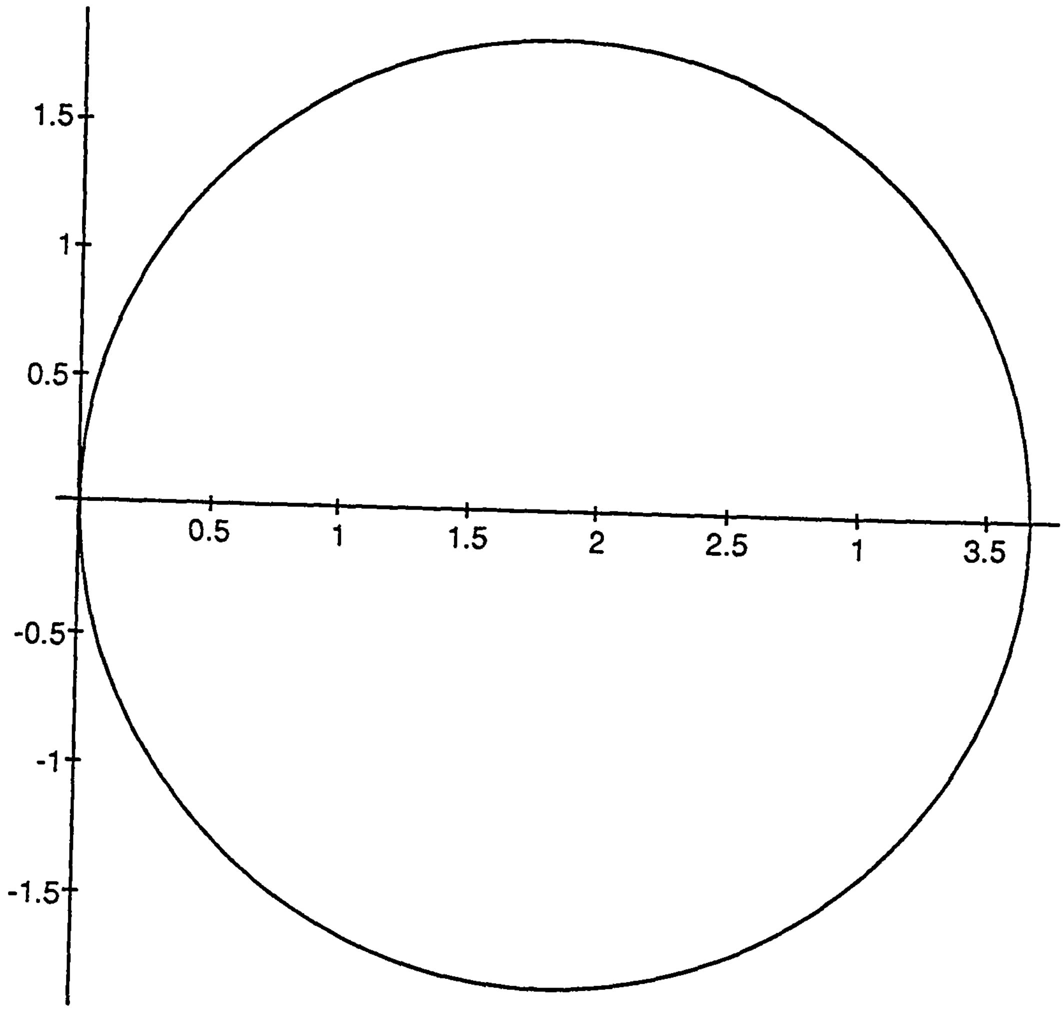

Figure 4.5:
Stability Regions for Adams–Moulton methods, where $h = 1$. Monomial basis $\{1, t, t^2\}$ is stable in the entire left–hand plane. Negative exponential basis $\{1, e^{-t}, e^{-2t}\}$ is stable inside the indicated region.

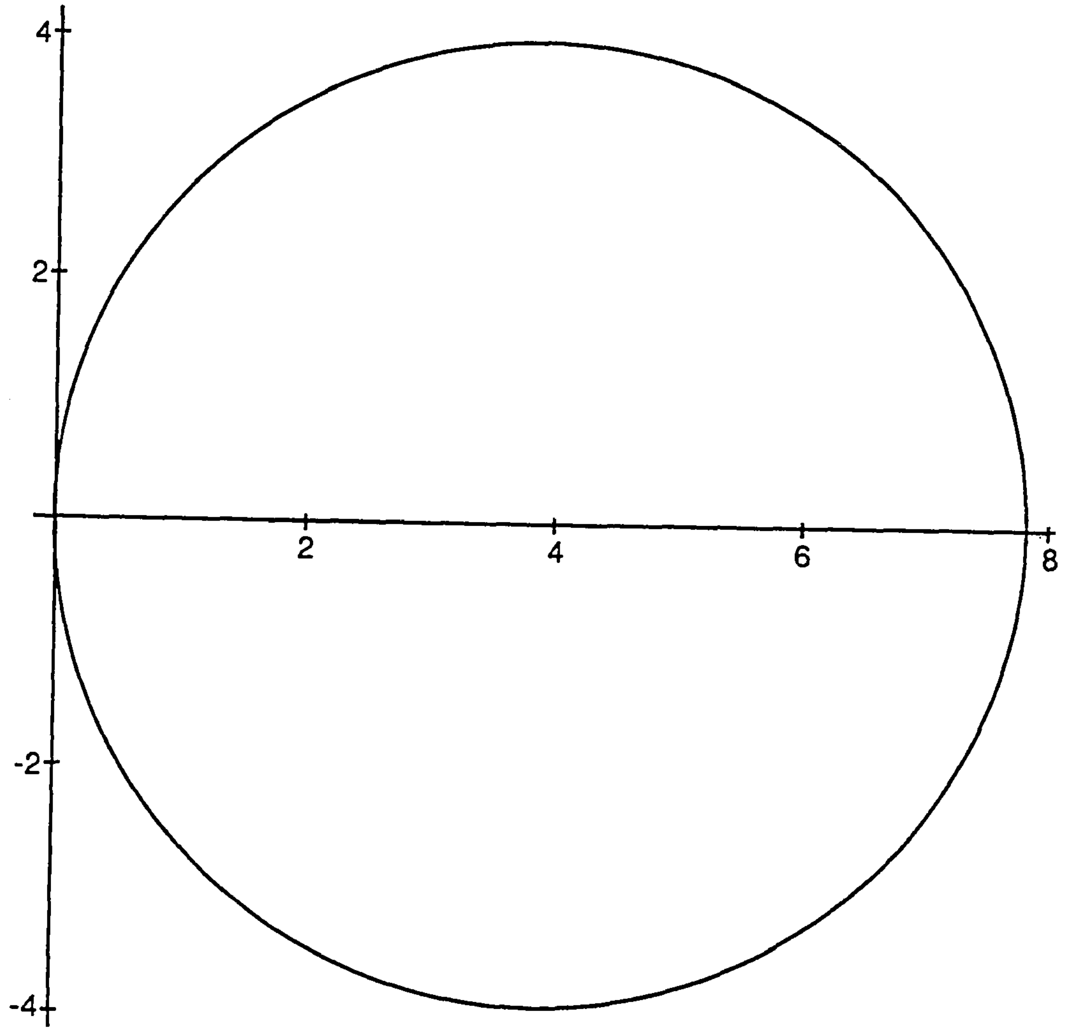

Figure 4.6:
Stability Regions for Adams–Moulton methods, where $h = .5$. Monomial basis $\{1, t, t^2\}$ is stable in the entire left-hand plane. Negative exponential basis $\{1, e^{-t}, e^{-2t}\}$ is stable inside the indicated region.

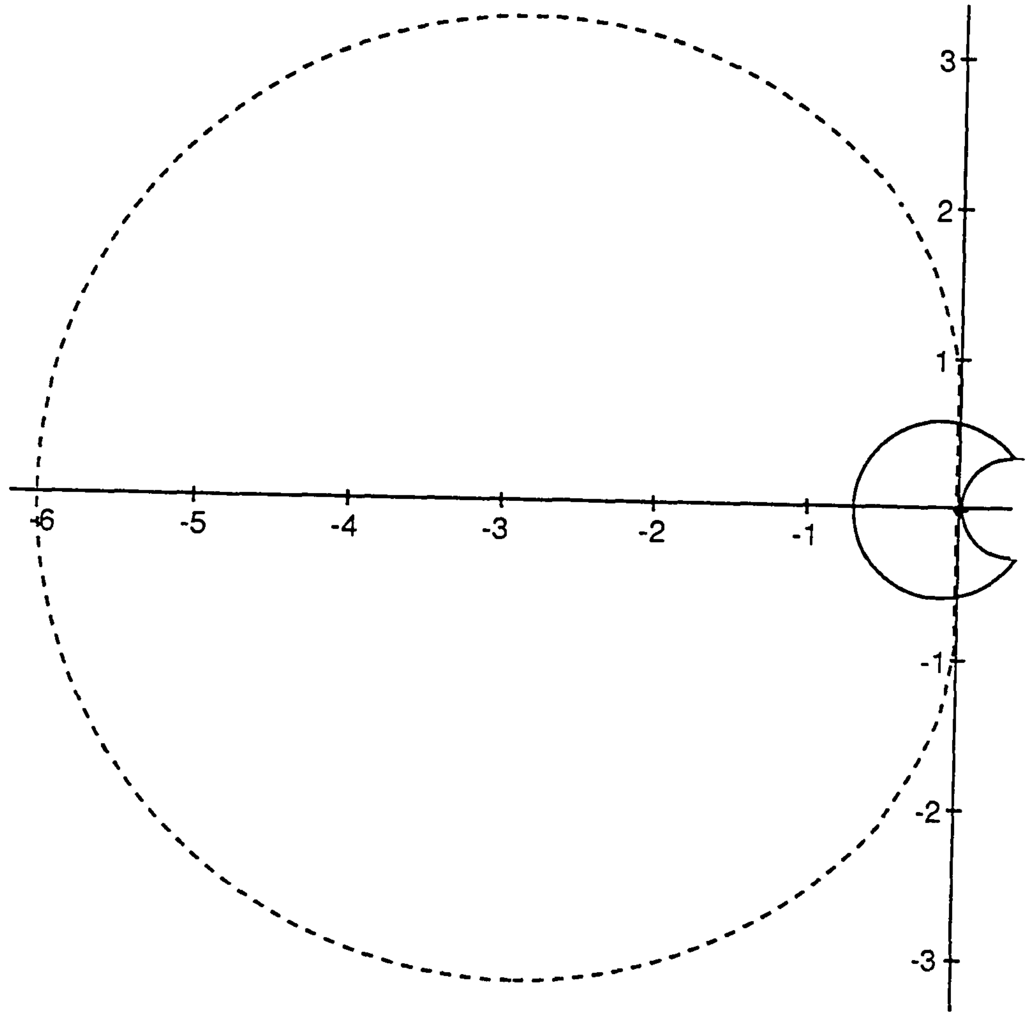

Figure 4.7:
Stability Regions for Adams–Moulton methods, where $h = 1$. Monomial basis $\{1, t, t^2, t^3\}$ is shown as a dashed curve. Positive exponential basis $\{1, e^t, e^{2t}, e^{3t}\}$ is shown as a solid curve. Methods are stable inside the indicated region.

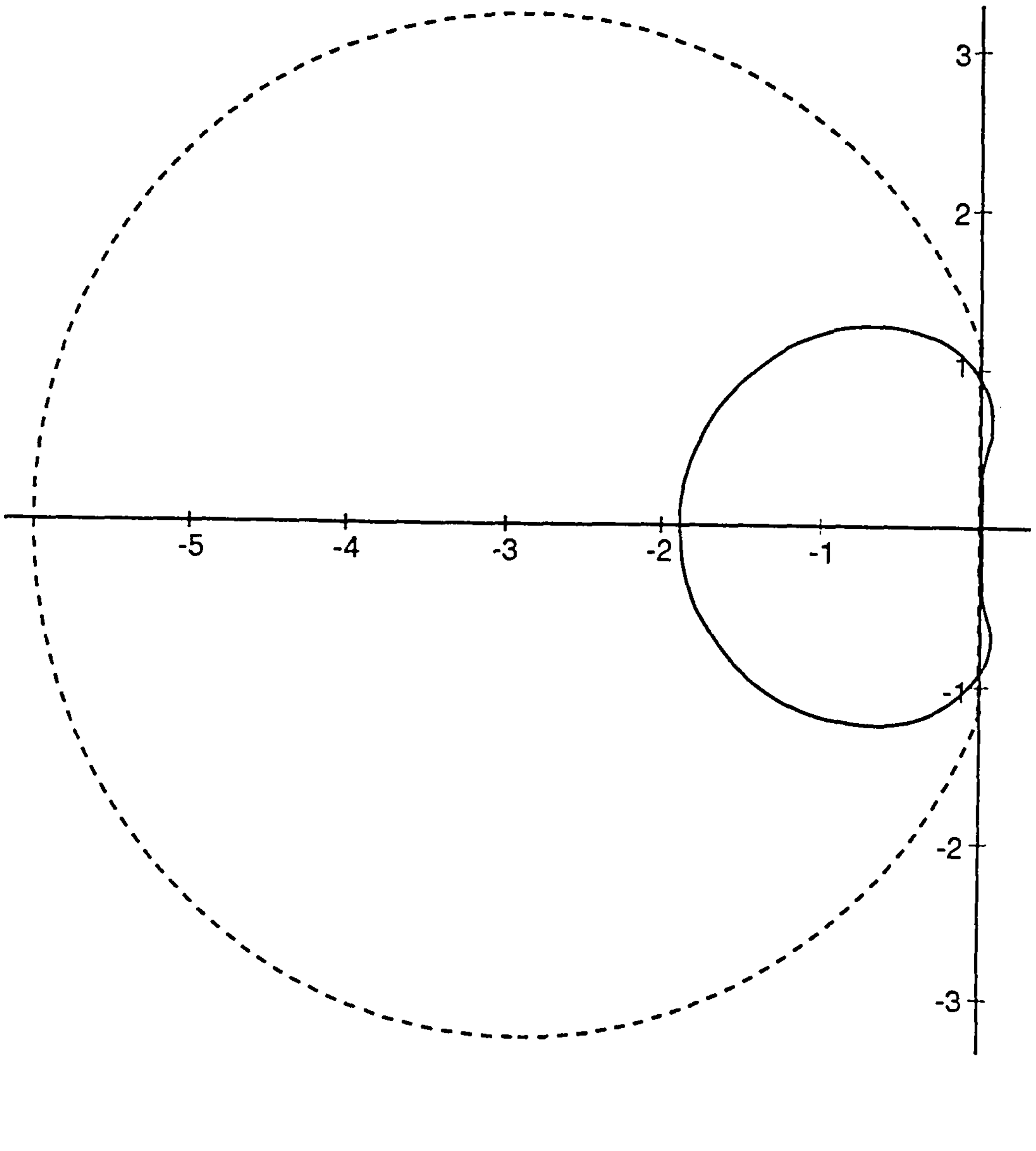

Figure 4.8:
Stability Regions for Adams–Moulton methods, where $h = .5$. Monomial basis $\{1, t, t^2, t^3\}$ is shown as a dashed curve. Positive exponential basis $\{1, e^t, e^{2t}, e^{3t}\}$ is shown as a solid curve. Methods are stable inside the indicated region.

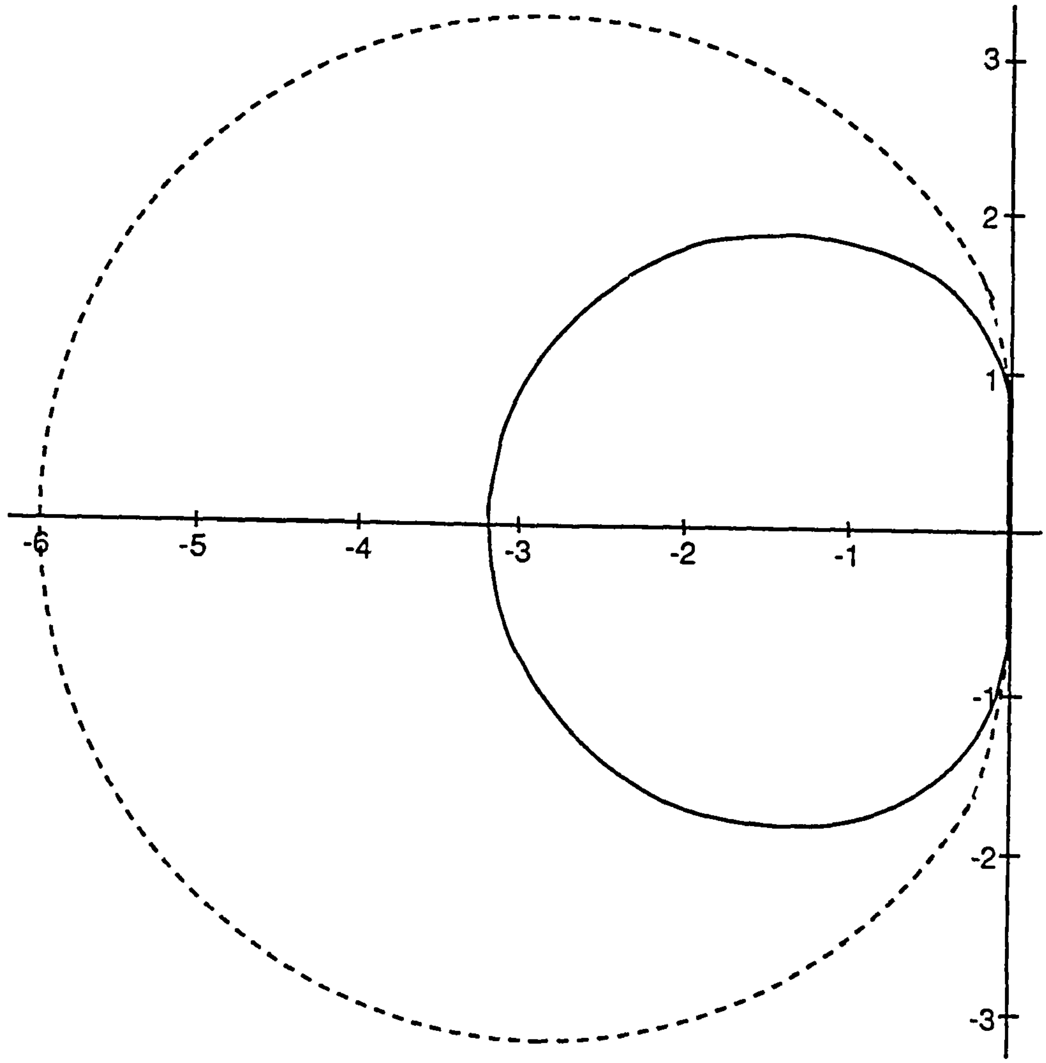

Figure 4.9:
Stability Regions for Adams–Moulton methods, where $h = .25$. Monomial basis $\{1, t, t^2, t^3\}$ is shown as a dashed curve. Positive exponential basis $\{1, e^t, e^{2t}, e^{3t}\}$ is shown as a solid curve. Methods are stable inside the indicated region.

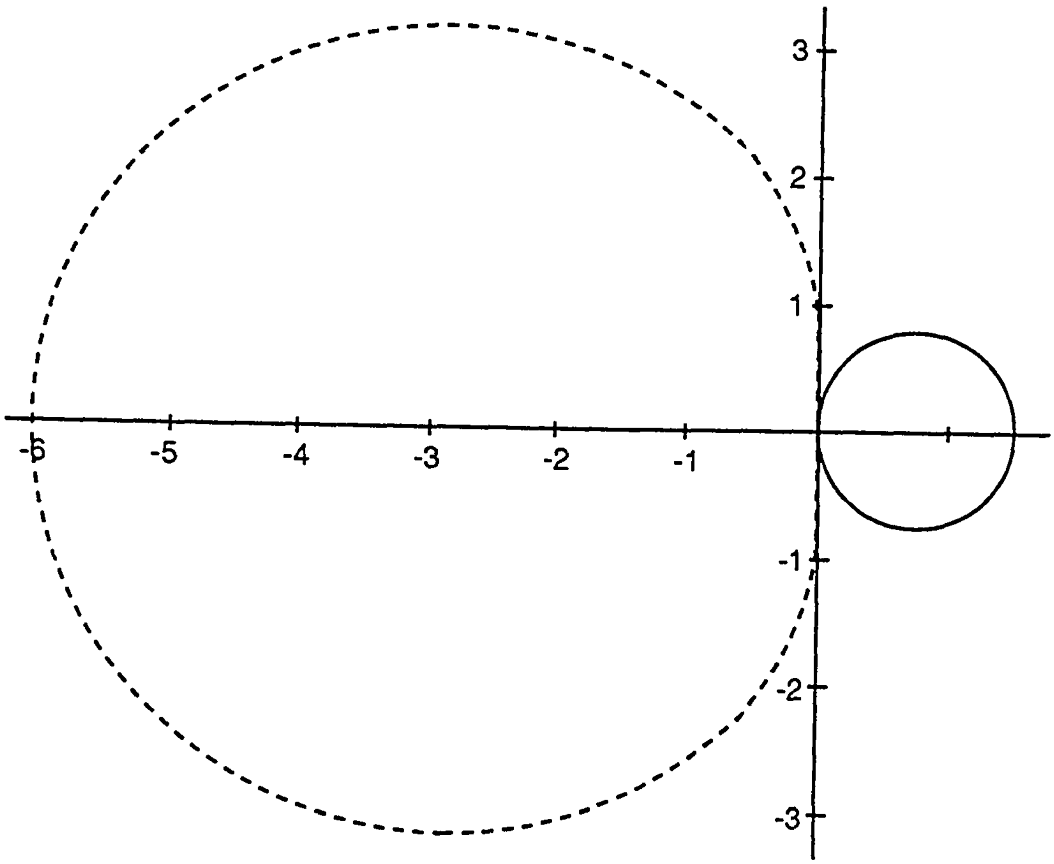

Figure 4.10:
Stability Regions for Adams–Moulton methods, where $h = 2$. Monomial basis $\{1, t, t^2, t^3\}$ is shown as a dashed curve. Negative exponential basis $\{1, e^{-t}, e^{-2t}, e^{-3t}\}$ is shown as a solid curve. Methods are stable inside the indicated region.

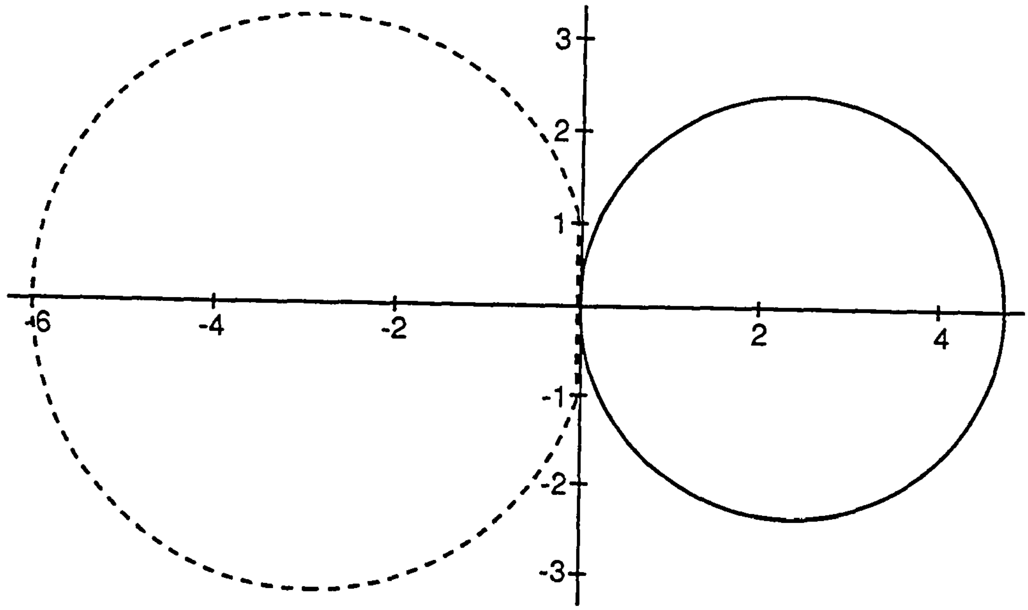

Figure 4.11:
Stability Regions for Adams–Moulton methods, where $h = 1$. Monomial basis $\{1, t, t^2, t^3\}$ is shown as a dashed curve. Negative exponential basis $\{1, e^{-t}, e^{-2t}, e^{-3t}\}$ is shown as a solid curve. Methods are stable inside the indicated region.

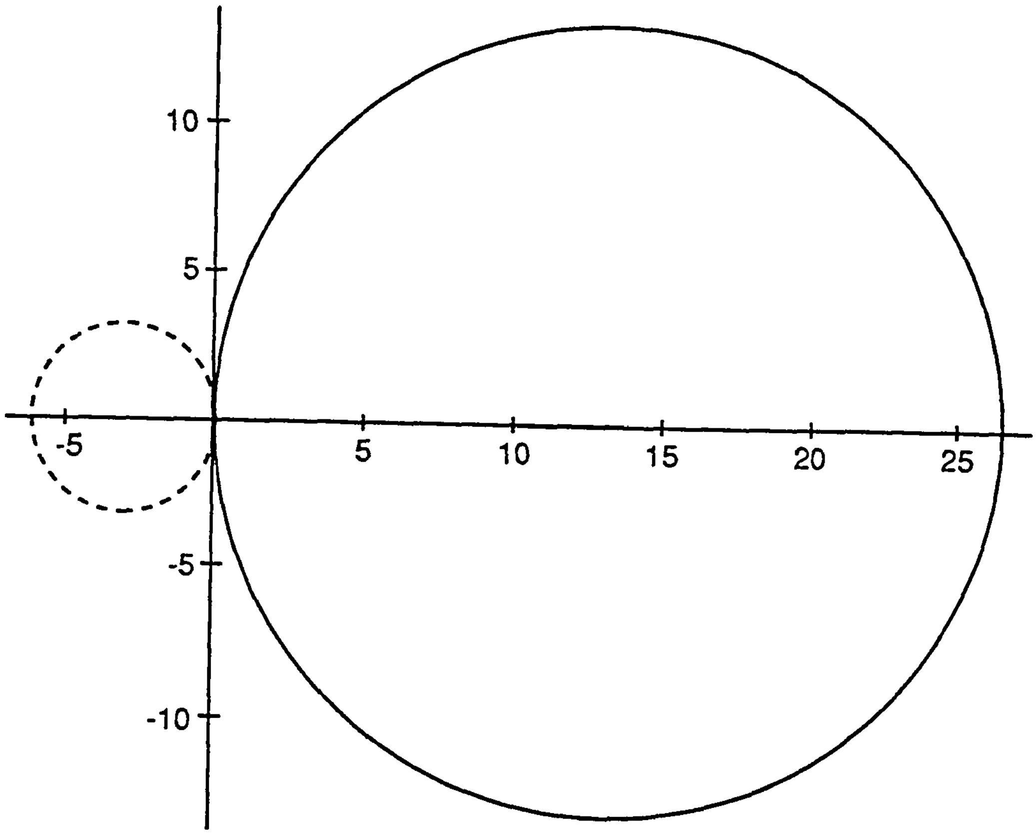

Figure 4.12:
Stability Regions for Adams–Moulton methods, where $h = .5$. Monomial basis $\{1, t, t^2, t^3\}$ is shown as a dashed curve. Negative exponential basis $\{1, e^{-t}, e^{-2t}, e^{-3t}\}$ is shown as a solid curve. Methods are stable inside the indicated region.

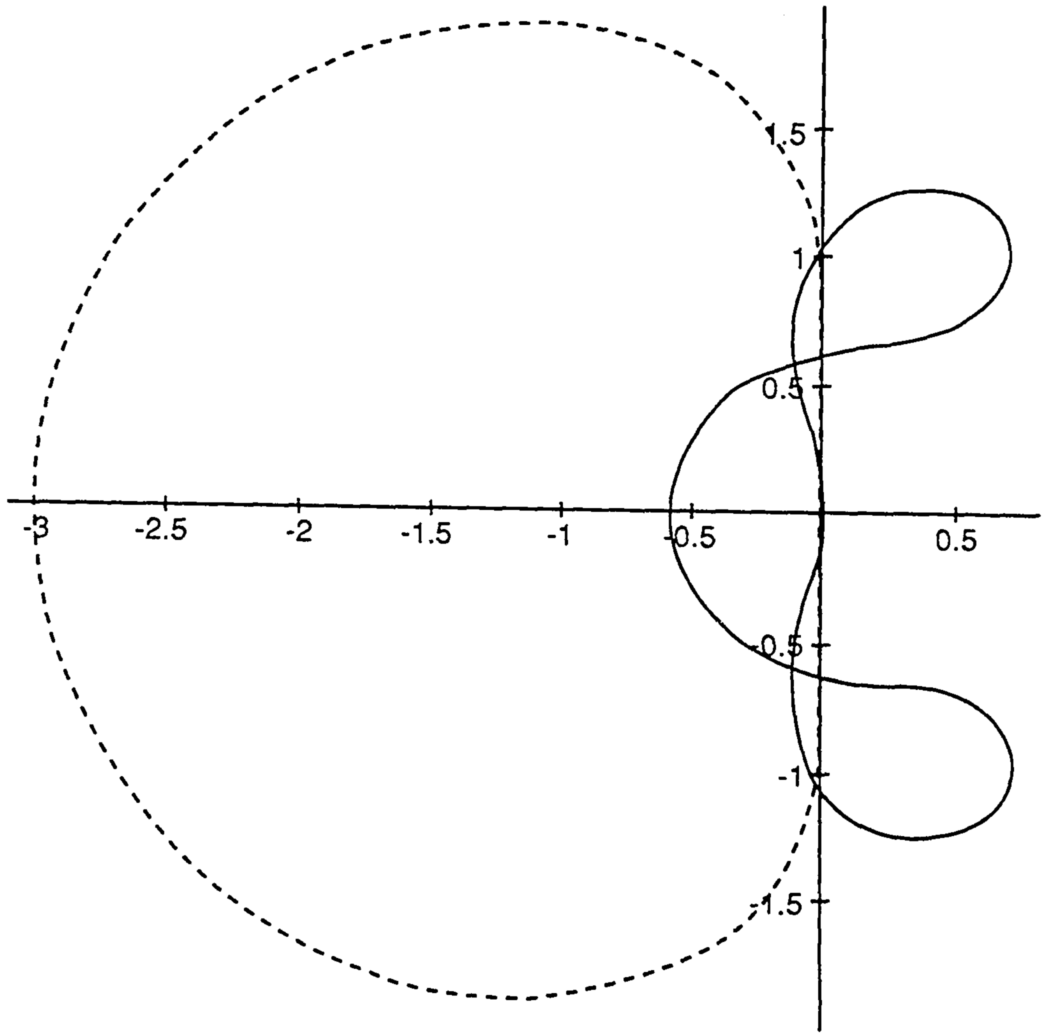

Figure 4.13:
Stability Regions for Adams–Moulton methods, where $h = .5$. Monomial basis $\{1, t, t^2, t^3, t^4\}$ is shown as a dashed curve. Positive exponential basis $\{1, e^t, e^{2t}, e^{3t}, e^{4t}\}$ is shown as a solid curve. Methods are stable inside the indicated region.

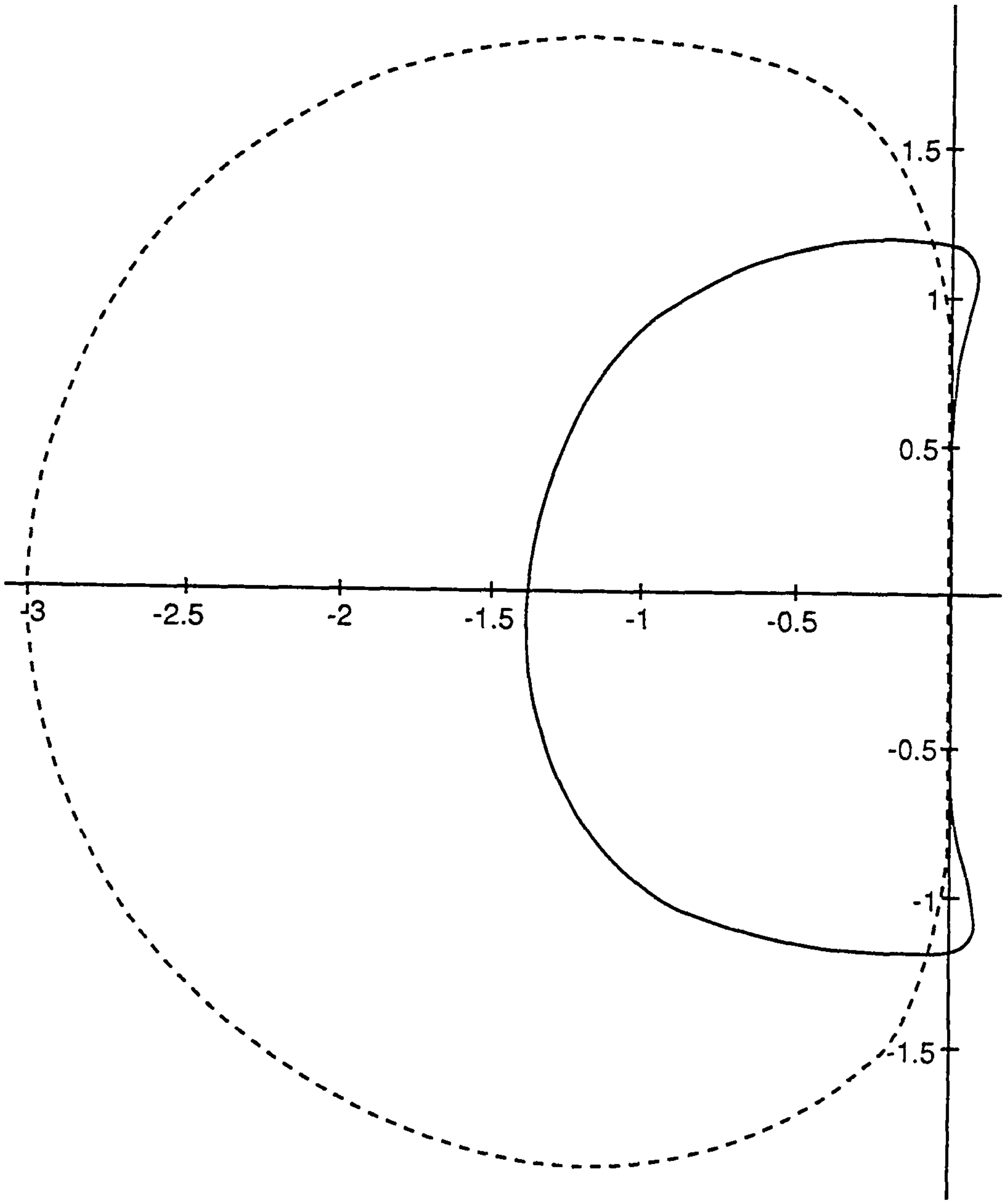

Figure 4.14:
Stability Regions for Adams–Moulton methods, where $h = .25$. Monomial basis $\{1, t, t^2, t^3, t^4\}$ is shown as a dashed curve. Positive exponential basis $\{1, e^t, e^{2t}, e^{3t}, e^{4t}\}$ is shown as a solid curve. Methods are stable inside the indicated region.

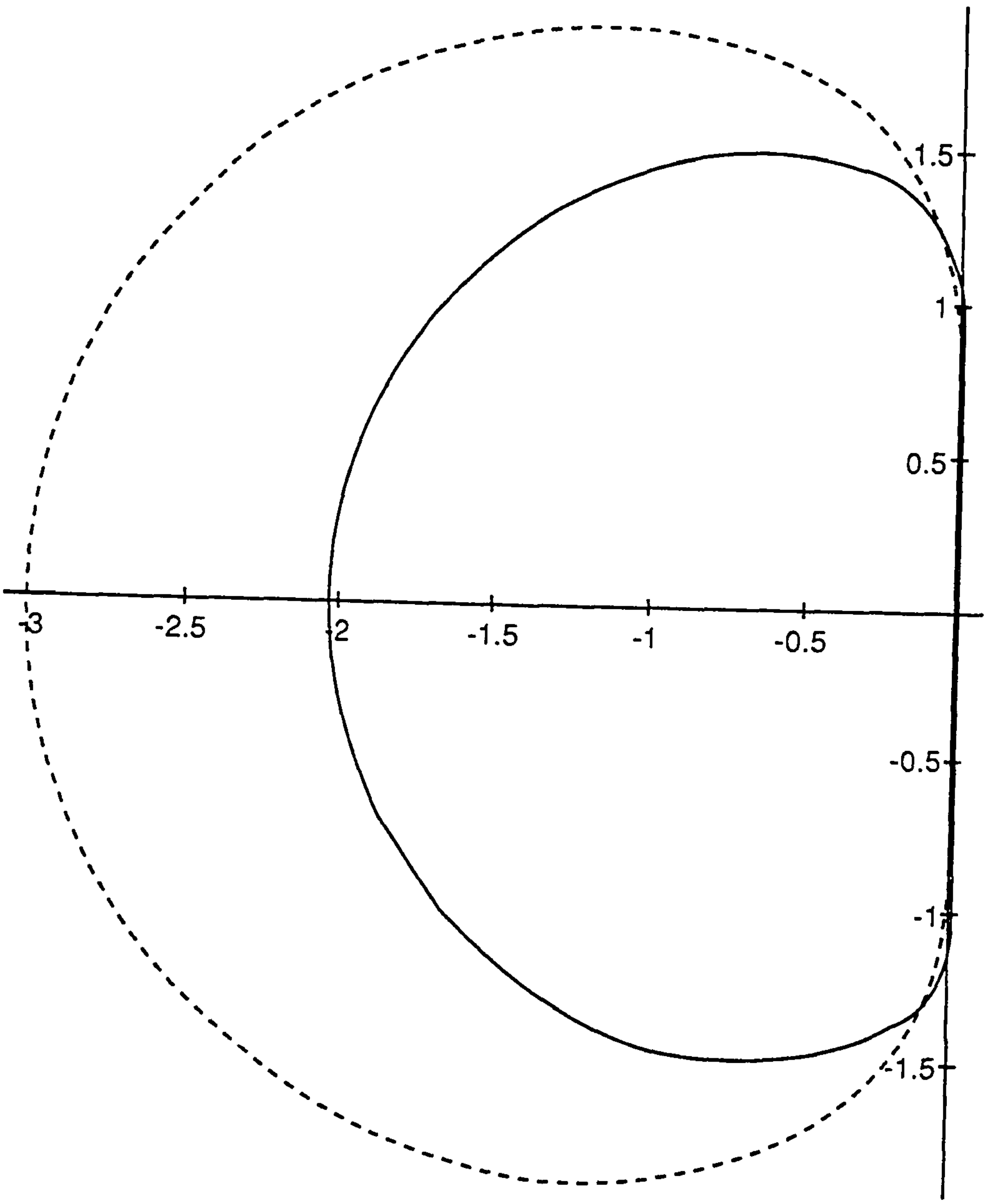

Figure 4.15:
Stability Regions for Adams–Moulton methods, where $h = .125$. Monomial basis $\{1, t, t^2, t^3, t^4\}$ is shown as a dashed curve. Positive exponential basis $\{1, e^t, e^{2t}, e^{3t}, e^{4t}\}$ is shown as a solid curve. Methods are stable inside the indicated region.

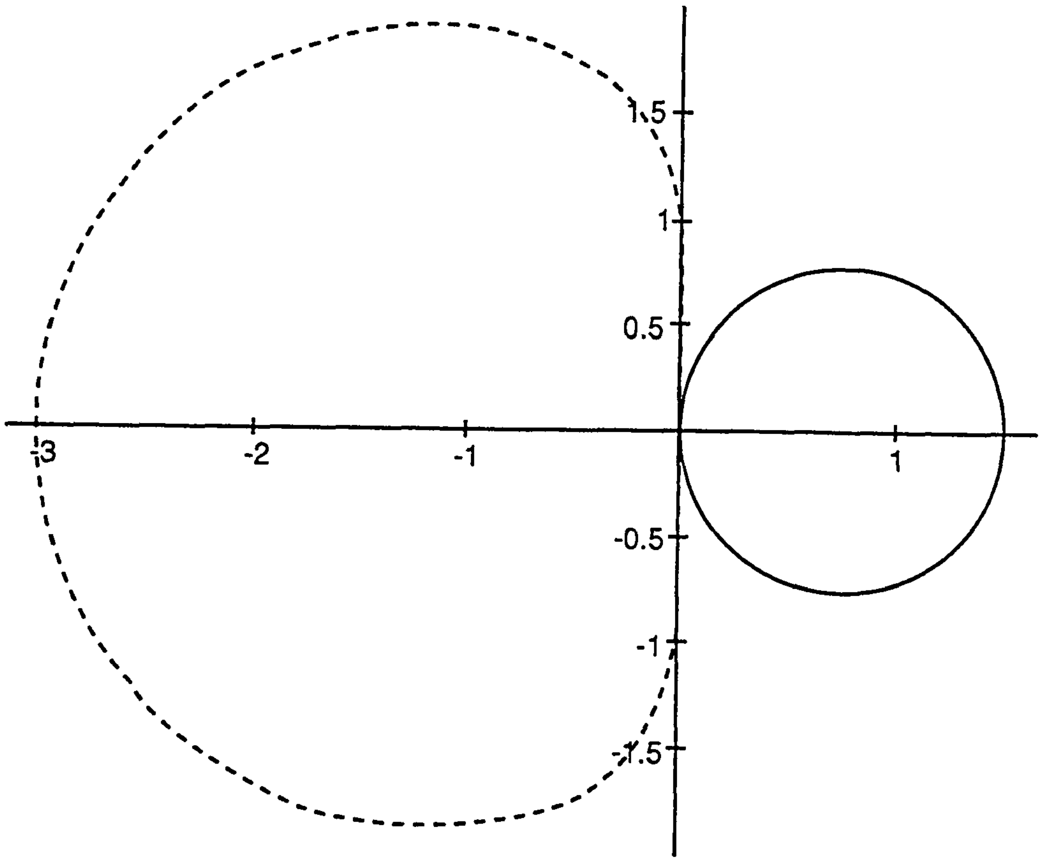

Figure 4.16:
Stability Regions for Adams–Moulton methods, where $h = 2$. Monomial basis $\{1, t, t^2, t^3, t^4\}$ is shown as a dashed curve. Negative exponential basis $\{1, e^{-t}, e^{-2t}, e^{-3t}, e^{-4t}\}$ is shown as a solid curve. Methods are stable inside the indicated region.

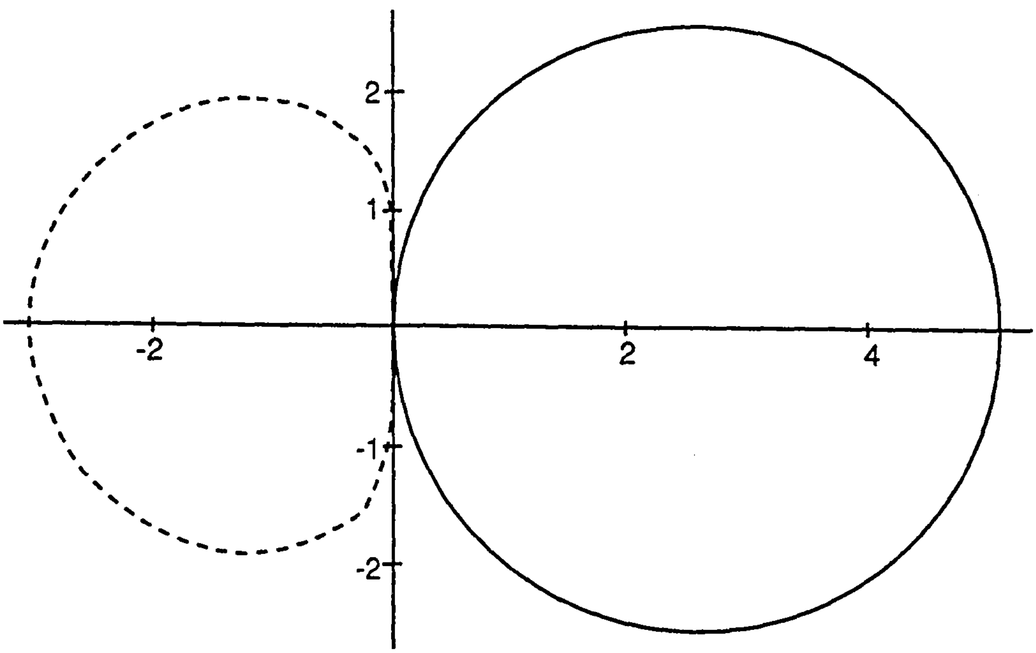

Figure 4.17:
Stability Regions for Adams–Moulton methods, where $h = 1$. Monomial basis $\{1, t, t^2, t^3, t^4\}$ is shown as a dashed curve. Negative exponential basis $\{1, e^{-t}, e^{-2t}, e^{-3t}, e^{-4t}\}$ is shown as a solid curve. Methods are stable inside the indicated region.

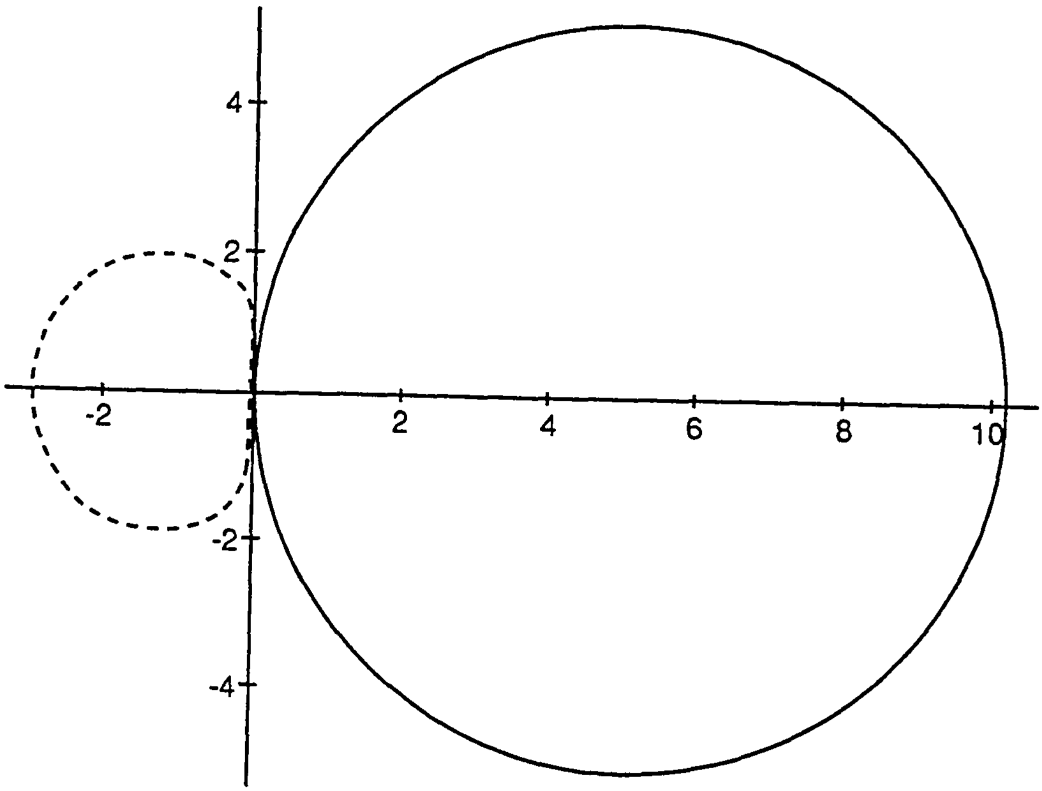

Figure 4.18:
Stability Regions for Adams–Moulton methods, where $h = .75$. Monomial basis $\{1, t, t^2, t^3, t^4\}$ is shown as a dashed curve. Negative exponential basis $\{1, e^{-t}, e^{-2t}, e^{-3t}, e^{-4t}\}$ is shown as a solid curve. Methods are stable inside the indicated region.

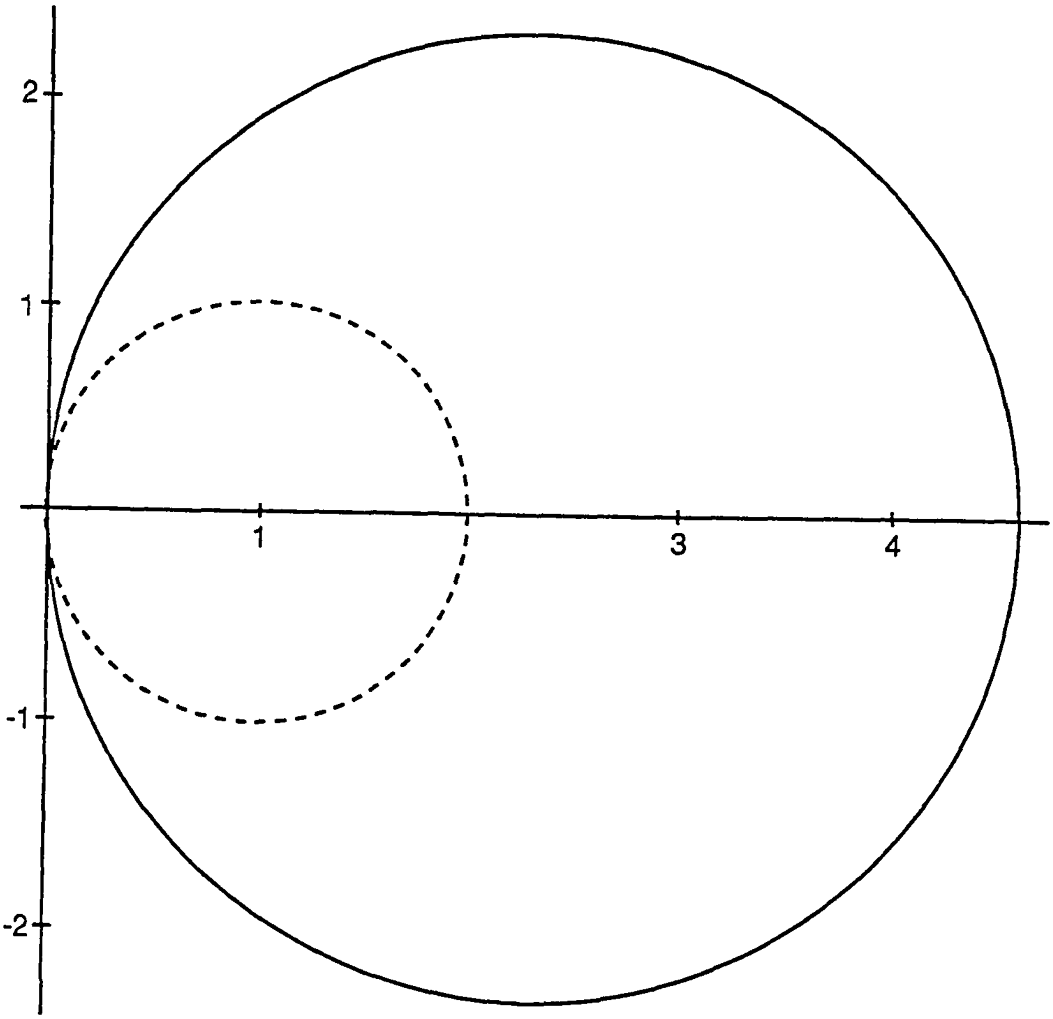

Figure 4.19:
Stability Regions for BDF methods, where $h = 2$. Monomial basis $\{1, t\}$ is shown as a dashed curve. Positive exponential basis $\{1, e^t\}$ is shown as a solid curve. Methods are stable outside the indicated region.

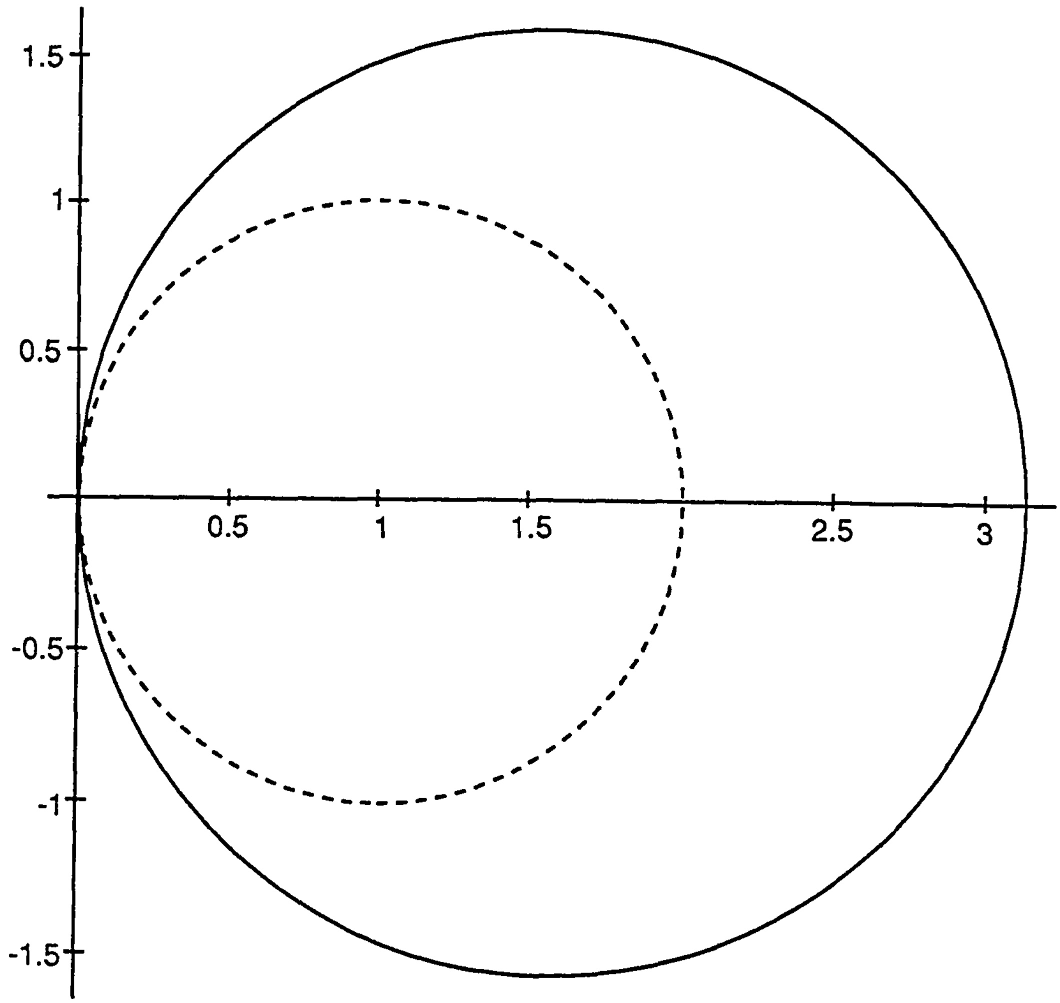

Figure 4.20:
Stability Regions for BDF methods, where $h = 1$. Monomial basis $\{1, t\}$ is shown as a dashed curve. Positive exponential basis $\{1, e^t\}$ is shown as a solid curve. Methods are stable outside the indicated region.

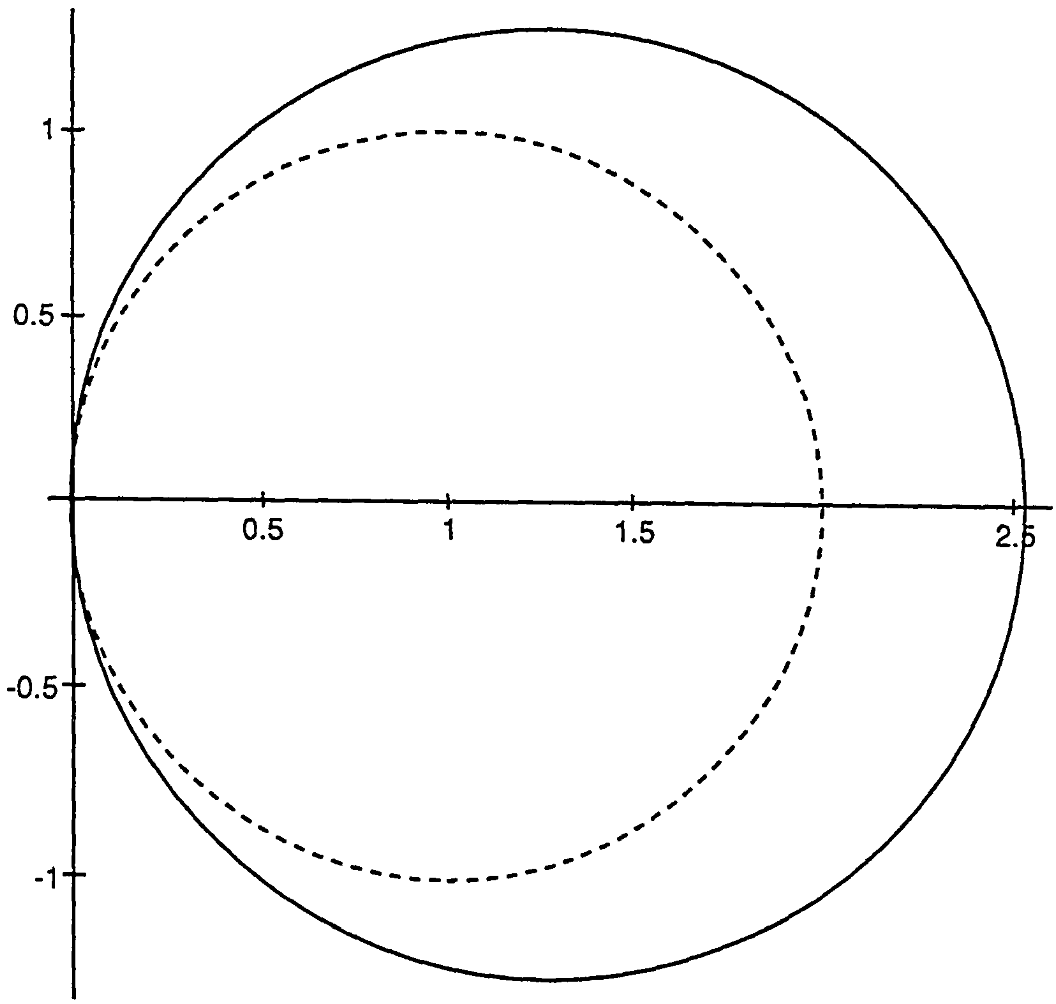

Figure 4.21:
Stability Regions for BDF methods, where $h = .5$. Monomial basis $\{1, t\}$ is shown as a dashed curve. Positive exponential basis $\{1, e^t\}$ is shown as a solid curve. Methods are stable outside the indicated region.

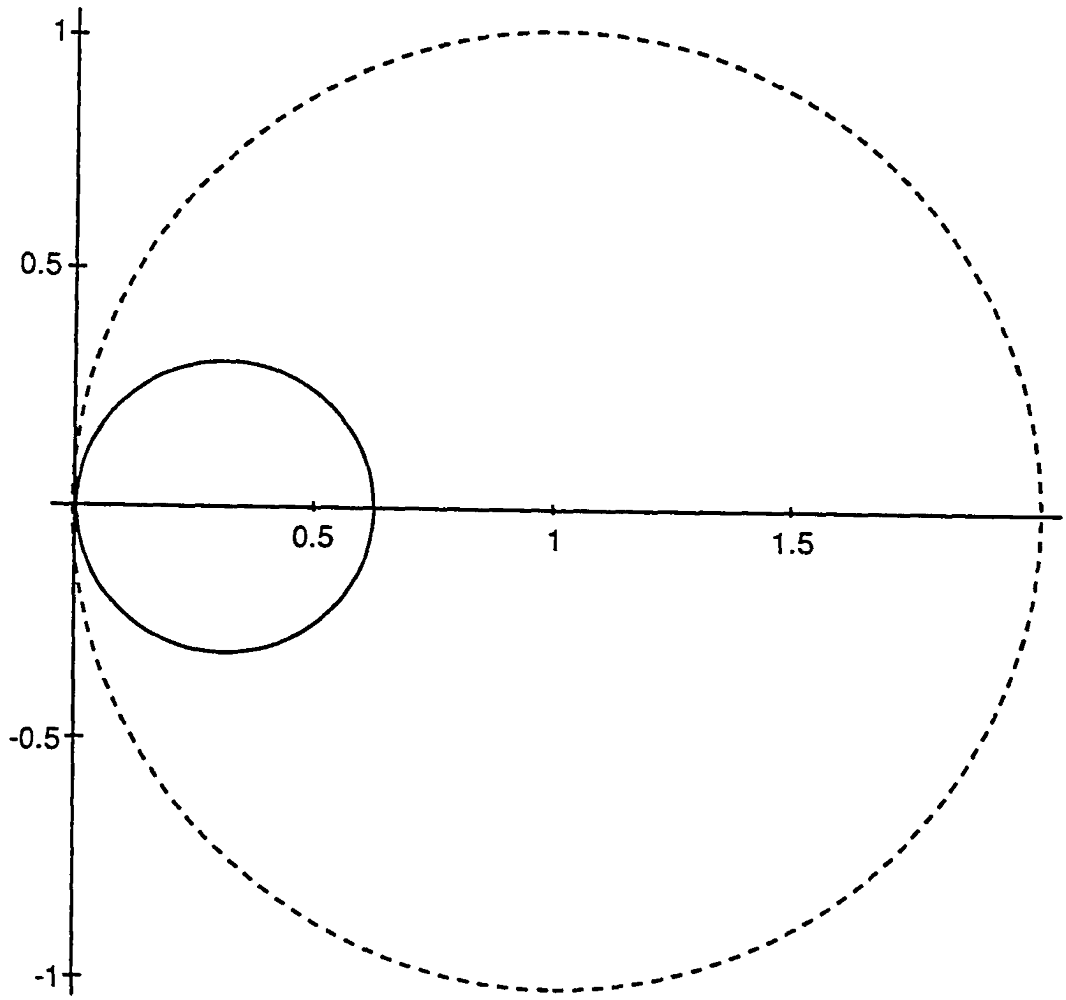

Figure 4.22:
Stability Regions for BDF methods, where $h = 2$. Monomial basis $\{1, t\}$ is shown as a dashed curve. Negative exponential basis $\{1, e^{-t}\}$ is shown as a solid curve. Methods are stable outside the indicated region.

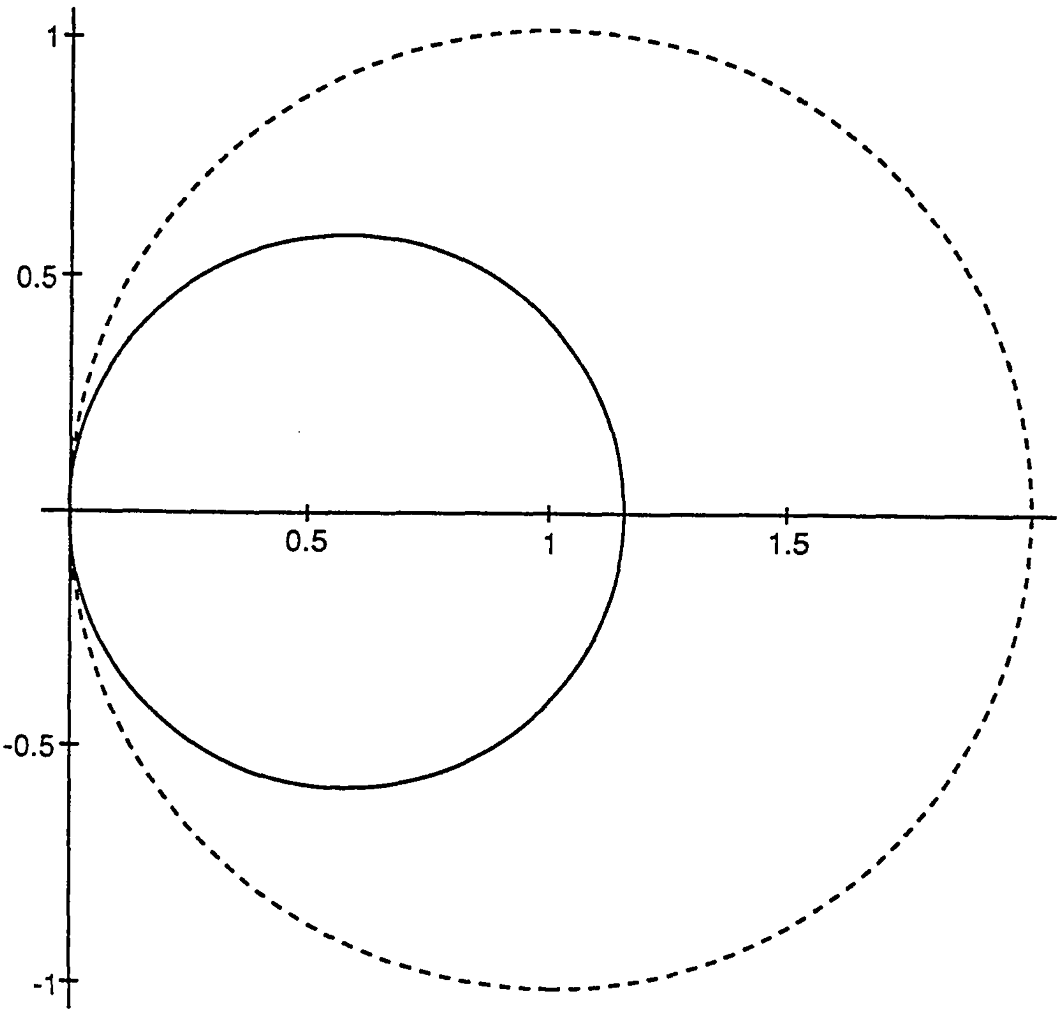

Figure 4.23:
Stability Regions for BDF methods, where $h = 1$. Monomial basis $\{1, t\}$ is shown as a dashed curve. Negative exponential basis $\{1, e^{-t}\}$ is shown as a solid curve. Methods are stable outside the indicated region.

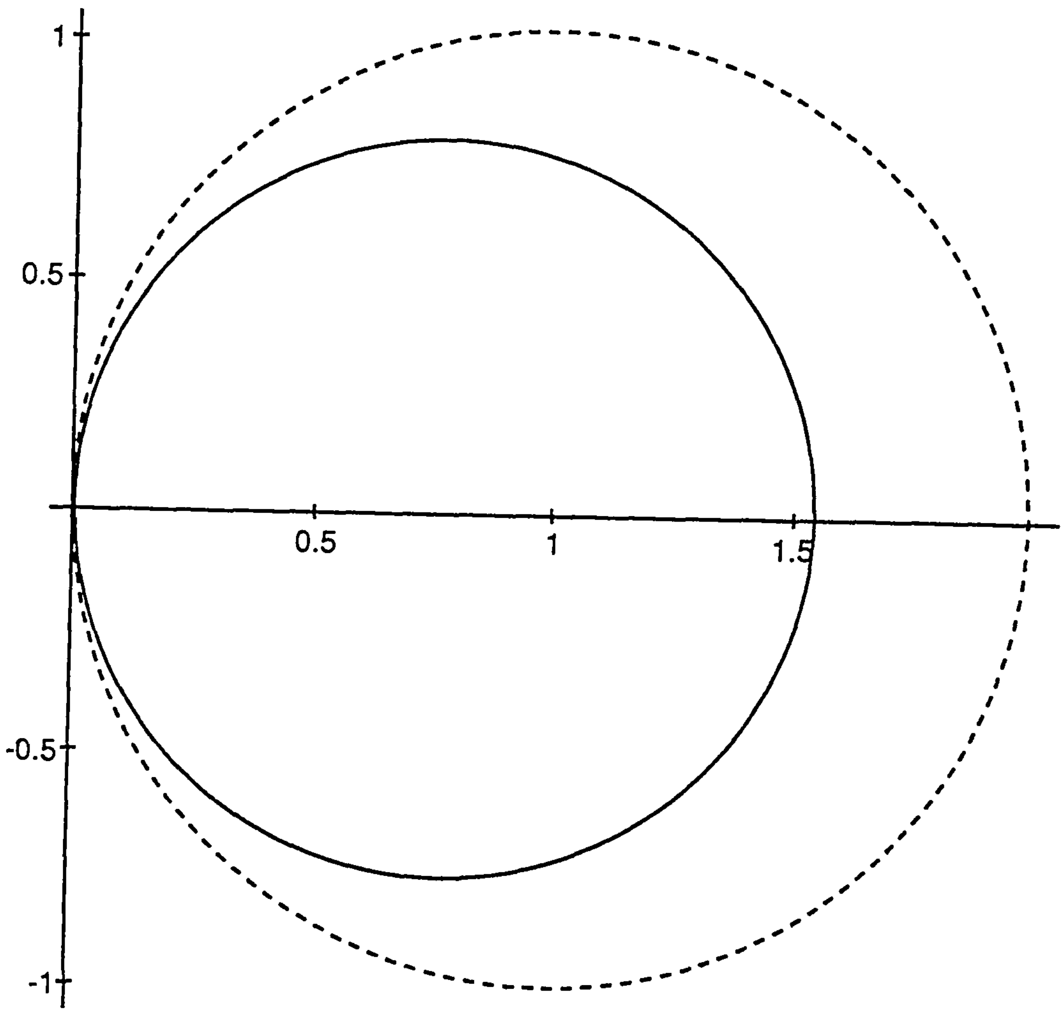

Figure 4.24:
Stability Regions for BDF methods, where $h = .5$. Monomial basis $\{1, t\}$ is shown as a dashed curve. Negative exponential basis $\{1, e^{-t}\}$ is shown as a solid curve. Methods are stable outside the indicated region.

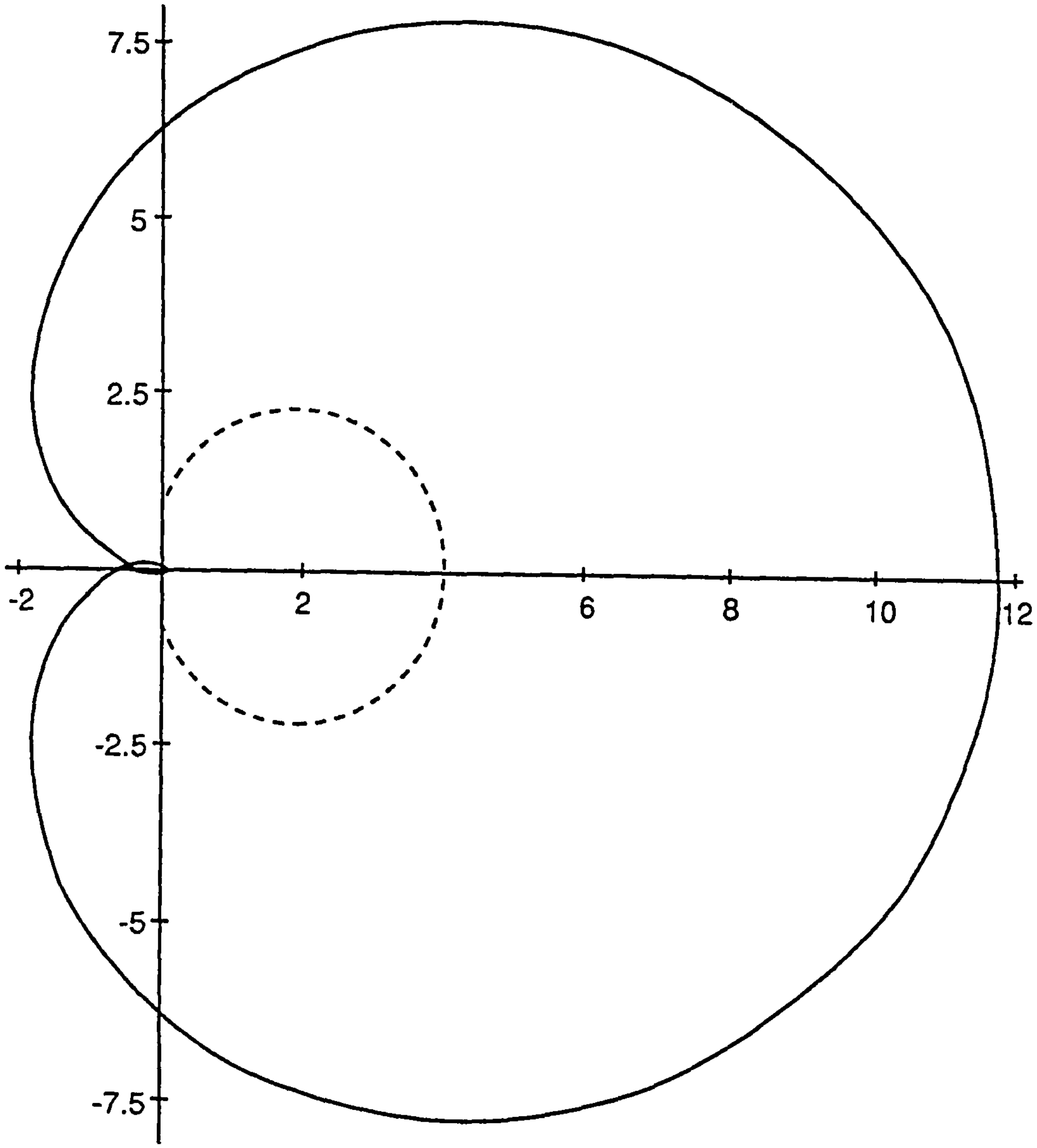

Figure 4.25:
Stability Regions for BDF methods, where $h = 1$. Monomial basis $\{1, t, t^2\}$ is shown as a dashed curve. Positive exponential basis $\{1, e^t, e^{2t}\}$ is shown as a solid curve. Methods are stable outside the indicated region.

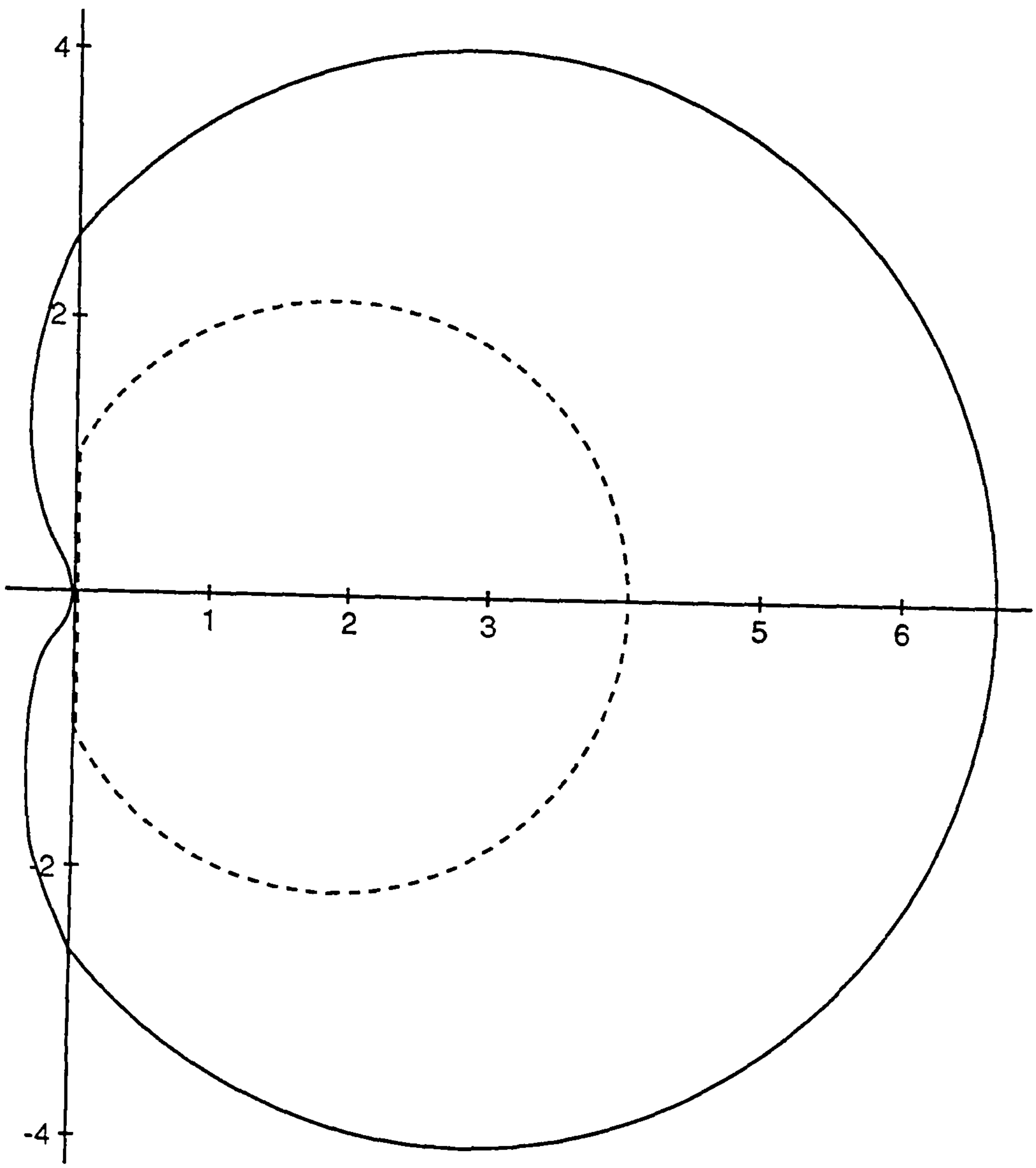

Figure 4.26:
Stability Regions for BDF methods, where $h = .5$. Monomial basis $\{1, t, t^2\}$ is shown as a dashed curve. Positive exponential basis $\{1, e^t, e^{2t}\}$ is shown as a solid curve. Methods are stable outside the indicated region.

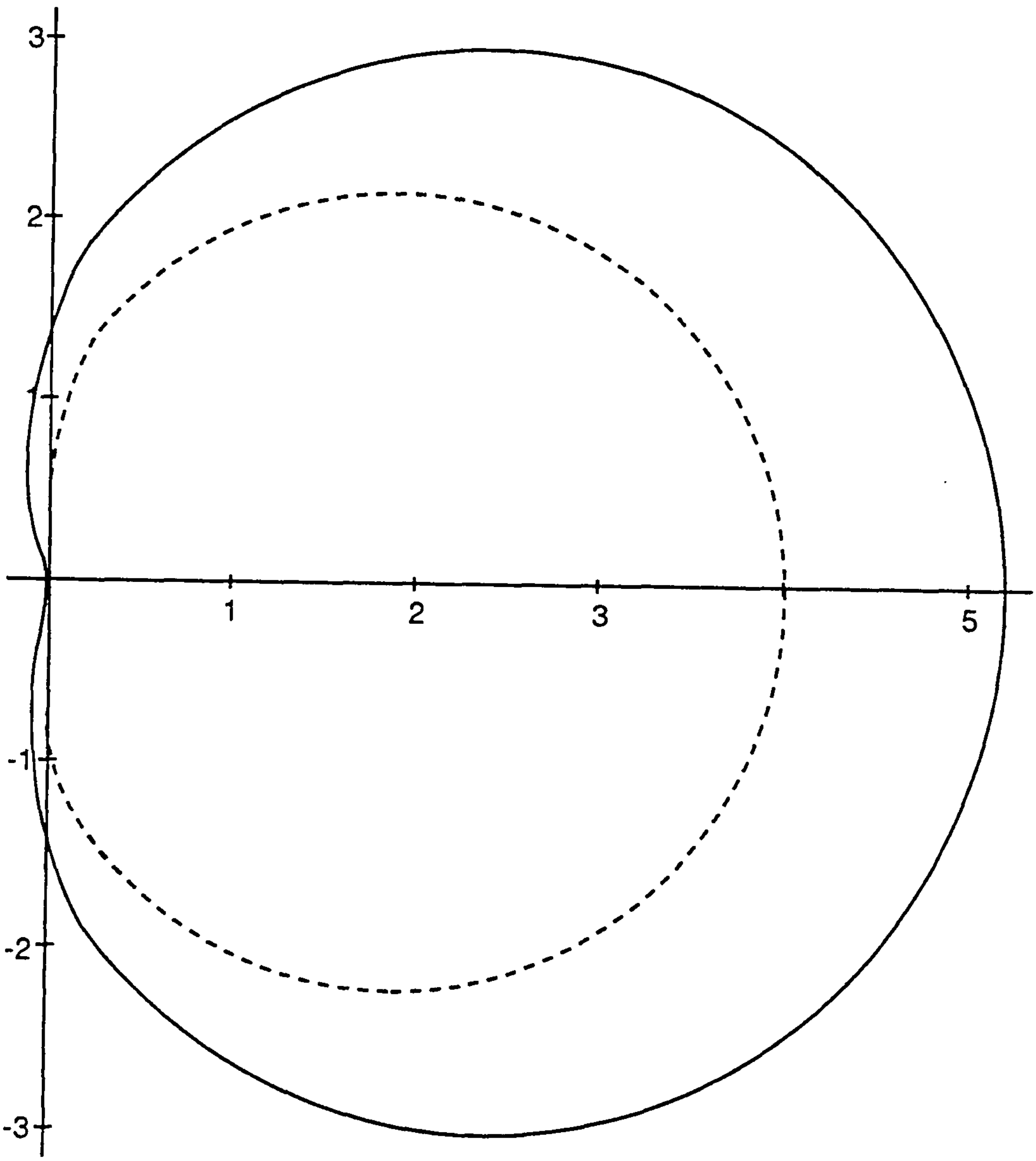

Figure 4.27:
Stability Regions for BDF methods, where $h = .25$. Monomial basis $\{1, t, t^2\}$ is shown as a dashed curve. Positive exponential basis $\{1, e^t, e^{2t}\}$ is shown as a solid curve. Methods are stable outside the indicated region.

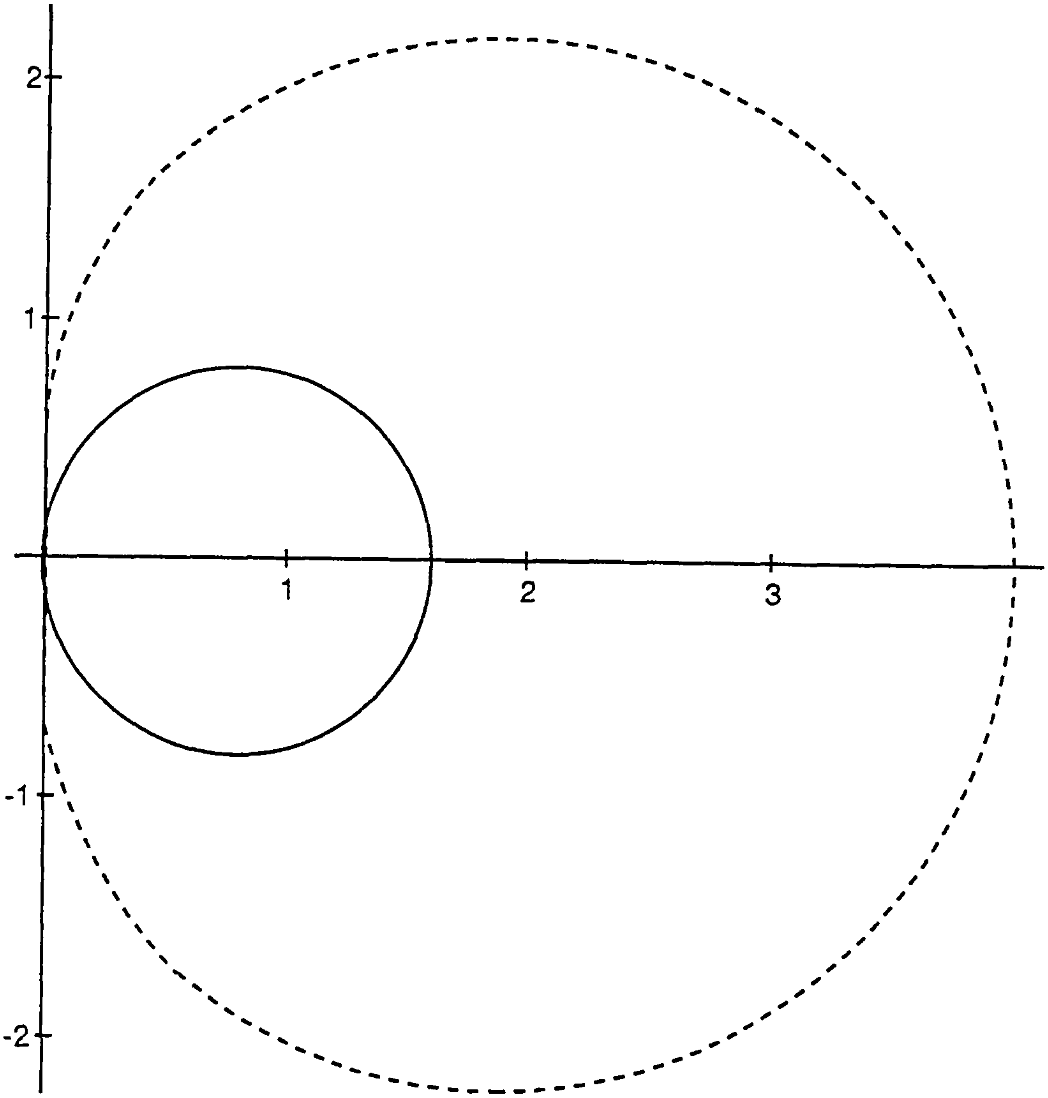

Figure 4.28:
Stability Regions for BDF methods, where $h = 1$. Monomial basis $\{1, t, t^2\}$ is shown as a dashed curve. Negative exponential basis $\{1, e^{-t}, e^{-2t}\}$ is shown as a solid curve. Methods are stable outside the indicated region.

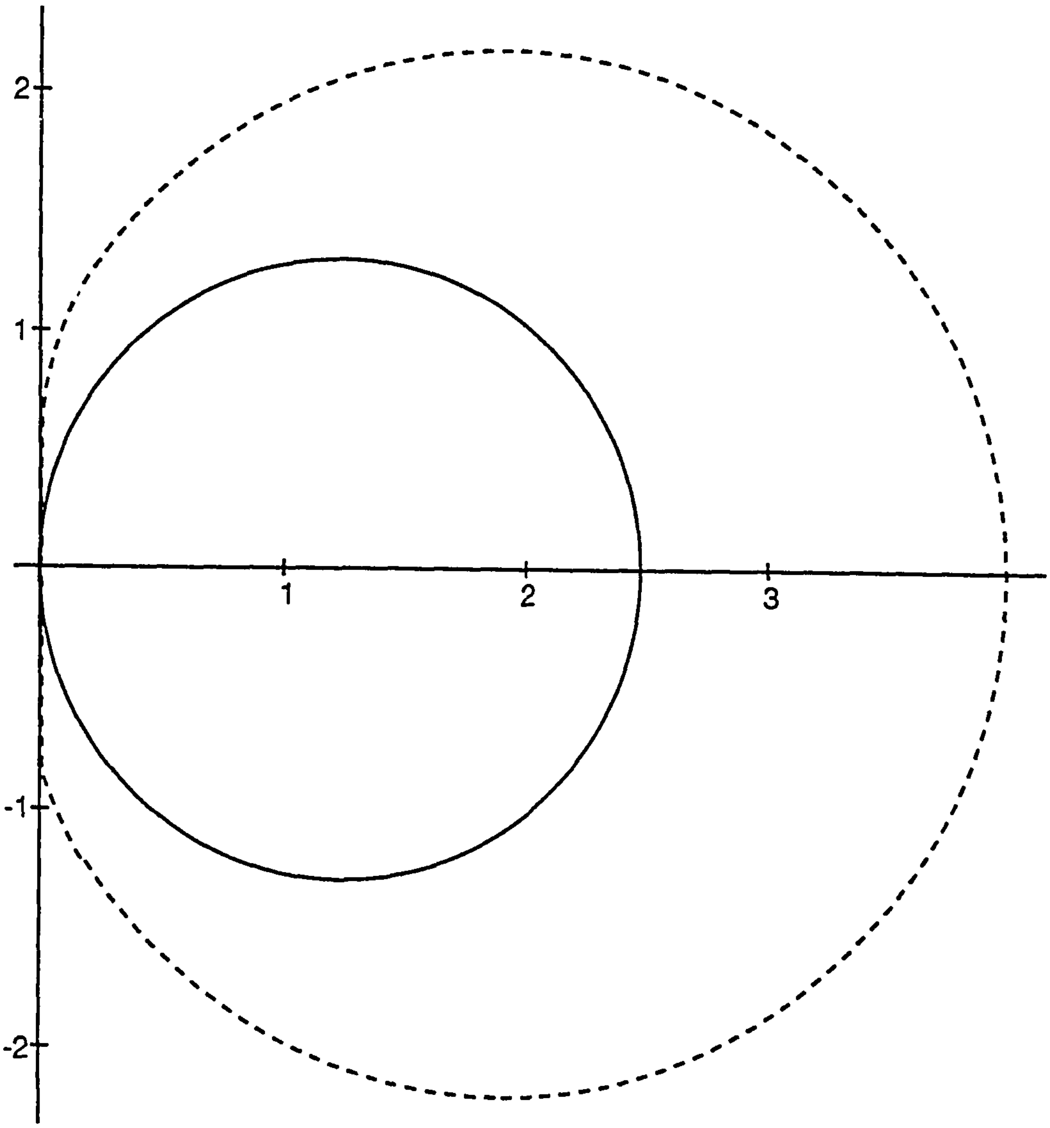

Figure 4.29:
Stability Regions for BDF methods, where $h = .5$. Monomial basis $\{1, t, t^2\}$ is shown as a dashed curve. Negative exponential basis $\{1, e^{-t}, e^{-2t}\}$ is shown as a solid curve. Methods are stable outside the indicated region.

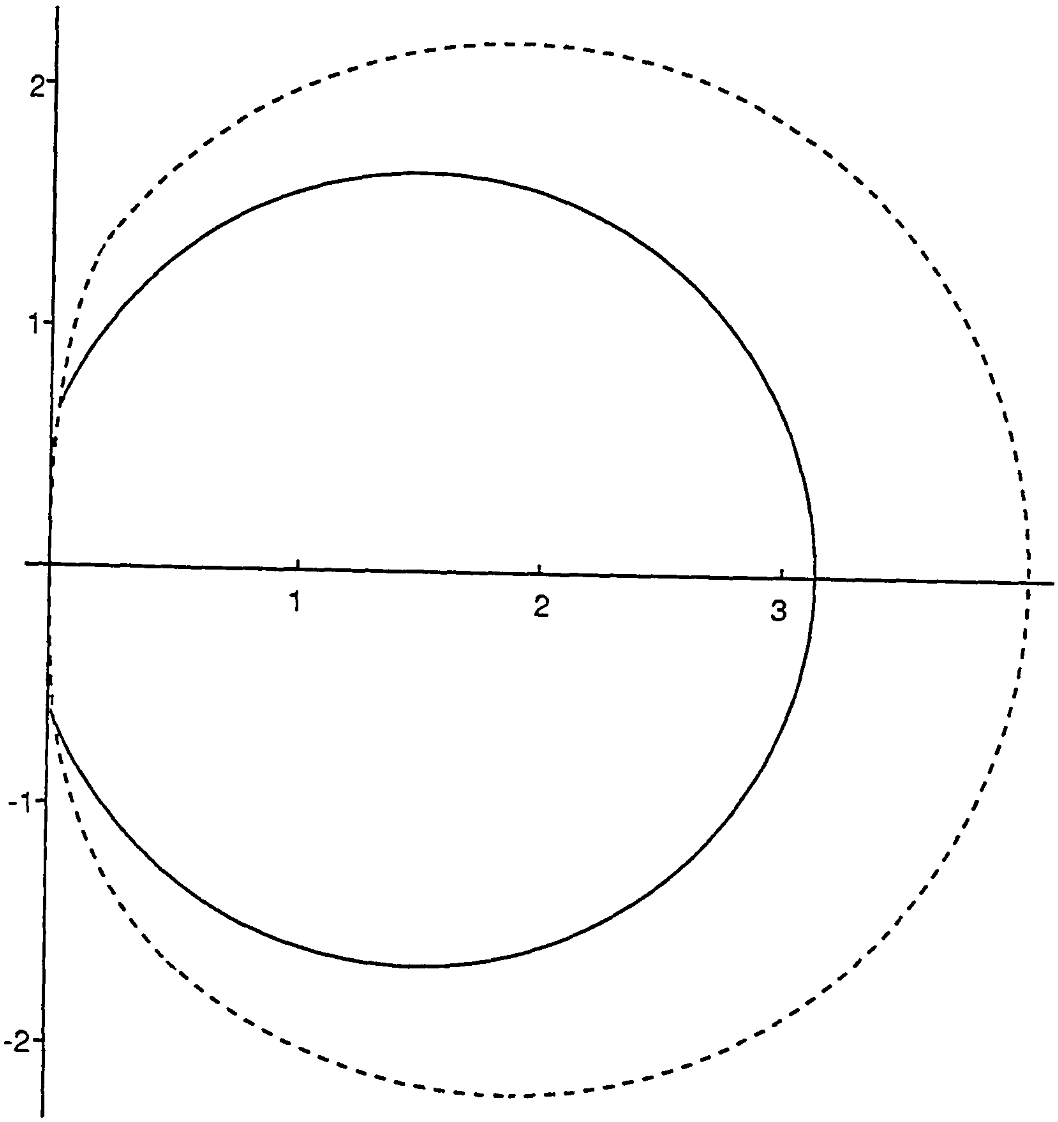

Figure 4.30:
Stability Regions for BDF methods, where $h = .25$. Monomial basis $\{1, t, t^2\}$ is shown as a dashed curve. Negative exponential basis $\{1, e^{-t}, e^{-2t}\}$ is shown as a solid curve. Methods are stable outside the indicated region.

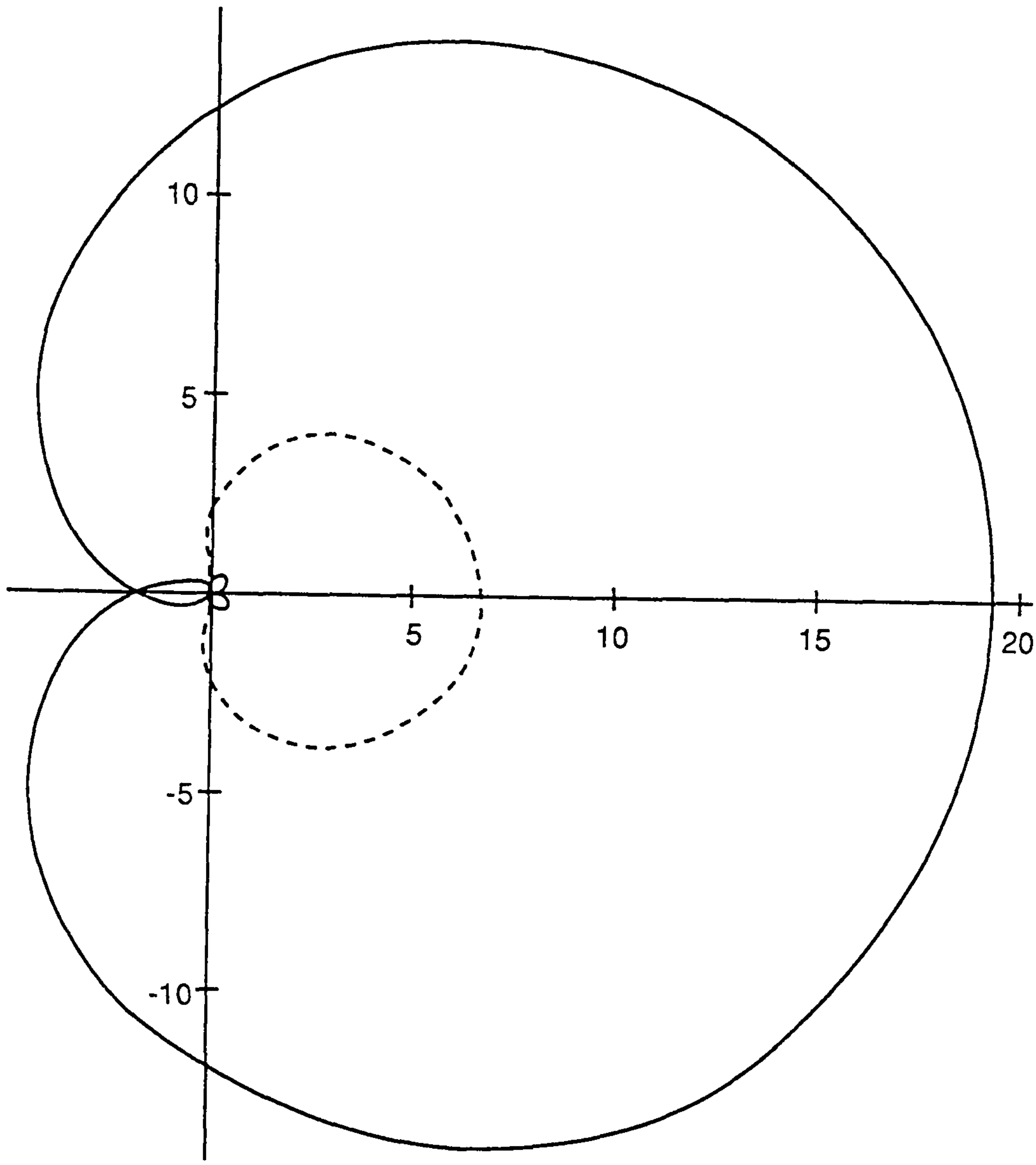

Figure 4.31:
Stability Regions for BDF methods, where $h = .5$. Monomial basis $\{1, t, t^2, t^3\}$ is shown as a dashed curve. Positive exponential basis $\{1, e^t, e^{2t}, e^{3t}\}$ is shown as a solid curve. Methods are stable outside the indicated region.

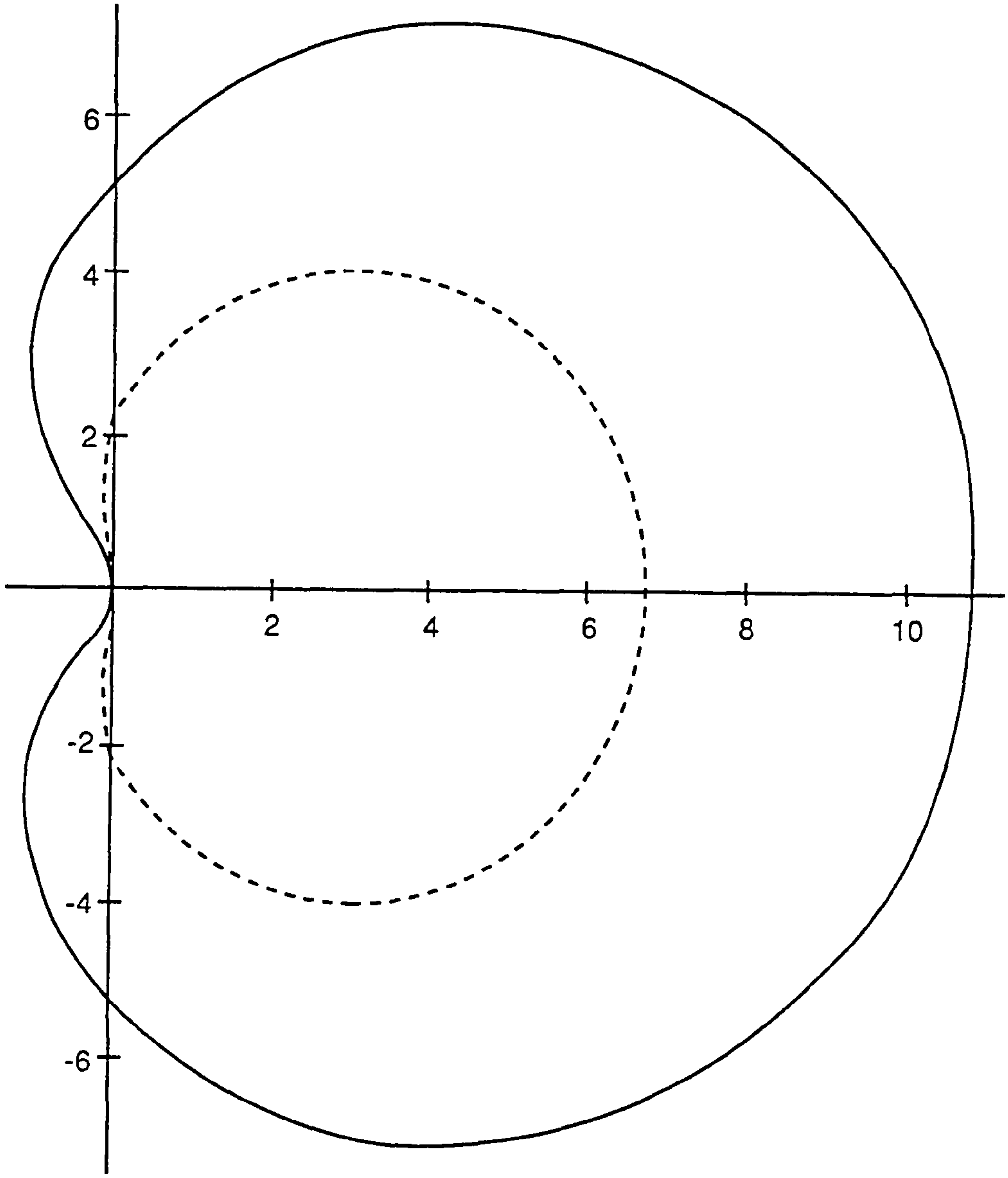

Figure 4.32:
Stability Regions for BDF methods, where $h = .25$. Monomial basis $\{1, t, t^2, t^3\}$ is shown as a dashed curve. Positive exponential basis $\{1, e^t, e^{2t}, e^{3t}\}$ is shown as a solid curve. Methods are stable outside the indicated region.

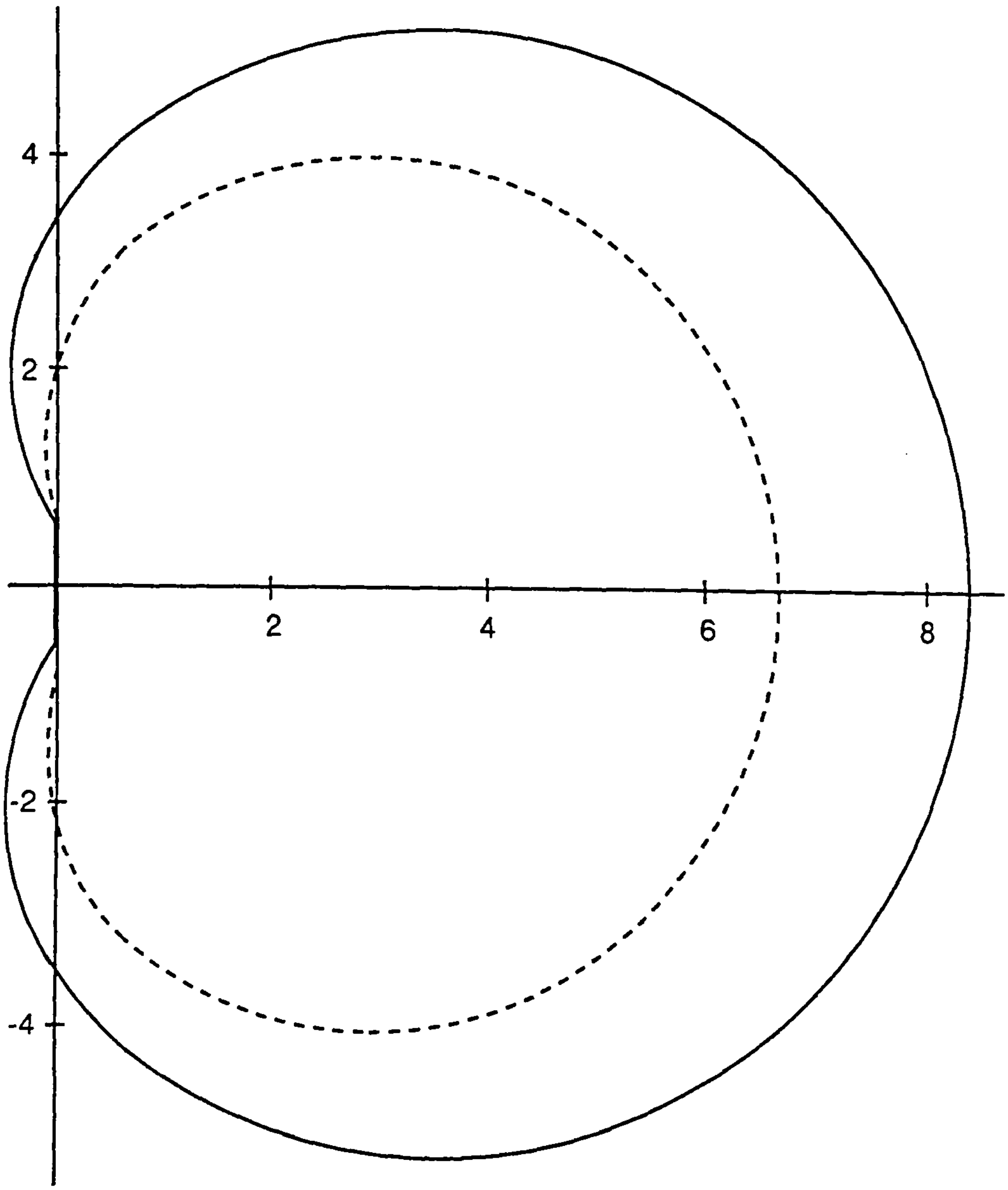

Figure 4.33:
Stability Regions for BDF methods, where $h = .125$. Monomial basis $\{1, t, t^2, t^3\}$ is shown as a dashed curve. Positive exponential basis $\{1, e^t, e^{2t}, e^{3t}\}$ is shown as a solid curve. Methods are stable outside the indicated region.

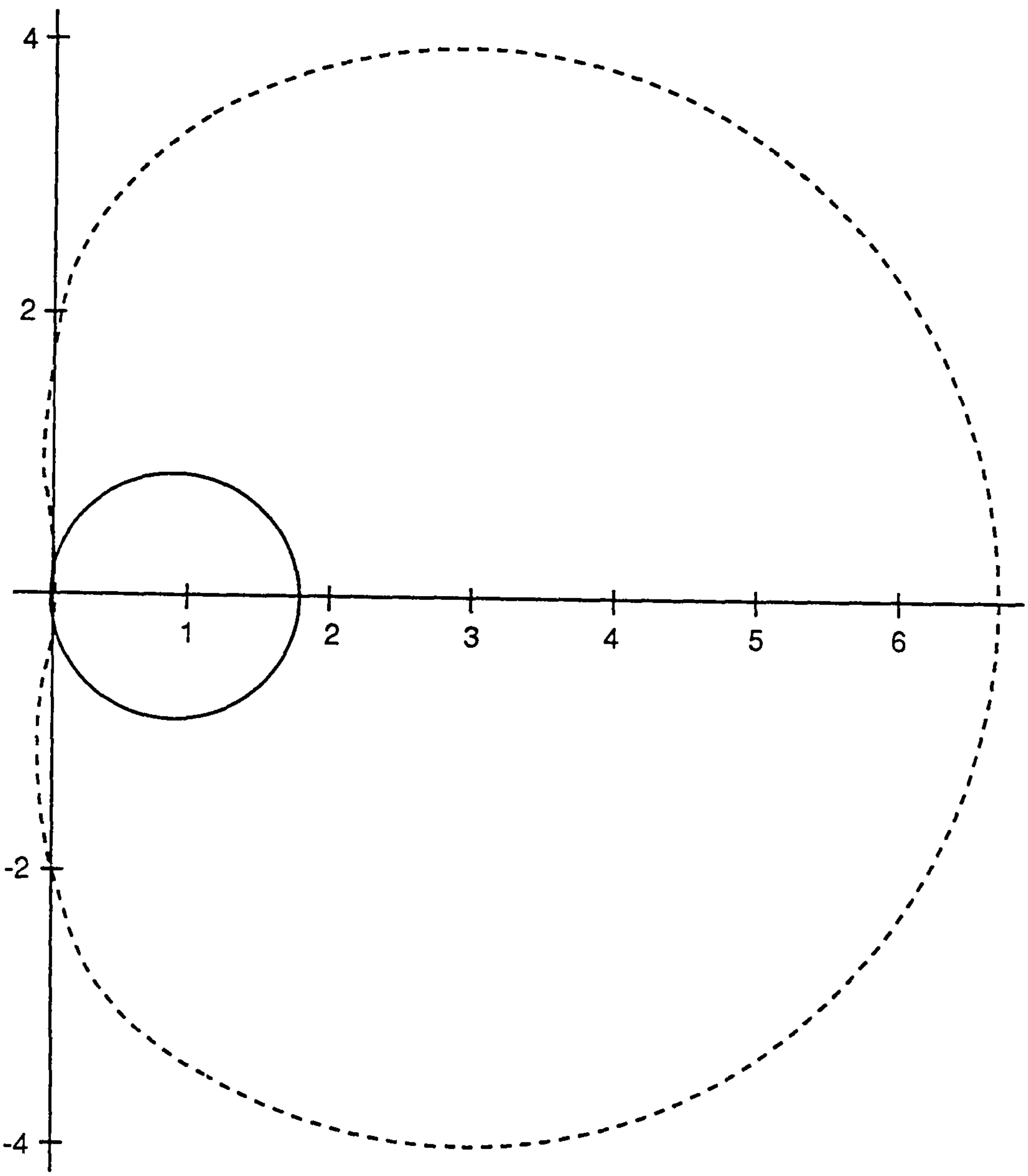

Figure 4.34:
Stability Regions for BDF methods, where $h = 1$. Monomial basis $\{1, t, t^2, t^3\}$ is shown as a dashed curve. Negative exponential basis $\{1, e^{-t}, e^{-2t}, e^{-3t}\}$ is shown as a solid curve. Methods are stable outside the indicated region.

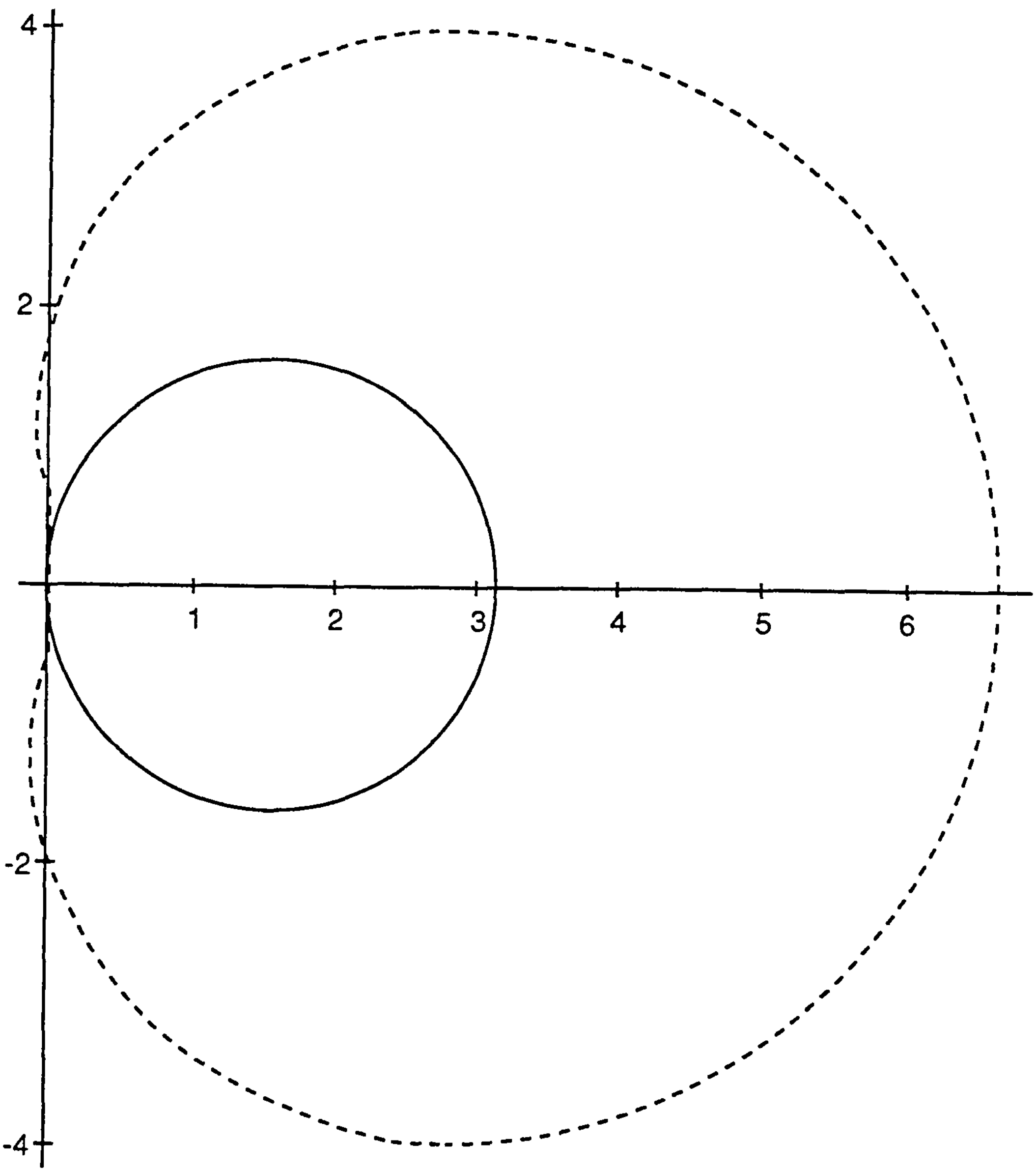

Figure 4.35:
Stability Regions for BDF methods, where $h = .5$. Monomial basis $\{1, t, t^2, t^3\}$ is shown as a dashed curve. Negative exponential basis $\{1, e^{-t}, e^{-2t}, e^{-3t}\}$ is shown as a solid curve. Methods are stable outside the indicated region.

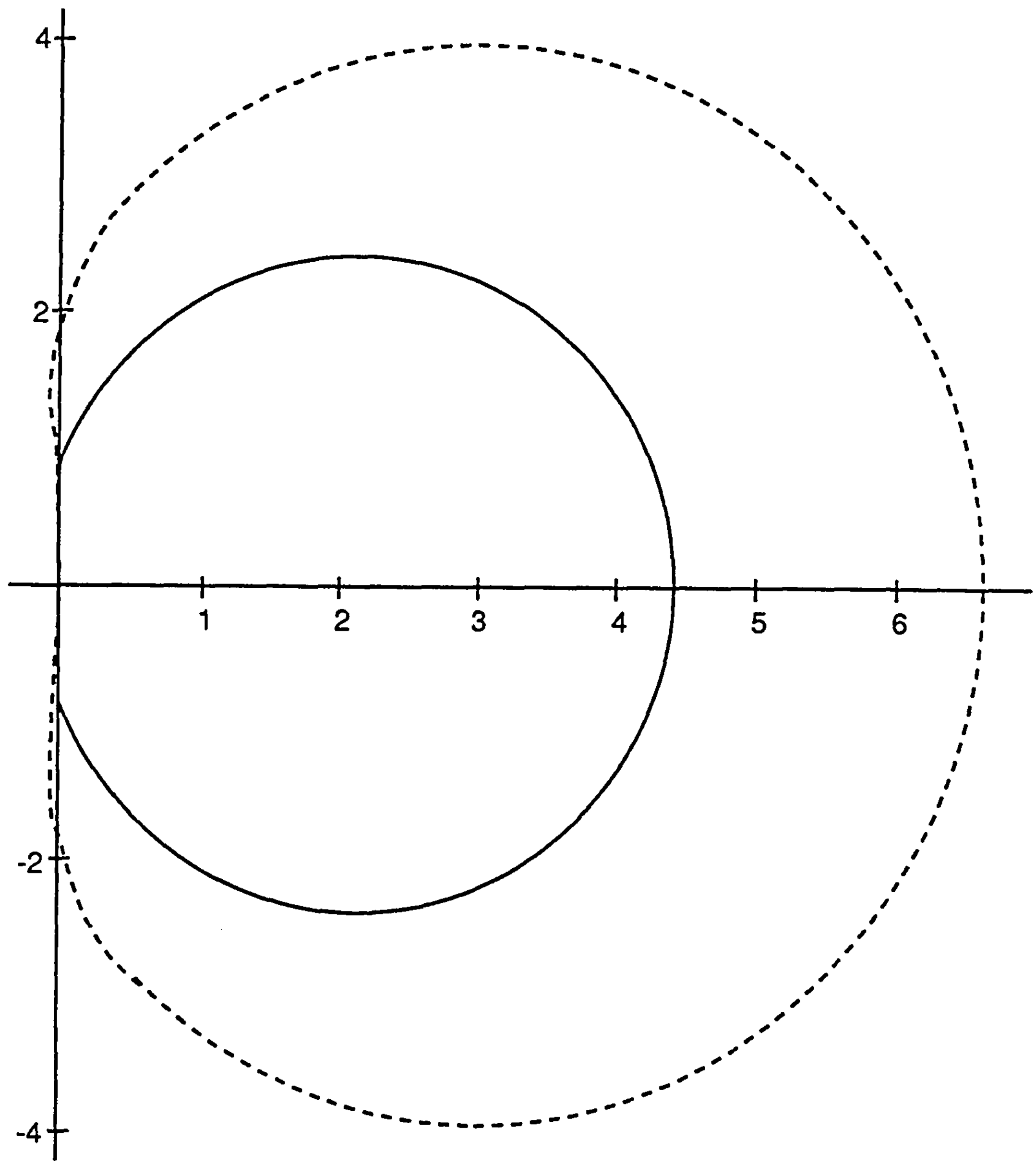

Figure 4.36:
Stability Regions for BDF methods, where $h = .25$. Monomial basis $\{1, t, t^2, t^3\}$ is shown as a dashed curve. Negative exponential basis $\{1, e^{-t}, e^{-2t}, e^{-3t}\}$ is shown as a solid curve. Methods are stable outside the indicated region.

Chapter 5

Application

5.1 Numerical Experiments

We now apply some of the methods generated in Chapter Two to several test problems. In order to conserve space, only four test problems along with results are given. These problems, however, possess behavior readily found in ordinary differential equations that model physical processes. The exact solutions to the first three test problems are known and hence direct comparisons can be made between the numerical and true solutions. Since the exact solution to the fourth problem is unknown, a baseline solution was generated by LSODA and Runge–Kutta–Fehlberg. The procedure used to solve these test problems is now discussed.

Recall the k step Adams–Moulton method

$$\vec{y}_n = \vec{y}_{n-1} + \sum_{j=0}^{k} \beta_j \vec{y}\,'_{n-j}$$

and the k step BDF method

$$\vec{y}_n = \sum_{j=1}^{k} \alpha_j \vec{y}_{n-j} + \beta_0 \vec{y}\,'_n$$

here written in vector form where

$$\vec{y}_{n-j} = \begin{pmatrix} y_{n-j}(1) \\ y_{n-j}(2) \\ \vdots \\ y_{n-j}(neq) \end{pmatrix},$$

and where neq is the number of equations in the system of differential equations being solved. Notice, both methods possess the implicit term

$$\beta_0 \vec{y}\,'_n = \vec{f}(\vec{y}_n),$$

and therefore in general a nonlinear difference equation in $\vec{y}_n$ must be solved.

The usual procedure for doing this is to construct a first prediction $\vec{y}_{n,(1)}$ to $\vec{y}_n$. A simple functional iterative procedure is then formed by

$$\vec{y}_{n,(m+1)} = \vec{y}_{n-1} + \sum_{j=1}^{k} \beta_j \vec{y}\,'_{n-j} + \beta_0 \vec{f}(\vec{y}_{n,(m)}) \qquad m = 1, \ldots, M$$

in the case of the Adams–Moulton methods and

$$\vec{y}_{n,(m+1)} = \sum_{j=1}^{k} \alpha_j \vec{y}_{n-j} + \beta_0 \vec{f}(\vec{y}_{n,(m)}) \qquad m = 1, \ldots, M$$

in the case of the BDF methods. The constant M is chosen large enough so that the sequence $\{\vec{y}_{n,(m)}\}_{m=1}^{M}$ has sufficiently converged. It is well known however that this simple functional scheme fails for stiff problems [14, 15, 24, 27, 28]. Specifically, convergence of the iterative scheme will not occur unless an excessively small step size h is chosen.

To alleviate this problem, write the Adams–Moulton and BDF methods in the form

$$\vec{y}_n = \beta_0 \vec{f}(\vec{y}_n) + \vec{g},$$

where

$$\vec{g} = \begin{pmatrix} y_{n-1}(1) & y'_{n-1}(1) & \cdots & y'_{n-k}(1) \\ y_{n-1}(2) & y'_{n-1}(2) & \cdots & y'_{n-k}(2) \\ \vdots & \vdots & & \vdots \\ y_{n-1}(neq) & y'_{n-1}(neq) & \cdots & y'_{n-k}(neq) \end{pmatrix} \begin{pmatrix} 1 \\ \beta_1 \\ \vdots \\ \beta_k \end{pmatrix}$$

for the Adams–Moulton methods, and

$$\vec{g} = \begin{pmatrix} y_{n-1}(1) & \cdots & y_{n-k}(1) \\ y_{n-1}(2) & \cdots & y_{n-k}(2) \\ \vdots & & \vdots \\ y_{n-1}(neq) & \cdots & y_{n-k}(neq) \end{pmatrix} \begin{pmatrix} 1 \\ \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_k \end{pmatrix}$$

for the BDF methods. Now instead of using functional iteration, use Newton iteration given by

$$\underset{\sim}{P}_{(m)} \vec{y}_{n,(m+1)} = \underset{\sim}{P}_{(m)} \vec{y}_{n,(m)} + \left(\beta_0 \vec{f}(\vec{y}_{n,(m)}) - \vec{y}_{n,(m)} + \vec{g} \right),$$

where

$$\underset{\sim}{P}_{(m)} = \underset{\sim}{I} - \beta_0 \underset{\sim}{J}(\vec{y}_{n,(m)})$$

and $\underset{\sim}{J}(\vec{y}_{n,(m)})$ is the Jacobian matrix

$$\underset{\sim}{J}(\vec{y}_{n,(m)}) = \frac{\partial \vec{f}(\vec{y}_{n,(m)})}{\partial \vec{y}}.$$

This linear system in $\vec{y}_{n,(m+1)}$ is then solved by an LU decomposition. In the special case where $\underset{\sim}{P}_{(m)}$ is numerically or analytically singular or very nearly singular, then a singular value decomposition

$$\underset{\sim}{P}_{(m)} = \underset{\sim}{U}_{(m)} \underset{\sim}{D}_{(m)} \underset{\sim}{V}^T_{(m)}$$

is used [30].

We now discuss how the predictor $\vec{y}_{n,(1)}$ is found. The usual procedure is to use an explicit method such as the Adams–Bashforth

$$\vec{y}_n = \vec{y}_{n-1} + \sum_{j=1}^{k} \beta_j \vec{y}^{\,\prime}_{n-j}$$

as a predictor for the implicit Adams–Moulton methods. In the case of the BDF methods, a linear extrapolation of the known data points $\vec{y}_{n-1}, \ldots, \vec{y}_{n-k}$ is used [15]. Alternatively, explicit Runge-Kutta methods of various orders could also be employed as predictors.

Recall in the previous error and stability analysis, no mention was given to the usage of a method in the more realistic predictor corrector setting. The reason for this omission is that the subtle interplay of error propagation with

truncation error in such a setting is to be considered in future research. As such, an ad hoc procedure was used.

As a first attempt to find a suitable predictor, various explicit methods were tried. Since the test problems to be described below have stiff behavior, it is readily apparent that either the predictor or corrector will limit the step size to be taken. For all the classical predictors used, the limiting factor was the poor stability characteristics of the explicit methods themselves. In fact upon closer examination, it was found that the solutions generated by the predictors were in such great error as to be useless unless the step size was reduced by more than a hundred fold that of the step size needed by the corrector. This resulted in a thrashing of errors.

The problem was eliminated by choosing

$$\vec{y}_{n,(1)} = \vec{y}_{n-1},$$

that is letting the prediction be the previous value. Furthermore, the need for a predictor method was also eliminated altogether. The justification for this being, if the actual solution is slowly varying on the interval $[t_{n-k}, t_n]$, then $\vec{y}_{n-1}$ is a better approximation to $\vec{y}_n$ when compared to an erratic predictor $\vec{y}_{n,(1)}$. It must be stated that this was a pragmatic solution and as such has no real theoretical backing. Future effort must be made to obtain suitable predictor methods.

In the following test problems, the error $e(t_n)$ is defined as

$$e(t_n) = \frac{1}{neq} \| \vec{y}(t_n) - \vec{y}_n \|_2,$$

where $\| \cdot \|_2$ denotes the l_2 norm.

5.1.1 Test Problem #1

Consider the stiff, constant coefficient linear problem [27]

$$\begin{aligned} x'(t) &= -.1x(t) - 49.9y(t) \\ y'(t) &= -50y(t) \\ z'(t) &= 70y(t) - 120z(t) \end{aligned}$$

with

$$x(0) = 2, \quad y(0) = 1, \quad z(0) = 2.$$

The exact solution is given by

$$\begin{aligned} x(t) &= e^{-.1t} + e^{-50t} \\ y(t) &= e^{-50t} \\ z(t) &= e^{-50t} + e^{-120t} \end{aligned}$$

Results of the integration using various basis functions, methods and stepsizes are given in Tables 5.1 thru 5.12.

Test Problem #1			
t	$\{1, t, t^2\}$	$\{1, e^t, e^{2t}\}$	$\{1, e^{-t}, e^{-2t}\}$
.2	4.61662(10^{-2})	7.07580(10^{-2})	3.12377(10^{-2})
.4	3.28230(10^{-2})	5.61641(10^{-2})	2.99512(10^{-2})
.6	3.96074(10^{-2})	5.38281(10^{-2})	4.09413(10^{-2})
.8	4.52355(10^{-2})	7.02083(10^{-2})	4.20559(10^{-2})
1.0	3.89346(10^{-2})	5.18704(10^{-2})	4.01609(10^{-2})
1.2	4.11833(10^{-2})	6.36913(10^{-2})	3.99817(10^{-2})
1.4	3.79677(10^{-2})	5.14082(10^{-2})	3.89215(10^{-2})
1.6	3.85894(10^{-2})	5.96528(10^{-2})	3.83688(10^{-2})
1.8	3.67585(10^{-2})	5.16630(10^{-2})	3.75667(10^{-2})
2.0	3.66459(10^{-2})	5.73444(10^{-2})	3.69214(10^{-2})

Table 5.1: Error for A-M with $h = .2$ and $k = 1$

Test Problem #1			
t	$\{1, t, t^2\}$	$\{1, e^t, e^{2t}\}$	$\{1, e^{-t}, e^{-2t}\}$
.2	$3.12441(10^{-2})$	$4.09830(10^{-2})$	$2.44107(10^{-2})$
.4	$1.70668(10^{-2})$	$1.94956(10^{-2})$	$1.64947(10^{-2})$
.6	$2.14540(10^{-2})$	$2.22234(10^{-2})$	$2.13944(10^{-2})$
.8	$2.09106(10^{-2})$	$2.12888(10^{-2})$	$2.09579(10^{-2})$
1.0	$2.04688(10^{-2})$	$2.07019(10^{-2})$	$2.05529(10^{-2})$
1.2	$2.00568(10^{-2})$	$2.02416(10^{-2})$	$2.01592(10^{-2})$
1.4	$1.96581(10^{-2})$	$1.98326(10^{-2})$	$1.97737(10^{-2})$
1.6	$1.92684(10^{-2})$	$1.94477(10^{-2})$	$1.93956(10^{-2})$
1.8	$1.88869(10^{-2})$	$1.90763(10^{-2})$	$1.90247(10^{-2})$
2.0	$1.85129(10^{-2})$	$1.87144(10^{-2})$	$1.86610(10^{-2})$

Table 5.2: Error for A-M with $h = .1$ and $k = 1$

Test Problem #1			
t	$\{1, t, t^2, t^3\}$	$\{1, e^t, e^{2t}, e^{3t}\}$	$\{1, e^{-t}, e^{-2t}, e^{-3t}\}$
.2	$3.62715(10^{-2})$	$5.35331(10^{-2})$	$3.26055(10^{-2})$
.4	$4.87516(10^{-2})$	$1.19224(10^{-1})$	$4.23725(10^{-2})$
.6	$6.38104(10^{-2})$	$3.30558(10^{-1})$	$5.15301(10^{-2})$
.8	$5.63663(10^{-2})$	$5.54368(10^{-1})$	$5.07733(10^{-2})$
1.0	$7.94025(10^{-2})$	1.05528	$4.95696(10^{-2})$
1.2	$8.35819(10^{-2})$	1.92638	$4.88207(10^{-2})$
1.4	$1.25022(10^{-1})$	3.64100	$4.76958(10^{-2})$
1.6	$1.59075(10^{-1})$	6.85924	$4.69623(10^{-2})$
1.8	$2.37429(10^{-1})$	$1.30836(10^{1})$	$4.59142(10^{-2})$
2.0	$3.32027(10^{-1})$	$2.50521(10^{1})$	$4.52051(10^{-2})$

Table 5.3: Error for A-M with $h = .2$ and $k = 2$

Test Problem #1			
t	$\{1, t, t^2, t^3\}$	$\{1, e^t, e^{2t}, e^{3t}\}$	$\{1, e^{-t}, e^{-2t}, e^{-3t}\}$
.2	$1.80969(10^{-2})$	$1.96639(10^{-2})$	$1.84221(10^{-2})$
.4	$2.27977(10^{-2})$	$4.43710(10^{-2})$	$2.09888(10^{-2})$
.6	$3.41773(10^{-2})$	$1.25639(10^{-1})$	$2.69907(10^{-2})$
.8	$4.42791(10^{-2})$	$2.74391(10^{-1})$	$2.64484(10^{-2})$
1.0	$6.53437(10^{-2})$	$6.09151(10^{-1})$	$2.60471(10^{-2})$
1.2	$1.04160(10^{-1})$	1.35816	$2.58064(10^{-2})$
1.4	$1.71622(10^{-1})$	3.03230	$2.57802(10^{-2})$
1.6	$2.86217(10^{-1})$	6.77372	$2.60630(10^{-2})$
1.8	$4.79299(10^{-1})$	$1.51351(10^{1})$	$2.67998(10^{-2})$
2.0	$8.03719(10^{-1})$	$3.38217(10^{1})$	$2.81946(10^{-2})$

Table 5.4: Error for A-M with $h = .1$ and $k = 2$

Test Problem #1			
t	$\{1, t, t^2, t^3, t^4\}$	$\{1, e^t, e^{2t}, e^{3t}, e^{4t}\}$	$\{1, e^{-t}, e^{-2t}, e^{-3t}, e^{-4t}\}$
.2	$2.03787(10^{-2})$	$3.51970(10^{-2})$	$1.45367(10^{-2})$
.4	$8.81137(10^{-2})$	$2.92203(10^{-1})$	$2.97799(10^{-2})$
.6	$5.61620(10^{-1})$	3.02841	$1.07933(10^{-1})$
.8	2.51110	$1.98346(10^{1})$	$3.27226(10^{-1})$
1.0	$1.12407(10^{1})$	$1.30014(10^{2})$	1.00189
1.2	$5.03238(10^{1})$	$8.52331(10^{2})$	3.07072
1.4	$2.25300(10^{2})$	$5.58770(10^{3})$	9.41252
1.6	$1.00867(10^{3})$	$3.66319(10^{4})$	$2.88521(10^{1})$
1.8	$4.51584(10^{3})$	$2.40152(10^{5})$	$8.84400(10^{1})$
2.0	$2.02175(10^{4})$	$1.57439(10^{6})$	$2.71094(10^{2})$

Table 5.5: Error for A-M with $h = .05$ and $k = 3$

Test Problem #1			
t	$\{1, t, t^2, t^3, t^4\}$	$\{1, e^t, e^{2t}, e^{3t}, e^{4t}\}$	$\{1, e^{-t}, e^{-2t}, e^{-3t}, e^{-4t}\}$
.2	$6.61361(10^{-3})$	$8.15190(10^{-3})$	$6.00669(10^{-3})$
.4	$7.67175(10^{-3})$	$1.16074(10^{-2})$	$6.99567(10^{-3})$
.6	$8.76886(10^{-3})$	$1.70597(10^{-2})$	$8.12650(10^{-3})$
.8	$8.62129(10^{-3})$	$2.42351(10^{-2})$	$7.94590(10^{-3})$
1.0	$8.47708(10^{-3})$	$3.57731(10^{-2})$	$7.77993(10^{-3})$
1.2	$8.33618(10^{-3})$	$5.37845(10^{-2})$	$7.62211(10^{-3})$
1.4	$8.19853(10^{-3})$	$8.15240(10^{-2})$	$7.46954(10^{-3})$
1.6	$8.06406(10^{-3})$	$1.23998(10^{-1})$	$7.32092(10^{-3})$
1.8	$7.93271(10^{-3})$	$1.88873(10^{-1})$	$7.17566(10^{-3})$
2.0	$7.80444(10^{-3})$	$2.87863(10^{-1})$	$7.03345(10^{-3})$

Table 5.6: Error for A-M with $h = .025$ and $k = 3$

Test Problem #1			
t	$\{1,t\}$	$\{1,e^t\}$	$\{1,e^{-t}\}$
.2	$2.14688(10^{-2})$	$1.93859(10^{-2})$	$2.38150(10^{-2})$
.4	$3.17845(10^{-2})$	$2.79242(10^{-2})$	$3.61517(10^{-2})$
.6	$4.11098(10^{-2})$	$3.55361(10^{-2})$	$4.73915(10^{-2})$
.8	$4.01986(10^{-2})$	$3.38176(10^{-2})$	$4.73760(10^{-2})$
1.0	$3.93072(10^{-2})$	$3.21511(10^{-2})$	$4.73408(10^{-2})$
1.2	$3.84354(10^{-2})$	$3.05354(10^{-2})$	$4.72867(10^{-2})$
1.4	$3.75827(10^{-2})$	$2.89690(10^{-2})$	$4.72145(10^{-2})$
1.6	$3.67487(10^{-2})$	$2.74509(10^{-2})$	$4.71251(10^{-2})$
1.8	$3.59329(10^{-2})$	$2.59796(10^{-2})$	$4.70191(10^{-2})$
2.0	$3.51351(10^{-2})$	$2.45539(10^{-2})$	$4.68973(10^{-2})$

Table 5.7: Error for BDF with $h = .2$ and $k = 1$

Test Problem #1			
t	$\{1,t\}$	$\{1,e^t\}$	$\{1,e^{-t}\}$
.2	$1.06937(10^{-2})$	$9.84431(10^{-3})$	$1.15928(10^{-2})$
.4	$1.62050(10^{-2})$	$1.46543(10^{-2})$	$1.78567(10^{-2})$
.6	$2.10603(10^{-2})$	$1.88019(10^{-2})$	$2.34625(10^{-2})$
.8	$2.05924(10^{-2})$	$1.78852(10^{-2})$	$2.34691(10^{-2})$
1.0	$2.01348(10^{-2})$	$1.69969(10^{-2})$	$2.34658(10^{-2})$
1.2	$1.96872(10^{-2})$	$1.61362(10^{-2})$	$2.34530(10^{-2})$
1.4	$1.92494(10^{-2})$	$1.53024(10^{-2})$	$2.34311(10^{-2})$
1.6	$1.88212(10^{-2})$	$1.44947(10^{-2})$	$2.34004(10^{-2})$
1.8	$1.84025(10^{-2})$	$1.37125(10^{-2})$	$2.33614(10^{-2})$
2.0	$1.79929(10^{-2})$	$1.29551(10^{-2})$	$2.33144(10^{-2})$

Table 5.8: Error for BDF with $h = .1$ and $k = 1$

Test Problem #1			
t	$\{1,t,t^2\}$	$\{1,e^t,e^{2t}\}$	$\{1,e^{-t},e^{-2t}\}$
.2	$3.31940(10^{-2})$	$3.40619(10^{-2})$	$3.33656(10^{-2})$
.4	$4.23820(10^{-2})$	$4.35747(10^{-2})$	$4.31615(10^{-2})$
.6	$5.15052(10^{-2})$	$5.34103(10^{-2})$	$5.26389(10^{-2})$
.8	$5.04867(10^{-2})$	$5.27159(10^{-2})$	$5.17916(10^{-2})$
1.0	$4.94881(10^{-2})$	$5.20278(10^{-2})$	$5.09570(10^{-2})$
1.2	$4.85094(10^{-2})$	$5.13469(10^{-2})$	$5.01352(10^{-2})$
1.4	$4.75501(10^{-2})$	$5.06727(10^{-2})$	$4.93258(10^{-2})$
1.6	$4.66097(10^{-2})$	$5.00048(10^{-2})$	$4.85288(10^{-2})$
1.8	$4.56880(10^{-2})$	$4.93433(10^{-2})$	$4.77439(10^{-2})$
2.0	$4.47845(10^{-2})$	$4.86881(10^{-2})$	$4.69710(10^{-2})$

Table 5.9: Error for BDF with $h = .2$ and $k = 2$

Test Problem #1			
t	$\{1, t, t^2\}$	$\{1, e^t, e^{2t}\}$	$\{1, e^{-t}, e^{-2t}\}$
.2	$1.71194(10^{-2})$	$1.73789(10^{-2})$	$1.70846(10^{-2})$
.4	$2.16747(10^{-2})$	$2.19043(10^{-2})$	$2.18399(10^{-2})$
.6	$2.64552(10^{-2})$	$2.68166(10^{-2})$	$2.66934(10^{-2})$
.8	$2.59314(10^{-2})$	$2.63705(10^{-2})$	$2.62189(10^{-2})$
1.0	$2.54183(10^{-2})$	$2.59322(10^{-2})$	$2.57529(10^{-2})$
1.2	$2.49153(10^{-2})$	$2.55009(10^{-2})$	$2.52951(10^{-2})$
1.4	$2.44222(10^{-2})$	$2.50765(10^{-2})$	$2.48453(10^{-2})$
1.6	$2.39390(10^{-2})$	$2.46589(10^{-2})$	$2.44034(10^{-2})$
1.8	$2.34653(10^{-2})$	$2.42480(10^{-2})$	$2.39692(10^{-2})$
2.0	$2.30009(10^{-2})$	$2.38437(10^{-2})$	$2.35426(10^{-2})$

Table 5.10: Error for BDF with $h = .1$ and $k = 2$

Test Problem #1			
t	$\{1, t, t^2, t^3\}$	$\{1, e^t, e^{2t}, e^{3t}\}$	$\{1, e^{-t}, e^{-2t}, e^{-3t}\}$
.2	$4.28415(10^{-2})$	$4.18148(10^{-2})$	$4.32050(10^{-2})$
.4	$5.24602(10^{-2})$	$5.18360(10^{-2})$	$5.27052(10^{-2})$
.6	$6.11476(10^{-2})$	$5.99853(10^{-2})$	$6.15407(10^{-2})$
.8	$5.99411(10^{-2})$	$5.86542(10^{-2})$	$6.03901(10^{-2})$
1.0	$5.87539(10^{-2})$	$5.73012(10^{-2})$	$5.92606(10^{-2})$
1.2	$5.75899(10^{-2})$	$5.59692(10^{-2})$	$5.81522(10^{-2})$
1.4	$5.64496(10^{-2})$	$5.46771(10^{-2})$	$5.70644(10^{-2})$
1.6	$5.53318(10^{-2})$	$5.34157(10^{-2})$	$5.59969(10^{-2})$
1.8	$5.42362(10^{-2})$	$5.21802(10^{-2})$	$5.49492(10^{-2})$
2.0	$5.31622(10^{-2})$	$5.09722(10^{-2})$	$5.39211(10^{-2})$

Table 5.11: Error for BDF with $h = .2$ and $k = 3$

Test Problem #1			
t	$\{1, t, t^2, t^3\}$	$\{1, e^t, e^{2t}, e^{3t}\}$	$\{1, e^{-t}, e^{-2t}, e^{-3t}\}$
.2	$2.19675(10^{-2})$	$2.18381(10^{-2})$	$2.20503(10^{-2})$
.4	$2.69407(10^{-2})$	$2.68198(10^{-2})$	$2.69879(10^{-2})$
.6	$3.15767(10^{-2})$	$3.14895(10^{-2})$	$3.16218(10^{-2})$
.8	$3.09497(10^{-2})$	$3.08341(10^{-2})$	$3.10059(10^{-2})$
1.0	$3.03372(10^{-2})$	$3.02049(10^{-2})$	$3.04023(10^{-2})$
1.2	$2.97364(10^{-2})$	$2.95853(10^{-2})$	$2.98104(10^{-2})$
1.4	$2.91476(10^{-2})$	$2.89791(10^{-2})$	$2.92300(10^{-2})$
1.6	$2.85704(10^{-2})$	$2.83852(10^{-2})$	$2.86610(10^{-2})$
1.8	$2.80047(10^{-2})$	$2.78034(10^{-2})$	$2.81030(10^{-2})$
2.0	$2.74502(10^{-2})$	$2.72336(10^{-2})$	$2.75559(10^{-2})$

Table 5.12: Error for BDF with $h = .1$ and $k = 3$

5.1.2 Test Problem #2

Consider the separably stiff, linear problem [25]

$$\begin{aligned} x'(t) &= -41x(t) + 59y(t) - \phi(t) \\ y'(t) &= 40x(t) - 60y(t) + \phi(t) \end{aligned}$$

where

$$\phi(t) = 2t^3(t^2 - 50t - 2)e^{-t^2},$$

and

$$x(0) = 9.9, \quad y(0) = 0.$$

The exact solution is given by

$$\begin{aligned} x(t) &= 4e^{-100t} + 5.9e^{-t} + t^4e^{-t^2} \\ y(t) &= -4e^{-100t} + 4e^{-t} - t^4e^{-t^2} \end{aligned}$$

Results of the integration using various basis functions, methods and stepsizes are given in Tables 5.13 thru 5.24.

Test Problem #2			
t	$\{1, t, t^2\}$	$\{1, e^t, e^{2t}\}$	$\{1, e^{-t}, e^{-2t}\}$
.2	$3.90786(10^{-1})$	$3.92497(10^{-1})$	$3.90056(10^{-1})$
.4	$4.70017(10^{-1})$	$4.71511(10^{-1})$	$4.69726(10^{-1})$
.6	$5.06556(10^{-1})$	$5.08320(10^{-1})$	$5.06204(10^{-1})$
.8	$4.17013(10^{-1})$	$4.18878(10^{-1})$	$4.16617(10^{-1})$
1.0	$3.43715(10^{-1})$	$3.45527(10^{-1})$	$3.43350(10^{-1})$
1.2	$2.79894(10^{-1})$	$2.81634(10^{-1})$	$2.79547(10^{-1})$
1.4	$2.25252(10^{-1})$	$2.26894(10^{-1})$	$2.24925(10^{-1})$
1.6	$1.81457(10^{-1})$	$1.82971(10^{-1})$	$1.81155(10^{-1})$
1.8	$1.47784(10^{-1})$	$1.49147(10^{-1})$	$1.47511(10^{-1})$
2.0	$1.21184(10^{-1})$	$1.22400(10^{-1})$	$1.20941(10^{-1})$

Table 5.13: Error for A-M with $h = .05$ and $k = 1$

Test Problem #2			
t	$\{1, t, t^2\}$	$\{1, e^t, e^{2t}\}$	$\{1, e^{-t}, e^{-2t}\}$
.2	$2.00276(10^{-1})$	$2.00508(10^{-1})$	$2.00231(10^{-1})$
.4	$2.44437(10^{-1})$	$2.44807(10^{-1})$	$2.44366(10^{-1})$
.6	$2.66046(10^{-1})$	$2.66489(10^{-1})$	$2.65961(10^{-1})$
.8	$2.19124(10^{-1})$	$2.19593(10^{-1})$	$2.19034(10^{-1})$
1.0	$1.80467(10^{-1})$	$1.80937(10^{-1})$	$1.80377(10^{-1})$
1.2	$1.46887(10^{-1})$	$1.47344(10^{-1})$	$1.46800(10^{-1})$
1.4	$1.18305(10^{-1})$	$1.18739(10^{-1})$	$1.18222(10^{-1})$
1.6	$9.54144(10^{-2})$	$9.58172(10^{-2})$	$9.53374(10^{-2})$
1.8	$7.77152(10^{-2})$	$7.80801(10^{-2})$	$7.76451(10^{-2})$
2.0	$6.36796(10^{-2})$	$6.40067(10^{-2})$	$6.36168(10^{-2})$

Table 5.14: Error for A-M with $h = .025$ and $k = 1$

Test Problem #2			
t	$\{1,t,t^2,t^3\}$	$\{1,e^t,e^{2t},e^{3t}\}$	$\{1,e^{-t},e^{-2t},e^{-3t}\}$
.2	$5.67843(10^{-1})$	$5.66816(10^{-1})$	$5.68096(10^{-1})$
.4	$6.07533(10^{-1})$	$6.07823(10^{-1})$	$6.07693(10^{-1})$
.6	$6.14125(10^{-1})$	$6.13520(10^{-1})$	$6.14000(10^{-1})$
.8	$5.06097(10^{-1})$	$5.05910(10^{-1})$	$5.05700(10^{-1})$
1.0	$4.16423(10^{-1})$	$4.16623(10^{-1})$	$4.15938(10^{-1})$
1.2	$3.39272(10^{-1})$	$3.39672(10^{-1})$	$3.38873(10^{-1})$
1.4	$2.73782(10^{-1})$	$2.74161(10^{-1})$	$2.73537(10^{-1})$
1.6	$2.20896(10^{-1})$	$2.21093(10^{-1})$	$2.20791(10^{-1})$
1.8	$1.79609(10^{-1})$	$1.79596(10^{-1})$	$1.79588(10^{-1})$
2.0	$1.46919(10^{-1})$	$1.46786(10^{-1})$	$1.46928(10^{-1})$

Table 5.15: Error for A-M with $h = .05$ and $k = 2$

Test Problem #2			
t	$\{1, t, t^2, t^3\}$	$\{1, e^t, e^{2t}, e^{3t}\}$	$\{1, e^{-t}, e^{-2t}, e^{-3t}\}$
.2	$2.96044(10^{-1})$	$2.96050(10^{-1})$	$2.96034(10^{-1})$
.4	$3.20823(10^{-1})$	$3.20804(10^{-1})$	$3.20826(10^{-1})$
.6	$3.26936(10^{-1})$	$3.26912(10^{-1})$	$3.26941(10^{-1})$
.8	$2.68905(10^{-1})$	$2.68886(10^{-1})$	$2.68906(10^{-1})$
1.0	$2.21135(10^{-1})$	$2.21111(10^{-1})$	$2.21139(10^{-1})$
1.2	$1.80221(10^{-1})$	$1.80197(10^{-1})$	$1.80225(10^{-1})$
1.4	$1.45655(10^{-1})$	$1.45633(10^{-1})$	$1.45659(10^{-1})$
1.6	$1.17732(10^{-1})$	$1.17711(10^{-1})$	$1.17735(10^{-1})$
1.8	$9.58007(10^{-2})$	$9.57821(10^{-2})$	$9.58039(10^{-2})$
2.0	$7.83531(10^{-2})$	$7.83363(10^{-2})$	$7.83560(10^{-2})$

Table 5.16: Error for A-M with $h = .025$ and $k = 2$

Test Problem #2			
t	$\{1, t, t^2, t^3, t^4\}$	$\{1, e^t, e^{2t}, e^{3t}, e^{4t}\}$	$\{1, e^{-t}, e^{-2t}, e^{-3t}, e^{-4t}\}$
.2	$7.52094(10^{-1})$	$7.75144(10^{-1})$	$7.43379(10^{-1})$
.4	$7.38316(10^{-1})$	1.10171	$7.27169(10^{-1})$
.6	1.58826	7.07598	$8.05087(10^{-1})$
.8	4.25859	$3.28297(10^{1})$	$8.82397(10^{-1})$
1.0	$1.32655(10^{1})$	$1.55384(10^{2})$	1.36550
1.2	$4.24071(10^{1})$	$7.37699(10^{2})$	2.65704
1.4	$1.36382(10^{2})$	$3.50431(10^{3})$	5.61379
1.6	$4.39263(10^{2})$	$1.66482(10^{4})$	$1.21666(10^{1})$
1.8	$1.41526(10^{3})$	$7.90933(10^{4})$	$2.65712(10^{1})$
2.0	$4.56009(10^{3})$	$3.75762(10^{5})$	$5.81472(10^{1})$

Table 5.17: Error for A-M with $h = .05$ and $k = 3$

Test Problem #2			
t	$\{1, t, t^2, t^3, t^4\}$	$\{1, e^t, e^{2t}, e^{3t}, e^{4t}\}$	$\{1, e^{-t}, e^{-2t}, e^{-3t}, e^{-4t}\}$
.2	$3.91164(10^{-1})$	$3.92525(10^{-1})$	$3.90428(10^{-1})$
.4	$3.94860(10^{-1})$	$3.93909(10^{-1})$	$3.95217(10^{-1})$
.6	$3.86513(10^{-1})$	$3.87172(10^{-1})$	$3.86411(10^{-1})$
.8	$3.18041(10^{-1})$	$3.18675(10^{-1})$	$3.17865(10^{-1})$
1.0	$2.61130(10^{-1})$	$2.61598(10^{-1})$	$2.61027(10^{-1})$
1.2	$2.12918(10^{-1})$	$2.13199(10^{-1})$	$2.12876(10^{-1})$
1.4	$1.72442(10^{-1})$	$1.72582(10^{-1})$	$1.72431(10^{-1})$
1.6	$1.39609(10^{-1})$	$1.39666(10^{-1})$	$1.39608(10^{-1})$
1.8	$1.13591(10^{-1})$	$1.13611(10^{-1})$	$1.13592(10^{-1})$
2.0	$9.28324(10^{-2})$	$9.28405(10^{-2})$	$9.28344(10^{-2})$

Table 5.18: Error for A-M with $h = .025$ and $k = 3$

Test Problem #2			
t	$\{1,t\}$	$\{1,e^t\}$	$\{1,e^{-t}\}$
.2	1.21424	1.12100	1.31557
.4	1.37227	1.24102	1.51029
.6	1.40898	1.27126	1.54926
.8	1.12160	$9.86017(10^{-1})$	1.25736
1.0	$9.17786(10^{-1})$	$7.88989(10^{-1})$	1.04495
1.2	$7.37426(10^{-1})$	$6.15925(10^{-1})$	$8.55527(10^{-1})$
1.4	$5.76450(10^{-1})$	$4.61618(10^{-1})$	$6.85864(10^{-1})$
1.6	$4.47352(10^{-1})$	$3.40929(10^{-1})$	$5.47015(10^{-1})$
1.8	$3.56491(10^{-1})$	$2.63162(10^{-1})$	$4.44095(10^{-1})$
2.0	$2.91172(10^{-1})$	$2.12448(10^{-1})$	$3.66548(10^{-1})$

Table 5.19: Error for BDF with $h = .2$ and $k = 1$

Test Problem #2			
t	$\{1,t\}$	$\{1,e^t\}$	$\{1,e^{-t}\}$
.2	$6.85183(10^{-1})$	$6.39358(10^{-1})$	$7.33284(10^{-1})$
.4	$7.98549(10^{-1})$	$7.29310(10^{-1})$	$8.70253(10^{-1})$
.6	$8.40660(10^{-1})$	$7.62432(10^{-1})$	$9.20631(10^{-1})$
.8	$6.74850(10^{-1})$	$5.95587(10^{-1})$	$7.55182(10^{-1})$
1.0	$5.46354(10^{-1})$	$4.69069(10^{-1})$	$6.24105(10^{-1})$
1.2	$4.35012(10^{-1})$	$3.60868(10^{-1})$	$5.08930(10^{-1})$
1.4	$3.39566(10^{-1})$	$2.69165(10^{-1})$	$4.08939(10^{-1})$
1.6	$2.64753(10^{-1})$	$1.99713(10^{-1})$	$3.28389(10^{-1})$
1.8	$2.10489(10^{-1})$	$1.53006(10^{-1})$	$2.67177(10^{-1})$
2.0	$1.69559(10^{-1})$	$1.19865(10^{-1})$	$2.19411(10^{-1})$

Table 5.20: Error for BDF with $h = .1$ and $k = 1$

Test Problem #2			
t	$\{1,t,t^2\}$	$\{1,e^t,e^{2t}\}$	$\{1,e^{-t},e^{-2t}\}$
.2	1.76956	1.81653	1.76699
.4	1.82050	1.90825	1.81117
.6	1.76289	1.86485	1.75009
.8	1.43468	1.53611	1.42168
1.0	1.19295	1.29062	1.17971
1.2	$9.78251(10^{-1})$	1.06977	$9.65649(10^{-1})$
1.4	$7.88084(10^{-1})$	$8.72619(10^{-1})$	$7.76332(10^{-1})$
1.6	$6.31816(10^{-1})$	$7.08515(10^{-1})$	$6.21004(10^{-1})$
1.8	$5.13478(10^{-1})$	$5.81365(10^{-1})$	$5.03711(10^{-1})$
2.0	$4.22906(10^{-1})$	$4.82039(10^{-1})$	$4.14234(10^{-1})$

Table 5.21: Error for BDF with $h = .2$ and $k = 2$

Test Problem #2			
t	$\{1, t, t^2\}$	$\{1, e^t, e^{2t}\}$	$\{1, e^{-t}, e^{-2t}\}$
.2	1.03625	1.04884	1.03503
.4	1.09676	1.11962	1.09353
.6	1.09007	1.11731	1.08581
.8	$8.95161(10^{-1})$	$9.23113(10^{-1})$	$8.90628(10^{-1})$
1.0	$7.38998(10^{-1})$	$7.66340(10^{-1})$	$7.34509(10^{-1})$
1.2	$6.03603(10^{-1})$	$6.29686(10^{-1})$	$5.99286(10^{-1})$
1.4	$4.87167(10^{-1})$	$5.11611(10^{-1})$	$4.83096(10^{-1})$
1.6	$3.92756(10^{-1})$	$4.15212(10^{-1})$	$3.88977(10^{-1})$
1.8	$3.19636(10^{-1})$	$3.39835(10^{-1})$	$3.16197(10^{-1})$
2.0	$2.62076(10^{-1})$	$2.80034(10^{-1})$	$2.58994(10^{-1})$

Table 5.22: Error for BDF with $h = .1$ and $k = 2$

Test Problem #2			
t	$\{1, t, t^2, t^3\}$	$\{1, e^t, e^{2t}, e^{3t}\}$	$\{1, e^{-t}, e^{-2t}, e^{-3t}\}$
.2	2.15416	2.11538	2.15802
.4	2.07327	2.00648	2.07447
.6	1.92562	1.85375	1.92628
.8	1.56435	1.49292	1.56584
1.0	1.29710	1.23030	1.29790
1.2	1.06061	$9.97130(10^{-1})$	1.06203
1.4	$8.54027(10^{-1})$	$7.95004(10^{-1})$	$8.55561(10^{-1})$
1.6	$6.84569(10^{-1})$	$6.31078(10^{-1})$	$6.85925(10^{-1})$
1.8	$5.55000(10^{-1})$	$5.07435(10^{-1})$	$5.56156(10^{-1})$
2.0	$4.55274(10^{-1})$	$4.13414(10^{-1})$	$4.56298(10^{-1})$

Table 5.23: Error for BDF with $h = .2$ and $k = 3$

Test Problem #2			
t	$\{1, t, t^2, t^3\}$	$\{1, e^t, e^{2t}, e^{3t}\}$	$\{1, e^{-t}, e^{-2t}, e^{-3t}\}$
.2	1.31494	1.31012	1.31542
.4	1.30167	1.29389	1.30175
.6	1.23962	1.23018	1.23982
.8	1.01691	1.00741	1.01699
1.0	$8.37047(10^{-1})$	$8.27391(10^{-1})$	$8.37380(10^{-1})$
1.2	$6.83329(10^{-1})$	$6.74188(10^{-1})$	$6.83617(10^{-1})$
1.4	$5.52064(10^{-1})$	$5.43476(10^{-1})$	$5.52346(10^{-1})$
1.6	$4.45348(10^{-1})$	$4.37456(10^{-1})$	$4.45600(10^{-1})$
1.8	$3.61885(10^{-1})$	$3.54757(10^{-1})$	$3.62110(10^{-1})$
2.0	$2.95977(10^{-1})$	$2.89605(10^{-1})$	$2.96181(10^{-1})$

Table 5.24: Error for BDF with $h = .1$ and $k = 3$

5.1.3 Test Problem #3

Consider the linear problem [26]

$$\begin{aligned} x'(t) &= (1+48t^2)x(t) - 49t^2y(t) - 24.5t^2z(t) + \phi_1(t) \\ y'(t) &= 98t^2x(t) + (1-99t^2)y(t) - 49t^2z(t) + \phi_2(t) \\ z'(t) &= (1-t^2)z(t) + \phi_3(t) \end{aligned}$$

with

$$x(1) = 1, \quad y(1) = 1 \quad z(1) = 1,$$

and where

$$\begin{aligned} \phi_1(t) &= 49t^3 - 48t^2 + 24.5t - 1 \\ \phi_2(t) &= 99t^3 - 98t^2 + 48t + 1 \\ \phi_3(t) &= \frac{t^3 - t - 1}{t^2}. \end{aligned}$$

The exact solution is given by

$$\begin{aligned} x(t) &= 1 \\ y(t) &= t \\ z(t) &= \frac{1}{t}. \end{aligned}$$

Results of the integration using various basis functions, methods and stepsizes are given in Tables 5.25 thru 5.36.

Test Problem #3			
t	$\{1,t,t^2\}$	$\{1,e^t,e^{2t}\}$	$\{1,e^{-t},e^{-2t}\}$
1.2	$1.89859(10^{-1})$	$1.87481(10^{-1})$	$1.93108(10^{-1})$
1.4	$2.57014(10^{-1})$	$2.66449(10^{-1})$	$2.51555(10^{-1})$
1.6	$2.07584(10^{-1})$	$2.02386(10^{-1})$	$2.12885(10^{-1})$
1.8	$1.98044(10^{-1})$	$2.08262(10^{-1})$	$1.95004(10^{-1})$
2.0	$1.53170(10^{-1})$	$1.46124(10^{-1})$	$1.58138(10^{-1})$
2.2	$1.39588(10^{-1})$	$1.48511(10^{-1})$	$1.38722(10^{-1})$
2.4	$1.46520(10^{-2})$	$2.14195(10^{-2})$	$1.48498(10^{-2})$
2.6	$4.10298(10^{-2})$	$5.29944(10^{-2})$	$3.94949(10^{-2})$
2.8	$3.69391(10^{-2})$	$5.48184(10^{-2})$	$3.41327(10^{-2})$
3.0	$3.01949(10^{-2})$	$5.75702(10^{-2})$	$2.58011(10^{-2})$

Table 5.25: Error for A-M with $h = .2$ and $k = 1$

Test Problem #3			
t	$\{1, t, t^2\}$	$\{1, e^t, e^{2t}\}$	$\{1, e^{-t}, e^{-2t}\}$
1.2	$1.24415(10^{-1})$	$1.26834(10^{-1})$	$1.22549(10^{-1})$
1.4	$1.12346(10^{-1})$	$1.13870(10^{-1})$	$1.11761(10^{-1})$
1.6	$9.92833(10^{-2})$	$1.00335(10^{-1})$	$9.92790(10^{-2})$
1.8	$8.33420(10^{-2})$	$8.40637(10^{-2})$	$8.36455(10^{-2})$
2.0	$6.98242(10^{-2})$	$7.02670(10^{-2})$	$7.02870(10^{-2})$
2.2	$6.21572(10^{-2})$	$6.24387(10^{-2})$	$6.26156(10^{-2})$
2.4	$6.84524(10^{-3})$	$6.95645(10^{-3})$	$7.06215(10^{-3})$
2.6	$1.65004(10^{-2})$	$1.55258(10^{-2})$	$1.77283(10^{-2})$
2.8	$1.74792(10^{-2})$	$1.81462(10^{-2})$	$1.75963(10^{-2})$
3.0	$1.09990(10^{-2})$	$9.45354(10^{-3})$	$1.21223(10^{-2})$

Table 5.26: Error for A-M with $h = .1$ and $k = 1$

Test Problem #3			
t	$\{1,t,t^2,t^3\}$	$\{1,e^t,e^{2t},e^{3t}\}$	$\{1,e^{-t},e^{-2t},e^{-3t}\}$
1.2	2.07389(10^{-1})	2.16486(10^{-1})	1.99865(10^{-1})
1.4	1.94829(10^{-1})	2.18447(10^{-1})	1.80867(10^{-1})
1.6	1.79391(10^{-1})	2.31187(10^{-1})	1.57039(10^{-1})
1.8	1.69419(10^{-1})	2.89460(10^{-1})	1.30557(10^{-1})
2.0	1.81660(10^{-1})	4.69834(10^{-1})	1.09738(10^{-1})
2.2	2.37895(10^{-1})	9.54344(10^{-1})	1.01292(10^{-1})
2.4	4.34368(10^{-1})	3.49293	5.19626(10^{-2})
2.6	1.34432	1.63736(10^{1})	1.19089(10^{-1})
2.8	4.30654	8.13509(10^{1})	2.06681(10^{-1})
3.0	1.47628(10^{1})	4.25056(10^{2})	5.04989(10^{-1})

Table 5.27: Error for A-M with $h = .1$ and $k = 2$

Test Problem #3			
t	$\{1, t, t^2, t^3\}$	$\{1, e^t, e^{2t}, e^{3t}\}$	$\{1, e^{-t}, e^{-2t}, e^{-3t}\}$
1.2	$8.55687(10^{-2})$	$8.70520(10^{-2})$	$8.44617(10^{-2})$
1.4	$7.75078(10^{-2})$	$7.84090(10^{-2})$	$7.70229(10^{-2})$
1.6	$6.53249(10^{-2})$	$6.62774(10^{-2})$	$6.49520(10^{-2})$
1.8	$5.11296(10^{-2})$	$5.27839(10^{-2})$	$5.06554(10^{-2})$
2.0	$3.97349(10^{-2})$	$4.37946(10^{-2})$	$3.88666(10^{-2})$
2.2	$3.43992(10^{-2})$	$4.65428(10^{-2})$	$3.24471(10^{-2})$
2.4	$7.78507(10^{-3})$	$6.71498(10^{-2})$	$4.91652(10^{-3})$
2.6	$2.03318(10^{-2})$	$3.88558(10^{-1})$	$6.81733(10^{-3})$
2.8	$5.75783(10^{-2})$	2.68464	$4.49676(10^{-2})$
3.0	$3.68958(10^{-1})$	$2.16993(10^{1})$	$1.44994(10^{-1})$

Table 5.28: Error for A-M with $h = .05$ and $k = 2$

Test Problem #3			
t	$\{1,t,t^2,t^3,t^4\}$	$\{1,e^t,e^{2t},e^{3t},e^{4t}\}$	$\{1,e^{-t},e^{-2t},e^{-3t},e^{-4t}\}$
1.2	$2.72932(10^{-2})$	$2.72935(10^{-2})$	$2.72935(10^{-2})$
1.4	$2.47716(10^{-2})$	$2.47722(10^{-2})$	$2.47722(10^{-2})$
1.6	$2.00554(10^{-2})$	$2.00561(10^{-2})$	$2.00561(10^{-2})$
1.8	$1.46905(10^{-2})$	$1.46912(10^{-2})$	$1.46912(10^{-2})$
2.0	$1.03513(10^{-2})$	$1.03520(10^{-2})$	$1.03520(10^{-2})$
2.2	$8.04307(10^{-3})$	$8.04359(10^{-3})$	$8.04360(10^{-3})$
2.4	$1.77529(10^{-3})$	$1.77560(10^{-3})$	$1.77559(10^{-3})$
2.6	$5.04459(10^{-2})$	$7.71864(10^{-2})$	$3.26448(10^{-2})$
2.8	2.68929	6.28057	1.13653
3.0	$4.85567(10^{2})$	$1.70910(10^{3})$	$1.36242(10^{2})$

Table 5.29: Error for A-M with $h = .0125$ and $k = 3$

Test Problem #3			
t	$\{1, t, t^2, t^3, t^4\}$	$\{1, e^t, e^{2t}, e^{3t}, e^{4t}\}$	$\{1, e^{-t}, e^{-2t}, e^{-3t}, e^{-4t}\}$
1.2	$1.36903(10^{-2})$	$1.36903(10^{-2})$	$1.36903(10^{-2})$
1.4	$1.23765(10^{-2})$	$1.23766(10^{-2})$	$1.23766(10^{-2})$
1.6	$9.96794(10^{-3})$	$9.96803(10^{-3})$	$9.96803(10^{-3})$
1.8	$7.25618(10^{-3})$	$7.25627(10^{-3})$	$7.25627(10^{-3})$
2.0	$5.09019(10^{-3})$	$5.09027(10^{-3})$	$5.09027(10^{-3})$
2.2	$3.96807(10^{-3})$	$3.96814(10^{-3})$	$3.96814(10^{-3})$
2.4	$8.98561(10^{-4})$	$8.98597(10^{-4})$	$8.98597(10^{-4})$
2.6	$1.71307(10^{-3})$	$1.71308(10^{-3})$	$1.71314(10^{-3})$
2.8	$1.73563(10^{-3})$	$1.73364(10^{-3})$	$1.73696(10^{-3})$
3.0	$1.30155(10^{-3})$	$1.24331(10^{-3})$	$1.33952(10^{-3})$

Table 5.30: Error for A-M with $h = .00625$ and $k = 3$

Test Problem #3			
1.2	2.01671(10^{-1})	1.92201(10^{-1})	2.12677(10^{-1})
1.4	2.26037(10^{-1})	2.12445(10^{-1})	2.42265(10^{-1})
1.6	2.12288(10^{-1})	1.95458(10^{-1})	2.31791(10^{-1})
1.8	1.87213(10^{-1})	1.69576(10^{-1})	2.06930(10^{-1})
2.0	1.62062(10^{-1})	1.45517(10^{-1})	1.80025(10^{-1})
2.2	1.42970(10^{-1})	1.28476(10^{-1})	1.58418(10^{-1})
2.4	1.72613(10^{-2})	6.52243(10^{-3})	3.13459(10^{-2})
2.6	3.84284(10^{-2})	2.82203(10^{-2})	4.76546(10^{-2})
2.8	3.66424(10^{-2})	2.95494(10^{-2})	4.31102(10^{-2})
3.0	3.03723(10^{-2})	2.46890(10^{-2})	3.56630(10^{-2})

Table 5.31: Error for BDF with $h = .2$ and $k = 1$

Test Problem #3			
t	$\{1, t\}$	$\{1, e^t\}$	$\{1, e^{-t}\}$
1.2	$1.07446(10^{-1})$	$1.03291(10^{-1})$	$1.12038(10^{-1})$
1.4	$1.09386(10^{-1})$	$1.01926(10^{-1})$	$1.17605(10^{-1})$
1.6	$9.95541(10^{-2})$	$9.00054(10^{-2})$	$1.09934(10^{-1})$
1.8	$8.62982(10^{-2})$	$7.63421(10^{-2})$	$9.70451(10^{-2})$
2.0	$7.46716(10^{-2})$	$6.56437(10^{-2})$	$8.44518(10^{-2})$
2.2	$6.69947(10^{-2})$	$5.94958(10^{-2})$	$7.51882(10^{-2})$
2.4	$7.69100(10^{-3})$	$3.42318(10^{-3})$	$1.56994(10^{-2})$
2.6	$2.24946(10^{-2})$	$1.63658(10^{-2})$	$2.83130(10^{-2})$
2.8	$2.23628(10^{-2})$	$1.78699(10^{-2})$	$2.66364(10^{-2})$
3.0	$1.84925(10^{-2})$	$1.48782(10^{-2})$	$2.19600(10^{-2})$

Table 5.32: Error for BDF with $h = .1$ and $k = 1$

Test Problem #3			
t	$\{1, t, t^2\}$	$\{1, e^t, e^{2t}\}$	$\{1, e^{-t}, e^{-2t}\}$
1.2	$2.94213(10^{-1})$	$2.91119(10^{-1})$	$3.00100(10^{-1})$
1.4	$3.51560(10^{-1})$	$3.63797(10^{-1})$	$3.48139(10^{-1})$
1.6	$3.16831(10^{-1})$	$3.30687(10^{-1})$	$3.14636(10^{-1})$
1.8	$2.64381(10^{-1})$	$2.77787(10^{-1})$	$2.63410(10^{-1})$
2.0	$2.09078(10^{-1})$	$2.19783(10^{-1})$	$2.09865(10^{-1})$
2.2	$1.63517(10^{-1})$	$1.69188(10^{-1})$	$1.66726(10^{-1})$
2.4	$2.13138(10^{-2})$	$2.30074(10^{-2})$	$2.52109(10^{-2})$
2.6	$4.15075(10^{-2})$	$4.55419(10^{-2})$	$4.27786(10^{-2})$
2.8	$3.53574(10^{-2})$	$3.78809(10^{-2})$	$3.67413(10^{-2})$
3.0	$2.64326(10^{-2})$	$2.70684(10^{-2})$	$2.84546(10^{-2})$

Table 5.33: Error for BDF with $h = .2$ and $k = 2$

Test Problem #3			
t	$\{1,t,t^2\}$	$\{1,e^t,e^{2t}\}$	$\{1,e^{-t},e^{-2t}\}$
1.2	1.66800(10^{-1})	1.68433(10^{-1})	1.66288(10^{-1})
1.4	1.60356(10^{-1})	1.62370(10^{-1})	1.60549(10^{-1})
1.6	1.39084(10^{-1})	1.41357(10^{-1})	1.39711(10^{-1})
1.8	1.11499(10^{-1})	1.13550(10^{-1})	1.12517(10^{-1})
2.0	8.59030(10^{-2})	8.71262(10^{-2})	8.74057(10^{-2})
2.2	6.90572(10^{-2})	6.95000(10^{-2})	7.07688(10^{-2})
2.4	1.01140(10^{-2})	1.07038(10^{-2})	1.10582(10^{-2})
2.6	2.31556(10^{-2})	2.41962(10^{-2})	2.37531(10^{-2})
2.8	2.09297(10^{-2})	2.14244(10^{-2})	2.15895(10^{-2})
3.0	1.57836(10^{-2})	1.58498(10^{-2})	1.65828(10^{-2})

Table 5.34: Error for BDF with $h = .1$ and $k = 2$

Test Problem #3			
t	$\{1,t,t^2,t^3\}$	$\{1,e^t,e^{2t},e^{3t}\}$	$\{1,e^{-t},e^{-2t},e^{-3t}\}$
1.2	$3.84268(10^{-1})$	$3.72839(10^{-1})$	$3.94195(10^{-1})$
1.4	$4.95404(10^{-1})$	$5.19396(10^{-1})$	$4.78059(10^{-1})$
1.6	$4.24796(10^{-1})$	$4.26336(10^{-1})$	$4.17818(10^{-1})$
1.8	$3.45456(10^{-1})$	$3.36899(10^{-1})$	$3.41760(10^{-1})$
2.0	$2.63564(10^{-1})$	$2.54892(10^{-1})$	$2.61540(10^{-1})$
2.2	$1.92080(10^{-1})$	$1.83299(10^{-1})$	$1.93540(10^{-1})$
2.4	$2.44110(10^{-2})$	$1.91712(10^{-2})$	$2.86358(10^{-2})$
2.6	$4.47768(10^{-2})$	$5.09385(10^{-2})$	$4.30836(10^{-2})$
2.8	$3.33507(10^{-2})$	$2.91374(10^{-2})$	$3.46391(10^{-2})$
3.0	$2.43618(10^{-2})$	$2.25883(10^{-2})$	$2.59638(10^{-2})$

Table 5.35: Error for BDF with $h = .2$ and $k = 3$

Test Problem #3			
t	$\{1,t,t^2,t^3\}$	$\{1,e^t,e^{2t},e^{3t}\}$	$\{1,e^{-t},e^{-2t},e^{-3t}\}$
1.2	$2.35708(10^{-1})$	$2.40906(10^{-1})$	$2.31628(10^{-1})$
1.4	$2.11373(10^{-1})$	$2.09808(10^{-1})$	$2.11795(10^{-1})$
1.6	$1.82919(10^{-1})$	$1.82867(10^{-1})$	$1.82451(10^{-1})$
1.8	$1.42543(10^{-1})$	$1.41956(10^{-1})$	$1.42488(10^{-1})$
2.0	$1.04323(10^{-1})$	$1.03856(10^{-1})$	$1.04573(10^{-1})$
2.2	$7.76870(10^{-2})$	$7.75119(10^{-2})$	$7.80602(10^{-2})$
2.4	$1.29735(10^{-2})$	$1.27671(10^{-2})$	$1.33148(10^{-2})$
2.6	$2.22564(10^{-2})$	$2.14266(10^{-2})$	$2.28007(10^{-2})$
2.8	$1.86228(10^{-2})$	$1.73241(10^{-2})$	$1.95524(10^{-2})$
3.0	$1.34654(10^{-2})$	$1.23937(10^{-2})$	$1.43953(10^{-2})$

Table 5.36: Error for BDF with $h = .1$ and $k = 3$

5.1.4 Test Problem #4

Consider the nonlinear problem [32]

$$\begin{aligned} x'(t) &= gx(t)(y(t)-1) \\ y'(t) &= \alpha - y(t)(x(t)+1) \end{aligned}$$

with

$$x(0) = .1, \quad y(0) = 1,$$

and where $g = 10^5$, $\alpha = 30$. Since no exact solution is known, a baseline solution was generated by Runge–Kutta–Fehlberg. Results of the integration using various basis functions, methods and stepsizes are given in Tables 5.37 thru 5.48.

Test Problem #4			
t	$\{1, t, t^2\}$	$\{1, e^t, e^{2t}\}$	$\{1, e^{-t}, e^{-2t}\}$
$1.0(10^{-3})$	$2.09672(10^{-4})$	$1.91341(10^{-2})$	$1.91341(10^{-2})$
$2.0(10^{-3})$	$3.79648(10^{-2})$	4.28648	4.28648
$3.0(10^{-3})$	$9.25534(10^{-2})$	$1.53199(10^{1})$	$1.53199(10^{1})$
$4.0(10^{-3})$	$1.08619(10^{-3})$	$2.00965(10^{-1})$	$2.00965(10^{-1})$
$5.0(10^{-3})$	$1.85961(10^{-5})$	$4.85592(10^{-3})$	$4.85592(10^{-3})$
$6.0(10^{-3})$	$1.11245(10^{-3})$	$1.42304(10^{-1})$	$1.42304(10^{-1})$
$7.0(10^{-3})$	$1.24369(10^{-1})$	$1.44137(10^{1})$	$1.44137(10^{1})$
$8.0(10^{-3})$	$1.66100(10^{-1})$	$4.02765(10^{1})$	$4.02765(10^{1})$
$9.0(10^{-3})$	$3.28059(10^{1})$	$3.22062(10^{1})$	$3.22062(10^{1})$
$1.0(10^{-3})$	$3.32985(10^{1})$	$3.32929(10^{1})$	$3.32929(10^{1})$

Table 5.37: Error for A-M with $h = 1.22070(10^{-7})$ and $k = 1$

Test Problem #4			
t	$\{1,t,t^2\}$	$\{1,e^t,e^{2t}\}$	$\{1,e^{-t},e^{-2t}\}$
$1.0(10^{-3})$	$1.04330(10^{-4})$	$1.54452(10^{1})$	$1.54452(10^{1})$
$2.0(10^{-3})$	$1.89391(10^{-2})$	$1.92469(10^{1})$	$1.92469(10^{1})$
$3.0(10^{-3})$	$4.62509(10^{-2})$	$3.95114(10^{1})$	$3.95114(10^{1})$
$4.0(10^{-3})$	$5.43505(10^{-4})$	$7.33368(10^{1})$	$7.33368(10^{1})$
$5.0(10^{-3})$	$9.46572(10^{-6})$	$8.87636(10^{-2})$	$8.87636(10^{-2})$
$6.0(10^{-3})$	$5.53763(10^{-4})$	$1.78533(10^{1})$	$1.78533(10^{1})$
$7.0(10^{-3})$	$6.19812(10^{-2})$	$2.51289(10^{1})$	$2.51289(10^{1})$
$8.0(10^{-3})$	$8.30028(10^{-2})$	$3.23340(10^{1})$	$3.23340(10^{1})$
$9.0(10^{-3})$	$3.28048(10^{1})$	8.83408	8.83408
$1.0(10^{-2})$	$3.32986(10^{1})$	$3.30004(10^{1})$	$3.30004(10^{1})$

Table 5.38: Error for A-M with $h = 6.10352(10^{-8})$ and $k = 1$

Test Problem #4			
t	$\{1, t, t^2, t^3\}$	$\{1, e^t, e^{2t}, e^{3t}\}$	$\{1, e^{-t}, e^{-2t}, e^{-3t}\}$
$1.0(10^{-3})$	$3.15052(10^{-4})$	$1.90378(10^{-2})$	$1.90378(10^{-2})$
$2.0(10^{-3})$	$5.06547(10^{-2})$	4.27654	4.27654
$3.0(10^{-3})$	$1.15685(10^{-1})$	$1.52894(10^{1})$	$1.52894(10^{1})$
$4.0(10^{-3})$	$1.30303(10^{-3})$	$2.00649(10^{-1})$	$2.00649(10^{-1})$
$5.0(10^{-3})$	$2.16319(10^{-5})$	$4.84844(10^{-3})$	$4.84844(10^{-3})$
$6.0(10^{-3})$	$1.27220(10^{-3})$	$1.42209(10^{-1})$	$1.42209(10^{-1})$
$7.0(10^{-3})$	$1.39980(10^{-1})$	$1.44067(10^{1})$	$1.44067(10^{1})$
$8.0(10^{-3})$	$1.84536(10^{-1})$	$4.02426(10^{1})$	$4.02426(10^{1})$
$9.0(10^{-3})$	$3.28061(10^{1})$	$3.22067(10^{1})$	$3.22067(10^{1})$
$1.0(10^{-2})$	$3.32985(10^{1})$	$3.32929(10^{1})$	$3.32929(10^{1})$

Table 5.39: Error for A-M with $h = 1.22070(10^{-7})$ and $k = 2$

Test Problem #4			
t	$\{1, t, t^2, t^3\}$	$\{1, e^t, e^{2t}, e^{3t}\}$	$\{1, e^{-t}, e^{-2t}, e^{-3t}\}$
$1.0(10^{-3})$	$1.56990(10^{-4})$	$1.54501(10^{1})$	$1.54501(10^{1})$
$2.0(10^{-3})$	$2.52776(10^{-2})$	$1.92471(10^{1})$	$1.92471(10^{1})$
$3.0(10^{-3})$	$5.78305(10^{-2})$	$3.95114(10^{1})$	$3.95114(10^{1})$
$4.0(10^{-3})$	$6.52077(10^{-4})$	$7.33199(10^{1})$	$7.33199(10^{1})$
$5.0(10^{-3})$	$1.09882(10^{-5})$	$8.87576(10^{-2})$	$8.87576(10^{-2})$
$6.0(10^{-3})$	$6.33506(10^{-4})$	$1.78584(10^{1})$	$1.78584(10^{1})$
$7.0(10^{-3})$	$6.97698(10^{-2})$	$2.51291(10^{1})$	$2.51291(10^{1})$
$8.0(10^{-3})$	$9.22438(10^{-2})$	$3.23338(10^{1})$	$3.23338(10^{1})$
$9.0(10^{-3})$	$3.28049(10^{1})$	8.84017	8.84017
$1.0(10^{-2})$	$3.32986(10^{1})$	$3.30004(10^{1})$	$3.30004(10^{1})$

Table 5.40: Error for A-M with $h = 6.10352(10^{-8})$ and $k = 2$

Test Problem #4			
t	$\{1, t, t^2, t^3, t^4\}$	$\{1, e^t, e^{2t}, e^{3t}, e^{4t}\}$	$\{1, e^{-t}, e^{-2t}, e^{-3t}, e^{-4t}\}$
$1.0(10^{-3})$	$4.20490(10^{-4})$	$1.89414(10^{-2})$	$1.89414(10^{-2})$
$2.0(10^{-3})$	$6.33553(10^{-2})$	4.26660	4.26660
$3.0(10^{-3})$	$1.38806(10^{-1})$	$1.52590(10^{1})$	$1.52590(10^{1})$
$4.0(10^{-3})$	$1.51980(10^{-3})$	$2.00333(10^{-1})$	$2.00333(10^{-1})$
$5.0(10^{-3})$	$2.46658(10^{-5})$	$4.84097(10^{-3})$	$4.84097(10^{-3})$
$6.0(10^{-3})$	$1.43207(10^{-3})$	$1.42114(10^{-1})$	$1.42114(10^{-1})$
$7.0(10^{-3})$	$1.55607(10^{-1})$	$1.43997(10^{1})$	$1.43997(10^{1})$
$8.0(10^{-3})$	$2.02967(10^{-1})$	$4.02087(10^{1})$	$4.02087(10^{1})$
$9.0(10^{-3})$	$3.28064(10^{1})$	$3.22072(10^{1})$	$3.22072(10^{1})$
$1.0(10^{-2})$	$3.32985(10^{1})$	$3.32929(10^{1})$	$3.32929(10^{1})$

Table 5.41: Error for A-M with $h = 1.22070(10^{-7})$ and $k = 3$

Test Problem #4			
t	$\{1,t,t^2,t^3,t^4\}$	$\{1,e^t,e^{2t},e^{3t},e^{4t}\}$	$\{1,e^{-t},e^{-2t},e^{-3t},e^{-4t}\}$
$1.0(10^{-3})$	$2.09663(10^{-4})$	$1.54551(10^1)$	$1.54551(10^1)$
$2.0(10^{-3})$	$3.16188(10^{-2})$	$1.92472(10^1)$	$1.92472(10^1)$
$3.0(10^{-3})$	$6.94074(10^{-2})$	$3.95113(10^1)$	$3.95113(10^1)$
$4.0(10^{-3})$	$7.60634(10^{-4})$	$7.33029(10^1)$	$7.33029(10^1)$
$5.0(10^{-3})$	$1.25106(10^{-5})$	$8.87517(10^{-2})$	$8.87517(10^{-2})$
$6.0(10^{-3})$	$7.13279(10^{-4})$	$1.78635(10^1)$	$1.78635(10^1)$
$7.0(10^{-3})$	$7.75623(10^{-2})$	$2.51293(10^1)$	$2.51293(10^1)$
$8.0(10^{-3})$	$1.01484(10^{-1})$	$3.23336(10^1)$	$3.23336(10^1)$
$9.0(10^{-3})$	$3.28050(10^1)$	8.84628	8.84628
$1.0(10^{-2})$	$3.32986(10^1)$	$3.30004(10^1)$	$3.30004(10^1)$

Table 5.42: Error for A-M with $h = 6.10352(10^{-8})$ and $k = 3$

Test Problem #4			
$1.0(10^{-3})$	$3.12702(10^{-4})$	$3.12649(10^{-4})$	$3.12754(10^{-4})$
$2.0(10^{-3})$	$6.68126(10^{-2})$	$6.67997(10^{-2})$	$6.68251(10^{-2})$
$3.0(10^{-3})$	$1.35442(10^{-1})$	$1.35407(10^{-1})$	$1.35476(10^{-1})$
$4.0(10^{-3})$	$7.48468(10^{-4})$	$7.48908(10^{-4})$	$7.48040(10^{-4})$
$5.0(10^{-3})$	$7.75021(10^{-4})$	$7.75025(10^{-4})$	$7.75017(10^{-4})$
$6.0(10^{-3})$	$5.47499(10^{-3})$	$5.47450(10^{-3})$	$5.47547(10^{-3})$
$7.0(10^{-3})$	$4.11600(10^{-1})$	$4.11544(10^{-1})$	$4.11655(10^{-1})$
$8.0(10^{-3})$	$4.59808(10^{-1})$	$4.59734(10^{-1})$	$4.59880(10^{-1})$
$9.0(10^{-3})$	$3.28050(10^{1})$	$3.28050(10^{1})$	$3.28050(10^{1})$
$1.0(10^{-2})$	$3.32961(10^{1})$	$3.32961(10^{1})$	$3.32961(10^{1})$

Table 5.43: Error for BDF with $h = 1.22070(10^{-7})$ and $k = 1$

Test Problem #4			
t	$\{1, t\}$	$\{1, e^t\}$	$\{1, e^{-t}\}$
$1.0(10^{-3})$	$1.55811(10^{-4})$	$1.55783(10^{-4})$	$1.55836(10^{-4})$
$2.0(10^{-3})$	$3.33424(10^{-2})$	$3.33358(10^{-2})$	$3.33485(10^{-2})$
$3.0(10^{-3})$	$6.77525(10^{-2})$	$6.77345(10^{-2})$	$6.77692(10^{-2})$
$4.0(10^{-3})$	$3.73929(10^{-4})$	$3.74154(10^{-4})$	$3.73720(10^{-4})$
$5.0(10^{-3})$	$3.86687(10^{-4})$	$3.86689(10^{-4})$	$3.86685(10^{-4})$
$6.0(10^{-3})$	$2.72780(10^{-3})$	$2.72755(10^{-3})$	$2.72803(10^{-3})$
$7.0(10^{-3})$	$2.05026(10^{-1})$	$2.04997(10^{-1})$	$2.05052(10^{-1})$
$8.0(10^{-3})$	$2.30730(10^{-1})$	$2.30692(10^{-1})$	$2.30765(10^{-1})$
$9.0(10^{-3})$	$3.28043(10^{1})$	$3.28043(10^{1})$	$3.28043(10^{1})$
$1.0(10^{-2})$	$3.32973(10^{1})$	$3.32973(10^{1})$	$3.32973(10^{1})$

Table 5.44: Error for BDF with $h = 6.10352(10^{-8})$ and $k = 1$

Test Problem #4			
t	$\{1, t, t^2\}$	$\{1, e^t, e^{2t}\}$	$\{1, e^{-t}, e^{-2t}\}$
$1.0(10^{-3})$	$3.15083(10^{-4})$	$3.15082(10^{-4})$	$3.15082(10^{-4})$
$2.0(10^{-3})$	$5.06660(10^{-2})$	$5.06658(10^{-2})$	$5.06659(10^{-2})$
$3.0(10^{-3})$	$1.15691(10^{-1})$	$1.15690(10^{-1})$	$1.15690(10^{-1})$
$4.0(10^{-3})$	$1.30320(10^{-3})$	$1.30319(10^{-3})$	$1.30319(10^{-3})$
$5.0(10^{-3})$	$2.16359(10^{-5})$	$2.16360(10^{-5})$	$2.16357(10^{-5})$
$6.0(10^{-3})$	$1.27239(10^{-3})$	$1.27239(10^{-3})$	$1.27239(10^{-3})$
$7.0(10^{-3})$	$1.40008(10^{-1})$	$1.40007(10^{-1})$	$1.40007(10^{-1})$
$8.0(10^{-3})$	$1.84556(10^{-1})$	$1.84555(10^{-1})$	$1.84556(10^{-1})$
$9.0(10^{-3})$	$3.28061(10^{1})$	$3.28061(10^{1})$	$3.28061(10^{1})$
$1.0(10^{-2})$	$3.32985(10^{1})$	$3.32985(10^{1})$	$3.32985(10^{1})$

Table 5.45: Error for BDF with $h = 1.22070(10^{-7})$ and $k = 2$

Test Problem #4			
t	$\{1,t,t^2\}$	$\{1,e^t,e^{2t}\}$	$\{1,e^{-t},e^{-2t}\}$
1.0(10^{-3})	1.56997(10^{-4})	1.56996(10^{-4})	1.56996(10^{-4})
2.0(10^{-3})	2.52804(10^{-2})	2.52802(10^{-2})	2.52802(10^{-2})
3.0(10^{-3})	5.78319(10^{-2})	5.78312(10^{-2})	5.78312(10^{-2})
4.0(10^{-3})	6.52120(10^{-4})	6.52111(10^{-4})	6.52111(10^{-4})
5.0(10^{-3})	1.09893(10^{-5})	1.09891(10^{-5})	1.09891(10^{-5})
6.0(10^{-3})	6.33554(10^{-4})	6.33544(10^{-4})	6.33545(10^{-4})
7.0(10^{-3})	6.97766(10^{-2})	6.97756(10^{-2})	6.97756(10^{-2})
8.0(10^{-3})	9.22490(10^{-2})	9.22475(10^{-2})	9.22475(10^{-2})
9.0(10^{-3})	3.28049(10^{1})	3.28049(10^{1})	3.28049(10^{1})
1.0(10^{-2})	3.32986(10^{1})	3.32986(10^{1})	3.32986(10^{1})

Table 5.46: Error for BDF with $h = 6.10352(10^{-8})$ and $k = 2$

Test Problem #4			
t	$\{1, t, t^2, t^3\}$	$\{1, e^t, e^{2t}, e^{3t}\}$	$\{1, e^{-t}, e^{-2t}, e^{-3t}\}$
$1.0(10^{-3})$	$4.20493(10^{-4})$	$4.20492(10^{-4})$	$4.20492(10^{-4})$
$2.0(10^{-3})$	$6.33555(10^{-2})$	$6.33553(10^{-2})$	$6.33553(10^{-2})$
$3.0(10^{-3})$	$1.38806(10^{-1})$	$1.38806(10^{-1})$	$1.38806(10^{-1})$
$4.0(10^{-3})$	$1.51980(10^{-3})$	$1.51980(10^{-3})$	$1.51980(10^{-3})$
$5.0(10^{-3})$	$2.46652(10^{-5})$	$2.46651(10^{-5})$	$2.46651(10^{-5})$
$6.0(10^{-3})$	$1.43207(10^{-3})$	$1.43207(10^{-3})$	$1.43207(10^{-3})$
$7.0(10^{-3})$	$1.55608(10^{-1})$	$1.55607(10^{-1})$	$1.55607(10^{-1})$
$8.0(10^{-3})$	$2.02967(10^{-1})$	$2.02966(10^{-1})$	$2.02967(10^{-1})$
$9.0(10^{-3})$	$3.28064(10^{1})$	$3.28064(10^{1})$	$3.28064(10^{1})$
$1.0(10^{-2})$	$3.32985(10^{1})$	$3.32985(10^{1})$	$3.32985(10^{1})$

Table 5.47: Error for BDF with $h = 1.22070(10^{-7})$ and $k = 3$

Test Problem #4			
t	$\{1, t, t^2, t^3\}$	$\{1, e^t, e^{2t}, e^{3t}\}$	$\{1, e^{-t}, e^{-2t}, e^{-3t}\}$
$1.0(10^{-3})$	$2.09664(10^{-4})$	$2.09663(10^{-4})$	$2.09663(10^{-4})$
$2.0(10^{-3})$	$3.16189(10^{-2})$	$3.16186(10^{-2})$	$3.16186(10^{-2})$
$3.0(10^{-3})$	$6.94075(10^{-2})$	$6.94068(10^{-2})$	$6.94068(10^{-2})$
$4.0(10^{-3})$	$7.60635(10^{-4})$	$7.60626(10^{-4})$	$7.60627(10^{-4})$
$5.0(10^{-3})$	$1.25104(10^{-5})$	$1.25103(10^{-5})$	$1.25103(10^{-5})$
$6.0(10^{-3})$	$7.13281(10^{-4})$	$7.13272(10^{-4})$	$7.13272(10^{-4})$
$7.0(10^{-3})$	$7.75625(10^{-2})$	$7.75614(10^{-2})$	$7.75615(10^{-2})$
$8.0(10^{-3})$	$1.01484(10^{-1})$	$1.01482(10^{-1})$	$1.01483(10^{-1})$
$9.0(10^{-3})$	$3.28050(10^{1})$	$3.28050(10^{1})$	$3.28050(10^{1})$
$1.0(10^{-2})$	$3.32986(10^{1})$	$3.32986(10^{1})$	$3.32986(10^{1})$

Table 5.48: Error for BDF with $h = 6.10352(10^{-8})$ and $k = 3$

Based upon the error results given in Tables 5.1 thru 5.48, we now tabulate the methods which gave the least amount of error over the entire range of integration. Notice that in almost all cases either the positive or the negative exponential bases gave the least total error.

Alternatively, test problem four shows that *all* methods fail to successfully integrate the system over the desired range of integration. This suggests a future search for alternative basis functions be made.

Error Results			
k	$\{1, t, t^2 \ldots . t^{k+1}\}$	$\{1, e^t, e^{2t}, \ldots, e^{(k+1)t}\}$	$\{1, e^{-t}, e^{-2t}, \ldots, e^{-(k+1)t}\}$
1			Best
2			Best
3			Best

Table 5.49: Adams–Moulton results for test problem #1.

Error Results			
k	$\{1, t, t^2 \ldots . t^k\}$	$\{1, e^t, e^{2t}, \ldots, e^{kt}\}$	$\{1, e^{-t}, e^{-2t}, \ldots, e^{-kt}\}$
1		Best	
2	Best		
3		Best	

Table 5.50: BDF results for test problem #1.

Error Results			
k	$\{1, t, t^2 \ldots . t^{k+1}\}$	$\{1, e^t, e^{2t}, \ldots, e^{(k+1)t}\}$	$\{1, e^{-t}, e^{-2t}, \ldots, e^{-(k+1)t}\}$
1			Best
2		Best	
3			Best

Table 5.51: Adams–Moulton results for test problem #2.

Error Results			
k	$\{1, t, t^2 \ldots . t^k\}$	$\{1, e^t, e^{2t}, \ldots, e^{kt}\}$	$\{1, e^{-t}, e^{-2t}, \ldots, e^{-kt}\}$
1		Best	
2			Best
3		Best	

Table 5.52: BDF results for test problem #2.

Error Results			
k	$\{1, t, t^2 \ldots . t^{k+1}\}$	$\{1, e^t, e^{2t}, \ldots, e^{(k+1)t}\}$	$\{1, e^{-t}, e^{-2t}, \ldots, e^{-(k+1)t}\}$
1			Best
2			Best
3		Best	

Table 5.53: Adams–Moulton results for test problem #3.

Error Results			
k	$\{1, t, t^2 \ldots . t^k\}$	$\{1, e^t, e^{2t}, \ldots, e^{kt}\}$	$\{1, e^{-t}, e^{-2t}, \ldots, e^{-kt}\}$
1		Best	
2	Best		
3		Best	

Table 5.54: BDF results for test problem #3.

5.2 Recommendations

In light of the results given here, we now make several recommendations for future research.

1. Clearly the monomial basis functions are not optimal for solving all systems of ordinary differential equations. Choosing an optimal basis would depend on stability, local truncation error and computational efficiency. In the examples provided above, the savings that an algorithm could attain due to an increase in stepsize resulting from changing to exponential basis functions might very well be offset by computational inefficiency. For example, computers require considerably more cpu time to evaluate exponentials when compared to monomials. Clearly some trade off exists between computational time and stepsize.

2. In order to exploit the increase in stepsizes that are possible when changing to other basis functions, some means of automatic step control is needed. The error analysis provided in Chapter Three required a fixed stepsize on the interval $[t_{n-k}, t_n]$. Future research should be made to obtain local truncation error estimates for variable stepsizes.

3. Consider the basis functions $\{1, e^{\lambda_1 t}, \ldots, e^{\lambda_k t}\}$, where the λ_i are some unknown parameters. Assuming the Jacobian matrix of the system is available, estimates of the eigenvalues could be made. By eliminating stiff

components from the linearized system, the λ_i could be chosen accordingly.

4. As was previously discussed, suitable predictor methods must be found in order for any practical gain to be achieved. For the Adams–Moulton methods, a reasonable predictor might be the Adams–Bashforth schemes as derived from the same basis function used in the construction of the corresponding Adams–Moulton method. For the BDF methods, no immediate candidates are available.

5. Using the ideas presented in this research, extensions to second derivative schemes such as Numerov's method are possible.

6. Recall the major assumption made in the stability analysis, namely, perturbations to the method are at worst linear. In the "real" world of numerical analysis this may or may not be true. As such, a simple example was constructed to observe the effects of a nonlinear perturbation. Euler's method was applied to Riccati's equation, with the coefficients varying in the complex plane. The regions where the method converged were plotted. The resulting plots were fractals, specifically the regions were Julia sets. Clearly, the introduction of nonlinear perturbations can result in drastically different behavior when compared to linear perturbations.

7. Theoretical aspects such as whether the methods derived are consistent and convergent have not been fully resolved. For simple basis functions some minimal results were obtained, however in the most general case no proofs were constructed. The difficulty arises due to the complex nature of the coefficients α_j and β_j.

Bibliography

[1] Acton, J. R.; Squire, P. T. *Solving Equations with Physical Understanding,* Adam Hilger Ltd, c.1985.

[2] Acton, Forman S., *Numerical Methods That Work,* Harper & Row, c.1970.

[3] Aiken, Richard C., *Stiff Computation,* Oxford University Press, c.1985.

[4] Atkinson, Kendall E., *An Introduction to Numerical Analysis,* Wiley Inc., c.1978.

[5] Ayres, Frank, *Schaum's Outline of Theory and Problems of Matrices,* McGraw–Hill Inc., c.1962.

[6] Bellman, Richard; Cooke, Kenneth L., *Differential–Difference Equations,* Academic Press Inc., c.1963.

[7] Bellman, Richard, *Introduction to Matrix Analysis,* McGraw–Hill Inc., c.1970.

[8] Bellman, Richard E.; Roth, Robert S., *Methods in Approximation: Techniques for Mathematical Modeling,* D. Reidel Publishing Company, c.1986.

[9] Burden, R. L.; Faires, S. T., *Numerical Analysis,* Prindle, Weber & Schmidt, c.1978.

[10] Cheney, E. W., *Introduction to Approximation Theory,* Chelsea Publishing Company, c.1982.

[11] Conte, S. D.; de Boor, Carl, *Elementary Numerical Analysis: An Algorithmic Approach,* McGraw–Hill Inc., c.1980.

[12] Cronin, Jane, *Differential Equations: Introduction and Qualitative Theory,* Marcel Dekker Inc., c.1980.

[13] Davis, Phillip J., *Interpolation and Approximation,* Dover Publications Inc., c.1975.

[14] Gear, C. W., 'The Automatic Integration of Ordinary Differential Equations', *Comm. ACM*, **14**, 176 (1971).

[15] Gear, C. W., *Numerical Initial Value Problems in Ordinary Differential Equations,* Prentice–Hall Inc., c.1971.

[16] Gladwell, I.; Sayers, D. K., *Computational Techniques for Ordinary Differential Equations,* Academic Press Inc., c.1980.

[17] Gleick, James, *Chaos: Making a New Science,* Penguin Books, c.1987.

[18] Henrici, Peter, *Discrete Variable Methods in Ordinary Differential Equations,* Wiley Inc., c.1962.

[19] Hindmarsh, A. C., 'GEAR: Ordinary Differential Equation Solver', *Lawrence Livermore Report UCID-30001, Rev. 3,* December (1974).

[20] Hindmarsh, A. C., 'Linear Multistep Methods for Ordinary Differential Equations: Method Formulations, Stability, and the Methods of Nordsieck and Gear', *Lawrence Livermore Report UCRL-51186, Rev. 1,* March (1972).

[21] Hindmarsh, A. C., 'Construction of Mathematical Software Part III: The Control of Error in the Gear Package for Ordinary Differential Equations', *Lawrence Livermore Report UCID-30050, Part 3,* August (1972).

[22] Kreyszig, Erwin, *Introductory Functional Analysis with Applications,* Wiley Inc., c.1978.

[23] Lakshmikantham, V.; Trigiante, D., *Theory of Difference Equations: Numerical Methods and Applications,* Academic Press Inc., c.1988.

[24] Lambert, J. D., *Computational Methods in Ordinary Differential Equations,* Wiley Inc., c.1983.

[25] Lambert, J. D., 'Safe Point Methods for Separably Stiff Systems of Ordinary Differential Equations', *SIAM J. Numer. Anal.* **18**, February (1981).

[26] Lambert, J. D., 'A Stable Sequence of Steplengths for Euler's Rule Applied to Stiff Systems of Differential Equations', *Comp. & Maths. with Appls.* **12B**, (1986).

[27] Lapidus, Leon; Seinfeld, John H., *Numerical Solution of Ordinary Differential Equations,* Academic Press Inc., c.1971.

[28] Miranker, Willard L., *Numerical Methods for Stiff Equations and Singular Perturbation Problems,* D. Reidel Publishing Company, c.1981.

[29] Moon, Francis C., *Chaotic Vibrations: An Introduction for Applied Scientists and Engineers,* John Wiley & Sons, c.1987.

[30] Press, William H.; Flannery, Brian P.; Teukolsky, Saul A.; Vetterling, William T., *Numerical Recipes: The Art of Scientific Computing,* Cambridge University Press, c.1988.

[31] Spiegel, Murray R., *Schaum's Outline of Theory and Problems of Calculus of Finite Differences and Difference Equations,* McGraw–Hill Inc., c.1971.

[32] Tarasov, L. V., *Laser Physics,* MIR Publishers, c.1983.

[33] Thompson, J. M. T.; Stewart, H. B., *Nonlinear Dynamics and Chaos: Geometrical Methods for Engineers and Scientists,* John Wiley & Sons, c.1986.

Biography

Leon Arriola was born on November 5, 1957 in Paterson, New Jersey. He graduated from Idaho State University in December 1981 with a B.S. in Physics. He was awarded a Ph.D in Mathematics from Old Dominion University in December 1990.

He has coauthored two papers entitled "Vibrational Excitation of CO by Blackbody Radiation" and "A Method for Analysis of Blackbody Diatomic–Triatomic Lasers" both of which are in "Lasant Materials for Blackbody–Pumped Lasers," NASA Technical Memorandum 87616, September 1985. This research was sponsored by the High Energy Science Branch at NASA Langley Research Center, grant number NAG–1–757.

He was awarded an ICAM fellowship from June 1985 through May 1987 and has taught undergraduate mathematics from 1982 to the present. He is currently a member of the Mathematics Association of America.

His hobbies include olympic weightlifting, furniture making, model railroading and amateur telescope making.