ON VARIABLE COEFFICIENT MULTISTEP METHODS

MIN HU

On Variable Coefficient Multistep Methods

by

©Min Hu

A thesis submitted to the School of Graduate Studies in partial fulfillment of the requirements for the degree of Doctor of Philosophy

Department of Mathematics and Statistics

Memorial University of Newfoundland

April 1993

St. John's Newfoundland Canada

Abstract

In this monograph, we develop a subclass of variable coefficient multistep (VCM) methods, which is A-contractive.

We introduce a set of simplifying conditions to relate VCM methods to the Padé approximants of the exponential function $\exp(z)$. We then proceed with the construction of the arbitrary order, A-contractive, variable stepsize VCM methods. Both linearly implicit and fully implicit families are considered.

The convergence properties of VCM methods are discussed in chapter 3. We show the stiff-independent convergence for VCM methods on general nonlinear dissipative problems. We also demonstrate convergence of VCM methods when applied to singular perturbation problems with the convergence being independent of the perturbation parameter.

Finally, in chapter 4 we report on a set of numerical experiments with fourth and fifth order linearly implicit and fully implicit methods.

Acknowledgements

Being at the end of my Ph. D. study, I would like to take this opportunity to acknowledge my indebtedness to a number of people who helped and encouraged me during my doctoral program.

I would like to thank my supervisor Dr. Richard Charron for proposing and discussing the subject of this thesis, for his many helpful suggestions and careful reading of this thesis, for his support during my stay in St. John's. Through his generosity, I have had the opportunity to attend a conference and to spend some time with him in Montréal to work on the final preparation of the thesis.

I am grateful to Dr. H. Brunner for his helpful suggestions and comments. He was always freely accessible for discussions and for providing help.

I would like to thank Dr. F. Chipman and Dr. E.A.D. Foster, who are external and internal examiners respectively, for their careful reading of the thesis and for their helpful comments.

Thanks are also due to Dr. P. P. Narayanaswami, Dr. E. Jespers, who are in my supervisor committee, and Dr. S. P. Singh, for all their help and encouragement of my degree work.

I would like to thank Dr. B. Shawyer and Dr. E. Goodaire who, during my stay, both served as Head of the department of mathematics and statistics, for making the facilities of the department available for the degree work.

I am grateful to the School of Graduate Studies and the Department of Mathematics and Statistics for their supporting me with a Memorial University Fellowship and the Teaching Assistantship.

My thesis work involves a large quantity of symbolic and numerical computations which were performed on the computers of both the Department of Mathematics and Statistics and the Computing and Communication Center. So I would like to thank Mr. J. Rochester, system manager of department of mathematics and statistics, and the staff in Computing and Communication Center, for their proficient systems maintaince and for their kind help.

Contents

List of Figures

List of Tables

Chapter 1

Preliminaries

1.1 Initial value problems and the concept of stiffness

The mathematical modelling of many problems in physics, engineering, chemistry, biology etc. gives rise to initial value problems for systems of ordinary differential equations

$$y' = f(x, y), \qquad y(x_0) = y_0 \tag{1.1.1}$$

where $x \in \mathbf{R}$, $y \in \mathbf{R}^n$ and $f : \mathbf{R} \times \mathbf{R}^n \to \mathbf{R}^n$. For f continuous in x, a sufficient condition to ensure (1.1.1) has a unique solution is the following Lipschitz condition:

$$|f(x, \bar{y}) - f(x, y)| \leq L|\bar{y} - y|, \qquad \text{for all } \bar{y}, y \in \mathbf{R}^n \text{ and } |x - x_0| < \delta. \tag{1.1.2}$$

There have been many studies on the numerical methods to solve initial value problems as described above. Among all initial value problems, the class of so-called "stiff" initial value problems commands more attention from numerical analysts. Nowadays, the most popular methods for computing solutions of stiff problems are implicit BDF (Backward Difference Formula) methods (see for example, Gear (1971), Lambert 1991)), which in part involve computing solutions to nonlinear systems of equations at each step of the computation.

We will develop a subclass of variable coefficient multistep methods that deals with stiffness. For the numerical solution of (1.1.1) we shall deal with finite difference equations solved in a step-by-step fashion. Only an approximation y_n to the solution y at grid points x_n is produced. These grid points are defined by $x_n = x_{n-1} + h_n$ $(1 \leq n \leq N)$ where the numbers $h_n > 0$ are the stepsizes and $x_N = b$. If all stepsizes are equal, say $h_n = h$ $(1 \leq n \leq N)$ the grid $\{x_n\}$ is said to be uniform.

During the numerical integration errors will inadvertently be introduced. Such errors are mainly caused by replacing the differential equation with a difference equation, which account for so-called local discretization errors. Errors will also be introduced in the computation by virtue of the computer arithmetic. We should always make efforts to keep these local errors small when integrating a differential equation. For example, if we have two sequences $\{\tilde{y}_n, y_n\}$ satisfying the same difference equation for different initial values, we would like to know whether $|\tilde{y}_n - y_n|$ remains small for all n when $|\tilde{y}_0 - y_0|$ is so. Furthermore, we would like to know whether the global error $|y(x_n) - y_n|$ remains small.

We consider the following simple illustration problem from Hundsdorfer (1984).

Example 1.1.1 Consider solving

$$y' = \lambda y, \quad \lambda = -10^6, \quad y(x_0) = y_0 \neq 0$$

using Euler's method, which on $y' = f(y)$ reads

$$y_n = y_{n-1} + hf(y_{n-1}). \qquad (1 \leq n \leq N) \tag{1.1.3}$$

Assuming u_{n-1} is exact, one can easily show that the error between exact and numerical solution would be

$$y(x_n) - y_n = \frac{h^2}{2}y''(\xi_n) = \frac{h^2\lambda^2}{2}e^{-\lambda\xi_n}, \quad x_{n-1} \le \xi_n \le x_n.$$

Away from $x = x_0$ where the solution becomes quite smooth, the error term above suggests small errors even with relatively large values of h. Moreover, we already know this problem is *contractive*, that is

$$|y(x_n + h) - \bar{y}(x_n + h)| < |y(x_n) - \bar{y}(x_n)|$$

for any $h > 0$ and any two solutions $y(x_n)$, $\bar{y}(x_n)$ of the differential equation. Substituting $f(y) = \lambda y$ into (1.1.3) and solving the resulting difference equation gives

$$y_n = (1 + h\lambda)^n y_0, \qquad 1 \le n \le N$$

for the problem $y' = \lambda y$ whose exact solution is $y(t) = e^{\lambda t}y_0$. For two sequences of approximations computed with different starting values $\bar{y}_0$, y_0, we have

$$|\bar{y}_n - y_n| = |1 + h\lambda|^n|\bar{y}_0 - y_0|, \qquad 1 \le n \le N.$$

When $h\lambda$ is large, as would be the case with say $h = 10^{-3}$, then $|1 + h\lambda| \approx 10^3$. Assuming a uniform grid, $N = h^{-1} = 10^3$, and we thus have the unfavorable result

$$|\bar{y}_N - y_N| \approx 1000^{1000}|\bar{y}_0 - y_0|$$

which does not correspond with the behavior of the exact solution. For this example, Euler's method preserves the problems' contractivity property only if $0 < h < 10^{-6}$; a severe restriction in light of considerations based only on the error term.

When employing the backward Euler method, i.e.

$$y_n = y_{n-1} + hf(y_n)$$

from which $\bar{y}_n - y_n = (1 - h\lambda)^{-n}(\bar{y}_0 - y_0)$ is obtained , and therefore

$$|\bar{y}_n - y_n| \leq |\bar{y}_0 - y_0|. \qquad 1 \leq n \leq N$$

for all step sizes $h > 0$. This is a completely different behavior compared with result by the explicit Euler method. The latter compares favorably with the behavior of the exact solution.

Stiffness in a differential system is the combination of many factors, such as the maximal eigenvalue, the ratio of the maximal and minimal eigenvalues of Jacobian matrix of $f(x, y)$, the integration interval. It is difficult to give a completely satisfying definition for there are many facets to the concept of stiff differential equations. The most important common feature is that when such equations are being solved with standard numerical methods (e.g., the Adams' methods), the step size h is forced to be extremely small in order to maintain stability — and far smaller than would appear to be necessary based on a consideration of the truncation error alone. For our purposes, we employ a definition from Shampine and Gear (1979, p.2):

> " *By a stiff problem we mean one for which no solution component is unstable (no eigenvalue of the Jacobian matrix has a real part which is at all large and positive) and at least some component is very stable (at least one eigenvalue has a real part which is large and negative). Further, we will not call a problem stiff unless its solution is slowly varying with*

respect to the most negative part of the eigenvalues... . Consequently a problem may be stiff for some intervals and not for others. "

Lambert (1991, p.217-221) also gives the following characteristics for a stiff problem:

" *1) all its eigenvalues have negative real parts and the stiffness ratio is large,*

2) stability requirements, rather than those of accuracy, constrain the steplength,

3) some components of the solution decay much more rapidly than others,

4) in a given interval, the neighbouring solution curves approach the solution curve at a rate which is very large in comparison with the rate at which the solution varies in that interval. "

1.2 The one-sided Lipschitz condition and the logarithmic norm

For the stability analysis as well as for existence and uniqueness of solution, it is often assumed that the function f appearing in the right-hand side of the differential equation satisfies a Lipschitz condition (1.1.2) which implies

$$|\tilde{y}(x+\triangle x) - y(x+\triangle x)| \leq e^{L\triangle x}|\tilde{y}(x) - y(x)|, \quad \text{for } x_0 \leq x < x + \triangle x \leq \bar{x} \quad (1.2.1)$$

for any two solutions $\tilde{y}$, y of the differential equation. There exists a rather satisfactory theory by means of which one can predict how well a numerical scheme will approximate the exact solution y of (1.1.1) provided that the product $\bar{x}L$ is not too large and hL is sufficiently small.

Many results concerning the stability, convergence and solvability of numerical methods for initial value problems are based on hL small . it is clear this condition is not practical for dealing with stiffness because stiff problems usually involve large values for L. So we turn our attention to the one-sided Lipschitz condition and the notion of logarithmic matrix norm to improve our analysis. On $\mathbf{R}^n$ let $<\cdot,\cdot>$ be an inner product and $\|\cdot\|$ the corresponding inner product norm defined by $\| u \|^2 := < u, u >$. Let $M_x \subset \mathbf{R}^n$ be a convex region on which the function $f(x,y)$ can be regarded as a function of y only.

Definition 1.2.1 The function $f(x,y)$ and the system $y' = f(x,y)$ are said to satisfy a *one-sided Lipschitz condition* if

$$< f(x,y) - f(x,\bar{y}),\ y - \bar{y} > \leq \nu(x) \| y - \bar{y} \|^2 \tag{1.2.2}$$

holds for all $y, \bar{y} \in M_x$ and for $a \leq x \leq b$. The function $\nu(x)$ is called an one-sided Lipschitz constant.

It is important to note that $\nu(x)$ need not be restricted to being positive. If $f(x,y)$ satisfies a Lipschitz condition, then it satisfies a one-sided Lipschitz condition. This can be seen from

$$< f(x,y) - f(x,\bar{y}), y - \bar{y} > \leq \|f(x,y) - f(x,\bar{y})\| \cdot \|y - \bar{y}\| \leq L\|y - \bar{y}\|^2.$$

Let $y(x), \bar{y}(x), x \in [x_0, \bar{x}]$, be any two solutions of $y' = f(x,y)$ with initial values $y_0, \bar{y}_0, y_0 \neq \bar{y}_0$. We introduce the function

$$\Phi(x) := \| \bar{y}(x) - y(x) \|^2 .$$

We have

$$\dot{\Phi}(x) = 2 < \dot{\bar{y}}(x) - \dot{y}(x), \bar{y}(x) - y(x) >,$$

and when (1.2.2) holds, it follows that Φ then satisfies the differential inequality

$$\dot{\Phi}(x) \leq 2\nu(x)\Phi(x), \qquad x \in [x_0, \bar{x}].$$

Multiplication of both sides with the integrating factor

$$\eta(x) = \exp(-2\int_{x_0}^{x} \nu(\tau)\,d\tau\ )$$

gives

$$\dot{\Phi}(x)\eta(x) + \Phi(x)\dot{\eta}(x) \leq 0$$

which in turn leads to the inequality

$$\frac{d}{dx}[\Phi(x)\eta(x)] \leq 0.$$

This means that $\Phi(x)\eta(x)$ decreases monotonically for $x \in [x_0, \bar{x}]$. So we have,

$$\| \bar{y}(x_2) - y(x_2) \| \leq \exp(\int_{x_1}^{x_2} \nu(\tau)d\tau) \| \bar{y}(x_1) - y(x_1) \|, \tag{1.2.3}$$

for all x_1, x_2 satisfying $x_0 \leq x_1 \leq x_2 \leq \bar{x}$. Thus if $\nu(x) \leq 0$ on $[x_0, \bar{x}]$, we see that the true solution to $y' = f(x, y)$ bears a contractivity property with respect to the norm in use, that is, from (1.2.3) we see that with $\nu(x)$ nonpositive

$$\| \bar{y}(x_2) - y(x_2) \| \leq \| \bar{y}(x_1) - y(x_1) \| . \tag{1.2.4}$$

An initial value problem with this property is normally referred to as being *dissipative.*

The results concerning the one sided-Lipschitz condition are based on the norms derived from inner products. Dahlquist (1959) further introduced the logarithmic matrix norm which is not restricted to the inner product norms. Contractivity results can then be obtained for arbitrary norms on $\mathbf{R}^n$ and they are related closely to the one-sided Lipschitz condition.

Definition 1.2.2 The *logarithmic norm* $\mu[A]$ of a square matrix A is defined by

$$\mu[A] := \lim_{\triangle \to 0^+} \frac{\| I + \triangle A \| - 1}{\triangle} \tag{1.2.5}$$

where I is the identity matrix and $\triangle \in \mathbf{R}$.

Note that $\mu[A]$ is well defined since the above limit exists for all norms. For the norms $\| \cdot \|_1$, $\| \cdot \|_2$ and $\| \cdot \|_\infty$ on $\mathbf{R}^n$, the fomulae to compute the corresponding logarithmic norm are as following,

$$\begin{aligned}
\mu_1[A] &= \max_j \Big(a_{jj} + \sum_{i \neq j} |a_{ij}| \Big), \\
\mu_2[A] &= \lambda_{\max} \Big(\frac{A + A^T}{2} \Big), \\
\mu_\infty[A] &= \max_i \Big(a_{ii} + \sum_{j \neq i} |a_{ij}| \Big).
\end{aligned}$$

The logarithmic norm is also useful tool in studying the contractivity of two solutions. The following theorem was derived by Dahlquist (1959):

Theorem 1.2.3 *Let $\| \cdot \|$ be a given norm. Let $\nu(x)$ be a piecewise continuous function such that*

$$\mu\Big[\frac{\partial f}{\partial y}(x, y)\Big] \leq \nu(x), \qquad \text{for all } x \in [a, b],\ y \in M_x.$$

Then, for any two solutions $y(x), \bar{y}(x)$ of $y' = f(x, y)$ satisfying initial conditions $y(x_0) = \eta$, $\bar{y}(x_0) = \bar{\eta}$, $\eta \neq \bar{\eta}$,

$$\| \bar{y}(x_2) - y(x_2) \| \leq \exp\Big(\int_{x_1}^{x_2} \nu(\tau) d\tau\Big) \| \bar{y}(x_1) - y(x_1) \|,$$

for all x_1, x_2 satisfying $x_0 \leq x_1 \leq x_2 \leq \bar{x}$.

We can also see the close connection between the one-sided Lipschitz condition for inner product norms and the logarithmic matrix norm from the following lemma of Hundsdorfer (1984) (see also Hairer & Wanner (1991,p.192)).

Lemma 1.2.4 *Let $D \subset \mathbf{R}^n$ be open and convex, and let $\nu \in \mathbf{R}$. Suppose f is differentiable on D. Then*

$$Re < f(x,\tilde{y}) - f(x,y), \tilde{y} - y > \leq \nu \parallel \tilde{y} - y \parallel^2, \qquad \textit{for all } (x,\tilde{y}),(x,y) \in \mathbf{R} \times D,$$

if and only if

$$\mu[f_y(x,y)] \leq \nu, \qquad \textit{for all } (x,y) \in \mathbf{R} \times D.$$

We have seen that the one-sided Lipschitz condition or logarithmic matrix norm ensure the dissipativity of the problem $y' = f(x,y)$. The classical Lipschitz condition cannot do this. Look at the simple examples $y' = y$ and $y' = -y$, both have the same Lipschitz constant $+1$, but the solution of the second is dissipative while the first is not.

1.3 Classical linear multistep methods

The standard form of a linear multistep method is

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = h \sum_{j=0}^{k} \beta_j f_{n+j}, \quad (k \geq 1) \tag{1.3.1}$$

where $h = x_n - x_{n-1}$, $n = 1,2,\ldots,$ $f_j = f(x_j, y_j)$, $x_j = x_0 + jh$, y_n is the numerical approximation to $y(x_n)$ and α_j, β_j are constants subject to the conditions

$$\alpha_k = 1, \qquad |\alpha_0| + |\beta_0| \neq 0.$$

More precisely, (1.3.1) is called linear k-step method because y_{n+k} is computed from the data $y_n, y_{n+1}, \cdots, y_{n+k-1}$. The methods are distinguished between explicit ($\beta_k = 0$) and implicit ($\beta_k \neq 0$). An implicit scheme necessarily involves solving a nonlinear system of equations at each x_{n+k}, $n \geq 0$.

As the numerical solution of a multistep method does not depend only on the initial value problem (1.1.1) but also on the choice of the starting values $\{y_1, y_2, \ldots, y_{k-1}\}$, we introduce the following definitions following closely Hairer, Nørsett & Wanner (1987).

Definition 1.3.1 The *local error* of the multistep method (1.3.1) is defined by

$$LE := y(x_k) - y_k$$

where $y(x)$ is the exact solution of $y' = f(x, y)$, $y(x_0) = y_0$ and y_k is the numerical solution obtained from (1.3.1) by using the exact starting values $y_j = y(x_j)$ for $j = 0, 1, \cdots, k-1$.

Now associate with (1.3.1) the linear differential operator L defined by

$$L(y, x, h) := \sum_{j=0}^{k} \Big[\alpha_j y(x + jh) - h\beta_j y'(x + jh)\Big].$$

Definition 1.3.2 The multistep method (1.3.1) is said to be of order p, if one of the following two conditions is satisfied:

1) for all sufficiently regular functions $y(x)$, we have $L(y, x, h) = O(h^{p+1})$;

2) the local error of (1.3.1) is $O(h^{p+1})$ for all sufficiently regular differential equations (1.1.1).

We define the first and second characteristic polynomials of (1.3.1) by

$$\rho(\zeta) := \sum_{j=0}^{k} \alpha_j \zeta^j, \qquad \sigma(\zeta) := \sum_{j=0}^{k} \beta_j \zeta^j \tag{1.3.2}$$

where $\zeta \in \mathbb{C}$ is a dummy variable. Now the linear multistep method (1.3.1) can be written in the form

$$\rho(E) y_n = h\sigma(E) f_n$$

where E is the forward shift operator defined by

$$EF_n := F_{n+1}, \quad E^2 F_n := E(EF_n) = F_{n+2}, \quad \text{etc.}$$

Theorem 1.3.3 (Hairer & Nørsett & Wanner 1987) *The multistep method (1.3.1) is of order p, if and only if one of the following equivalent conditions is satisfied:*

1) $\displaystyle\sum_{j=0}^{k} \alpha_j = 0, \qquad \sum_{j=0}^{k} \alpha_j j^m - m \sum_{j=0}^{k} \beta_j j^{m-1} = 0 \qquad \text{for} \quad m = 1, \ldots, p.$

2) $\rho(e^h) - h\sigma(e^h) = O(h^{p+1}) \qquad \text{for} \quad h \to 0.$

3) $\displaystyle\frac{\rho(\zeta)}{\log \zeta} - \sigma(\zeta) = O((\zeta - 1)^p) \qquad \text{for} \quad \zeta \to 1.$

For a given numerical method, we are concerned about not only its convergence but also the speed of convergence. We introduce the following notation for convenience, given x and h such that $\dfrac{x - x_0}{h} = n$ is a fixed integer, denote the numerical solution:

$$y_h(x) := y_n \qquad \text{when} \quad x - x_0 = nh.$$

A minimal requirement would be that $y_h(x)$ converges to the exact solution $y(x)$ as $h \to 0$. Furthermore, when f is smooth, it is natural to expect the rate of convergence to be roughly comparable to the order of the method. Let $D = \{(x, y) | x \in$

$[x_0, \bar{x}], \| y(x) - y \| \leq b\}$ where $y(x)$ is the exact solution of (1.1.1) and b is some positive number.

Definition 1.3.4 (Convergence) The linear multistep method (1.3.1) is called *convergent*, if for all initial value problems (1.1.1) satisfying the Lipschitz condition (1.1.2) on D and f is continuous on D,

$$y(x) - y_h(x) \to 0 \qquad \text{for} \quad h \to 0, \ x \in [x_0, \bar{x}]$$

whenever the starting values satisfy

$$y(x_0 + jh) - y_h(x_0 + jh) \to 0 \qquad \text{for} \quad h \to 0, \ j = 0, 1, \ldots, k-1.$$

Method (1.3.1) is convergent of order p, if for any problem (1.1.1) with f sufficiently differentiable, there exists a positive h_0 such that

$$\| y(x) - y_h(x) \| \leq Ch^p \qquad \text{for} \quad h \leq h_0$$

whenever the starting values satisfy

$$\| y(x_0 + jh) - y_h(x_0 + jh) \| \leq C_0 h^p \qquad \text{for} \quad h \leq h_0, \ j = 0, 1, \ldots, k-1.$$

We assume that a unique solution of (1.1.1) exits on $[x_0, \bar{x}]$.

Definition 1.3.5 The multistep method (1.3.1) is said to be *consistent* if, for all initial value problems satisfying Lipschitz condition (1.1.2),

$$\lim_{h \to 0} \frac{1}{h} L(y, x, h) = 0, \qquad x = x_0 + nh.$$

Definition 1.3.6 The multistep method (1.3.1) is said to be *zero-stable* if, for all initial value problems satisfying Lipschitz condition (1.1.2), there exist constants K and h_0 such that

$$\|y_n - \tilde{y}_n\| \leq K\|y_0 - \tilde{y}_0\|$$

for all $x_0 \leq \bar{x}$ and all $h \in (0, h_0)$, where y_n, $\tilde{y}_n$ are two numerical solutions.

Dahlquist (1956) was the first to find the equivalency of consistency, zero-stable and convergence.

Theorem 1.3.7 *Necessary and sufficient conditions for the multistep method (1.3.1) to be convergent are that it be both consistent and zero-stable.*

The most popular convergent multistep methods are Adams methods and BDF methods. These methods have a long history and are efficient in integrating differential equations. Adams methods are derived through numerical integration from the identity

$$y(x_{n+k}) = y(x_{n+k-1}) + \int_{x_{n+k-1}}^{x_{n+k}} y'(t)dt.$$

Replace $y'(x)$ by $f(x, y)$ and deal with the integration term by polynomial interpolation at

$$(x_n, f_n), (x_{n+1}, f_{n+1}), \ldots, (x_{n+k-1}, f_{n+k-1}).$$

Using Newton backward difference formula, we obtain the explicit Adams-Bashforth formula:

$$y_{n+k} = y_{n+k-1} + h\sum_{j=0}^{k-1} \gamma_j \nabla^j f_{n+k-1}$$

where

$$\gamma_j = (-1)^j \int_0^1 \binom{-s}{j} ds.$$

The following implicit Adams-Moulton method

$$y_{n+k} = y_{n+k-1} + h\sum_{j=0}^{k} \gamma_j^* \nabla^j f_{n+k}$$

where

$$\gamma_j^* = (-1)^j \int_0^1 \binom{-s+1}{j} ds$$

can be obtained similarly but with the interpolation at

$$(x_n, f_n), (x_{n+1}, f_{n+1}), \ldots, (x_{n+k}, f_{n+k}).$$

For instance, taking $k = 3$ in the above, we have:

$$y_{n+3} = y_{n+2} + h\left[\frac{23}{12} f_{n+2} - \frac{16}{12} f_{n+1} + \frac{5}{12} f_n\right], \quad \text{Adams-Bashforth (explicit);}$$

and

$$y_{n+3} = y_{n+2} + h\left[\frac{9}{24} f_{n+3} + \frac{19}{24} f_{n+2} - \frac{5}{24} f_{n+1} + \frac{1}{24} f_n\right]. \quad \text{Adams-Moulton (implicit).}$$

Backward Differentiation Formulae (BDF) have the form

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = h\beta_k f_{n+k}. \tag{1.3.3}$$

Adams methods are based on numerical integration whereas BDF methods are based on numerical differentiation of a given function. Assume that the approximations $y_n, \ldots, y_{n+k-1}$ to the exact solution of (1.1.1) are known. We consider the polynomial $q(x)$ which interpolates the values $\{(x_i, y_i) | i = n, \ldots, n+k-1\}$ to derive a formula for y_{n+k}. Express this polynomial in form of backward differences, that is,

$$q(x) = q(x_n + sh) = \sum_{j=0}^{k} (-1)^j \binom{-s+1}{j} \Delta^j \, y_{n+k}$$

The unknown value y_{n+k} will now be determined in such a way that the polynomial $q(x)$ satisfies the differential equation at at least one grid-point, i.e.,

$$q'(x_{n+k-r}) = f(x_{n+k-r}, y_{n+k-r})$$

When $r = 0$, we obtain the implicit formula

$$\sum_{j=0}^{k} \frac{1}{j} \Delta^j y_{n+k} = h f_{n+k}. \tag{1.3.4}$$

For $k = 1, 2, \ldots, 6$, BDF methods are convergent and zero-stable. We list BDF coefficients as follow:

$$k = 1, \quad y_{n+1} - y_n = h f_{n+1}$$

$$k = 2, \quad \tfrac{3}{2} y_{n+2} - 2 y_{n+1} + \tfrac{1}{2} y_n = h f_{n+2}$$

$$k = 3, \quad \tfrac{11}{6} y_{n+3} - 3 y_{n+2} + \tfrac{3}{2} y_{n+1} - \tfrac{1}{3} y_n = h f_{n+3}$$

$$k = 4, \quad \tfrac{25}{12} y_{n+4} - 4 y_{n+3} + 3 y_{n+2} - \tfrac{4}{3} y_{n+1} + \tfrac{1}{4} y_n = h f_{n+4}$$

$$\begin{aligned} k = 5, \quad & \tfrac{137}{60} y_{n+5} - 5 y_{n+4} + 5 y_{n+3} - \tfrac{10}{3} y_{n+2} \\ & + \tfrac{5}{4} y_{n+1} - \tfrac{1}{5} y_n = h f_{n+5} \end{aligned}$$

$$\begin{aligned} k = 6, \quad & \tfrac{147}{60} y_{n+6} - 6 y_{n+5} + \tfrac{15}{2} y_{n+4} - \tfrac{20}{3} y_{n+3} \\ & + \tfrac{15}{4} y_{n+2} - \tfrac{6}{5} y_{n+1} + \tfrac{1}{6} y_n = h f_{n+6} \end{aligned}$$

1.4 A-stability and A-contractivity

For integrating stiff problems, one cannot be satisfied with the zero-stability and convergence which requires the product of stepsize and Lipschitz constant be kept small. In the 1950's, people began noting inefficiencies when integrating some problems despite using convergent methods. Dahlquist (1963) formalized the notion of A-stability for dealing with stiffness and provided several important contributions in this subject area.

Consider once again the linear equation

$$y' = \lambda y, \qquad Re(\lambda) \leq 0. \tag{1.4.1}$$

Apply (1.3.1) to (1.4.1), we get a difference equation

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = h\lambda \sum_{j=0}^{k} \beta_j y_{n+j}, \tag{1.4.2}$$

whose characteristic equation is

$$\sigma(\zeta) - \mu\rho(\zeta) = 0, \quad \mu := h\lambda$$

So the stability region of (1.3.1) can be regarded as the following set in $\mathbf{C}$

$$S_1 := \{\mu \in \mathbf{C} : |\zeta(\mu)| < 1\}$$

where $\zeta(\mu)$ are the roots of the characteristic equation. The solution of the difference equation (1.4.2) is bounded when $|\zeta(\mu)| < 1$.

Generally, we can define the stability region of a numerical method to be the set in $\mathbf{C}$

$$S := \{\mu \in \mathbf{C} : \text{ the numerical approximation to (1.4.1) are bounded for arbitrary } n\}$$

where $\mu = h\lambda$, λ is a complex constant and h is the stepsize used to get the approximation.

Definition 1.4.1 (A-stability) A numerical method is said to be *A-stable* if all numerical approximations are bounded for arbitrary n when it is applied to the test equation (1.4.1) with a fixed positive h and a (complex) constant λ with a negative real part, i.e., $S \supset \mathbf{C}_-$.

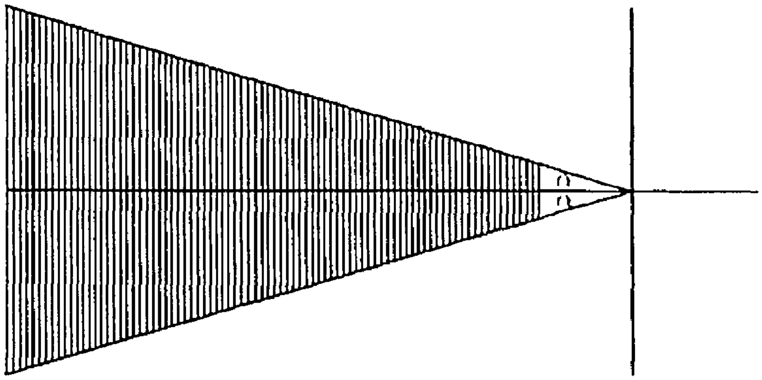

Figure 1.4.1: A(α)-stability region on $h\lambda$ plane

Theorem 1.4.2 (Dahlquist barrier) *An A-stable linear multistep method (1.3.1) must be of order $p \leq 2$. Furthermore, an A-stable multistep method cannot be explicit.*

This restrictive result indicates that if linear multistep methods are to be used, the requirement of A-stability has to be relaxed. Widlund (1967) defined A(α)-stability as follows:

Definition 1.4.3 (A(α)-stability) A numerical method is called *$A(\alpha)$-stable*, for some $0 < \alpha < \frac{\pi}{2}$, if all numerical approximations to (1.4.1) are bounded for arbitrary n with h fixed and λ satisfying $|arg(-\lambda)| < \alpha$, $|\lambda| \neq 0$. i.e.,

$$S \supset S_\alpha = \{\mu : |arg(-\mu)| < \alpha\}.$$

A typical region of A(α)-stability is shown in Figure 1.4.1. When $\alpha = \frac{\pi}{2}$, we have A-stability.

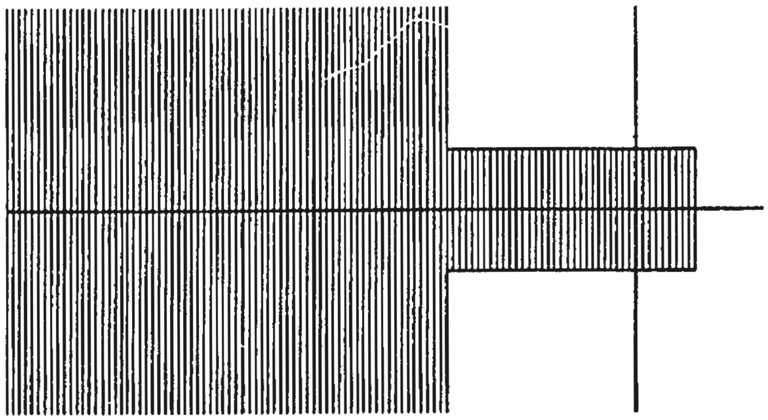

Figure 1.4.2: Stiff-stability region on $h\lambda$ plane

An alternative weakening of A-stability was introduced by Gear (1969):

Definition 1.4.4 (Stiff-stability) The method (1.3.1) is called *stiffly-stable* if $S \supset \{\mu : Re(\mu) < -D\}$ for some $D > 0$ and that the method is accurate in a rectangle

$$-D \leq \mathrm{Re}(\mu) \leq a, \qquad -\theta \leq \mathrm{Im}(\mu) \leq \theta$$

for some θ, $a > 0$.

A typical region of stiff-stability is shown in Figure 1.4.2.

The region of absolute stability of the Adams-Moulton methods, though reasonably sized, turn out to be inadequate to cope with the problem of stiffness, where stability rather than accuracy is paramount. A class of implicit linear k-step methods with regions of absolute stability large enough to make them relevant to the problem of stiffness is the class of Backward Differentiation Formulae (1.3.3). BDF methods are widely used for the integration of stiff differential equations. They

Table 1.4.1: Angles of A(α)-stability for BDF methods

k	1	2	3	4	5	6
α	90°	90°	88°	73°	52°	19°

were introduced by Curtiss and Hirschfelder (1952). For $k = 1, 2$, BDF methods are A-stable and for $3 \leq k \leq 6$, BDF methods are A(α)-stable and the corresponding α values are shown in Table 1.4.1.

Nevanlinna & Liniger (1978, 1979) point out however that in the context of a variable step length method, A-contractivity may be a more appropriate property than A-stability. Let

$$Y_m := (y_{m+k-1}, y_{m+k-2}, \ldots, y_m)^T.$$

Definition 1.4.5 A multistep method is called A-contractive if

$$\| Y_{n+1} \| \leq \| Y_n \|, \qquad \text{for all} \quad n \geq 0$$

when the method is applied to test equation (1.4.1).

Clearly, contractivity of a method at z implies stability at z; if a method is A-contractive, then, by induction, $\| y_n \| \leq \| y_0 \|$ for all n, i.e., the discrete solutions of the test equation computed by such a method are globally non-increasing whereas those generated by a stable method which is not contractive *may* grow, boundedly. In the variable step size case, stability is more difficult to characterize. Stability is a global property, while contractivity is a local property. So contractivity results as opposed to A-stability results are easier to generalize to time-dependent and nonlinear systems, or to discrete solutions computed with variable steps. By virtue of Dahlquist's order barrier theorem, there exist no linear multistep method of order $p \geq 2$ which are A-contractive and an explicit linear multistep method cannot be A-contractive.

1.5 Variable stepsize multistep methods

For reasons of efficiency one needs to be able to change the stepsize as the integration proceeds. There are two basic methods to do this. The first is to interpolate the previously calculated back points and use this to determine the new back points at a new uniform grid spacing. The main problem with this method is that frequent changes in the stepsize may cause instability in the calculated solution. The second method is to allow the stepsize to vary and maintain the proper order by adjusting the coefficients in the multistep method. This method is generally superior to the first method in terms of stability. There is no reason to expect those important results (such as stability, convergence) derived for fixed stepsize, still to hold when a stepsize change is in effect. So we need to reconsider the stability and convergence properties of a method during the step size changing.

Gear & Tu (1974) study convergence and stability of variable stepsize multistep methods. In general, let $h_j = x_{n+j} - x_{n+j-1}$, $j = 1, 2, \ldots, k$. Now the coefficients α_i, β_i of multistep methods (1.3.1) depend on h_j. For high order methods, the coefficients become complicated and difficult to analyze, since k is large. So they consider simultaneously another strategy for variable stepsize multistep methods. Namely, for k step methods, keep h fixed for at least $k-1$ consecutive steps. Let

$r := h_2/h_1$ and for $u \in \{1, \ldots, k-1\}$, let

$$x_{n+j} - x_{n+j-1} = h_1, \quad j = 1, \ldots, k-u;$$

$$x_{n+j} - x_{n+j-1} = h_2, \quad j = k-u+1, \ldots, k.$$

Results by Gear & Tu (1974) and Gear & Watanabe (1974) show that a variable step-size variable order algorithm based on Adams-Bashforth-Moulton (ABM) method with step-changing achieved by a variable coefficient technique is always zero-stable and convergent. Calvo & Lisbona & Montijano (1987) obtained zero-stability under the condition the step ratio $r \leq r_k^*$ for k-order BDF method with

$$r_2^* = 3.0, \; r_3^* = 2.781, \; r_4^* = 1.971, \; r_5^* = 1.681, \; r_6^* = 1.312.$$

These bounds are for keeping stepsize fixed at least k steps. The bounds are much smaller when stepsize changes are allowed at every step. Obtaining an upper bound r for step ratios of zero-stable and convergence methods is the underlying idea in Gear & Tu (1974), Grigorieff (1983) and Skeel & Jackson (1983).

Moreover, Rockswold (1988) points out that the BDF methods do not necessarily remain zero-stable (even for a fixed order) when the stepsize is varied. When changing the stepsize, the region of absolute stability of a BDF method decreases according to a rule dependent on stepsize ratios h_n/h_{n-1}. He gives the $A(\alpha)$-stability region with varying r for BDF and so-called α-type methods (see table 1.5.1).

1.6 Results about Padé approximations and matrix functions

In the analysis of stability and contractivity of numerical methods for ODE's, the Padé approximation to the exponential function e^z plays a pivotal role. A Padé ap-

Table 1.5.1: Angles of A(α)-stability for variable stepsize BDF methods

BDF -2	r	0.5	1	1.5	2	2.5
	α	90°	90°	76°	52°	–
BDF -3	r	0.5	1	1.5	2	2.5
	α	90°	86°	26°	–	–
BDF -4	r	0.5	1	1.1	1.15	2
	α	90°	73°	55°	42°	–

A dash means no positive value for α. $r = h_2/h_1$.

proximant to a function f is defined by a rational function $[m/n](z) := P_m(z)/Q_n(z)$, where P_m and Q_n are polynomials of order m and n respectively, satisfying

$$\frac{P_m(z)}{Q_n(z)} - f(z) = O(z^{m+n+1}).$$

The Padé approximantions to the exponential function e^z is well-known and bears a good number of important properties. The following explicit representation is in Perron (1913). More accessible references are Baker & Graves-Morris (1981, p.8-14) and Butcher (1987, p.75).

Lemma 1.6.1 *The $[m/n]$ member of the Padé table for the exponential function e^z is given by P_m/Q_n, where*

$$\begin{aligned} P_m(z) &= \sum_{i=0}^{m} \frac{m!(m+n-i)!}{(m+n)!i!(m-i)!} z^i, \\ Q_n(z) &= \sum_{i=0}^{n} \frac{n!(m+n-i)!}{(m+n)!i!(n-i)!} (-z)^i. \end{aligned}$$

In the stability analysis of most numerical methods for initial value problems applied to the linear test equation $y' = Ay$, where A is an $n \times n$ matrix, one derives recursion relations of the form

$$\phi_0(hA)y_{n+k} = \phi_1(hA)y_{n+k-1} + \ldots + \phi_k(hA)y_n$$

where each $\phi_i(z)$ is a polynomial. The size of the ratios $\phi_0(hA)^{-1}\phi_i(hA)$ becomes important and these rational functions furthermore tend to be intrinsically linked to rational approximants of the natural exponential. We give the following two results pertaining to this issue and these will later play an important role in our analysis.

Lemma 1.6.2 (Wanner, Hairer and Nørsett, 1978) *The $[m/n]$ Padé approximants of e^z with $n-2 \leq m \leq n$ are strictly bounded by 1 in modulus for all $z \in \mathbb{C}_-$ with $Re(z) < 0$.*

The A-stability analysis for systems is usually based on the transformation of the Jacobian $J = \partial f/\partial y$ to diagonal form. For large dimensional systems, however, the matrix which performs this transformation may be badly conditioned and destroy all the nice estimations which have been obtained, that is, in the study of A-contractivity one simply cannot diagonalize the system under consideration and expect that results applying to scalar problems will automatically transfer over to the higher dimensional case. For instance, we consider

$$R(z) = \frac{1 + z/2}{1 - z/2} \qquad \text{and} \qquad A = \begin{bmatrix} -2 & 10 \\ 0 & -4 \end{bmatrix}.$$

Note that $|R(z)| \leq 1$ for all $z \in \mathbb{C}_-$ and that A is diagonalizable, has negative eigenvalues, but $\|R(A)\|_p \geq 10/6$ for all p satisfying $1 \leq p \leq \infty$. Thus we need to consider the stability function directly in matrix norm. Let $\|\cdot\|_2$ be the Euclidean norm and $<\cdot,\cdot>$ be the corresponding scalar product.

Lemma 1.6.3 (von Neumann, 1951) *Let the rational function $W(z), z \in \mathbf{C}$ be bounded for $Re(z) \leq \nu$.*

1) Assume the matrix A satisfies

$$Re < v, Av > \leq 0, \qquad \text{for all} \quad v \in \mathbf{C}^n$$

Then in the matrix norm corresponding to the scalar product we have

$$\| W(A) \|_2 \leq \sup_{Re(z)\leq 0} |W(z)|.$$

2) Assume the matrix A satisfies

$$Re < v, Av > \leq \nu \| v \|_2^2, \qquad \text{for all} \quad v \in \mathbf{C}^n$$

Then

$$\| W(A) \|_2 \leq \sup_{Re(z)\leq \nu} |W(z)|.$$

As a direct consequence, we have the important result:

Corollary 1.6.4 *The $[m/n]$ Padé approximants to the matrix exponential e^A with $n-2 \leq m \leq n$ and $\mu_2[A] \leq 0$ satisfy*

$$\|Q_n(A)^{-1} P_m(A)\|_2 \leq 1.$$

This result unfortunately cannot be extended to any other norm. The study on contractivity of matrix functions in general norms has been carried out mainly by Spijker (1983, 1985, 1987). Spijker (1983) proved the order of A-contractive numerical methods (linear multistep methods, Runge-Kutta methods, Rosenbrock methods) cannot exceed $p = 1$, when applied to the system $y' = Ay$ with arbitrary norms. Studies on the so-called threshold factor which describe the size of the contractivity region allow the comparison between methods with order $p > 1$ possible (see Hairer & Wanner (1991)).

where $b_j^{(s)} := 0$ for each j. A method of class (2.1.1) is said to be linearly implicit if $b_k^{(i)} = 0$, $i = 0, 1, \ldots, s-1$, otherwise is said to be fully implicit. Fully implicit methods require the solution of a non-linear system at each step, while for a linearly implicit method the system of equations to be solved is linear.

A-, A(α)- and stiff-stability of VCM methods are investigated by Lambert & Sigurdsson (1972) and Sanz-Serna (1981) by applying (2.1.1) to the test equation $y' = Ay$, A an $m \times m$ matrix with all its eigenvalues in the left half plane and with $Q_n = A$. When dealing with stiffness, VCM methods enjoy the following potential advantages:

1) For any $p \geq 1$, there exist A-stable VCM methods with order p. This is in sharp contrast to the situation with linear multistep methods,

2) furthermore such high order A-stable methods can be found even if we require them to be linearly implicit and therefore avoid the expensive Newton iteration necessitated by implicit methods,

3) the order of the method, being independent of the choice of Q_n, does not suffer if Q_n is a poor approximation to the Jacobian of the initial value problem.

We remark however that a poor approximation to the Jacobian does affect the stability properties of the method. Sanz-Serna (1981) summarizes the following three interesting theorems:

Theorem 2.1.1 (An order barrier) *A convergent, A-stable VCM method has order* $p \leq 2s$.

Theorem 2.1.2 *Given a convergent linear k-step method of order* $p \leq k+1$, *there exist convergent, linearly implicit VCM methods with the same order and stability* *region with* $s = 1$ *and at most* $k+1$ *steps.*

Theorem 2.1.3 *Given a convergent linear k-step method of order k (such as the BDF methods,* $k \leq 6$*), there exists a convergent, linearly implicit VCM method with* $s = 1$*, and the same step number and order, such that both methods generate the same numerical solution when applied to the test system*

$$y' = Ay,$$

when Q_n *is chosen to be* $-A$*. (And hence they have the same region of absolute stability.)*

Another interpretation of VCM methods presents itself if we simply gather the terms in (2.1.1) in a different way as

$$\sum_{i=0}^{s-1} h^i Q_n^i \sum_{j=0}^{k} [a_j^{(i)} y_{n+j} - h b_j^{(i)} f_{n+j}] + \sum_{j=0}^{k} a_j^{(s)} h^s Q_n^s y_{n+j} = 0. \qquad (2.1.3)$$

Thus a VCM method can be interpreted as a combination of classical linear multistep methods. If the VCM method has order p, then the linear multistep method

$$\sum_{j=0}^{k} [a_j^{(i)} y_{n+j} - h b_j^{(i)} f_{n+j}]$$

has order not less than $p - i$. Combinations such as (2.1.3) are christened *blended linear multistep methods* by Skeel & Kong (1977), who develop a variable stepsize variable order (VSVO) algorithm based on blends of the Adams-Moulton and BDF methods with $s = 1$.

The idea to put the Jacobian directly into the coefficients of a numerical method was first proposed by Rosenbrock (1963) in the context of Runge-Kutta methods.

Rosenbrock methods have been extensively developed in recent years, and various forms have been studied. One can regard Rosenbrock methods as either a modification of an explicit Runge-Kutta method or a linearization of a semi-implicit Runge-Kutta method. Rosenbrock methods are linearly implicit and A-stable (or nearly A-stable); methods of order up to 6 have been constructed (Kaps & Wanner, 1981). Convergence results and application to singular perturbation problems can be found in Hairer & Wanner (1991).

There is a short summary of VCM methods in Lambert (1991, p.253-254). The A-stability properties of VCM methods with fixed step length have been studied in Lambert & Sigurdsson (1972) and Sanz-Serna (1981). However, there are to date no studies on contractivity, nor on stiff-independent convergence, nor on variable stepsize formulation of VCM methods. It is necessary in practice to work with a variable stepsize formulation. As stated earlier in section 1.5, the results obtained by Gear & Tu (1974) and Rockswold (1988) indicate that the for BDF methods the stepsize ratio $r = h_{n+1}/h_n$ are restricted to a value near 1 in order to maintain zero-stability and A(α)-stability of corresponding fixed stepsize formulae. This is undesirable in the context of a stiff problem.

2.2 Simplifying conditions and contractivity function

For the balance of this monograph, we use the following strategy for varying stepsizes: we restrict the exposition to situations where there are two stepsizes in use within the range of steps covered by the k-step formula. Without the additional use of interpolation formulae, this therefore allows for a step-size change to take place after every k successive steps, which is practical since we should avoid frequent stepsize changes in order to not incur excessive computing effort. For formula (2.1.1) with variable stepsize, let u be any integer between 1 and $k-1$, it becomes

$$\sum_{j=0}^{k-u}\left[\sum_{i=0}^{s} a_j^{(i)} h^i Q_n^i\right] y_{n+j} + \sum_{j=k-u+1}^{k}\left[\sum_{i=0}^{s} a_j^{(i)} (rh)^i Q_n^i\right] y_{n+j}$$
$$= h\sum_{j=0}^{k-u}\left[\sum_{i=0}^{s} b_j^{(i)} h^i Q_n^i\right] f_{n+j} + rh\sum_{j=k-u+1}^{k}\left[\sum_{i=0}^{s} b_j^{(i)} (rh)^i Q_n^i\right] f_{n+j}$$

where

$$\begin{aligned} x_{n+j} - x_{n+j-1} &= h, \qquad j = 1,\ldots,k-u; \\ x_{n+j} - x_{n+j-1} &= rh, \qquad j = k-u+1,\ldots,k. \end{aligned}$$

We now note that this equation can be rewritten in the form (2.1.1) with $h_2 := rh$ in place of h if the $a_j^{(i)}$ and the $b_j^{(i)}$ coefficients are redefined to include scale factors consisting of reciprocal powers of r , i.e. $a_j^{(i)}$ becomes $\frac{a_j^{(i)}}{r^i}$ and $b_j^{(i)}$ becomes $\frac{b_j^{(i)}}{r^{i+1}}$ for $j = 0, 1, \ldots, k-u$ and all i. Now the variable stepsize VCM can be written as

$$\sum_{j=0}^{k}\left[\sum_{i=0}^{s} a_j^{(i)} h_2^i Q_n^i\right] y_{n+j} = h_2\sum_{j=0}^{k}\left[\sum_{i=0}^{s-1} b_j^{(i)} h_2^i Q_n^i\right] f_{n+j}. \tag{2.2.1}$$

When $r = 1$, $h_2 = h$, this equation reduce to (2.1.1). Expand y_{n+j}, f_{n+j} about $x = x_{n+k-u}$, we then obtain the order conditions for (2.2.1) to be of order p

$$\sum_{j=0}^{k} a_j^{(i)} = 0, \qquad i = 0, 1, \ldots, \min(p, s), \tag{2.2.2}$$

$$\frac{1}{m!}\left[\sum_{j=0}^{k-u}(j-k+u)^m a_j^{(i)} + r^m\sum_{j=k-u+1}^{k}(j-k+u)^m a_j^{(i)}\right]$$
$$= \frac{r}{(m-1)!}\left[\sum_{j=0}^{k-u}(j-k+u)^{m-1} b_j^{(i)} + r^{m-1}\sum_{j=k-u+1}^{k}(j-k+u)^{m-1} b_j^{(i)}\right], \tag{2.2.3}$$

$m = 1, 2, \ldots, p-i, \quad i = 0, 1, \ldots, \min(p, s).$

For convenience, define $b_j^{(s)} := 0$ and $b_j^{(-1)} := 0$ for each j.

Definition 2.2.1 A VCM method is of type (k, p, q) if it is k-step, order p with $s = q$.

From order conditions (2.2.2) and (2.2.3), we can see the coefficients $\{a_j^{(i)}, b_j^{(i)}\}$ depend on r and u. For $r = 1$, we can choose any integer $0 \le u \le k$ to get the fixed stepsize formula. When $r \neq 1$, this means $h_2 \neq h_1$, we need $k - 1$ steps to complete the stepsize change from h_1 to h_2, so we need $k - 1$ sets of coefficients with respect to $u = 1, 2, \ldots, k - 1$. For given u, the coefficients will only depend on r. As we will see shortly, the order conditions to be imposed on the VCM methods leaves too many degrees of freedom for our analysis. With a view to rigorous contractivity properties, we add on the following restriction to the VCM methods:

$$\begin{aligned} a_j^{(0)} &= 0 & & j = 0, 1, \ldots, k-2, \\ a_j^{(i)} &= b_j^{(i-1)} & & j = 0, 1, \ldots, k-2, \quad i = 1, 2, \ldots, s. \end{aligned} \tag{2.2.4}$$

With these, a VCM method applied to the test equation (1.4.1) with $Q_n = \lambda$ for all n, gives the relation

$$\left[\sum_{i=0}^{s}(a_k^{(i)} - b_k^{(i-1)})h^i\lambda^i\right] y_{n+k} + \left[\sum_{i=0}^{s}(a_{k-1}^{(i)} - b_{k-1}^{(i-1)})h^i\lambda^i\right] y_{n+k-1} = 0.$$

Thus, locally on $[x_n, x_{n+k}]$, contractivity is assured if the contractivity function

$$R(z) := -\frac{\sum_{i=0}^{s}[a_{k-1}^{(i)} - b_{k-1}^{(i-1)}]z^i}{\sum_{i=0}^{s}[a_k^{(i)} - b_k^{(i-1)}]z^i}$$

with $z := \lambda h$ satisfies $|R(z)| \le 1$, and this same condition assures $\|Y_{n+1}\| \le \|Y_n\|$ for $n \ge k - 1$. In practice, one uses a succession of low order schemes to generate the approximations to the solution over $[x_0, x_k]$ that are needed to start a k-step formula and if they are all A-contractive, then the condition $|R(z)| \leq 1$ will, for all practical purposes, ensure the formula indeed A-contractive. We introduce the following assumption.

Assumption A: *the initial data valu.* $\{y_1, y_2, \ldots, y_{n+k-1}\}$ *are generated using A-contractive methods.*

Throughout this monograph, it is always assumed that assumption A holds.

For scalar test problem (1.4.1) and the linear system

$$y' = Ay, \quad A \in C^{m \times m} \tag{2.2.5}$$

with matrix A being normal, the contractivity results hold for all norms. When considering a general matrix A which may not be normal, the contractivity results depend on the famous theorem of von Neumann (lemma 1.6.3). However, the theorem only holds for the Euclidean norm with $< \cdot, \cdot >$ denoting the corresponding inner product. So the contractivity results we derive can be extended to the more general linear problem (2.2.5) provided Re $< y, Ay > \leq 0 \ \forall y \in C^m$ in Euclidean norm. For other norms the extension to (2.2.5) of the A-contractivity results we derive in this chapter is not possible.

Although the matrix Q_n in (2.2.1) is independent of the order conditions, we will set Q_n to be the Jacobian of $f(x, y)$ computed at (x_{n+k-1}, y_{n+k-1}).

The case with $s = 1$ needs to be treated separately. By employing a Taylor series expansion about the point $x = x_n$ one can easily verify that the one-step method

$$(-1 + \alpha h Q_n) y_n + (1 - \alpha h Q_n) y_{n+1} = h f_n \tag{2.2.6}$$

is generally of order one, and is of order two in the special instance when $\alpha =$ 0.5, (1.1.1) is autonomous and $Q_n = f_y(y_n)$. Furthermore, method (2.2.6) is A-contractive so long as the real parameter $\alpha \geq \frac{1}{2}$. In fact, when Q_n is the Jacobian of $f(x, y)$ (as we've chosen it) then (2.2.6) corresponds to a one-stage Rosenbrock method whose nonlinear stability characteristics have been well-studied (e.g. Hundsdorfer, 1981).

2.3 Existence of A-contractive variable stepsize VCM methods

In this section, we will prove there indeed exist arbitrary order, variable stepsize, fully implicit, and linearly implicit VCM methods and give a constructive proof which gives the procedure for computing the corresponding coefficients efficiently. First define for any natural number $\ell \leq q$

$$\alpha_i = (-1)^i \frac{q!(\ell + q - i)!}{(\ell + q)!i!(q - i)!}, \quad i = 0, \dots, q \tag{2.3.1}$$

$$\beta_i = \begin{cases} -\dfrac{\ell!(\ell + q - i)!}{(\ell + q)!i!(\ell - i)!} & i = 0, \dots, \ell, \\ 0 & i = \ell + 1, \ell + 2, \dots, q. \end{cases} \tag{2.3.2}$$

Lemma 2.3.1 *For any real number u the equations*

$$\sum_{\theta=0}^{q} \frac{1}{(m+\theta)!}[u^{m+\theta}\alpha_{q-\theta} + (u-1)^{m+\theta}\beta_{q-\theta}] = 0, \quad m = 1, 2, \dots, \ell, \tag{2.3.3}$$

$$\sum_{\theta=0}^{i} \frac{1}{\theta!}[u^{\theta}\alpha_{i-\theta} + (u-1)^{\theta}\beta_{i-\theta}] = 0, \quad i = 0, 1, 2, \dots, q, \tag{2.3.4}$$

hold if α_i, β_i $(i = 0, 1, \dots, q)$ satisfy (2.3.1)-(2.3.2).

Proof. Let $u = 1 + v$ and rewrite (2.3.3) as

$$\sum_{\theta=0}^{q}\sum_{j=0}^{m+\theta}\binom{m+\theta}{j}\frac{\alpha_{q-\theta}}{(m+\theta)!}v^j+\sum_{\theta=0}^{q}\frac{\beta_{q-\theta}}{(m+\theta)!}v^{m+\theta}=0,\quad m=1,2,\ldots,\ell.$$

Comparing coefficients of equal powers of v, we have

$$\sum_{\theta=0}^{q}\binom{m+\theta}{j}\frac{\alpha_{q-\theta}}{(m+\theta)!}=0,\qquad \begin{array}{rcl} j &=& 0,1,\ldots,m-1 \\ m &=& 1,2,\ldots,\ell \end{array} \tag{2.3.5}$$

and

$$\sum_{\theta=j}^{q}\binom{m+\theta}{m+j}\frac{\alpha_{q-\theta}}{(m+\theta)!}+\frac{\beta_{q-j}}{(m+j)!}=0,\qquad \begin{array}{rcl} j &=& 0,1,\ldots,q \\ m &=& 1,2,\ldots,\ell \end{array} \tag{2.3.6}$$

Similarly from (2.3.4) we have

$$\sum_{\theta=0}^{i}\sum_{j=0}^{\theta}\binom{\theta}{j}\frac{\alpha_{i-\theta}}{\theta!}v^j+\sum_{\theta=0}^{i}\frac{\beta_{i-\theta}}{\theta!}v^\theta=0,\quad i=0,1,\ldots,q$$

so that comparing coefficients of equal powers of v we get

$$\sum_{\theta=0}^{i}\binom{\theta}{j}\frac{\alpha_{i-\theta}}{\theta!}+\frac{\beta_{i-j}}{j!}=0,\qquad \begin{array}{rcl} j &=& 0,1,\ldots,i \\ i &=& 1,\ldots,q \end{array} \tag{2.3.7}$$

Now it is clear that the lemma holds if equations (2.3.5), (2.3.6) and (2.3.7) hold. The left hand side of (2.3.5) becomes

$$\begin{aligned}\sum_{\theta=0}^{q}\binom{m+\theta}{j}\frac{\alpha_{q-\theta}}{(m+\theta)!} &= \sum_{\theta=0}^{q}\binom{m+\theta}{j}\frac{(-1)^{q-\theta}q!(\ell+\theta)!}{(m+\theta)!(\ell+q)!(q-\theta)!\theta!} \\ &= \frac{(\ell-m+j)!}{j!(\ell+q)!}\sum_{\theta=0}^{q}(-1)^{q-\theta}\binom{q}{q-\theta}\binom{\ell+\theta}{m+\theta-j} \\ &= \frac{(-1)^{q+m-j}(\ell-m+j)!}{j!(\ell+q)!}\sum_{\theta=0}^{q}\binom{q}{q-\theta}\binom{m-j-\ell-1}{m-j+\theta}.\end{aligned}$$

Because $q-\ell\ge 0$, $m-j-1\ge 0$, therefore

$$\sum_{\theta=0}^{q}\binom{q}{q-\theta}\binom{m-j-\ell-1}{m-j+\theta}=\binom{q+m-j-\ell-1}{q+m-j}=0.$$

This means (2.3.5) holds. Likewise,

$$\begin{aligned}
\sum_{\theta=j}^{q}\binom{m+\theta}{m+j}\frac{\alpha_{q-\theta}}{(m+\theta)!} &= \frac{(\ell+j)!}{(m+j)!(\ell+q)!}\sum_{\theta=j}^{q}(-1)^{q-\theta}\binom{q}{q-\theta}\binom{\ell+\theta}{\theta-j} \\
&= \frac{(-1)^{q-j}(\ell+j)!}{(m+j)!(\ell+q)!}\sum_{\theta=0}^{q-j}\binom{q}{q-\theta-j}\binom{-j-\ell-1}{\theta} \\
&= \frac{(-1)^{q-j}(\ell+j)!}{(m+j)!(\ell+q)!}\binom{q-j-\ell-1}{q-j} \\
&= \frac{(-1)^{q-j}(\ell+j)!}{(m+j)!(\ell+q)!}(-1)^{q-j}\binom{\ell}{q-j} \\
&= \begin{cases} 0 & \text{if } \; q-j>\ell \\ \dfrac{(\ell+j)!\ell!}{(m+j)!(\ell+q)!(q-j)!(\ell-q+j)!} & \text{if } \; q-j\leq\ell \end{cases} \\
&= -\frac{\beta_{q-j}}{(m+j)!}.
\end{aligned}$$

This proves equation (2.3.6) holds. Furthermore,

$$\begin{aligned}
\sum_{\theta=0}^{i}\binom{\theta}{j}\frac{\alpha_{i-\theta}}{\theta!} &= \frac{(-1)^{i-j}(\ell+q-i+j)!}{j!(\ell+q)!}\sum_{\theta=0}^{i}\binom{q}{i-\theta}\binom{i-\ell-q-j-1}{\theta-j} \\
&= \frac{(-1)^{i-j}(\ell+q-i+j)!}{j!(\ell+q)!}\binom{i-\ell-j+1}{i-j} \\
&= \frac{(-1)^{i-j}(\ell+q-i+j)!}{j!(\ell+q)!}(-1)^{i-j}\binom{\ell}{i-j} \\
&= \begin{cases} 0 & \text{if } \; i-j>\ell \\ \dfrac{(\ell+q-i+j)!\ell!}{j!(\ell+q)!(i-j)!(\ell-i+j)!} & \text{if } \; i-j\leq\ell \end{cases} \\
&= -\frac{\beta_{i-j}}{j!}.
\end{aligned}$$

This is just equation (2.3.7). □

We are now in the position to establish our main results. Hereafter FIM and LIM mean fully implicit and linearly implicit methods respectively.

Theorem 2.3.2 *For any integer $q \geq 2$ there are A-contractive, variable stepsize FIM methods (2.2.1) of type $(2q-2, 2q-2, q)$, $(2q-1, 2q-1, q)$, $(2q, 2q, q)$. Their contractivity functions $R(z)$ are respectively the $[q-2/q]$, $[q-1/q]$ and $[q/q]$ Padé approximants of e^z. The coefficients depend on the step changing ratio $r = h_2/h_1$ and each method has $q(q+1)/2$ degree of freedom.*

Proof. In each of the three cases, let k denote both order and number of steps in the method, $k \in \{2q-2, 2q-1, 2q\}$. In addition to the simplifying conditions (2.2.4) let

$$\begin{aligned} a_k^{(i)} - b_k^{(i-1)} &= \alpha_i, \\ a_{k-1}^{(i)} - b_{k-1}^{(i-1)} &= \beta_i, \end{aligned} \qquad i = 0, 1, \ldots, q. \tag{2.3.8}$$

with $\ell = k - q$. From known formulae (see for instance, Butcher (1987), p.75), it is clear that $R(z)$ resulting from the selection (2.3.8) equals the $[\ell/q]$ Padé approximant of e^z from which A-contractivity of the resulting method (2.2.1) is an immediate consequence since $q-2 \leq \ell \leq q$. There remains to determine the terms $\{a_j^{(i)}, b_j^{(i)}\}$ not already specified and to prove that the order conditions (2.2.2)-(2.2.3) are satisfied by this selection.

For $i = 0$, $\{a_j^{(0)}\}$ are determined by equations (2.2.4) and (2.3.8), i.e., $a_j^{(0)} = 0$, for $j = 0, 1, \ldots, k-2$ and $a_{k-1}^{(0)} = \beta_0 = -1, a_k^{(0)} = \alpha_0 = 1$ (the latter is sometimes referred as normalization condition). Thus (2.2.3) represents a full rank linear Vandermonde system with k equations for the $k+1$ unknowns $\{b_j^{(0)}\}_{j=0}^k$. Under this condition, a solution therefore exists with 1 degree of freedom.

Now for $i = 1$, $\{a_j^{(1)}\}_{j=0}^{k-2}$, $a_{k-1}^{(1)}$ and $a_k^{(1)}$ are determined by (2.2.4) and (2.3.8) respectively. Equations (2.2.3) then specifies a full rank linear Vandermonde system with $k-1$ equations for the $k+1$ unknowns $\{b_j^{(1)}\}_{j=0}^{k}$ for which a solution with 2 degrees of freedom exists.

For $i = 2, \ldots, q-1$, we proceed in a completely analogous manner solving for $\{a_j^{(i)}, b_j^{(i)}\}_{j=0}^{k}$ with $i+1$ degrees of freedom each time. For $i = q$, we only need to determine $\{a_j^{(q)}\}_{j=0}^{k}$ according to (2.2.4) and (2.3.8) because we already know $b_j^{(q)} := 0$. We have so far determined $\{a_j^{(i)}, b_j^{(i)}\}$ for $i = 0, 1, \ldots, q$, $j = 0, 1, \ldots, k$ with $q(q+1)/2$ degrees of freedom. From the above procedure, it is clear the conditions (2.2.3) with $i < q$ holds. What remains to be proved is that (2.2.2) for $i = 0, 1, \ldots, q$ and (2.2.3) with $i = q$ is compatible with our selection.

Recalling that $b_j^{(q)} := 0$ for each j, condition (2.2.3) with $i = q$ becomes

$$\frac{1}{m!}\Big[\sum_{j=0}^{k-u}(j-k+u)^m a_j^{(q)} + r^m \sum_{j=k-u+1}^{k}(j-k+u)^m a_j^{(q)}\Big] = 0, \qquad m = 1, 2, \ldots, p-q. \tag{2.3.9}$$

Using (2.2.4), we have for $0 \le i \le q$

$$\begin{aligned}
&\frac{1}{m!}\Big[\sum_{j=0}^{k-u}(j-k+u)^m a_j^{(i)} + r^m \sum_{j=k-u+1}^{k}(j-k+u)^m a_j^{(i)}\Big] \\
&= \frac{r^m}{m!}\Big[u^m a_k^{(i)} + (u-1)^m a_{k-1}^{(i)}\Big] \\
&\quad + \frac{1}{m!}\Big[\sum_{j=0}^{k-u}(j-k+u)^m b_j^{(i-1)} + r^m \sum_{j=k-u+1}^{k-2}(j-k+u)^m b_j^{(i-1)}\Big] \\
&= \frac{r^m}{m!}\Big[u^m(a_k^{(i)} - b_k^{(i-1)}) + (u-1)^m(a_{k-1}^{(i)} - b_{k-1}^{(i-1)})\Big]
\end{aligned}$$

$$+\frac{1}{m!}\left[\sum_{j=0}^{k-u}(j-k+u)^m b_j^{(i-1)} + r^m \sum_{j=k-u+1}^{k}(j-k+u)^m b_j^{(i-1)}\right]. \quad (2.3.10)$$

Substituting into the last line of (2.3.10) relation (2.2.3) with $i < q$, we therefore obtain the recursion relation

$$\frac{1}{m!}\left[\sum_{j=0}^{k-u}(j-k+u)^m a_j^{(i)} + r^m \sum_{j=k-u+1}^{k}(j-k+u)^m a_j^{(i)}\right]$$

$$= \frac{1}{r(m+1)!}\left[\sum_{j=0}^{k-u}(j-k+u)^{m+1} a_j^{(i-1)} + r^{m+1} \sum_{j=k-u+1}^{k}(j-k+u)^{m+1} a_j^{(i-1)}\right]$$

$$+\frac{r^m}{m!}\left[u^m(a_k^{(i)} - b_k^{(i-1)}) + (u-1)^m(a_{k-1}^{(i)} - b_{k-1}^{(i-1)})\right] \quad (2.3.11)$$

for $i = 1, \ldots, q$. Using this recursion relation, the left hand side of (2.3.9) becomes

$$\frac{1}{m!}\left[\sum_{j=0}^{k-u}(j-k+u)^m a_j^{(q)} + r^m \sum_{j=k-u+1}^{k}(j-k+u)^m a_j^{(q)}\right]$$

$$= \frac{1}{r(m+1)!}\left[\sum_{j=0}^{k-u}(j-k+u)^{m+1} a_j^{(q-1)} + r^{m+1} \sum_{j=k-u+1}^{k}(j-k+u)^{m+1} a_j^{(q-1)}\right]$$

$$+\frac{r^m}{m!}\left[u^m(a_k^{(q)} - b_k^{(q-1)}) + (u-1)^m(a_{k-1}^{(q)} - b_k^{(q-1)})\right]$$

$$= \quad \cdots$$

$$= \frac{1}{r^q(m+q)!}\left[\sum_{j=0}^{k-u}(j-k+u)^{m+q} a_j^{(0)} + r^{m+q} \sum_{j=k-u+1}^{k}(j-k+u)^{m+q} a_j^{(0)}\right]$$

$$+\sum_{\theta=0}^{q-1}\frac{r^m}{(m+\theta)!}\left[u^{m+\theta}(a_k^{(q-\theta)} - b_k^{(q-\theta-1)}) + (u-1)^{m+\theta}(a_{k-1}^{(q-\theta)} - b_{k-1}^{(q-\theta-1)})\right]$$

$$= r^m \sum_{\theta=0}^{q}\frac{1}{(m+\theta)!}\left[u^{m+\theta}(a_k^{(q-\theta)} - b_k^{(q-\theta-1)}) + (u-1)^{m+\theta}(a_{k-1}^{(q-\theta)} - b_{k-1}^{(q-\theta-1)})\right],$$

since $a_j^{(0)} = 0$ for $j = 0, 1, \ldots, k-2$. This we know equals zero for the selection (2.3.8) by virtue of Lemma 2.3.1 since $\ell = k - q$.

To establish that (2.2.2) also holds, using (2.2.4) we have the relation

$$\sum_{j=0}^{k} a_j^{(i)} = a_k^{(i)} - b_k^{(i-1)} + a_{k-1}^{(i)} - b_{k-1}^{(i-1)} + \sum_{j=0}^{k} b_j^{(i-1)}, \quad i = 0, 1, \ldots, q.$$

Substituting the last term $\sum_{j=0}^{k} b_j^{(i-1)}$ in the above by (2.2.3) with $m = 1$, then use (2.3.11) in the same recursive fashion as before, we eventually derive

$$\begin{aligned}
\sum_{j=0}^{k} a_j^{(i)} &= a_k^{(i)} - b_k^{(i-1)} + a_{k-1}^{(i)} - b_{k-1}^{(i-1)} \\
&\quad + \frac{1}{r}\left[\sum_{j=0}^{k-u}(j-k+u)a_j^{(i-1)} + r\sum_{j=k-u+1}^{k}(j-k+u)a_j^{(i-1)}\right] \\
&= a_k^{(i)} - b_k^{(i-1)} + a_{k-1}^{(i)} - b_{k-1}^{(i-1)} \\
&\quad + \frac{1}{r}\left\{r\left[u(a_k^{(i-1)} - b_k^{(i-2)}) + (u-1)(a_{k-1}^{(i-1)} - b_{k-1}^{(i-2)})\right]\right. \\
&\qquad \left. + \frac{1}{r2!}\left[\sum_{j=0}^{k-u}(j-k+u)^2 a_j^{(i-2)} + r^2 \sum_{j=k-u+1}^{k}(j-k+u)^2 a_j^{(i-2)}\right]\right\} \\
&= \sum_{\theta=0}^{1}\frac{1}{\theta!}\left[u^\theta(a_k^{(i-\theta)} - b_k^{(i-\theta-1)}) + (u-1)^\theta(a_{k-1}^{(i-\theta)} - b_{k-1}^{(i-\theta-1)})\right] \\
&\quad + \frac{1}{r^2 2!}\left[\sum_{j=0}^{k-u}(j-k+u)^2 a_j^{(i-2)} + r^2\sum_{j=k-u+1}^{k}(j-k+u)^2 a_j^{(i-2)}\right] \\
&= \quad \cdots \\
&= \sum_{\theta=}^{i}\frac{1}{\theta!}\left[u^\theta(a_k^{(i-\theta)} - b_k^{(i-\theta-1)}) + (u-1)^\theta(a_{k-1}^{(i-\theta)} - b_{k-1}^{(i-\theta-1)})\right] \qquad (2.3.12)
\end{aligned}$$

for $i = 1, 2, \ldots, q$. which also equals zero for the selection (2.3.8) by virtue of Lemma 2.3.1. □

We have the following similar result for LIM methods.

Theorem 2.3.3 *For any integer $q \geq 2$ there are A-contractive, variable stepsize LIM methods (2.2.1) of type $(2q-2, 2q-2, q)$, $(2q-1, 2q-1, q)$, $(2q, 2q, q)$. Their contractivity functions $R(z)$ are respectively the $[q-2/q]$, $[q-1/q]$ and $[q/q]$ Padé approximant of e^z. The coefficients depend on the step changing ratio $r = h_2/h_1$ and each method has $q(q-1)/2$ degrees of freedom.*

Proof. Let $b_k^{(i)} = 0$ for $i = 0, 1, \ldots, q-1$, other coefficients are determined in the same way as in the proof of the previous theorem except for losing one degree of freedom while solving $\{b_j^{(i)}\}$ for given i. So the total degree of freedom of coefficients will be $q(q-1)/2$ which is q degrees less than the FIM case.

The remainder of the proof is exactly the same as the proof in the previous theorem. □

When $r = 1$, $u = k$, the fixed step size FIM and LIM methods are obtained

Corollary 2.3.4 *For any integer $q \geq 2$ there are A-contractive FIM and LIM methods (2.1.1) of type $(2q-2, 2q-2, q)$, $(2q-1, 2q-1, q)$, $(2q, 2q, q)$. Their contractivity functions $R(z)$ are respectively the $[q-2/q]$, $[q-1/q]$ and $[q/q]$ Padé approximant of e^z. The coefficients are constants with $q(q+1)/2$ or $q(q-1)/2$ degrees of freedom respectively for FIM and LIM methods.*

Listed in the appendix are examples of such methods along with their respective truncation error terms. Also listed in the appendix are the coefficients of the $(4, 4, 2)$ algorithms. The parameter α is totally free and would normally be chosen in such a way as to minimize local truncation errors.

Note that unlike stability results for BDF methods (see Rockswold, 1988), there is no restriction on h_2/h_1 for preserving A-contractivity. The coefficients $\{a_j^{(i)}, b_j^{(i)}\}$ are continuous functions of r on $0 < r < \infty$, so they are bounded on $\omega \leq r \leq \Omega$ for any $0 < \omega < 1$ and $1 < \Omega < \infty$. For norms other than the Euclidean norm, the boundedness of $\|R(hA)\|$ by one with $R(z)$ being a Padé approximant of e^z is a complicated problem. The reader can consult Hairer & Wanner (1991, p.185-188) for more details, particularly theorem 11.10 and the section on threshold factors.

2.4 Local error terms of VCM methods

As seen in the previous section, there still remains some degree freedom in choosing the coefficients of A-contractive VCM methods for given order $p = k$. Usually these are chosen such that the local error terms are as small as possible. Here we give a way to fix all coefficients in a given method in such a way as to lead to a manageable local error term which will in turn facilitate our convergence analysis of general nonlinear problems. For simplicity, we consider fixed stepsize which means $u = k$, $r = 1$ in our notation of the previous section. The results can then be extended to variable stepsize without difficulties. Define the local error of VCM methods to be

$$TE_{(k,p,s)} := \sum_{j=0}^{k}\left[\sum_{i=0}^{s} a_j^{(i)} h^i Q_n^i\right] y(x_{n+j}) - h\sum_{j=0}^{k}\left[\sum_{i=0}^{s-1} b_j^{(i)} h^i Q_n^i\right] y'(x_{n+j}) \tag{2.4.1}$$

and define the Peano kernel by,

$$K_p^{(i)}(\tau) := \sum_{j=0}^{k}\left[a_j^{(i)}\frac{(j-\tau)_+^p}{p!} - b_j^{(i)}\frac{(j-\tau)_+^{p-1}}{(p-1)!}\right]$$

where

$$(j-\tau)_+^p = \begin{cases} 0 & \text{if } j-\tau < 0 \\ (j-\tau)^p & \text{otherwise} \end{cases}$$

Substituting the Taylor series derived about $x = x_{n+k-n} = x_n$ for y, and y' with integral representation for the remainder into the right hand side of (2.4.1), we obtain

$$TE_{(k,p,q)} =$$

$$\sum_{i=0}^{q}\sum_{j=0}^{k} a_j^{(i)} h^i Q^i \Big[\sum_{m=0}^{p} \frac{j^m}{m!} h^m y^{(m)}(x_n) + h^{p+1}\int_0^j \frac{(j-\tau)^p}{(p)!} y^{(p+1)}(x_n+\tau h)\,d\tau\Big]$$
$$-\sum_{i=0}^{q-1}\sum_{j=0}^{k} b_j^{(i)} h^i Q^i \Big[\sum_{m=1}^{p} \frac{j^{m-1}}{(m-1)!} h^m y^{(m)}(x_n) + h^{p+1}\int_0^j \frac{(j-\tau)^{p-1}}{(p-1)!} y^{(p+1)}(x_n+\tau h)\,d\tau\Big].$$

That is,

$$\begin{aligned}TE_{(k,p,q)} &= \sum_{i=0}^{q}\sum_{j=0}^{k} a_j^{(i)} h^i Q^i y^{(0)}(x_n)\\
&+ \sum_{m=1}^{p}\Big[\frac{1}{m!}\sum_{i=0}^{q}\sum_{j=0}^{k} j^m a_j^{(i)} h^i Q^i - \frac{1}{(m-1)!}\sum_{i=0}^{q-1}\sum_{j=0}^{k} j^{m-1} b_j^{(i)} h^i Q^i\Big] h^m y^{(m)}(x_n)\\
&+h^{p+1}\Big[\sum_{i=0}^{q}\sum_{j=0}^{k} a_j^{(i)} h^i Q^i \int_0^j \frac{(j-\tau)^p}{p!} y^{(p+1)}(x_n+\tau h)\,d\tau\\
&\quad -\sum_{i=0}^{q-1}\sum_{j=0}^{k} b_j^{(i)} h^i Q^i \int_0^j \frac{(j-\tau)^{p-1}}{(p-1)!} y^{(p+1)}(x_n+\tau h)\,d\tau\Big].\end{aligned} \tag{2.4.2}$$

From the above it can be seen that if

$$\sum_{j=0}^{k} a_j^{(i)} = 0, \qquad i = 0,\ldots,q \tag{2.4.3}$$

$$\frac{1}{m!}\sum_{j=0}^{k} j^m a_j^{(i)} - \frac{1}{(m-1)!}\sum_{j=0}^{k} j^{m-1} b_j^{(i)} = 0, \quad \begin{array}{l} m = 1,2,\ldots,k, \\ i = 0,1,\ldots,q. \end{array} \tag{2.4.4}$$

hold, where the parameter k can take on each of the values $2q-2$, $2q-1$, $2q$, then we have the following result.

Theorem 2.4.1 *For A-contractive FIM methods of type* $(2q-2, 2q-2, q)$, $(2q-1, 2q-1, q)$, $(2q, 2q, q)$ *the local error can be written as*

$$TE_{(k,p,q)} = h^{p+1}\sum_{i=0}^{l} h^i Q^i \int_0^k K_{p+1}^{(i)}(\tau)y^{(p+1)}(x_n+\tau h)d\tau$$

Remark **2.4.2** We give an explanation for the suggested way to reconstruct the coefficients $\{a_j^{(i)}, b_j^{(i)}\}$ such that equations (2.4.3) and (2.4.4) hold. For example, consider the (4,4,2) method , i.e., $q = 2$, $k = p = 4$. We recommend Table 2.4.1 as an outline for proof of the theorem. Let $[i, m]$ denote the system (2.4.4) with certain i, m, which have $(q+1)\cdot k = 12$ equations. Adding the $(q+1)$ - 3 equations in (2.4.3), the total number of equations to be solved is 15. Due to the simplifying conditions (2.2.4) and (2.3.8), noting $b_j^{(q)} = b_j^{(-1)} = 0$, for $j = 0, \ldots, k$, we have only $q\cdot(k+1) = 10$ unknown $\{b_j^{(i)}\}$ which is far less than the total number of the equations. Thus any solution will not be straightforwardly solved. Fortunately, we know from the proof of theorem 2.3.2 that equations $[2,1], [2,2]$ and all three equations in (2.4.3) hold automatically when $\{b_j^{(i)}\}$ satisfy $[0,1], [0,2], [0,3], [0,4], [1,1], [1,2], [1,3]$ and conditions (2.2.4), (2.3.8) hold. Therefore we really have ten equations for ten unknowns. Our next step is to express $[1,4], [2,3], [2,4]$ as $[0,5], [1,4], [1,5]$ respectively through an index substitution. The focus therefore will be on the nonsingular character of this reduced linear system of equations.

Proof of Theorem 2.4.1. We need to choose $\{a_j^{(i)}, b_j^{(i)}\}$ such that systems (2.4.3) and (2.4.4) hold. We know from the proof of theorem 2.3.2 that if conditions (2.4.4) hold for $i = 0, \ldots, q-1$, $m = 1, \ldots, k-i$ together with (2.2.4) and (2.3.8), then (2.4.3) and (2.4.4) with $i = q$, $m = 1, \ldots, k-q$ also hold. So we need only to prove that there exist $\{b_j^{(i)}\}$ satisfying (2.4.4) with $i = 0, \ldots, q-1$, $m = 1, \ldots, k-i$ and with

Table 2.4.1: An example fixing the coefficients of method (4,4,2)

	m=1	m=2	m=3	m=4	
i=0	[0,1]	[0,2]	[0,3]	[0,4]	
i=1	[1,1]	[1,2]	[1,3]	[1,4]	
i=2	[2,1]	[2,2]	[2,3]	[2,4]	

	m=1	m=2	m=3	m=4	m=5
i=0	[0,1]	[0,2]	[0,3]	[0,4]	[0,5]
i=1	[1,1]	[1,2]	[1,3]	[1,4]	[1,5]

$i = 0, \ldots, q$, $m = k - i + 1, \ldots, k$. Using (2.2.4) and (2.3.8), the condition (2.4.4) can be rewritten as

$$\frac{1}{(m-1)!} \sum_{j=0}^{k} j^{m-1} b_j^{(i)} = \frac{1}{m!} \sum_{j=0}^{k} j^m b_j^{(i-1)} + \frac{1}{m!}\left[(k-1)^m \beta_i + k^m \alpha_i\right] \qquad (2.4.5)$$

for $i = 0, 1, \ldots, q$, $m = 1, 2, \ldots, k$. Now we consider $m = 1, 2, \ldots, k - i$ and $m = k - i + 1, k - i + 2, \ldots, k$ separately.

For (2.4.5) with $i = 1, 2, \ldots, q$, $m = k - i + 1, k - i + 2, \ldots, k$, change i to $i + 1$, m to $m - 1$, it becomes

$$\begin{aligned} &\frac{1}{(m-1)!} \sum_{j=0}^{k} j^{m-1} b_j^{(i)} \\ &\quad = \frac{1}{(m-2)!} \sum_{j=0}^{k} j^{m-2} b_j^{(i+1)} - \frac{1}{(m-1)!}\left[(k-1)^{m-1} \beta_{i+1} + k^{m-1} \alpha_{i+1}\right] \end{aligned}$$

for $i = 0, 1, \ldots, q - 1$, $m = k - i + 1, k - i + 2, \ldots, k + 1$.

Now put together with $m = 1, 2, \ldots, k-i$, we have,

$$\sum_{j=0}^{k} j^{m-1} b_j^{(i)} = \begin{cases} \dfrac{1}{m}\displaystyle\sum_{j=0}^{k} j^m b_j^{(i-1)} + \dfrac{1}{m}\Big[(k-1)^m \beta_i + k^m \alpha_i\Big], \\ \qquad \text{for } m = 1, 2, \ldots, k-i, \\ \\ (m-1)\displaystyle\sum_{j=0}^{k} j^{m-2} b_j^{(i+1)} - \Big[(k-1)^{m-1}\beta_{i+1} + k^{m-1}\alpha_{i+1}\Big], \\ \qquad \text{for } m = k-i+1, k-i+2, \ldots, k+1, \end{cases} \tag{2.4.6}$$

with $i = 0, 1, \ldots, q-1$. The system has $q \cdot (k+1)$ equations with $q \cdot (k+1)$ variables $\{b_j^{(i)}\}$. It remains to prove that the coefficient matrix has full rank, then we have a unique solution. Let $A_{i,j}$ be $(k+1) \times (k+1)$ matrices and O be the $(k+1) \times (k+1)$ null matrix. Noting $b_j^{(-1)} := b_j^{(q)} := 0$ for each j, we can write (2.4.6) as

$$\bar{A}\mathbf{b} = \mathbf{c}$$

where

$$\mathbf{b} = (b_0^{(0)}, \ldots, b_k^{(0)}, b_0^{(1)}, \ldots, b_k^{(1)}, \ldots, b_0^{(q-1)}, \ldots, b_k^{(q-1)})^T, \quad \mathbf{c} = (\ldots)^T$$

and

$$\bar{A} = \begin{pmatrix} A_{0,0} & A_{0,1} & O & O & \cdots & O & O & O & O \\ A_{1,0} & A_{1,1} & A_{1,2} & O & \cdots & O & O & O & O \\ O & A_{2,1} & A_{2,2} & A_{2,3} & \cdots & O & O & O & O \\ O & O & A_{3,2} & A_{3,3} & \cdots & O & O & O & O \\ \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots \\ O & O & O & O & \cdots & A_{q-4,q-4} & A_{q-4,q-3} & O & O \\ O & O & O & O & \cdots & A_{q-3,q-4} & A_{q-3,q-3} & A_{q-3,q-2} & O \\ O & O & O & O & \cdots & O & A_{q-2,q-3} & A_{q-2,q-2} & A_{q-2,q-1} \\ O & O & O & O & \cdots & O & O & A_{q-1,q-2} & A_{q-1,q-1} \end{pmatrix}$$

with

$$A_{0,1} = \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & -k & -k\ 2^{k-1} & \cdots & -k \cdot (k-1)^{k-1} & -k \cdot k^{k-1} \end{pmatrix}.$$

$$A_{1,0} = \begin{pmatrix} 0 & -1 & -2 & \cdots & -(k-1) & -k \\ 0 & -\frac{1}{2} & -\frac{2^2}{2} & \cdots & -\frac{(k-1)^2}{2} & -\frac{k^2}{2} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & -\frac{1}{k-2} & -\frac{2^{k-2}}{k-2} & \cdots & -\frac{(k-1)^{k-2}}{k-2} & -\frac{k^{k-2}}{k-2} \\ 0 & -\frac{1}{k-1} & -\frac{2^{k-1}}{k-1} & \cdots & -\frac{(k-1)^{k-1}}{k-1} & -\frac{k^{k-1}}{k-1} \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 \end{pmatrix}$$

and in general, we have the following expressions for the block matrices $A_{i,i}$, $A_{i,i+1}$, $A_{i+1,i}$ with $i = 0, 1, \ldots, q-1$.

$$A_{i,i} = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 & 1 \\ 0 & 1 & 2 & \cdots & k-1 & k \\ 0 & 1 & 2^2 & \cdots & (k-1)^2 & k^2 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 1 & 2^{k-1} & \cdots & (k-1)^{k-1} & k^{k-1} \\ 0 & 1 & 2^k & \cdots & (k-1)^k & k^k \end{pmatrix}.$$

$$A_{i,i+1} = \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & -(k-i) & -(k-i)\cdot 2^{k-i-1} & \cdots & -(k-i)\cdot(k-i)^{k-i-1} & -(k-i)\cdot k^{k-i-1} \\ 0 & -(k-i-1) & -(k-i-1)\cdot 2^{k-i-2} & \cdots & -(k-i-1)\cdot(k-i-1)^{k-i-2} & -(k-i-1)\cdot k^{k-i-2} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & -(k-1) & -(k-1)\cdot 2^{k-2} & \cdots & -(k-1)\cdot(k-1)^{k-2} & -(k-1)\cdot k^{k-2} \\ 0 & -k & -k\cdot 2^{k-1} & \cdots & -k\cdot(k-1)^{k-1} & -k\cdot k^{k-1} \end{pmatrix}.$$

$$A_{i+1,i} = \begin{pmatrix} 0 & -1 & -2 & \cdots & -(k-1) & -k \\ 0 & -\frac{1}{2} & -\frac{2^2}{2} & \cdots & -\frac{(k-1)^2}{2} & -\frac{k^2}{2} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & -\frac{1}{k-i-1} & -\frac{2^{k-i-1}}{k-i-1} & \cdots & -\frac{(k-1)^{k-i-1}}{k-i-1} & -\frac{k^{k-i-1}}{k-i-1} \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & -\frac{1}{k-i} & -\frac{2^{k-i}}{k-i} & \cdots & -\frac{(k-1)^{k-i}}{k-i} & -\frac{k^{k-i}}{k-i} \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 \end{pmatrix}.$$

Noting the special structure of the matrix $\tilde{A}$, we manipulate it by elementary matrix operations in the following order. From $i = 0$ to $i = q-2$, the $[i(k+1)+j]$-th row of $\tilde{A}$ multiplied by $\frac{1}{j-1}$ is added to the $[(i+1)(k+1)+j-1]$-th row of $\tilde{A}$ for

$j = 2, 3, \ldots, k-i$. The matrix $\bar{A}$ becomes

$$\bar{A} = \begin{pmatrix} A_{0,0} & A_{0,1} & O & \cdots & O & O \\ O & A_{1,1} & A_{1,2} & \cdots & O & O \\ O & O & A_{2,2} & \cdots & O & O \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ O & O & O & \cdots & A_{q-2,q-2} & A_{q-2,q-1} \\ O & O & O & \cdots & O & A_{q-1,q-1} \end{pmatrix}.$$

whose determinant is the qth power of the Vandemonde determinant $|A_{0,0}|$ which is not zero. Therefore with this selection of parameters, we have

$$\begin{aligned} TE_{(k,p,q)} &= h^{p+1}\Big[\sum_{i=0}^{q}\sum_{j=0}^{k} a_j^{(i)} h^i Q^i \int_0^j \frac{(j-s)^{p+1}}{(p+1)!} y^{(p+1)}(x_n + sh)\, ds \\ &\quad - \sum_{i=0}^{q-1}\sum_{j=0}^{k} b_j^{(i)} h^i Q^i \int_0^j \frac{(j-s)^{p}}{(p)!} y^{(p+1)}(x_n + sh)\, ds\Big]. \end{aligned}$$

□

For LIM methods, there are fewer free parameters since $b_k^{(i)} = 0$, $i = 0, 1, \ldots, q$. Instead of (2.4.3) and (2.4.4), we have

$$\sum_{j=0}^{k} a_j^{(i)} = 0, \qquad i = 0, \ldots, q \tag{2.4.7}$$

$$\frac{1}{m!}\sum_{j=0}^{k} j^m a_j^{(i)} - \frac{1}{(m-1)!}\sum_{j=0}^{k} j^{m-1} b_j^{(i)} = 0, \qquad \begin{array}{l} m = 1, 2, \ldots, k, \text{ for } i = 0, \\ m = 1, 2, \ldots, k-1, \text{ for } i = 1, \ldots, q. \end{array} \tag{2.4.8}$$

With essentially the same construction as that of Theorem 2.4.1 we also have:

Theorem 2.4.3 *For A-contractive LIM methods of type $(2q-2, 2q-2, q)$, $(2q-1, 2q-1, q)$, $(2q, 2q, q)$ the local error can be written as*

$$TE_{(k,p,q)} = h^p \sum_{i=0}^{q} h^i Q^i \int_0^k K_p^{(i)}(s) y^{(p)}(x_n + sh) ds$$

where p is the order of the corresponding method.

Remark **2.4.4** The theorems in this section also hold for variable stepsize formulae. We can see this by writing (2.4.6) in the following way:

$$\sum_{j=0}^{k-u} (j-k+u)^{m-1} b_j^{(i)} + r^{m-1} \sum_{j=k-u+1}^{k} (j-k+u)^{m-1} b_j^{(i)}$$

$$= \begin{cases} \dfrac{1}{m}\left[\dfrac{1}{r}\displaystyle\sum_{j=0}^{k-u}(j-k+u)^m b_j^{(i-1)} + r^{m-1}\sum_{j=k-u+1}^{k}(j-k+u)^m b_j^{(i-1)}\right] \\ +\dfrac{r^{m-1}}{m}\left[(u-1)^m \beta_i + u^m \alpha_i\right], \qquad \text{for } m = 1, 2, \dots, k-i, \\ \\ (m-1)\left[r\displaystyle\sum_{j=0}^{k-u}(j-k+u)^{m-2} b_j^{(i+1)} + r^{m-1}\sum_{j=k-u+1}^{k}(j-k+u)^{m-2} b_j^{(i+1)}\right] \\ -r^{m-1}\left[(u-1)^{m-1}\beta_i + u^{m-1}\alpha_i\right], \qquad \text{for } m = k-i+1, k-i+2, \dots, k+1. \end{cases}$$

In order to simplify the presentation, we consider $u = 2$, but note that a similar argument is valid for any integer $u \in \{0, 1, \dots, k$. With the same notation for the coefficient matrix $\tilde{A}$, the block matrices $A_{i,i}$, $-A_{i,i+1}$, $-A_{i+1,i}$ are now respectively as follows,

$$\begin{pmatrix} 1 & 1 & \cdots & 1 & 1 & 1 & 1 \\ (u-k) & (u-k+1) & \cdots & (u-3) & 0 & r(u-1) & ru \\ (u-k)^2 & (u-k+1)^2 & \cdots & (u-3)^2 & 0 & r^2(u-1)^2 & r^2u^2 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots \\ (u-k)^{k-1} & (u-k+1)^{k-1} & \cdots & (u-3)^{k-1} & 0 & r^{k-1}(u-1)^{k-1} & r^{k-1}u^{k-1} \\ (u-k)^k & (u-k+1)^k & \cdots & (u-3)^k & 0 & r^k(u-1)^k & r^ku^k \end{pmatrix}.$$

$$
\begin{pmatrix}
0 & 0 & \cdots & 0 & 0 & 0 & 0 \\
\vdots & \vdots & & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & \cdots & 0 & 0 & 0 & 0 \\
(k-i)r(u-k)^{k-i-1} & (k-i)r(u-k+1)^{k-i-1} & \cdots & (k-i)r(u-3)^{k-i-1} & 0 & (k-i)r^{k-i}(u-1)^{k-i-1} & (k-i)r^{k-i}u^{k-i-1} \\
(k-i-1)r(u-k)^{k-i-2} & (k-i-1)r(u-k+1)^{k-i-2} & \cdots & (k-i-1)r(u-3)^{k-i-2} & 0 & (k-i-1)r^{k-i-1}(u-1)^{k-i-2} & (k-i-1)r^{k-i-1}u^{k-i-2} \\
\vdots & \vdots & & \vdots & \vdots & \vdots & \vdots \\
(k-1)r(u-k)^{k-2} & (k-1)r(u-k+1)^{k-2} & \cdots & (k-1)r(u-3)^{k-2} & 0 & (k-1)r^{k-1}(u-1)^{k-2} & (k-1)r^{k-1}u^{k-2} \\
kr(u-k)^{k-1} & kr(u-k+1)^{k-1} & \cdots & kr(u-3)^{k-1} & 0 & kr^{k}(u-1)^{k-1} & kr^{k}u^{k-1}
\end{pmatrix}
$$

and

$$\begin{pmatrix}
\frac{u-k}{r} & \frac{u-k+1}{r} & \cdots & \frac{u-3}{r} & 0 & (u-1) & u \\
\frac{(u-k)^2}{2r} & \frac{(u-k+1)^2}{2r} & \cdots & \frac{(u-3)^2}{2r} & 0 & \frac{r(u-1)^2}{2} & \frac{ru^2}{2} \\
\vdots & \vdots & & \vdots & \vdots & \vdots & \vdots \\
\frac{(u-k)^{k-i-1}}{(k-i-1)r} & \frac{(u-k+2)^{k-i-1}}{(k-i-1)r} & \cdots & \frac{(u-3)^{k-i-1}}{(k-i-1)r} & 0 & \frac{r^{k-i-2}(u-1)^{k-i-1}}{k-i-1} & \frac{r^{k-i-2}u^{k-i-1}}{k-i-1} \\
\frac{(u-k)^{k-i}}{(k-i)r} & \frac{(u-k+2)^{k-i}}{(k-i)r} & \cdots & \frac{(u-3)^{k-i}}{(k-i)r} & 0 & \frac{r^{k-i-1}(u-1)^{k-i}}{k-i} & \frac{r^{k-i-1}u^{k-i}}{k-i} \\
0 & 0 & \cdots & 0 & 0 & 0 & 0 \\
\vdots & \vdots & & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & \cdots & 0 & 0 & 0 & 0
\end{pmatrix}$$

With the similar operations in page 48 we again reduce matrix $\bar{A}$ to the equivalent coefficient matrix:

$$\begin{pmatrix}
A_{0,0} & O & O & \cdots & O & O \\
O & A_{1,1} & O & \cdots & O & O \\
O & O & A_{2,2} & \cdots & O & O \\
\vdots & \vdots & \vdots & & \vdots & \vdots \\
O & O & O & \cdots & A_{q-2,q-2} & O \\
O & O & O & \cdots & O & A_{q-1,q-1}
\end{pmatrix}$$

whose determinant is qth power of the Vandemonde determinant $|A_{0,0}|$ which is not zero.

Remark **2.4.5** One should not overlook the fact that the local error is dependent on the Jacobian matrix whose norm can be quite large with a stiff problem. The error term derived here is useful mostly as a tool for our analysis. We postpone this issue to Chapter 3.

2.5 Examples

We borrow from Byrne and Hindmarsh (1987) the following two problems.

Example 2.5.1 Robertson's problem

$$\begin{aligned} y_1' &= -0.04y_1 + 10^4 y_2 y_3 \\ y_2' &= 0.04y_1 - 10^4 y_2 y_3 - 3 \times 10^7 y_2 y_2 \\ y_3' &= 3 \times 10^7 y_2 y_2 \end{aligned} \tag{2.5.1}$$

with the initial data

$$y_1(0) = 1, \quad y_2(0) = 0, \quad y_3(0) = 0$$

whose solution over the interval $[0, 10^7]$ is desired.

Example 2.5.2 The Field-Noyes chemical oscillator

$$\begin{aligned} y_1' &= s(y_2 - y_1 y_2 + y_1 - q y_1^2) \\ y_2' &= (y_3 - y_2 - y_1 y_2)/s \\ y_3' &= w(y_1 - y_3). \end{aligned} \tag{2.5.2}$$

where

$$s = 77.27, \quad w = 0.1610, \quad q = 8.375 \times 10^{-6}.$$

with the initial data

$$y_1(0) = 4.0, \quad y_2(0) = 1.1, \quad y_3(0) = 4.0$$

whose solution over $[0, 600]$ is desired.

Table 2.5.1: Results for examples 1–2 of FIM (4,4,2) method

	Tol	Nstep	Nsc	Hmin	Hmax	Rmin	Rmax	Eabs
Example 1.	10^{-6}	363	91	1.00d-5	1.24d+6	0.35	3.50	1.31×10^{-5}
	10^{-3}	130	43	1.00d-5	5.84d+6	0.91	3.50	2.69×10^{-3}
Example 2.	10^{-6}	2083	567	1.00d-5	5.61	0.26	3.50	1.31×10^{-5}
	10^{-3}	608	183	1.00d-5	24.6	0.23	3.50	1.40×10^{-2}

We defer to chapter 4 the complete description of the algorithm used, the results are obtained by the code labelled as code (c) there. But we briefly report on some statistics and observations here.

The results are summarized in table 2.5.1 and figures 2.5.1 – 2.5.2. The notations "E+x" and "E-x" mean logscale in the figure. The notations in the table have the following meanings:

* Tol: local error tolerance (we use scalar error control here and set both absolute and relative error tolerance equal to Tol),

* Nstep: number of steps,

* Nsc: number of times stepsizes changed,

* Hmin: smallest value of stepsize,

* Hmax: largest stepsize,

* Rmin: smallest value of h_{n+1}/h_n,

* Rmax: largest value of h_{n+1}/h_n,

* Eabs: actual error at the endpoint.

Robertson's problem is a typical stiff problem with its Jacobian having an eigenvalue of large negative real part.

$$\lambda_1 = 0, \ \lambda_2 \rightarrow 0-, \ \lambda_3 \rightarrow -10^4, \quad \text{for } x \rightarrow \infty.$$

Its solution components are quite smooth, so the stepsizes used in the code grows very fast (around 10^6 when $x > 10^7$). This means the product of stepsizes and the eigenvalue can reach $|h\lambda| = 10^{10}$, and the stepsize is clearly not hampered by the stiffness of the problem (see also figure 2.5.1). The second example has several narrow transient regions which are about 10% of the total integration interval. The code spends around 50% of the total computing efforts to go through these transient regions.

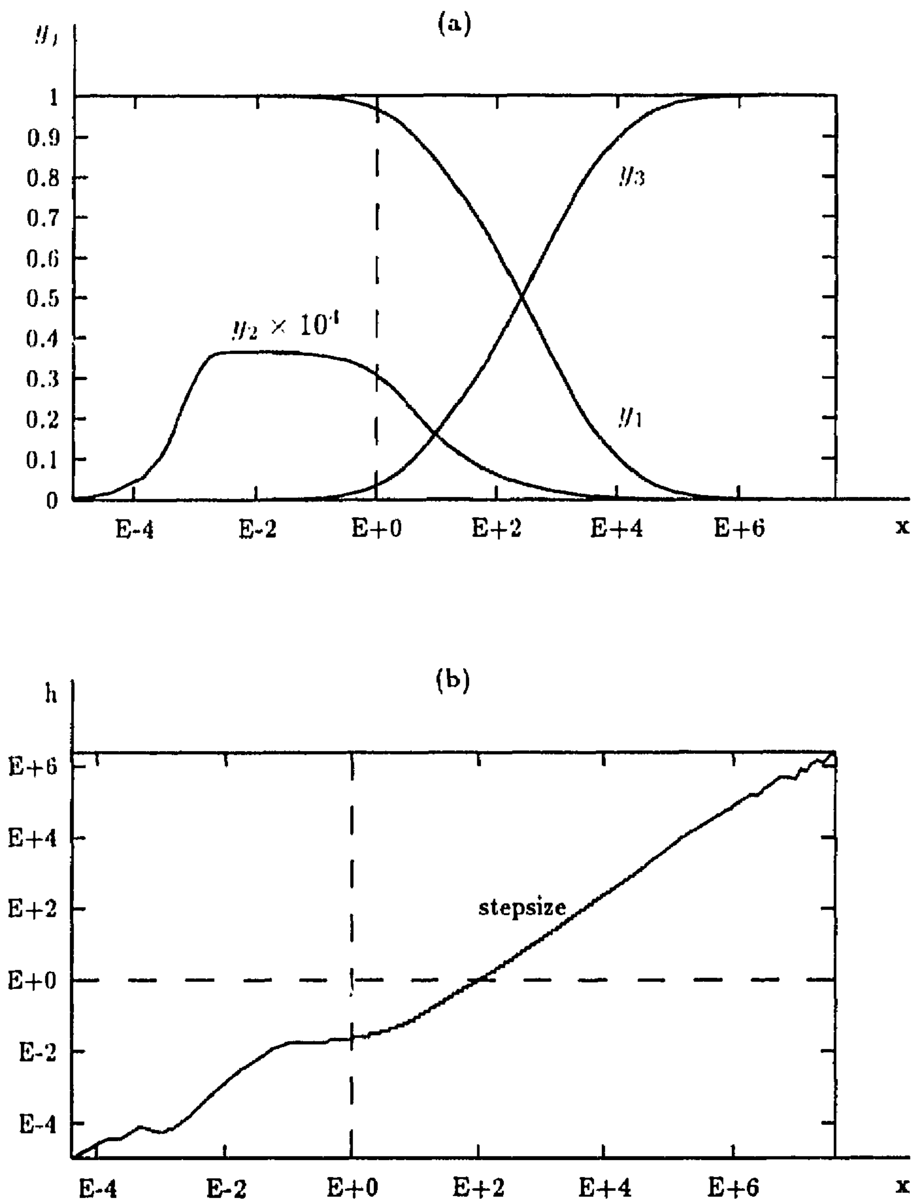

Figure 2.5.1: Robertson's problem

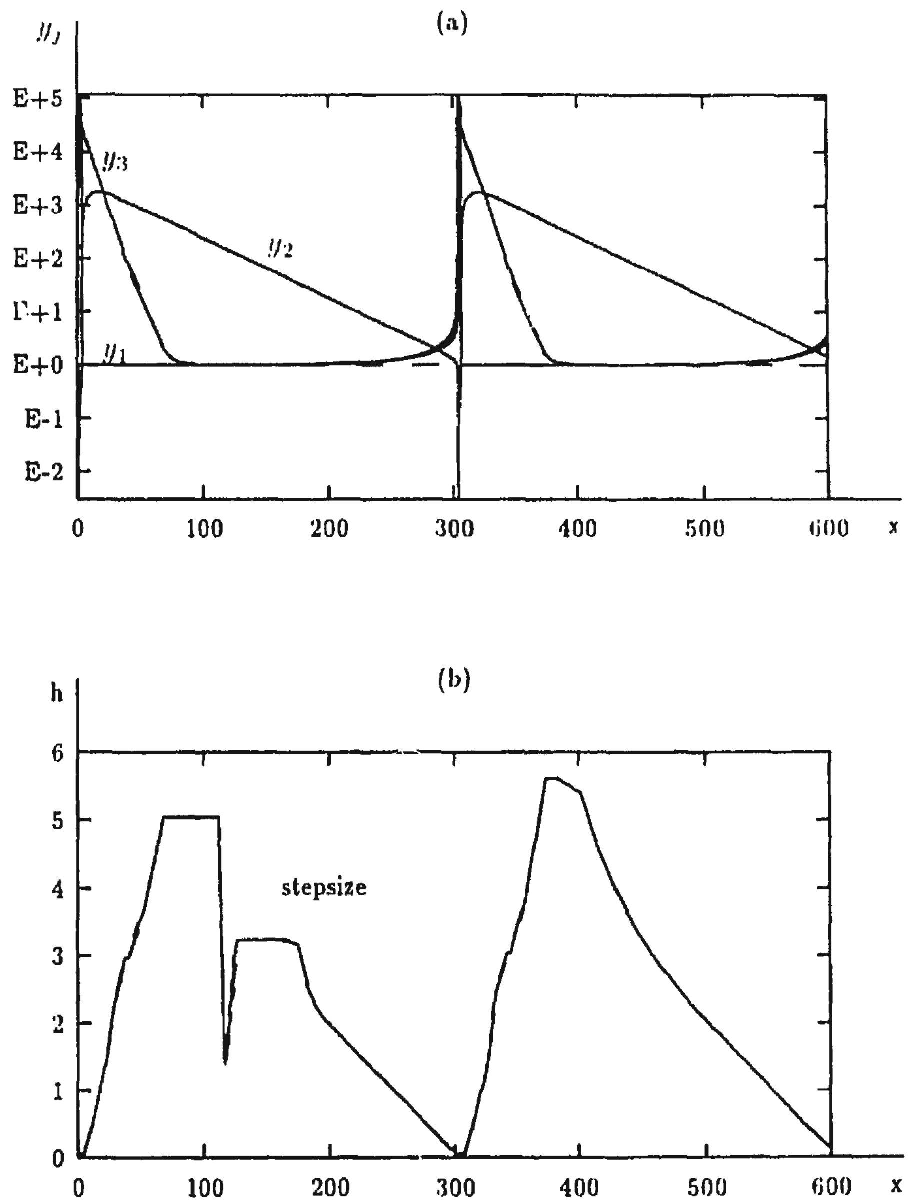

Figure 2.5.2: The Field-Noyes chemical oscillator

Chapter 3

Convergence analysis of VCM methods

3.1 Introduction

Prothero & Robinson (1974) were the first to notice an order reduction phenomenon in the context of stiff problems. They found that the adequate stability properties together with classical order results would not suffice to characterize the global error behavior of numerical methods. Frank & Schneid & Ueberhuber (1981, 1985a, 1985b) studied intensively B-convergence properties for Runge-Kutta methods. Their idea was to study the global error bound of a numerical method

$$\|y_n - y(x_n)\| \leq Ch^p, \text{ for all } h \in [x_0, \bar{x}]$$

with the constants C and $\bar{x}$ independent of stiffness of the problem, then the method is said to be B-convergent of order p. For avoiding dependency on the Lipschitz constant, instead, they considered the one-sided Lipschitz condition or logarithmic matrix norm for the convergence analysis of integration methods for nonlinear problems. In an analogous manner, Lubich (1991) and Hundsdorfer & Steininger (1991) considered the convergence properties for linear multistep methods.

In the definition of B-convergence, the global error bound Ch^p should not depend on $\|f_y\|$ and other derivatives f_x, f_{xx}, f_{yy}, ... However, Frank et al (1985b) also pointed out that a successful application of Newton's method can be guaranteed only for problems with moderately sized second derivative of f (or similar properties of f). Lubich (1991) studied convergence for nonlinear stiff problems with a condition of the form

$$\|A_1^{-1}\left(\frac{\partial f}{\partial y}(y) - A\right)\| \leq C$$

The assumption on continuity of f_y is reasonable, since in real computation most codes tend to keep the Jacobian fixed for as many s · as possible.

In this chapter we consider convergence of VCM methods (2.1.1). We analyze the convergence properties of VCM methods for nonlinear stiff problems in section 3.3 and derive the global error bound Ch^p with constant C independent of the classical Lipschitz constant but dependent on the continuity of the derivative f_y.

In section 3.4 we consider solving autonomous singular perturbation problems

$$\begin{aligned} y' &= f(y, z), \qquad y(x_0) = y_0 \\ \varepsilon z' &= g(y, z), \qquad z(x_0) = z_0, \end{aligned} \tag{3.1.1}$$

where y, $z \in \mathbf{R}^m$, by VCM methods of the form (2.1.1).

Singular perturbation problems (SPP) form a special class of problems containing a parameter ε. When this parameter is small, the corresponding differential system is stiff; when ε tends to zero, the system loses some of its highest derivatives and then becomes a differential algebraic system. SPP have several origins in applied mathematics such as fluid dynamics, nonlinear oscillations with large parameters and chemical kinetics with slow and fast reactions (cf. van der Pol 1926, Dorondicyn 1947). A typical singular perturbation problem is van der Pol's equation (van der

Pol 1926).

Example 3.1.1

$$\begin{aligned} y_1' &= y_2 \\ y_2' &= \mu(1-y_1^2)y_2 - y_1. \end{aligned} \tag{3.1.2}$$

Rescale the solutions by introducing $t = \frac{x}{\mu}$, $z_1(t) = y_1(x)$, $z_2(t) = \mu y_2(x)$. In the resulting equation the factor μ^2 multiplies the entire second line of f. Substituting again y for z, x for t and $\mu^2 = \frac{1}{\varepsilon}$ we obtain

$$\begin{aligned} y_1' &= y_2 \\ \varepsilon y_2' &= (1-y_1^2)y_2 - y_1. \end{aligned} \tag{3.1.3}$$

For moderate values of μ or ε, Van der Pol's equation is easily integrated. But when μ is large, say greater than 500, (i.e., ε is small), the problem becomes stiff. Hairer and Wanner (1991) point out the predictor-corrector Adams code DEABM of Shampine and Watts computes 451 steps and stops at $x = 8.61 \times 10^{-4}$ with the message "the problem appears to be stiff" for initial values $y_1(0) = 2$, $y_2(0) = 0$ and $Atol = 10^{-7}$, $Rtol = 10^{-2}$. We should note that DEABM is designed for non-stiff problems, so is not suitable for singular perturbation problems.

3.2 Estimation of the local error and related rational functions

Since the von Neumann theorem will be cited in the following sections, we always assume the norm $\|\cdot\|$ is the inner product norm $\|\cdot\|_2$ and the logarithmic matrix norm $\mu[\cdot]$ is $\mu_2[\cdot]$ from now on. Insert the exact solution of the initial value problem (1.1.1) into VCM method (2.1.1) and obtain

$$\sum_{j=0}^{k}\left[\sum_{i=0}^{s} a_j^{(i)} h^i Q_{n+k-1}^i\right] y(x_{n+j})$$

$$= \; h\sum_{j=0}^{k}\Big[\sum_{i=0}^{s-1} b_j^{(i)} h^i Q_{n+k-1}^i\Big] f(x_{n+j}, y(x_{n+j})) + d_{n+k} \qquad (3.2.1)$$

where the perturbation terms d_{n+k} are determined by theorem 2.4.1 (theorem 2.4.3 in the case of a linearly implicit method) with Q replaced by Q_{n+k-1}.

We now connect the local error terms with the contractivity function $R(z)$. Subtraction of (3.2.1) from (2.1.1) yields for $n \geq 0$

$$\sum_{j=0}^{k}\Big[\sum_{i=0}^{s} a_j^{(i)} h^i Q_{n+k-1}^i\Big] \triangle y_{n+j} = h\sum_{j=0}^{k}\Big[\sum_{i=0}^{s-1} b_j^{(i)} h^i Q_{n+k-1}^i\Big] \triangle f_{n+j} + d_{n+k} \qquad (3.2.2)$$

where the global errors

$$\triangle y_{n+j} := y(x_{n+j}) - y_{n+j}, \quad \triangle f_{n+j} := f(x_{n+j}, y(x_{n+j})) - f(x_{n+j}, y_{n+j}).$$

As usual we define for $j < 0$

$$\triangle y_j := 0, \quad \triangle f_j := 0$$

and for convenience we further define additional quantities $d_0, d_1, \ldots, d_{k-1}$ according to (3.2.2) for negative values of n.

Noting the simplifying condition (2.2.4), we have from (3.2.2) after some manipulation

$$\begin{aligned}
&\sum_{i=0}^{s}[a_k^{(i)} - b_k^{(i-1)}] h^i Q_{n+k-1}^i \triangle y_{n+k} \\
&\quad = -\sum_{i=0}^{s}[a_{k-1}^{(i)} - b_{k-1}^{(i-1)}] h^i Q_{n+k-1}^i \triangle y_{n+k-1} \\
&\qquad + h\sum_{j=0}^{k}\sum_{i=0}^{s-1} b_j^{(i)} h^i Q_{n+k-1}^i [\triangle f_{n+j} - Q_{n+k-1} \triangle y_{n+j}] + d_{n+k}. \qquad (3.2.3)
\end{aligned}$$

Now define over $\mathbf{R}^{m\times m}$

$$w(h, Q) \; := \; \Big[\sum_{i=0}^{s}[a_k^{(i)} - b_k^{(i-1)}] h^i Q^i\Big]^{-1}.$$

$$r(h,Q) := -\left[\sum_{i=0}^{s}[a_k^{(i)} - b_k^{(i-1)}]h^iQ^i\right]^{-1}\sum_{i=0}^{s}[a_{k-1}^{(i)} - b_{k-1}^{(i-1)}]h^iQ^i. \tag{3.2.4}$$

$$q_j(h,Q) := \left[\sum_{i=0}^{s}[a_k^{(i)} - b_k^{(i-1)}]h^iQ^i\right]^{-1}\sum_{i=0}^{s-1}b_j^{(i)}h^iQ^i, \text{ for } j = 0,1,ldots,k.$$

and on $\mathbb{R}^m$

$$\begin{aligned}\triangle F_{n+k} &:= h\sum_{j=0}^{k}q_j(h,Q_{n+k-1})[\triangle f_{n+j} - Q_{n+k-1}\triangle y_{n+j}] \\ &\quad + w(h,Q_{n+k-1})d_{n+k}.\end{aligned}$$

By multiplying both sides of (3.2.3) with $w(h,Q_{n+k-1})$, we can rewrite it in the form

$$\triangle y_{n+k} = r(h,Q_{n+k-1})\triangle y_{n+k-1} + \triangle F_{n+k}.$$

Define

$$\prod_{\ell=n}^{n-1}r(h,Q_\ell) = 1.$$

The recursion relations with respect to $\triangle y_j$ imply

$$\begin{aligned}\triangle y_n &= \sum_{m=0}^{n}\left[\prod_{\ell=m}^{n-1}r(h,Q_\ell)\right]\triangle F_m \\ &= \sum_{m=0}^{n}\left[\prod_{\ell=m}^{n-1}r(h,Q_\ell)\right]h\sum_{j=0}^{k}q_j(h,Q_{m-1})[\triangle f_{m-k+j} - Q_{m-1}\triangle y_{m-k+j}] \\ &\quad + \sum_{m=0}^{n}\left[\prod_{\ell=m}^{n-1}r(h,Q_\ell)\right]w(h,Q_{m-1})d_m. \end{aligned} \tag{3.2.5}$$

When we consider LIM methods with $q_k(h,Q) = 0$ since $b_k^{(i)} = 0$, $i = 0,1,\ldots,s$, (3.2.5) becomes

$$\begin{aligned}\triangle y_n &= \sum_{m=0}^{n}\left[\prod_{\ell=m}^{n-1}r(h,Q_\ell)\right]h\sum_{j=0}^{k-1}q_j(h,Q_{m-1})[\triangle f_{m-k+j} - Q_{m-1}\triangle y_{m-k+j}] \\ &\quad + \sum_{m=0}^{n}\left[\prod_{\ell=m}^{n-1}r(h,Q_\ell)\right]w(h,Q_{m-1})d_m. \end{aligned} \tag{3.2.6}$$

For general VCM methods, it is better to write (3.2.5) as

$$
\begin{aligned}
\triangle y_n &= hq_k(h,Q_{n-1})[\triangle f_n - Q_{n-1} \triangle y_n] \\
&+h \sum_{j=0}^{k-1} q_j(h,Q_{n-1})[\triangle f_{n-k+j} - Q_{n-1} \triangle y_{n-k+j}] \\
&+ \sum_{m=0}^{n-1}\left[\prod_{\ell=m}^{n-1} r(h,Q_\ell)\right] h \sum_{j=0}^{k} q_j(h,Q_{m-1})[\triangle f_{m-k+j} - Q_{m-1} \triangle y_{m-k+j}] \\
&+ \sum_{m=0}^{n-1}\left[\prod_{\ell=m}^{n-1} r(h,Q_\ell)\right] w(h,Q_{m-1})d_m + w(h,Q_{n-1})d_n. \qquad (3.2.7)
\end{aligned}
$$

We now consider $r(h,Q)$, $q_j(h,Q)$ and $w(h,Q)$ in closer detail with $<\cdot,\cdot>$ still denoting the inner product corresponding to the Euclidean norm.

Lemma 3.2.1 *Let* $\{a_j^{(i)}, b_j^{(i)}\}$ *be the respective coefficients of* $(2s-2,2s-2,s)$, $(2s-1,2s-1,s)$, $(2s,2s,s)$ *VCM methods satisfying (2.1.2), (2.2.4) and (2.3.8) with* $s>1$. *If a sequence of matrices* $\{Q_\ell\}$ *defined over* $\mathbf{R}^{m\times m}$ *satisfies*

$$Re < v, Q_\ell v > \leq 0, \quad \textit{for all} \quad v \in \mathbf{C}^m \textit{ and all } \ell,$$

then

$$\|r(h,Q_\ell)\| \leq 1 \textit{ for all } Q_\ell,$$

and

$$\|q_j(h,Q_\ell)\| \leq C_1, \quad \|q_j(h,Q_\ell)hQ_\ell\| \leq C_2, \quad j=1,2,\ldots,k.$$

for some constants C_1, C_2. *Furthermore, if*

$$Re < v, Q_\ell v > \leq \nu\|v\|^2, \quad \textit{for some } \nu < 0, \quad \textit{for all } v \in \mathbf{R}^m \textit{ and all } \ell,$$

then there exists a constant κ, $0<\kappa<1$, *such that*

$$\|r(h,Q_\ell)\| \leq \kappa \textit{ for all } Q_\ell$$

for the $(2s-2,2s-2,s)$, $(2s-1,2s-1,s)$ *VCM methods.*

Proof. For each of the cases considered in the hypothesis, let p denote both order and number of steps in the method. By hypothesis, $R(z)$ defined on page 31 is the $[p-s/s]$ Padé approximant of e^z. From lemmas 1.6.2 and 1.6.3 we then have

$$\|r(h,Q_\ell)\| = \|R(hQ_\ell)\| \le \sup_{\mathrm{Re}(z)\le 0} |R(z)| \le 1. \qquad \text{when } 2s-2 \le p \le 2s.$$

The rational functions $q_j(h,z)$ are bounded along the imaginary axis and are analytic in $\mathbb{C}_-$ since they have no poles in $\mathbb{C}_-$. Therefore by lemma 1.6.3 (von Neumann theorem), there exist constants C_1, C_2 such that

$$\|q_j(h,Q_\ell)\| \le \sup_{\mathrm{Re}(z)\le 0} |q_j(h,z)| \le C_1,$$

and

$$\|q_j(h,Q_\ell)hQ_\ell\| \le \sup_{\mathrm{Re}(z)\le 0} |q_j(h,z)z| \le C_2.$$

When $p = 2s-1$ or $p = 2s-2$, $R(z)$ correspondingly is the $[s-2/s]$ or $[s-1/s]$ Padé approximant of e^z thus

$$\sup_{\mathrm{Re}(z)\le \nu} |R(z)| \le \kappa$$

with $0 \le \kappa < 1$. The last inequality of this result can be now obtained noting

$$\|r(h,Q_\ell)\| = \|R(Q_\ell)\| \le \sup_{\mathrm{Re}(z)\le \nu} |R(z)| \le \kappa.$$

□

Lemma 3.2.2 *Let $\{a_j^{(i)}, b_j^{(i)}\}$ be the respective coefficients of $(2s-2,2s-2,s)$, $(2s-1,2s-1,s)$, $(2s,2s,s)$ FIM methods satisfying (2.4.3), (2.4.4), (2.2.4) and (2.3.8) with $s>1$. If a sequence of matrices $\{Q_\ell\}$ defined over $\mathbb{C}^{m\times m}$ satisfies*

$$Re < v, Q_\ell v > \le 0, \quad \textit{for all } v \in \mathbb{C}^m \textit{ and all } \ell,$$

then for some constant C

$$\|w(h, Q_{\ell-1})d_\ell\| \;\le \begin{cases} Ch^p \int_{x_{\ell-k}}^{x_\ell} \|y^{(p+1)}(x)\|\,dx, & \text{for } \ell \ge k \\ C\Big(\max\limits_{0\le\ell<k} \|\triangle y_\ell\| + \max\limits_{0\le\ell<k} \|\triangle f_\ell\|\Big), & \text{for } 0 \le \ell < k \end{cases}$$

where p denotes the corresponding order of the method.

Proof. By definition

$$\begin{aligned} \|w(h, Q_{\ell-1})d_\ell\| &= \|h^{p+1}\sum_{i=0}^{s} w(h, Q_{\ell-1})h^iQ_{\ell-1}^i \int_0^k K_p^{(i)}(s)y^{(p+1)}(x_{\ell-k}+sh)ds\| \\ &\le h^{p+1}\sum_{i=0}^{s} C_i\|w(h, Q_{\ell-1})h^iQ_{\ell-1}^i\| \int_0^k \|y^{(p+1)}(x_{\ell-k}+sh)\|ds \\ &\le Ch^p \int_{x_{\ell-k}}^{x_\ell} \|y^{(p+1)}(x)\|dx \end{aligned}$$

We go from the second to the third inequality using the fact that $w(h,\zeta)h^i\zeta^i$ is analytic in the left half plane and bounded along the imaginary axis for $i \le s$. For $\ell < k$, we get the result by substituting d_ℓ into (3.2.3) directly. □

With essentially the same construction we also have:

Lemma 3.2.3 *Let* $\{a_j^{(i)}, b_j^{(i)}\}$ *be the respective coefficients of* $(2s-2, 2s-2, s)$, $(2s-1, 2s-1, s)$, $(2s, 2s, s)$ *LIM methods satisfying (2.4.7), (2.4.8), (2.2.4) and (2.3.8) with* $s > 1$. *If a sequence of matrices* $\{Q_\ell\}$ *defined over* $C^{m\times m}$ *satisfies*

$$Re < v, Q_\ell v > \le 0, \quad \text{for all } v \in C^m \text{ and all } \ell,$$

then for some constant C

$$\|w(h, Q_{\ell-1})d_\ell\| \;\le \begin{cases} Ch^{p-1} \int_{x_{\ell-k}}^{x_\ell} \|y^{(p)}(x)\|\,dx, & \text{for } \ell \ge k \\ C\Big(\max\limits_{0\le\ell<k} \|\triangle y_\ell\| + \max\limits_{0\le\ell<k} \|\triangle f_\ell\|\Big), & \text{for } 0 \le \ell < k \end{cases}$$

where p denotes the corresponding order of the method.

3.3 Convergence for nonlinear dissipative problems

We can now formulate convergence results for nonlinear problems.

Theorem 3.3.1 *Suppose the initial value problem (1.1.1) satisfies*

$$\mu_2[f_y(x,y)] \leq 0.$$

Assume further that the starting values are in a small neighborhood of the exact solution. Then the error for FIM methods of types $(2s-2, 2s-2, s)$, $(2s-1, 2s-1, s)$, $(2s, 2s, s)$ *satisfying (2.4.3), (2.4.4), (2.2.4) and (2.3.8) is bounded by*

$$\|y_n - y(x_n)\| \leq C\Big(\max_{0\leq j<k} \|\triangle y_j\| + \max_{0\leq j<k} \|\triangle f_j\| + h^p \int_{x_0}^{x_n} \|y^{(p+1)}(x)\|\, dx\Big)$$

provided that $hC_1M < 1$ *where the constant* C_1 *depends on the coefficients of the method and* M *depends on the continuity of the Jacobian. The constant* C *depends on* C_1, M *and the length of the integration interval.*

Proof. Denote $\tilde{d}_j := \|w(h, Q_{j-1})d_j\|$. From (3.2.7), using the mean value theorem for $\triangle f_{m-k+j}$, noting the fact that $Q_{m-1} = f_y(x_{m-1}, y_{m-1})$, and lemma 3.2.1, there exist a constant C_1 dependent on the coefficients of the method such that

$$\begin{aligned}
\|\triangle y_n\| \leq\ & hC_1\|\triangle f_n - Q_{n-1}\triangle y_n\| + hC_1\sum_{j=0}^{k-1}\|\triangle f_{n-k+j} - Q_{n-1}\triangle y_{n-k+j}\| \\
& + hC_1\sum_{m=0}^{n-1}\sum_{j=0}^{k}\|\triangle f_{m-k+j} - Q_{m-1}\triangle y_{m-k+j}\| + \sum_{m=0}^{n}\tilde{d}_m \\
\leq\ & hC_1\|f_y(x_n, y(x_n) + t_n\triangle y_n) - f_y(x_{n-1}, y_{n-1})\| \cdot \|\triangle y_n\| \\
& + hC_1\sum_{n}\sum_{j=0}^{k}\|f_y(x_{m-k+j}, y(x_{m-k+j}) + t_{m-k+j}\triangle y_{m-k+j}) \\
& \qquad - f_y(x_{m-1}, y_{m-1})\| \cdot \|\triangle y_{m-k+j}\| + \sum_{m=0}^{n}\tilde{d}_m.
\end{aligned}$$

Suppose there exists a constant M such that for $0 < t_j < 1$,

$$\| f_y(x_{m-k+j}, y(x_{m-k+j}) + t_{m-k+j} \triangle y_{m-k+j}) - f_y(x_{m-1}, y_{m-1}) \| \leq M,$$

then

$$\| \triangle y_n \| \leq hC_1 M \| \triangle y_n \| + h(k+1) C_1 M \sum_{m=0}^{n-1} \| \triangle y_m \| + \sum_{m=0}^{n} \bar{d}_m.$$

From the assumption $hC_1 M < 1$, we have

$$\| \triangle y_n \| \leq h \frac{(k+1) C_1 M}{1 - hC_1 M} \sum_{m=0}^{n-1} \| \triangle y_m \| + \frac{1}{1 - hC_1 M} \sum_{m=0}^{n} \bar{d}_m.$$

Let $M^* := \dfrac{(k+1) C_1 M}{1 - hC_1 M}$ and $L^* := \dfrac{1}{1 - hC_1 M} \sum_{m=0}^{n} \bar{d}_m$, the above can then be written as,

$$\| \triangle y_n \| \leq hM^* \sum_{m=0}^{n-1} \| \triangle y_m \| + L^*.$$

By induction, and noting $nh = x_n - x_0$ and for $j \leq n$, $\dfrac{1}{1 - hC_1 M} \sum_{m=0}^{j} \bar{d}_m \leq \dfrac{1}{1 - hC_1 M} \sum_{m=0}^{n} \bar{d}_m = L^*$,

$$\begin{aligned}
\| \triangle y_n \| &\leq (hM^* + 1)^n (hM^* \| \triangle y_0 \| + L^*) \\
&\leq [\exp(hM^*)]^n (hM^* \| \triangle y_0 \| + L^*) \\
&= \exp[(x_n - x_0) M^*] (hM^* \| \triangle y_0 \| + L^*) \\
&= \frac{\exp[(x_n - x_0) M^*]}{1 - hC_1 M} \Big(hkC_1 M \| \triangle y_0 \| + \sum_{m=0}^{n} \bar{d}_m \Big).
\end{aligned}$$

The result of the theorem now can be obtained by lemma 3.2.2. □

Similarly as above, we can get the theorem for LIM methods.

Theorem 3.3.2 *Suppose the initial value problem (1.1.1) satisfies*

$$\mu_2\left[f_y(x,y)\right] \le 0.$$

Assume further that starting values are in a small neighborhood of the exact solution. Then the error for LIM methods of types $(2s-2, 2s-2, s)$, $(2s-1, 2s-1, s)$, $(2s, 2s, s)$ *satisfying (2.4.7), (2.4.8), (2.2.4) and (2.3.8) is bounded by*

$$\|y_n - y(x_n)\| \le C\left(\max_{0\le j<k} \| \triangle\, y_j\| + \max_{0\le j<k} \| \triangle\, f_j \| + h^{p-1} \int_{x_0}^{x_n} \|y^{(p+1)}(x)\|\, dx\right)$$

where the constant C depends on the continuity of the Jacobian of the problem, the coefficients of the methods and the length of the integration interval.

Proof. Similar as proof of theorem 3.3.1, but use (3.2.6) instead of (3.2.7),

$$\| \triangle\, y_n\| \le hC_1 \sum_{m=0}^{n} \sum_{j=0}^{k-1} \|f_y\left(x_{m-k+j}, y(x_{m-k+j}) + t_{m-k+j} \triangle\, y_{m-k+j}\right)$$
$$- f_y(x_{m-1}, y_{m-1})\| \cdot \| \triangle\, y_{m-k+j}\| + \sum_{m=0}^{n} \bar{d}_m.$$

Suppose there exists a constant M such that for $0 < t_j < 1$,

$$\|f_y\left(x_{m-k+j}, y(x_{m-k+j}) + t_{m-k+j} \triangle\, y_{m-k+j}\right) - f_y(x_{m-1}, y_{m-1})\| \le M,$$

then

$$\| \triangle\, y_n\| \le hkC_1 M \sum_{m=0}^{n-1} \| \triangle\, y_m\| + \sum_{m=0}^{n} \bar{d}_m.$$

Let $M^* := kC_1 M$ and $L^* := \sum_{m=0}^{n} \bar{d}_m$, the above then can be written as,

$$\| \triangle\, y_n\| \le hM^* \sum_{m=0}^{n-1} \| \triangle\, y_m \| + L^*.$$

By induction,

$$\begin{aligned} \| \triangle y_n\| &\leq (hM^* + 1)^n (hM^* \| \triangle y_0\| + L^*) \\ &\leq \exp[(x_n - x_0)M^*](hM^* \| \triangle y_0\| + L^*). \end{aligned}$$

The result of the theorem now can then be obtained by lemma 3.2.3. □

As we noted in remark 2.4.4, the coefficients of VCM methods are continous functions of $r = h_2/h_1$ when two stepsize are in use. We can choose positive constants ω, Ω with $\omega < 1$ and $\Omega > 1$ such that all coefficients are bounded if we restrict the stepsize ratio $\omega \leq r \leq \Omega$. So $h_2/h_1 \geq \omega/\Omega$ when $h_2 \to 0$. The theorems in this section can be extended to variable stepsize VCM methods without difficulty.

3.4 Convergence for singular perturbation problems

We consider convergence when VCM methods are applied to the autonomous singular perturbation problems (3.1.1). In the event that the Lipschitz assumption on $f(y, z)$ cannot be met, it becomes impossible to provide convergence estimates independent of ε. One could tailor the convergence results of section 3.3, albeit they would provide little insight. This is due to the dependence on ε that results from this approach. So we suppose that the singular perturbation problem has the properties

- $f(y, z)$ satisfies a Lipschitz condition with moderate Lipschitz constant.
- $\mu_2[g_z] \leq \nu < 0$.

Let $u := (y, z)^T \in \mathbf{R}^{2m}$ and $F(u) := (f(u), \varepsilon^{-1}g(u))^T : \mathbf{R}^{2m} \to \mathbf{R}^{2m}$. The problem (3.1.1) can be written simply as

$$u' = F(u), \qquad u(x_0) = u_0. \tag{3.4.1}$$

The Jacobian matrix of the problem is

$$F_u = \begin{pmatrix} f_y & f_z \\ \varepsilon^{-1}g_y & \varepsilon^{-1}g_z \end{pmatrix} = \begin{pmatrix} I & 0 \\ 0 & \varepsilon^{-1}I \end{pmatrix} \begin{pmatrix} f_y & f_z \\ g_y & g_z \end{pmatrix}.$$

When ε is small, the eigenvalues of g_z may be the dominant eigenvalues of the system (3.1.1) (Hairer and Wanner (1991, pp 411)). Since Q in the VCM formulae does not affect the order, we base our analysis on the alternative selection

$$Q := \begin{pmatrix} 0 & 0 \\ 0 & \varepsilon^{-1}g_z \end{pmatrix}$$

for capturing the characteristic of the problems. It is equivalent to the following scheme:

$$\begin{aligned} \sum_{j=0}^{k} a_j^{(0)} y_{n+j} &= h \sum_{j=0}^{k} b_j^{(0)} f_{n+j} \\ \sum_{j=0}^{k} \left[\sum_{i=0}^{s} a_j^{(i)} h^i Q_{2,n+k-1}^i \right] z_{n+j} &= \frac{h}{\varepsilon} \sum_{j=0}^{k} \left[\sum_{i=0}^{s-1} b_j^{(i)} h^i Q_{2,n+k-1}^i \right] g_{n+j} \end{aligned} \tag{3.4.2}$$

where

$$Q_{2,n+k-1} = \frac{1}{\varepsilon} g_z(y_{n+k-1}, z_{n+k-1}),$$

$$f_{n+j} = f(y_{n+j}, z_{n+j}), \quad g_{n+j} = g(y_{n+j}, z_{n+j}).$$

Since

$$a_j^{(0)} = 0 \quad \text{for } j = 0, 1, \ldots, k-2, \quad a_{k-1}^{(0)} = -1, \quad a_k^{(0)} = 1,$$

then y_n is being solved by a classical Adams method which is zero-stable.

We insert the exact solution of (3.1.1) into the method (3.4.2) and by manipulations similar to those in section 3.2, the relations for $\triangle y_n$ and $\triangle z_n$ can be written as

$$\triangle y_n = \sum_{m=0}^{n}\left[\prod_{\ell=m}^{n-1} r(h,O)\right]\sum_{j=0}^{k} q_j(h,O)\triangle f_{m-k+j} + \sum_{m=0}^{n}\left[\prod_{\ell=m}^{n-1} r(h,O)\right] w(h,O)d_m, \qquad (3.4.3)$$

and

$$\triangle z_n = \sum_{m=0}^{n}\left[\prod_{\ell=m}^{n-1} r(h,Q_{2,\ell})\right]\frac{h}{\varepsilon}\sum_{j=0}^{k} q_j(h,Q_{2,m-1})[\triangle g_{m-k+j} - g_z(y_{m-1},z_{m-1})\triangle z_{m-k+j}] + \sum_{m=0}^{n}\left[\prod_{\ell=m}^{n-1} r(h,Q_{2,\ell})\right] w(h,Q_{2,m-1})e_m \qquad (3.4.4)$$

where the relation for $\triangle y_n$ can be explained as chosing the zero matrix O in the corresponding VCM formula. The following definitions are similar to those in section 3.2:

$$\triangle y_{n+j} := y(x_{n+j}) - y_{n+j}, \qquad \triangle z_{n+j} := z(x_{n+j}) - z_{n+j}$$

and

$$\triangle f_{n+j} := f(y(x_{n+j}), z(x_{n+j})) - f(y_{n+j}, z_{n+j})$$

$$\triangle g_{n+j} := g(y(x_{n+j}), z(x_{n+j})) - g(y_{n+j}, z_{n+j}).$$

As usual we define for $j < 0$

$$\triangle y_j := 0, \quad \triangle f_j := 0, \quad \triangle z_j := 0, \quad \triangle g_j := 0.$$

Lemma 3.4.1 *Let $\{a_j^{(i)}, b_j^{(i)}\}$ be the respective coefficients of $(2s-2,2s-2,s)$, $(2s-1,2s-1,s)$ VCM methods satisfying (2.1.2), (2.2.4) and (2.3.8) with $s > 1$. If* *the logarithmic norm of the sequence of matrices* $\{Q_\varepsilon\}$ *defined over* $\mathbb{C}^{m\times m}$ *satisfies* $\mu[Q_\varepsilon] \le \nu < 0$, *then for any constants* h, ε, $h \ge \varepsilon$ *there exist constants* C, κ, $0 < \kappa < 1$, *such that*

$$\|r(h, \frac{1}{\varepsilon}Q_\varepsilon)\| \le \kappa, \qquad \|\frac{h}{\varepsilon}q_j(h, \frac{1}{\varepsilon}Q_\varepsilon)\| \le C, \quad j = 1, 2, \ldots, k.$$

Proof. Using lemma 1.6.3 (von Neumann theorem) and theorem 2.3.1 of Dekker & Verwer (1984, p. 43), we have

$$\|r(h, \frac{1}{\varepsilon}Q_\varepsilon)\| \le \|R(\frac{h}{\varepsilon}Q_\varepsilon)\| \le \sup_{Re(z)\le h\nu/\varepsilon} |R(z)| \le \sup_{Re(z)\le\nu} |R(z)| \le \kappa < 1, \qquad \text{for} \quad h \ge \varepsilon.$$

From the condition $< v, Q_\varepsilon v > \le \mu_2[Q_\varepsilon] \le \nu < 0$ we know Q_ε^{-1} exists,

$$\begin{aligned}
\frac{h}{\varepsilon}\|q_j(h, \frac{1}{\varepsilon}Q_\varepsilon)\| &= \frac{h}{\varepsilon}\|\Big[\sum_{i=0}^{s}(a_k^{(i)} - b_k^{(i-1)})h^i(\frac{1}{\varepsilon}Q_\varepsilon)^i\Big]^{-1}\sum_{i=0}^{s-1} b_j^{(i)}h^i(\frac{1}{\varepsilon}Q_\varepsilon)^i\| \\
&= \|\Big[\sum_{i=0}^{s}(a_k^{(i)} - b_k^{(i-1)})h^i(\frac{1}{\varepsilon}Q_\varepsilon)^i\Big]^{-1}\sum_{i=0}^{s-1} b_j^{(i)}h^{i+1}(\frac{1}{\varepsilon}Q_\varepsilon)^{i+1}Q_\varepsilon^{-1}\| \\
&\le \|Q_\varepsilon^{-1}\| \sup_{\mathrm{Re}(z)\le 0} |zq_j(h, z)|.
\end{aligned}$$

The second inequality is now proved since $\|Q_\varepsilon^{-1}\|$ is bounded as $\|Q_\varepsilon^{-1}\| \le -(\mu_2[Q_\varepsilon])^{-1} \le -\nu^{-1}$, ν is a negative constant, $q_j(h, z)$ has no poles in $\mathbb{C}_-$ and $|zq_j(h, z)|$ is also bounded in $\mathbb{C}_-$. $\square$

Lubich (1991) gives a convergence result with restrictions on the eigenvalues of $g_z(y, z)$ for reasons of stability of the employed underlying multistep method. We can give essentially the same result except relax the eigenvalue restrictions given the stronger stability properties of the VCM methods considered here.

Theorem 3.4.2 *Assume the logarithmic matrix norm of g_z satisfies*

$$\mu_2[g_z(y,z)] \leq \nu < 0.$$

Assume further that starting values are in a sufficiently small h– and ε– independent neighborhood of the exact solution and that $h \leq h_0$ with h_0 sufficiently small but independent of ε. Then the error for FIM methods of types $(2s-2, 2s-2, s)$, $(2s-1, 2s-1, s)$ satisfying (2.4.3), (2.4.4), (2.2.4) and (2.3.8) is bounded for $h \geq \varepsilon$ by

$$\begin{aligned}
&\|y_n - y(x_n)\| + \|z_n - z(x_n)\| \\
&\leq C\Big(h^p \int_{x_0}^{x_n} \|y^{(p+1)}(x)\|\,dx + h^p \int_{x_0}^{x_n} \|z^{(p+1)}(x)\|\,dx \\
&\qquad + \max_{0\leq j<k} \|\triangle y_j\| + \max_{0\leq j<k} \|\triangle f_j\| + (h+\rho^n)\Big[\max_{0\leq j<k} \|\triangle z_j\| + \max_{0\leq j<k} \|\triangle g_j\|\Big]\Big)
\end{aligned}$$

with $0 < \rho < 1$. The constants C and ρ are independent of ε and h.

Proof. Define

$$\tilde{d}_j := w(h, O)d_j, \qquad \tilde{e}_j := w(h, Q_{2,j-1})e_j.$$

From (3.4.3), using the assumed smoothness of $f(y,z)$ and lemma 3.2.1, there exist constants M, N such that

$$\|\triangle y_n\| \leq h\sum_{j=0}^{n}\Big(M\|\triangle y_j\| + N\|\triangle z_j\|\Big) + \sum_{j=0}^{n}\|\tilde{d}_j\|. \tag{3.4.5}$$

From (3.4.4), using lemma 3.4.1, there exist constants L, ε_1, κ such that

$$\|\triangle z_n\| \leq \sum_{j=0}^{n}\kappa^{n-j}\Big(L\|\triangle y_j\| + \varepsilon_1\|\triangle z_j\|\Big) + \sum_{j=0}^{n}\kappa^{n-j}\|\tilde{e}_j\|. \tag{3.4.6}$$

The rest of the proof is along the similar lines as in Hairer & Wanner (1991, p.412-415). Now define the sequence $\{u_n\}$, $\{v_n\}$ by

$$\begin{aligned}
u_n &:= h\sum_{j=0}^{n}\Big(Mu_j + Nv_j\Big) + \sum_{j=0}^{n}\|\tilde{d}_j\|, \\
v_n &:= \sum_{j=0}^{n}\kappa^{n-j}\Big(Lu_j + \varepsilon_1 v_j\Big) + \sum_{j=0}^{n}\kappa^{n-j}\|\tilde{e}_j\|.
\end{aligned} \tag{3.4.7}$$

By induction one can show that

$$\| \triangle y_n \| \leq u_n, \qquad \| \triangle z_n \| \leq v_n.$$

provided $\varepsilon_1 < 1$, and $h < h_0$. Rewrite (3.4.7) as

$$\begin{aligned} u_n &= u_{n-1} + hMu_n + hNv_n + \| \tilde{d}_n \| \\ v_n &= \kappa v_{n-1} + Lu_n + \varepsilon_1 v_n + \| \tilde{e}_n \| \end{aligned} \tag{3.4.8}$$

Solving for u_n, v_n, we get

$$\begin{pmatrix} u_n \\ v_n \end{pmatrix} = A(h) \begin{pmatrix} u_{n-1} \\ v_{n-1} \end{pmatrix} + \begin{pmatrix} \hat{d}_n \\ \hat{e}_n \end{pmatrix} \tag{3.4.9}$$

where

$$A(h) = \begin{pmatrix} 1 + O(h) & O(h) \\ O(1) & \rho + O(h) \end{pmatrix}$$

and

$$\| \hat{d}_n \| \leq C_1 \Big(\| \tilde{d}_n \| + \| \tilde{e}_n \| \Big), \qquad \| \hat{e}_n \| \leq C_2 \Big(\| \tilde{d}_n \| + \| \tilde{e}_n \| \Big).$$

Inserting to (3.4.9) repeatedly we have

$$\begin{pmatrix} u_n \\ v_n \end{pmatrix} = \sum_{j=0}^{n} A(h)^{n-j} \begin{pmatrix} \hat{d}_j \\ \hat{e}_j \end{pmatrix} \tag{3.4.10}$$

If ε_1 is small enough so that $\rho := \dfrac{\kappa}{(1-\varepsilon_1)} < 1$ and if $h \leq h_0$, then the eigenvalues of $A(h)$ are distinct and $A(h)$ can be diagonalized as

$$A(h) = T^{-1}(h) \begin{pmatrix} 1 + O(h) & 0 \\ 0 & \rho + O(h) \end{pmatrix} T(h), \qquad T(h) = \begin{pmatrix} 1 & O(h) \\ O(h) & 1 \end{pmatrix}$$

Inserted into (3.4.10) the latter yields

$$u_n + v_n \leq \text{constant} \cdot \left(\sum_{j=1}^{n} \hat{d}_j + \sum_{j=1}^{n} (h + \rho^{n-j}) \hat{e}_j \right)$$

where $\rho = \kappa/(1-\varepsilon_1)$. This is done using the continuity of $g_z(y, z)$ and knowing ε_1 can be made arbitrarily small by assumption provided h itself is sufficiently small.

The statement of the theorem now follows from lemma 3.2.2 and the definitions of $\bar{d}_j$, $\bar{e}_j$. □

Similarly we have

Theorem 3.4.3 *Under the assumptions of theorem 3.4.2, the error for LIM methods of types* $(2s-2, 2s-2, s)$, $(2s-1, 2s-1, s)$ *satisfying (2.4.7), (2.4.8), (2.2.4) and (2.3.8) is bounded for* $h \geq \varepsilon$ *by*

$$\begin{aligned}
&\|y_n - y(x_n)\| + \|z_n - z(x_n)\| \\
&\leq C\Big(h^{p-1}\int_{x_0}^{x_n}\|y^{(p+1)}(x)\|\,dx + h^{p-1}\int_{x_0}^{x_n}\|z^{(p)}(x)\|\,dx \\
&\qquad + \max_{0\leq j<k}\|\triangle y_j\| + \max_{0\leq j<k}\|\triangle f_j\| + (h+\rho^n)\Big[\max_{0\leq j<k}\|\triangle z_j\| + \max_{0\leq j<k}\|\triangle g_j\|\Big]\Big)
\end{aligned}$$

with $0 < \rho < 1$. *The constants* C *and* ρ *are independent of* ε *and* h.

Remark **3.4.4** Theorems 3.4.2–3.4.3 also hold for corresponding variable step size methods in as much as $h_2 \to 0$ while the ratio $h_2/h_1 \geq \omega > 0$.

3.5 Example

We have tested the $(5, 5, 3)$ method on van der Pol's equation

$$\begin{aligned}
y_1' &= y_2 & y_1(0) &= 2 \\
\varepsilon y_2' &= (1 - y_1^2)y_2 - y_1 & y_2(0) &= -0.66.
\end{aligned} \tag{3.5.1}$$

integrated over $[0, 2]$ and set $\varepsilon = 10^{-6}$. We note that the assumptions on smoothness of the functions f and g in the begining of section 3.4 are satisfied by this particular SPP in all of the integration interval except $\mu_2[g_z] > 0$ in two narrow transient intervals.

Table 3.5.1: Results of FIM (5,5,3) code for van der Pol's equation

Selection	Tol	Nstep	Nsc	Nfe	Nje	Nlu	Eabs
SQ_1	10^{-6}	1281	294	2614	296	604	1.42×10^{-7}
	10^{-3}	553	137	1478	139	370	3.23×10^{-4}
SQ_2	10^{-6}	1316	301	2700	303	622	7.32×10^{-7}
	10^{-3}	546	129	1424	131	344	3.23×10^{-4}

The description of code is again left to chapter 4 where we always choose the matrix Q in VCM formulae as the full Jacobian of the corresponding problem. For van der Pol's equation, the full Jacobian is

$$Q = \begin{pmatrix} f_y & f_z \\ \varepsilon^{-1} g_y & \varepsilon^{-1} g_z \end{pmatrix} .$$

Whereas we also test here the setting

$$Q = \begin{pmatrix} 0 & 0 \\ 0 & \varepsilon^{-1} g_z \end{pmatrix}$$

according to our analysis in section 3.4. Denote the former and the latter setting Q respectively as SQ_1, SQ_2. Illustrated in figure 3.5.1 is the y_1 component of the solution and the stepsizes used in the computation. We list the results in table 3.5.1.

The entries noted in the table correspond to:

- Tol: local error tolerance (we use scalar error control here and set both absolute and relative error tolerance equal to Tol),
- Nstep: number of steps,
- Nsc: number of stepsize changes,
- Nfe: number of function evaluations,

- Nje: number of Jacobian evaluations,
- Nlu: number of LU-decomposition,
- Eabs: actual absolute error at end point.

We can see from the table the numbers of function and Jacobian evaluations and of LU-decompositions are similar for both selections. But the code with selection SQ_2 is overall much faster given the simpler structure of Q for this case.

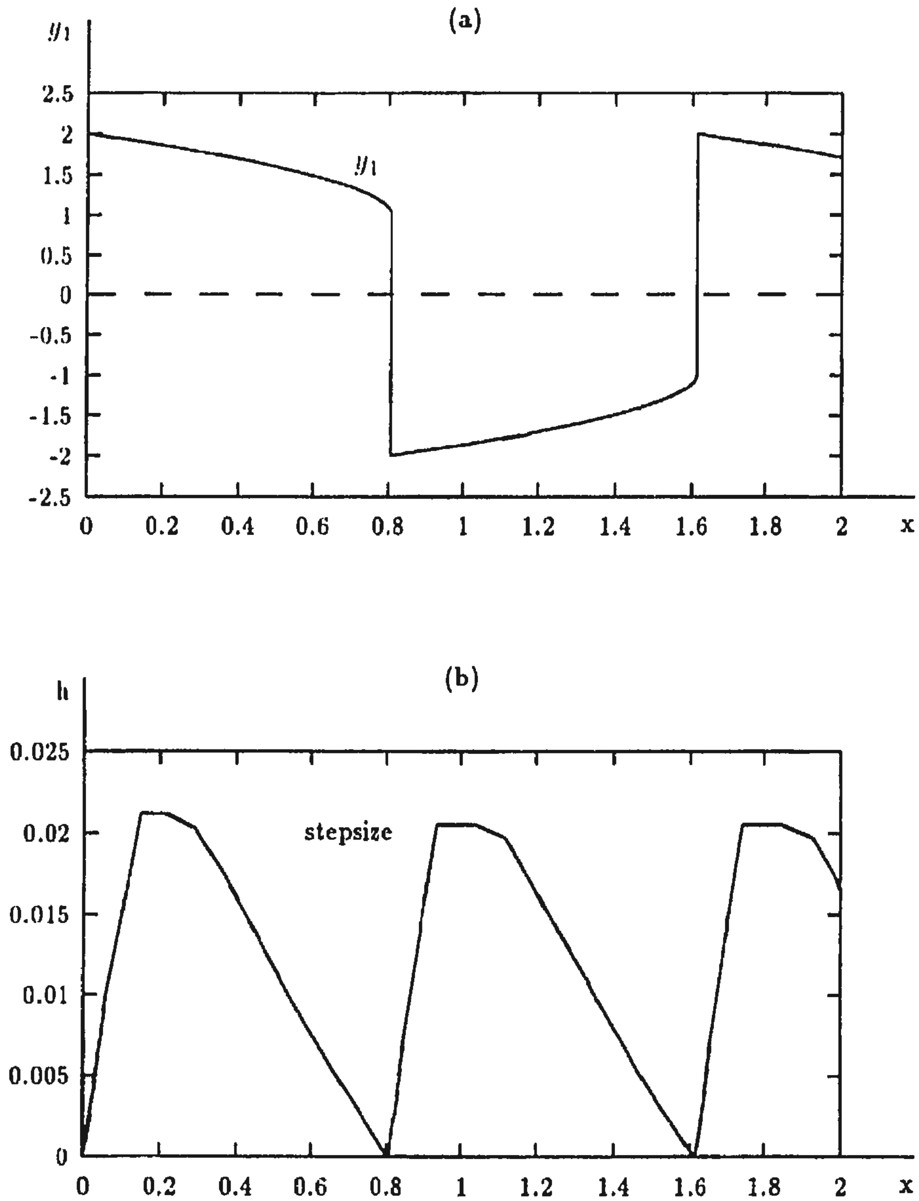

Figure 3.5.1: van der Pol's equation

Chapter 4

Implementation and numerical testing

4.1 Implementation

To use the VCM methods efficiently, we implement them in variable stepsize form and an increase in stepsize is considered only if the stepsize has been constant for $k-1$ steps (where k is the number of steps in the formula being used).

Even though the powers of the Jacobian matrix appear in the formula, one should always avoid matrix multiplication in a practical implementation. The number of operations in matrix multiplication is approximately n^3 while the work in a real LU-decomposition is about $n^3/3$ operations. Moreover, as is often in licated in the literature (eg. see Enright 1973) matrix multiplication destroys sparseness, whereas LU-decomposition retains sparsity. Since many large dimensional problems appear in sparse form, retaining this structure is quite important.

We can avoid matrix multiplication by factoring the denominator polynomial of Padé approximant to e^z. For implementation, we rewrite (2.1.1) in the form

$$\left[\sum_{i=0}^{s} a_k^{(i)} h^i Q_n^i\right] y_{n+k} = \sum_{i=0}^{s} h^i Q_n^i \sum_{j=0}^{k-1} \left[-a_j^{(i)} y_{n+j} + b_j^{(i)} h f_{n+j}\right]. \tag{4.1.1}$$

Furthermore, we factor the left hand side of (4.1.1) and represent the right hand side using the nested multiplication scheme, that is,

$$\begin{aligned}
& w_0(hQ_n - w_1 I)(hQ_n - w_2 I)\cdots(hQ_n - w_s I) y_{n+k} \\
&= \sum_{j=0}^{k-1}\left[-a_j^{(0)} y_{n+j} + b_j^{(0)} h f_{n+j}\right] \\
&\quad + hQ_n\Bigg\{\sum_{i=0}^{k-1}\left[-a_j^{(1)} y_{n+j} + b_j^{(1)} h f_{n+j}\right] \\
&\qquad + hQ_n\Bigg\{\sum_{i=0}^{k-1}\left[-a_j^{(2)} y_{n+j} + b_j^{(2)} h f_{n+j}\right] \\
&\qquad\quad + \cdots \\
&\qquad\qquad + hQ_n \sum_{j=0}^{k-1}\left[-a_j^{(s)} y_{n+j} + b_j^{(s)} h f_{n+j}\right]\Bigg\}\cdots\Bigg\}
\end{aligned} \tag{4.1.2}$$

Now for the right hand side, there are only matrix-vector multiplications involved. To solve the above linear system, we use the approach suggested by Willoughby (see Enright (1973)). For a complex pair w, $\bar{w}$, w with non-zero imaginary part, consider solving the problem

$$(hQ - wI)(hQ - \bar{w}I)y = b \tag{4.1.3}$$

where y and b are real vectors, Q is a real matrix. This equation can be solved using a single complex LU decomposition. Setting

$$(hQ - \bar{w}I)y = z,$$

we can determine z by solving the complex system

$$(hQ - wI)z = b$$

Since both hQ and y are real, we can equate imaginary parts of (4.1.3) and obtain

$$y = \mathrm{Im}(z)/\mathrm{Im}(w).$$

Thus for each real root, we need a real LU-decomposition. For a pair of complex roots, we need a single complex LU-decomposition which is four times as much work as a real LU-decomposition. If the corresponding formula has s_1 real roots and s_2 complex pairs of roots, we can solve (4.1.1) by s_1 real LU-decompositions, s_2 complex LU-decompostions and $s_1 + s_2$ backsolves. For example, we need one real LU-decomposition for a method with $s = 1$, one complex LU-decomposition for a method with $s = 2$, one real and one complex when $s = 3$.

Finally, in our codes Jacobian updates are performed only when the stepsize is changed.

To implement a variable stepsize method efficiently, one needs a stepsize changing strategy and an esvimate of the local error at each step or at several steps.

As is commonly done, we use the following formula for chosing a new stepsize.

$$hnew = hold \cdot fac \cdot \left(\frac{tol}{err}\right)^{\frac{1}{k+1}}$$

where $hold, hnew$ are old and new stepsizes respectively, tol is the error tolerance, err is the estimated local error and fac is a safety factor which is usually chosen between 0 and 1 (we use $fac = 0.8$).

Two popular ways to implement variable stepsizes method are:

- use one formula with two stepsize sequences, obtain the error estimate by local extrapolation;

- use a pair of methods which belong to a family with differer.t order (eg., some classical RK pairs) or with same order (eg. predictor-corrector methods).

We implement the above two different strategies for the fourth and fifth order LIM methods, we also implement FIM methods of the fourth and fifth order in the predictor-corrector form. We then test the three codes on a set of 17 problems. Now we begin with a description of these codes.

Code a: This approach is based on the classical idea of Richardson extrapolation (see for example, Atkinson 1989). Let p denote the order of the method in use. For given stepsize h and x_{n+k-1} at which the solution has been accepted, the code will compute two steps at stepsize h to give an estimate

$$y_{n+k+1} = y(x_{n+k} + 2h) + C_1 h^{p+1}.$$

Then a double step from x_n of size $2h$ is taken to give another estimate

$$\hat{y}_{n+k+1} = y(x_{n+k} + 2h) + C_1 (2h)^{p+1}.$$

If the above two estimates are used to solve for the error term, we obtain

$$C_1 h^{p+1} = \frac{\hat{y}_{n+k+1} - y_{n+k+1}}{2^{p+1} - 1}.$$

The above can serve as an estimate of the local error and we can also *locally extrapolate* the solution to

$$\bar{y}_{n+k+1} = y_{n+k+1} + \frac{\hat{y}_{n+k+1} - y_{n+k+1}}{2^{p+1} - 1}.$$

If the error tolerances are met on every second step, the intermediate steps are also accepted without a second explicit error check for them. Thus the overall method cannot have order greater than $p+1$ since local extrapolation is performed only every second step. Methods of type (4, 4, 2) and (5, 5, 3) have been coded. Each of these uses methods of orders one and two for generating the necessary start-up data and thus are self-starting. For the (4, 4, 2) code, stepsize changes are considered after every four steps and the Richardson extrapolation scheme blends in well with this requirement. In order to allow for stepsize change to occur after every five steps in the (5, 5, 3) code, a slight modification to the strategy is needed. Our approach used here is to compute Richardson extrapolation with two half steps/one full step combination for the first step, then a pair of double step calculations thus generating a complement of five consecutive steps with no more than two stepsizes in use. Finally, at order p, a stepsize change is considered after p consecutive steps only if the stepsize control formula suggests a step ratio greater than 1.1 for the next set of p steps. In the data presented later, we note the large number of LU factorizations this code necessitates. In actual fact, about half of these factorizations could be spared with further allocation of memory to the code. The codes used here are a modification of those presented in Charron (1993) which in their original design permit stepsize changes to occur as frequently as on every step.

Code b: The local error is estimated by a pair of LIM methods which are both k step methods with different free parameters chosen to determine coefficients of the methods. The coefficients of one method are determined according to Theorem 2.4.1–2.4.3 to minimize the complexity of the error term and this is used as the main method to calculate the approximation values of y_n. A second selection of parameters gives a formula whose approximations are used only for error estimation. According to theorem 2.4.3 with the usual localizing assumption ($y_{n+j} = y(x_{n+j}), j = 0, 1, \ldots, k-1.$)

in place, we have

$$y(x_{n+k}) - y_{n+k} = \sum_{i=0}^{q} h^i Q_n^i (c_i h^p y^{(p)}(x_{n+k}) + \mathcal{O}(h^{p+1})) \tag{4.1.4}$$

whereas with $\tilde{y}_{n+k}$

$$y(x_{n+k}) - \tilde{y}_{n+k} = \sum_{i=0}^{q} h^i Q_n^i (\tilde{c}_i h^{p-i} y^{(p-i)}(x_{n+k}) + \mathcal{O}(h^{p-i+1})) \tag{4.1.5}$$

since the selection of parameters for $\tilde{y}_{n+k}$ satisfies (2.2.2)–(2.2.3) only. When working with stiff problems, $\|Q_n\|$ is not of moderate size and therefore the principle contributions to the local error term in (4.1.5) originate with the terms whose index satisfies $i \geq 1$. Thus the difference $\tilde{y}_{n+k} - y_{n+k}$ obtained from the difference of equations (4.1.4) and (4.1.5) above will provide an estimate for the local truncation error expressed in (4.1.5). We can see from (4.1.2) that the left hand side is independent of the parameter selection, so the pair can share the same LU decompositions. The codes start from low order methods $(1, 2, \ldots, k-1)$, so they are also self-starting.

Code c: Charron (1993) tested VSFIM methods, which are variable stepsize FIM methods without the restriction on only two stepsizes in the current k-steps, in the predictor-corrector form where the nonlinear equations in VSFIM formulae are solved by fixed point iteration. He noticed VSFIM performs better than VSLIM code and has more reliable local error estimate as

$$\left[w_0 \prod_{i=0}^{q} (hQ_n - w_i I)\right](y(x_n) - y_n) = \sum_{i=0}^{q} h^i Q_n^i \left[\sum_{j=0}^{k-i} a_j^{(i)} (y_n - \phi_{n-1,q}^{(i)})\right]$$

where $\phi_{n-1,q}^{(i)}$ can be expressed as an integration of the polynomial to the data $\{(x_{n-j}, f_{n-j})\}$, $j = 0, 1, \ldots, q \leq k-1$.

Here we consider implementing VCM methods in the predictor-corrector form, with the nonlinear equations solved by simplified Newton iteration. For VCM methods, (4.1.1) can be adjusted to

$$\left[\sum_{i=0}^{s} a_k^{(i)} h^i Q_n^i\right] y_{n+k} - h\left[\sum_{i=0}^{s-1} b_k^{(i)} h^i Q_n^i\right] f_{n+k}$$
$$= \sum_{i=0}^{s} h^i Q_n^i \sum_{j=0}^{k-1}\left[-a_j^{(i)} y_{n+j} + b_j^{(i)} h f_{n+j}\right]. \qquad (4.1.6)$$

This is a nonlinear system for y_{n+k} since $f_{n+k} = f(x_{n+k}, y_{n+k})$. Applying Newton iteration to (4.1.6), we have

$$\left\{\left[\sum_{i=0}^{s} a_k^{(i)} h^i Q_n^i\right] - h\left[\sum_{i=0}^{s-1} b_k^{(i)} h^i Q_n^i\right] f_y(x_{n+k}, y_{n+k})\right\}\left(y_{n+k}^{[m+1]} - y_{n+k}^{[m]}\right)$$
$$= -\left[\sum_{i=0}^{s} a_k^{(i)} h^i Q_n^i\right] y_{n+k}^{[m]} + h\left[\sum_{i=0}^{s-1} b_k^{(i)} h^i Q_n^i\right] f_{n+k}^{[m]} + \sum_{i=0}^{s} h^i Q_n^i \sum_{j=0}^{k-1}\left[-a_j^{(i)} y_{n+j} + b_j^{(i)} h f_{n+j}\right].$$

We replace the Jacobian $f_y(x_{n+k}, y_{n+k})$ evaluated at x_{n+k} by Q_n, which is an approximation of the Jacobian. Then the simplified Newton iterations for (4.1.6) becomes

$$\left[\sum_{i=0}^{s}(a_k^{(i)} - b_k^{(i-1)}) h^i Q_n^i\right]\left(y_{n+k}^{[m+1]} - y_{n+k}^{[m]}\right)$$
$$= \sum_{i=0}^{s} h^i Q_n^i\left[-a_k^{(i)} y_{n+k}^{[m]} + b_k^{(i)} h f_{n+k}^{[m]}\right] + \sum_{i=0}^{s} h^i Q_n^i \sum_{j=0}^{k-1}\left[-a_j^{(i)} y_{n+j} + b_j^{(i)} h f_{n+j}\right] \qquad (4.1.7)$$

Note that if the predictor and corrector are of the same order, then the matrices in the left hand side of (4.1.1) and (4.1.7) are equal. We use a LIM as predictor and the same order FIM as corrector, so the code can share the LU-decomposition. For simplicity, we adopt the PECE mode described in Lambert (1991, p.104) where only one iteration for the corrector is allowed at each step.

Error control: All three codes use a weighted error control as in the popular code LSODE. Define the error weight as

$$ewt[j] := atol[j] + rtol[j] \cdot abs(y[j]), \quad \text{for } j = 1, 2, \ldots, m$$

where $atol$ and $rtol$ are absolute and relative error tolerances respectively. If in the current step, $err[j]/ewt[j] \leq 1$ for $j = 1, 2, \ldots, m$, we accept the result at that step, otherwise, it is a rejected step. When all $atol[j]$ are equal and $rtol[j]$ are equal, we essentially have scalar error control. Scalar error control is suitable for most of our test problems. For simplicity, we always set $atol[j] = rtol[j] = Tol$ for all test problems in next section except for Robertson problem.

4.2 Numerical testing results

The first extensive test set for stiff problems was presented by Enright et al (1975), and was later supplemented by Enright & Hull (1976). Byrne & Hindmarsh (1987) presented another test set which contains 10 test problems. We choose all 10 problems from Enright & Hull (1976) and the first 6 problems from Byrne & Hindmarsh (1987).

Problem 1. (chemical pyrolysis)

$$
\begin{aligned}
y_1' &= -7.89 \times 10^{-10} y_1 - 1.1 \times 10^7 y_1 y_3 & y_1(0) &= 1.76 \times 10^{-3} \\
y_2' &= 7.89 \times 10^{-10} y_1 - 1.13 \times 10^9 y_2 y_3 & y_2(0) &= 0 \\
y_3' &= 7.89 \times 10^{-10} y_1 - 1.1 \times 10^7 y_1 y_3 & & \\
&\quad +1.13 \times 10^3 y_4 - 1.13 \times 10^9 y_2 y_3 & y_3(0) &= 0 \\
y_4' &= 1.1 \times 10^7 y_1 y_3 - 1.13 \times 10^3 y_2 y_3 & y_4(0) &= 0
\end{aligned}
$$

Integration interval: $0 \leq t \leq 1000$.

Problem 2. (chemistry: Robertson (1966))

$$
\begin{aligned}
y_1' &= -0.04 y_1 + 0.01 y_2 y_3 & y_1(0) &= 1 \\
y_2' &= 400 y_1 - 100 y_2 y_3 - 3000 y_2^2 & y_2(0) &= 0 \\
y_3' &= 30 y_2^2 & y_3(0) &= 0
\end{aligned}
$$

Integration interval: $0 \leq t \leq 40$. This is the scaled Robertson problem obtained from Problem 11 by transform

$$y_1 = y_1,\ y_2 = 10^4 y_2,\ y_3 = 10^2 y_3.$$

Problem 3. (chemistry: Bjurel et al (1970))

$$\begin{aligned}
y_1' &= y_3 - 100 y_1 y_2 & y_1(0) &= 1\\
y_2' &= y_3 + 2y_4 - 100 y_1 y_2 - 2 \times 10^4 y_2^2 & y_2(0) &= 1\\
y_3' &= -y_3 + 100 y_1 y_2 & y_3(0) &= 0\\
y_4' &= -y_4 + 10^4 y_2^2 & y_4(0) &= 0
\end{aligned}$$

Integration interval: $0 \leq t \leq 20$.

Problem 4. (chemistry: Gear (1969))

$$\begin{aligned}
y_1' &= -0.013 y_1 - 1000 y_1 y_3 & y_1(0) &= 1\\
y_2' &= -2500 y_2 y_3 & y_2(0) &= 1\\
y_3' &= -0.013 y_1 - 1000 y_1 y_3 - 2500 y_2 y_3 & y_3(0) &= 0
\end{aligned}$$

Integration interval: $0 \leq t \leq 50$.

Problem 5. (reactor kinetics: Liniger & Willoughby (1967))

$$\begin{aligned}
y_1' &= 0.01 - [1 + (y_1 + 1000)(y_1 + 1)](0.01 + y_1 + y_2) & y_1(0) &= 0\\
y_2' &= 0.01 - (1 + y_2^2)(0.01 + y_1 + y_2) & y_1(0) &= 0
\end{aligned}$$

Integration interval: $0 \leq t \leq 100$.

Problem 6. (dynamics of a catalytic fluidized bed: Luss & Amundson (1968))

$$\begin{aligned}
y_1' &= 1.3(y_3 - y_1) + 10400 k y_2 & y_1(0) &= 761\\
y_2' &= 1880[y_4 - y_2(1 + k)] & y_2(0) &= 0\\
y_3' &= 1752 - 269 y_3 + 267 y_1 & y_3(0) &= 600\\
y_4' &= 0.1 + 320 y_2 - 321 y_4 & y_4(0) &= 0.1
\end{aligned}$$

where $k = e^{20.7-1500/y_1}$. Integration interval: $0 \leq t \leq 1000$.

Problem 7. (thermal decomposition of ozone: Lapidus et al (1974))

$$y_1' = -y_1 - y_1 y_2 + 294 y_2 \quad y_1(0) = 1$$
$$y_2' = y_1(1 - y_2)/98 - 3y_2 \quad y_2(0) = 0$$

Integration interval: $0 \leq t \leq 240$.

Problem 8. (nuclear reactor theory: Liniger & Willoughby (1967))

$$y_1' = 0.2(y_2 - y_1) \qquad y_1(0) = 0$$
$$y_2' = 10y_1 - (60 - 0.125y_3)y_2 + 0.125y_3 \quad y_2(0) = 0$$
$$y_3' = 1 \qquad y_3(0) = 0$$

Integration interval: $0 \leq t \leq 400$.

Problem 9. (oscillating chemical system: Field & Noyes (1974))

$$y_1' = s(y_2 - y_1 y_2 + y_1 - q y_1^2) \quad y_1(0) = 4$$
$$y_2' = (y_3 - y_2 - y_1 y_2)/s \qquad y_2(0) = 1.1$$
$$y_3' = w(y_1 - y_3) \qquad y_3(0) = 4$$

where

$$s = 77.27, \quad w = 0.1610, \quad q = 8.375 \times 10^{-6}.$$

whose solution over [0, 300] is desired.

Problem 10. (enzyme kinetics: Garfinkel et al (1966))

$$y_1' = 10^{11}(-3y_1 y_2 + 0.0012y_4 - 9y_1 y_3) \quad y_1(0) = 3.365 \times 10^{-7}$$
$$y_2' = -3 \times 10^{11} y_1 y_2 + 2 \times 10^7 y_4 \qquad y_2(0) = 8.261 \times 10^{-3}$$
$$y_3' = 10^{11}(-9y_1 y_3 + 0.001y_4) \qquad y_3(0) = 1.642 \times 10^{-3}$$
$$y_4' = 10^{11}(3y_1 y_2 - 0.0012y_4 + 9y_1 y_3) \qquad y_4(0) = 9.38 \times 10^{-6}$$

Integration interval: $0 \leq t \leq 100$.

The following are six problems from Byrne & Hindmarsh (1987).

Problem 11. Robertson's Problem

$$y_1' = -0.04y_1 + 10^4 y_2 y_3 \qquad y_1(0) = 1$$

$$y_2' = 0.04y_1 - 10^4 y_2 y_3 - 3 \times 10^7 y_2 y_2 \qquad y_2(0) = 0$$

$$y_3' = 3 \times 10^7 y_2 y_2 \qquad y_3(0) = 0$$

Integration interval: $0 \le t \le 4.0d7$.

Problem 12. The Field Noyes chemical oscillator. This is the same problem as Problem 9 except the integration is from 0 to 600.

Problem 13. Two Species Diurnal Kinetics

$$y_1' = -k_1 y_1 y_3 - k_2 y_1 y_2 + 2k_3(t) y_3 + k_4(t) y_2 \qquad y_1(0) = 10^6$$

$$y_2' = k_1 y_1 y_3 - k_2 y_1 y_2 - k_4(t) y_2 \qquad y_2(0) = 10^{12}$$

with

$$y_3 = 3.7 \times 10^{16}$$

$$k_1 = 1.63 \times 10^{-15}$$

$$k_2 = 4.66 \times 10^{-16}$$

$$k_i = \begin{cases} \exp[-a_i / \sin \omega t], & \sin \omega t > 0 \\ 0, & \sin \omega t \le 0 \end{cases} \qquad i = 3, 4$$

$$a_3 = 22.62, \qquad a_4 = 7.601$$

$$\omega = \pi/43200$$

and the integration interval is $0 \le t \le 8.64 \times 10^5$, or 10 days.

Problem 14. A Kidney Model

$$y_1' = a(y_3 - y_1)y_1/y_2 \qquad y_1(0) = 1.0$$

$$y_2' = -a(y_3 - y_1) \qquad y_2(0) = 1.0$$

$$y_3' = [b - c(y_3 - y_5) - ay_3(y_3 - y_1)]/y_4 \qquad y_3(0) = 1.0$$

$$y_4' = a(y_3 - y_1) \qquad y_4(0) = -10$$

$$y_5' = -c(y_5 - y_3)/d \qquad y_5(0) = 0.9$$

with

$$a = 100, \quad b = 0.9, \quad c = 1000, \quad d = 10.$$

Problem 15. A Laser Oscillator Model

$$\dot{n} = -n(\alpha\phi + \beta) + \gamma \qquad n(0) = -1$$

$$\dot{\phi} = \phi(\rho n - \sigma) + \tau(1 + n) \quad \phi(0) = 0$$

with

$$\alpha = 1.5 \times 10^{-18}, \quad \beta = 2.5 \times 10^{-6}$$

$$\gamma = 2.1 \times 10^{-6}, \quad \rho = 0.6$$

$$\sigma = 0.18, \quad \tau = 0.016.$$

The interval is $0 \le t \le 0.7 \times 10^{6}$.

Problem 16. Burgers' Equation

$$u_t + uu_x = \nu u_{xx}, \qquad 0 \le x \le 1, \quad t \ge 0$$

This is a partial differential equation with travelling wave solutions. It is discretized along the x axis with a uniform mesh and we replace all spatial derivatives by centered finite difference analogues:

$$\dot{u}_i = -(u_i/2\triangle)(u_{i+1} - u_{i-1}) + (\nu/2\triangle^2)(u_{i+1} - 2u_i + u_{i-1}), \quad i = 1, 2, \ldots, N$$

where $\triangle = \dfrac{1}{N+1}$. Initial and boundary conditions are:

$$u_i(0) = [1 + \exp(i \triangle /2\nu)]^{-1}, \qquad i = 1, 2, \ldots, N$$

$$u_0(t) = [1 + \exp(-t/4\nu)]^{-1},$$

$$u_{N+1}(t) = \left[1 + \exp\left(\tfrac{1}{2\nu} - \tfrac{t}{4\nu}\right)\right]^{-1}.$$

We chose $\nu = 0.04$, $N = 50$ in the test.

The last problem we tested is the van der Pol equation which can be classified as singular perturbation problem.

Problem 17. (van der Pol's equation: (van der Pol 1926))

$$y_1' = y_2 \qquad y_1(0) = 2$$

$$y_2' = ((1 - y_1^2)y_2 - y_1)/\varepsilon \quad y_2(0) = -0.66$$

with $\varepsilon = 10^{-6}$ on the interval $[0, 2]$.

The test results are shown on table 4.2.1– 4.2.12. The notations in the tables have the following meanings:

- Tol: local error tolerance (see page 86 for detail about the error control),
- Nstep: number of steps,
- Nsc: number of stepsizes changed,
- Nfe: number of function evaluations,
- Nje: number of Jacobian evaluations,
- Nlu: number of LU-decomposition,
- Eabs: absolute error at end point.
- Erel: relative error according to the component with the largest abstract value at end point.

The last two items are estimated by comparing our solution to that obtained by calling IMSL subroutines with a tolerance of 1.0E-10 (problem 13 is a challenging problem, only one significant figure can be obtained by calling both IMSL and NAG subroutines). We restrict the stepsize ratio $r \leq 3.5$. The numerical test is performed on a MIPS M/120s machine in double precision.

Table 4.2.1: Results of LIM (4,4,2) on problems 1–5

Problem	Code	Tol	Nstep	Nsc	Nfe	Nje	Nlu	Eabs	Erel
P1	(a)	10^{-6}	68	18	74	18	68	2.69×10^{-7}	1.66×10^{-4}
		10^{-3}	44	12	50	12	44	7.32×10^{-7}	4.53×10^{-4}
	(b)	10^{-6}	46	15	49	17	18	3.23×10^{-7}	2.07×10^{-4}
		10^{-3}	43	14	46	16	17	1.00×10^{-5}	6.63×10^{-3}
P2	(a)	10^{-6}	192	43	198	43	192	4.53×10^{-4}	1.59×10^{-5}
		10^{-3}	64	17	73	17	70	8.03×10^{-2}	2.83×10^{-3}
	(b)	10^{-6}	457	31	460	33	34	2.94×10^{-7}	1.03×10^{-8}
		10^{-3}	73	23	76	25	26	1.50×10^{-3}	5.08×10^{-5}
P3	(a)	10^{-6}	192	48	203	48	202	2.57×10^{-8}	1.08×10^{-8}
		10^{-3}	84	22	99	22	102	3.12×10^{-6}	1.03×10^{-6}
	(b)	10^{-6}	165	47	168	49	50	2.80×10^{-8}	9.61×10^{-9}
		10^{-3}	66	2[illegible]	69	23	24	8.67×10^{-6}	3.40×10^{-6}
P4	(a)	10^{-6}	56	14	62	15	56	3.85×10^{-5}	2.74×10^{-5}
		10^{-3}	36	10	42	10	36	2.00×10^{-5}	1.42×10^{-5}
	(b)	10^{-6}	32	10	35	12	13	5.56×10^{-6}	3.84×10^{-6}
		10^{-3}	31	10	34	12	13	9.31×10^{-4}	5.40×10^{-4}
P5	(a)	10^{-6}	100	22	130	24	140	3.28×10^{-4}	2.90×10^{-4}
		10^{-3}	40	11	46	11	40	1.06×10^{-2}	1.06×10^{-2}
	(b)	10^{-6}	107	33	122	35	40	5.60×10^{-6}	4.85×10^{-6}
		10^{-3}	34	11	40	13	15	1.37×10^{-1}	1.20×10^{-1}

Table 4.2.2: Results of LIM (4,4,2) on problems 6–10

Problem	Code	Tol	Nstep	Nsc	Nfe	Nje	Nlu	Eabs	Erel
P6	(a)	10^{-6}	212	29	225	31	222	1.73×10^{-2}	1.42×10^{-5}
		10^{-3}	64	16	73	16	70	2.34×10^{-2}	1.93×10^{-5}
	(b)	10^{-6}	578	167	647	169	192	1.91×10^{-2}	1.58×10^{-5}
		10^{-3}	61	19	64	21	22	1.24×10^{-1}	1.03×10^{-4}
P7	(a)	10^{-6}	88	22	94	22	88	4.22×10^{-6}	1.08×10^{-5}
		10^{-3}	44	12	50	12	44	2.36×10^{-4}	6.03×10^{-4}
	(b)	10^{-6}	50	15	53	17	18	1.36×10^{-5}	3.78×10^{-5}
		10^{-3}	34	11	37	13	14	6.12×10^{-4}	1.63×10^{-3}
P8	(a)	10^{-6}	372	18	394	26	398	2.34×10^{-5}	1.21×10^{-14}
		10^{-3}	64	10	84	16	86	2.79×10^{-1}	2.13×10^{-15}
	(b)	10^{-6}	880	82	883	84	85	6.03×10^{-7}	1.18×10^{-14}
		10^{-3}	121	38	124	40	41	1.57×10^{-3}	1.14×10^{-15}
P9	(a)	10^{-6}	1044	95	1229	152	1340	8.10×10^{-5}	1.83×10^{-5}
		10^{-3}	212	33	295	49	334	2.58×10^{0}	1.41×10^{0}
	(b)	10^{-6}	1763	400	1772	402	405	1.51×10^{-5}	3.42×10^{-6}
		10^{-3}	331	103	379	105	121	7.22×10^{-3}	1.57×10^{-3}
P10	(a)	10^{-6}	3772	180	4033	264	0	4.09×10^{-4}	2.86×10^{-2}
	(b)	10^{-6}	573	164	633	166	186	9.84×10^{-6}	1.59×10^{-3}
		10^{-3}	149	18	152	20	21	3.56×10^{-4}	2.06×10^{-2}

Table 4.2.3: Results of LIM (4,4,2) on problems 11-15

Problem	Code	Tol	Nstep	Nsc	Nfe	Nje	Nlu	Eabs	Erel
P11*	(a)	10^{-6}	540	109	554	112	552	3.11×10^{-3}	2.30×10^{-3}
	(b)	10^{-6}	1039	107	1045	109	111	1.87×10^{-6}	1.87×10^{-6}
P12	(a)	10^{-6}	2080	181	2456	300	2688	8.77×10^{-5}	2.64×10^{-5}
		10^{-3}	448	63	667	102	796	1.80×10^{0}	1.18×10^{0}
	(b)	10^{-6}	3546	807	3561	809	814	1.19×10^{-5}	3.58×10^{-6}
		10^{-3}	651	201	744	203	234	7.12×10^{-3}	2.13×10^{-3}
P13	(a)	10^{-6}	22240	823	23735	1060	24860	4.57×10^{11}	2.59×10^{-1}
		10^{-3}	3788	520	5476	607	6976	4.57×10^{11}	2.59×10^{-1}
	(b)	10^{-6}	77779	3115	78190	3135	3290	4.57×10^{11}	2.59×10^{-1}
		10^{-3}	7549	1337	7873	1350	1469	4.57×10^{11}	2.59×10^{-1}
P14	(a)	10^{-6}	152	28	170	37	174	3.96×10^{-1}	$6.79 \times 10^{-\text{[illegible]}}$
		10^{-3}	144	29	213	35	242	7.76×10^{3}	1.53×10^{-1}
	(b)	10^{-6}	229	58	235	60	124	7.93×10^{0}	1.04×10^{-1}
		10^{-3}	147	46	177	48	116	2.95×10^{1}	3.20×10^{-1}
P15	(a)	10^{-6}	6964	346	7492	522	7752	2.05×10^{10}	4.14×10^{-3}
		10^{-3}	1844	157	2162	270	2336	9.37×10^{12}	6.53×10^{-1}
	(b)	10^{-6}	8610	1329	8730	1331	1371	1.37×10^{11}	2.74×10^{-2}
		10^{-3}	2249	593	2465	595	667	1.18×10^{13}	[illegible].73 $\times 10^{1}$

* For Robertson problem, we set error tolerance: $rtol = 10^{-6}$, and $atol = (10^{-6}, 10^{-10}, 10^{-6})$ (see Byrne & Hindmarch (1987)).

Table 4.2.4: Results of LIM (4,4,2) on problems 16-17

Problem	Code	Tol	Nstep	Nsc	Nfe	Nje	Nlu	Eabs	Erel
P16	(a)	10^{-6}	232	53	289	55	316	1.94×10^{-7}	1.20×10^{-12}
		10^{-3}	48	12	58	12	54	4.27×10^{-5}	3.03×10^{-9}
	(b)	10^{-6}	410	54	413	56	57	2.96×10^{-8}	1.87×10^{-13}
		10^{-3}	97	23	103	25	27	3.72×10^{-5}	2.37×10^{-10}
P17	(a)	10^{-6}	1988	95	2349	215	2558	3.18×10^{-4}	2.18×10^{-5}
		10^{-3}	420	47	669	95	826	1.54×10^{-2}	5.87×10^{-3}
	(b)	10^{-6}	4048	807	4063	809	814	3.84×10^{-6}	1.91×10^{-6}
		10^{-3}	603	190	669	192	214	1.01×10^{-2}	4.68×10^{-3}

Table 4.2.5: Results of LIM (5,5,3) on problems 1–5

Problem	Code	Tol	Nstep	Nsc	Nfe	Nje	Nlu	Eabs	Erel
P1	(a)	10^{-6}	93	17	125	18	212	3.35×10^{-5}	2.11×10^{-2}
		10^{-3}	58	13	83	13	128	2.11×10^{-5}	1.29×10^{-2}
	(b)	10^{-6}	69	17	77	19	44	1.83×10^{-5}	1.15×10^{-2}
		10^{-3}	57	14	61	16	36	8.42×10^{-4}	3.51×10^{-1}
P2	(a)	10^{-6}	198	40	251	40	464	1.27×10^{-5}	4.46×10^{-7}
		10^{-3}	73	16	104	16	170	6.37×10^{-4}	2.24×10^{-5}
	(b)	10^{-6}	117	29	121	31	66	1.47×10^{-4}	4.95×10^{-6}
		10^{-3}	57	14	61	16	36	1.10×10^{-2}	3.78×10^{-4}
P3	(a)	10^{-6}	188	39	244	39	450	2.05×10^{-9}	2.05×10^{-9}
		10^{-3}	98	21	140	21	242	3.59×10^{-9}	5.62×10^{-9}
	(b)	10^{-6}	154	38	158	40	84	2.36×10^{-11}	3.68×10^{-11}
		10^{-3}	74	18	78	20	44	2.95×10^{-10}	4.61×10^{-10}
P4	(a)	10^{-6}	73	16	101	16	164	3.36×10^{-6}	2.38×10^{-6}
		10^{-3}	48	11	71	11	104	8.98×10^{-6}	6.40×10^{-6}
	(b)	10^{-6}	41	10	45	12	28	1.06×10^{-5}	6.66×10^{-7}
		10^{-3}	41	10	45	12	28	4.88×10^{-4}	2.67×10^{-4}
P5	(a)	10^{-6}	98	20	157	20	288	2.08×10^{-4}	2.09×10^{-4}
		10^{-3}	53	12	77	12	116	8.97×10^{-3}	8.96×10^{-3}
	(b)	10^{-6}	49	12	53	14	32	1.11×10^{-1}	9.97×10^{-2}
		10^{-3}	41	10	45	12	28	5.86×10^{-2}	5.54×10^{-2}

Table 4.2.6: Results of LIM (5,5,3) on problems 7–9

Problem	Code	Tol	Nstep	Nsc	Nfe	Nje	Nlu	Eabs	Erel
P7	(a)	10^{-6}	93	20	125	20	212	2.54×10^{-6}	6.48×10^{-6}
		10^{-3}	53	12	77	12	116	1.18×10^{-5}	3.01×10^{-5}
	(b)	10^{-6}	60	14	4	16	36	3.55×10^{-6}	1.06×10^{-5}
		10^{-3}	45	11	49	13	30	3.12×10^{-4}	9.46×10^{-4}
P8	(a)	10^{-6}	323	18	446	27	872	2.47×10^{-5}	1.93×10^{-13}
		10^{-3}	73	13	127	15	228	1.98×10^{-1}	2.37×10^{-14}
	(b)	10^{-6}	237	59	241	61	126	1.24×10^{-6}	1.28×10^{-14}
		10^{-3}	87	14	99	16	40	1.48×10^{-2}	3.12×10^{-15}
P9	(a)	10^{-6}	818	68	1261	122	2604	1.90×10^{-6}	4.30×10^{-7}
		10^{-3}	218	29	415	43	876	2.78×10^{0}	1.69×10^{0}
	(b)	10^{-6}	891	188	931	190	402	2.87×10^{-6}	6.34×10^{-7}
		10^{-3}	480	106	564	108	260	4.76×10^{-2}	9.85×10^{-3}

Table 4.2.7: Results of LIM (5,5,3) on problems 12-15

Problem	Code	Tol	Nstep	Nsc	Nfe	Nje	Nlu	Eabs	Erel
P12	(a)	10^{-6}	1593	117	2441	230	5088	1.02×10^{-5}	3.06×10^{-6}
		10^{-3}	438	55	878	84	1936	3.46×10^{-1}	1.16×10^{-1}
	(b)	10^{-6}	1771	375	1859	377	800	2.59×10^{-6}	7.71×10^{-7}
		10^{-3}	974	215	1158	217	528	2.44×10^{-1}	7.12×10^{-2}
P13	(a)	10^{-6}	20098	589	25635	787	52896	4.57×10^{11}	2.59×10^{-1}
		10^{-3}	2498	288	4192	386	9760	4.57×10^{11}	2.59×10^{-1}
	(b)	10^{-6}	10954	1390	11374	1395	3014	4.57×10^{11}	2.59×10^{-1}
		10^{-3}	769	164	961	170	454	3.38×10^{11}	1.98×10^{-1}
P14	(a)	10^{-6}	163	29	240	33	460	8.64×10^{-1}	1.48×10^{-6}
		10^{-3}	148	20	280	27	588	9.06×10^{2}	1.58×10^{-2}
	(b)	10^{-6}	689	158	857	160	406	6.31×10^{0}	1.06×10^{-1}
		10^{-3}	297	68	345	70	166	1.39×10^{5}	9.02×10^{-1}
P15	(a)	10^{-6}	4878	216	6523	359	13332	1.66×10^{11}	3.44×10^{-2}
		10^{-3}	1643	129	2510	229	5260	4.98×10^{12}	2.59×10^{1}
	(b)	10^{-6}	6058	929	6294	931	1982	9.16×10^{9}	1.84×10^{-3}
		10^{-3}	97	22	101	24	52	4.31×10^{13}	2.48×10^{7}

Table 4.2.8: Results of LIM (5,5,3) on problems 16-17

Problem	Code	Tol	Nstep	Nsc	Nfe	Nje	Nlu	Eabs	Erel
P16	(a)	10^{-6}	183	34	276	35	540	3.73×10^{-8}	2.05×10^{-13}
		10^{-3}	48	9	77	11	120	1.79×10^{-5}	5.81×10^{-11}
	(b)	10^{-6}	252	27	264	29	66	3.09×10^{-8}	2.03×10^{-13}
		10^{-3}	84	14	88	16	36	3.44×10^{-5}	2.29×10^{-10}
P17	(a)	10^{-6}	1518	67	2340	181	4872	5.91×10^{-6}	3.46×10^{-6}
		10^{-3}	[illegible]	40	899	76	2000	2.59×10^{-3}	9.20×10^{-4}
	(b)	10^{-6}	1250	279	1342	281	610	4.90×10^{-4}	5.11×10^{-5}
		10^{-3}	324	79	488	82	252	5.59×10^{0}	1.32×10^{0}

Table 4.2.9: Results of FIM codes on problems 1-5

Problem	Code	Order	Tol	Nstep	Nsc	Nfe	Nje	Nlu	Eabs	Erel
P1	(c)	(5,5,3)	10^{-6}	71	17	154	19	44	3.95×10^{-9}	2.45×10^{-6}
			10^{-3}	61	15	134	17	40	6.85×10^{-5}	4.67×10^{-2}
		(4,4,2)	10^{-6}	46	15	94	17	18	2.73×10^{-8}	1.84×10^{-5}
			10^{-3}	43	14	88	16	17	3.70×10^{-7}	2.44×10^{-4}
P2	(c)	(5,5,3)	10^{-6}	144	29	292	31	66	2.41×10^{-7}	8.28×10^{-9}
			10^{-3}	66	16	136	18	40	1.71×10^{-4}	5.21×10^{-6}
		(4,4,2)	10^{-6}	179	33	366	35	37	9.07×10^{-7}	3.15×10^{-8}
			10^{-3}	56	18	114	20	21	5.05×10^{-4}	1.70×10^{-5}
P3	(c)	(5,5,3)	10^{-6}	172	40	176	42	88	1.78×10^{-12}	2.68×10^{-12}
			10^{-3}	81	20	85	22	48	3.09×10^{-12}	4.81×10^{-12}
		(4,4,2)	10^{-3}	70	23	142	25	26	2.95×10^{-8}	3.89×10^{-9}
			10^{-6}	184	46	376	48	50	4.23×10^{-10}	6.62×10^{-10}
P4	(c)	(5,5,3)	10^{-6}	41	10	86	12	28	4.24×10^{-8}	2.77×10^{-8}
			10^{-3}	41	10	86	12	28	9.24×10^{-5}	5.07×10^{-5}
		(4,4,2)	10^{-6}	34	11	70	13	14	1.37×10^{-7}	9.22×10^{-8}
			10^{-3}	31	10	64	12	13	8.24×10^{-5}	4.93×10^{-5}
P5	(c)	(5,5,3)	10^{-6}	86	21	216	23	60	3.21×10^{-6}	2.65×10^{-6}
			10^{-3}	85	21	222	25	74	2.44×10^{-1}	3.15×10^{-1}
		(4,4,2)	10^{-6}	86	25	216	27	35	5.93×10^{-6}	5.20×10^{-6}
			10^{-3}	34	11	76	13	15	1.95×10^{-2}	1.92×10^{-2}

Table 4.2.10: Results of FIM codes on problems 6–10

Problem	Code	Order	Tol	Nstep	Nsc	Nfe	Nje	Nlu	Eabs	Erel
P6	(c)	(4,4,2)	10^{-6}	291	83	614	85	91	2.90×10^{-5}	2.39×10^{-8}
			10^{-3}	56	18	114	20	21	1.34×10^{-3}	1.08×10^{-6}
P7	(c)	(5,5,3)	10^{-6}	61	14	126	16	36	1.15×10^{-8}	2.93×10^{-8}
			10^{-3}	45	11	94	13	30	7.87×10^{-6}	2.16×10^{-5}
		(4,4,2)	10^{-6}	56	15	114	17	18	3.31×10^{-8}	8.48×10^{-8}
			10^{-3}	37	12	76	14	15	2.56×10^{-5}	7.99×10^{-5}
P8	(c)	(5,5,3)	10^{-6}	178	43	360	45	94	4.98×10^{-7}	5.67×10^{-16}
			10^{-3}	88	14	196	16	40	2.40×10^{-3}	0.0
		(4,4,2)	10^{-6}	266	88	534	90	91	2.67×10^{-7}	8.51×10^{-16}
			10^{-3}	82	18	178	20	23	6.32×10^{-3}	4.19×10^{-16}
P9	(c)	(5,5,3)	10^{-6}	776	174	1636	176	376	1.71×10^{-6}	3.79×10^{-7}
			10^{-3}	337	82	902	84	228	9.94×10^{-4}	2.17×10^{-4}
		(4,4,2)	10^{-6}	1041	276	2108	278	283	1.01×10^{-5}	2.23×10^{-6}
			10^{-3}	309	92	770	94	120	1.58×10^{-2}	3.44×10^{-3}
P10	(c)	(4,4,2)	10^{-6}	1589	345	3186	347	349	1.61×10^{-7}	8.47×10^{-6}
			10^{-3}	142	18	286	20	21	2.79×10^{-3}	4.29×10^{-1}

Table 4.2.11: Results of FIM codes on problems 11–15

Problem	Code	Order	Tol	Nstep	Nsc	Nfe	Nje	Nlu	Eabs	Erel
P11*	(c)	(4,4,2)	10^{-6}	486	106	980	108	110	8.92×10^{-7}	8.74×10^{-7}
			10^{-3}	149	48	300	50	51	1.30×10^{-3}	1.13×10^{-3}
P12	(c)	(5,5,3)	10^{-6}	1537	343	3214	345	728	1.60×10^{-6}	4.61×10^{-7}
			10^{-3}	627	153	1642	155	410	2.83×10^{-3}	7.39×10^{-4}
		(4,4,2)	10^{-6}	2083	567	4204	569	576	1.31×10^{-5}	3.92×10^{-6}
			10^{-3}	608	183	1524	185	237	1.40×10^{-2}	3.98×10^{-3}
P13	(c)	(5,5,3)	10^{-6}	8126	1126	17224	1130	2514	4.57×10^{11}	2.59×10^{-1}
			10^{-3}	2441	506	5798	511	1266	4.57×10^{11}	2.59×10^{-1}
		(4,4,2)	10^{-6}	15932	2023	32580	2025	2145	4.57×10^{11}	2.59×10^{-1}
			10^{-3}	3160	719	6970	723	834	4.57×10^{11}	2.59×10^{-1}
P14	(c)	(5,5,3)	10^{-6}	221	49	470	51	112	5.12×10^{-2}	8.01×10^{-7}
			10^{-3}	150	36	352	38	92	3.12×10^{3}	5.15×10^{-2}
		(4,4,2)	10^{-6}	148	41	352	43	53	1.22×10^{0}	1.55×10^{-5}
			10^{-3}	87	28	194	30	34	4.55×10^{2}	4.11×10^{-3}
P15	(c)	(5,5,3)	10^{-6}	6739	978	13802	980	2044	1.90×10^{8}	3.37×10^{-5}
			10^{-3}	2364	531	5348	533	1224	4.48×10^{10}	7.81×10^{-3}
		(4,4,2)	10^{-6}	9085	1465	18466	1467	1517	1.90×10^{8}	3.35×10^{-5}
			10^{-3}	2523	648	5564	650	737	8.74×10^{11}	1.29×10^{-1}

* For Robertson problem, we set error tolerance: $rtol = 10^{-6}$, $atol = (10^{-6}, 10^{-10}, 10^{-6})$ and $rtol = 10^{-3}$, $atol = (10^{-3}, 10^{-7}, 10^{-3})$ respectively.

Table 4.2.12: Results of FIM codes on problems 16–17

Problem	Code	Order	Tol	Nstep	Nsc	Nfe	Nje	Nlu	Eabs	Erel
P16	(c)	(5,5,3)	10^{-6}	199	11	402	13	30	7.26×10^{-10}	5.33×10^{-15}
			10^{-3}	69	11	142	13	30	5.93×10^{-7}	4.24×10^{-12}
		(4,4,2)	10^{-6}	246	24	494	26	27	2.63×10^{-9}	1.79×10^{-14}
			10^{-3}	64	12	130	14	15	2.22×10^{-6}	2.47×10^{-11}
P17	(c)	(5,5,3)	10^{-6}	1281	294	2614	296	608	1.42×10^{-7}	4.25×10^{-9}
			10^{-3}	553	137	1478	139	374	3.23×10^{-4}	3.49×10^{-6}
		(4,4,2)	10^{-6}	1823	543	3690	545	553	9.59×10^{-7}	5.68×10^{-7}
			10^{-3}	562	167	1480	169	229	1.40×10^{-3}	1.31×10^{-4}

4.3 Discussions on numerical test

In this section, we group the results obtained in section 4.2 to bring the data into a more accessible form, such that the comparison between codes becomes easier. Since the numbers of function and Jacobian evaluation are generally considered to be most important factor for a integration method, we focus on them.

Comparison between the fourth and the fifth order codes: Table 4.3 13 shows (5,5,3) codes are more efficient than the (4,4,2) codes on most problems. However, the (4,4,2) codes are more robust, which work on all 17 test problems whereas the (5,5,3) codes failed to solve problems 6, 10, 11. We believe the reason for (5,5,3) codes' failing is due to the larger error constant in higher order methods, and the three problems are poorly scaled.

Comparison between the FIM and LIM codes: For predictor-corrector codes, we refer to Lambert (1991, p.104), where the mode $P(EC)^{\mu}E^{1-t}$ is discussed for positive integer μ and $t = 0$ or 1. But we only implement the mode PECE which has two function evaluations at each step. The FIM code (c) is always more accurate than the LIM codes (a), (b) at the expense by one more function evaluation each step. If we only evaluate the function once at each step, i.e., in PEC scheme, then the accuracy will be similar for code (a), (b), (c).

Comparison examples for VCM, BDF and Radau5 codes: We do not compare VCM methods with other common methods such as BDF and Runge-Kutta methods systematically. However, we compute two problems by VCM5, IMSL and Radau5. VCM5 is the fifth order predictor-corrector code, in which the predictor is LIM (6,5,3) and the correct is FIM (5,5,3), the coefficients of both satisfy the condition (2.4.3)-(2.4.4), the implementation is same as code (c). IMSL is the subroutine divpag in IMSL Fortran library based on BDF methods and Radau5 is the code developped by Hairer & Wanner (1991, p.547). We present in table 4.3.14–4.3.15 comparative data only for function calls, Jacobian evaluation calls and number of steps for given accuracy tolerances. Comparison of CPU times with well established codes would be inappropriate at this time since these codes are finely tuned and our codes are still in early stages of development.

Possible improvements: In order to develop VCM methods as a practical package, it is important to vary their order as well as stepsize so that we can take advantages of both the high order methods' efficiency and the lower order methods' robustness. This is the strategy used in variable-order variable-step BDF methods. Another possible improvement is adopting the idea in Enright (1978), factoring $(Q + w_1 I) = LHL^{-1}$, where H is upper Hessenberg and L is unit low triangular. The decomposition of each remaining matrices only needs n additions noting the relation

$$(Q + w_i I) = L(H + (w_i - w_1)I)L^{-1}.$$

One half of LU-decompositions for (5,5,3) codes in section 4.2 can be eliminated if we adopt this strategy.

The test results indicate that the VCM codes work well on stiff problems. This confirms our stability and convergence analysis in the preceding two chapters.

Table 4.3.13: Comparison for problems 1–17

Problem	LIM (4,4,2)		LIM (5,5,3)		FIM	
	(a)	(b)	(a)	(b)	(4,4,2)	(5,5,3)
P1	(74, 18)	(49,17)	(125, 18)	(77,19)	(94, 17)	(154,19)
P2	(198, 43)	(460, 33)	(251, 40)	(121, 31)	(366, 35)	(292, 31)
P3	(203,48)	(168, 49)	(244, 39)	(158, 40)	(142, 25)	(176, 42)
P4	(62, 15)	(35, 12)	(101, 16)	(45, 12)	(70, 13)	(86, 12)
P5	(130, 24)	(122, 35)	(157, 20)	(53, 14)	(216, 27)	(216,23)
P6	(225, 31)	(647, 169)			(614, 85)	
P7	(94, 22)	(53, 17)	(125, 20)	(64, 16)	(114, 17)	(126, 16)
P8	(394, 26)	(883, 84)	(446, 27)	(241, 61)	(534, 90)	(360, 45)
P9	(1229, 152)	(1772, 402)	(1261, 122)	(931, 190)	(2108, 278)	(1636, 176)
P10	(4033, 264)	(633, 166)			(3186, 347)	
P11	(554, 112)	(1045, 109)			(980, 108)	
P12	(2456, 300)	(3561, 809)	(2441, 230)	(1859, 377)	(4204, 569)	(3214, 345)
P13	(23735, 1060)	(78190, 3135)	(25635, 787)	(11374, 1395)	(32580, 2025)	(17224, 1130)
P14	(170, 37)	(235, 60)	(240, 33)	(857, 160)	(352, 43)	(470, 51)
P15	(7492, 522)	(8730, 1331)	(6523, 359)	(6294, 931)	(18466, 1467)	(13802, 980)
P16	(289, 55)	(413, 56)	(276, 35)	(264, 29)	(494, 26)	(402, 13)
P17	(2349, 215)	(4063, 809)	(2340, 181)	(1342, 281)	(3690, 545)	(2614, 296)

The entries in the table are numbers of function and Jacobian evaulation (Nfe, Nje) under the $Tol = 10^{-}$[illegible].

Table 4.3.14: Comparison results for problem 12

Method	Tol	Nstep	Nfe	NJe	Eabs
VCM5	10^{-5}	1168	2922	222	6.76×10^{-6}
	10^{-6}	1450	3356	274	2.40×10^{-6}
	10^{-7}	1837	3890	339	2.71×10^{-7}
IMSL	10^{-6}	1648	2768	191	1.23×10^{-4}
	10^{-7}	2236	3563	207	5.54×10^{-6}
	10^{-8}	3184	4725	264	1.44×10^{-7}
RADAU5	10^{-5}	581	4433	434	7.54×10^{-7}
	10^{-6}	968	6691	587	1.22×10^{-7}
	10^{-7}	1694	10809	760	4.80×10^{-8}

Table 4.3.15: Comparison results for problem 17

Method	Tol	Nstep	Nfe	NJe	Eabs
VCM5	10^{-5}	1042	2530	200	2.29×10^{-6}
	10^{-6}	1175	2636	220	6.34×10^{-7}
	10^{-7}	1501	3088	275	8.30×10^{-8}
IMSL	10^{-6}	1299	1923	160	1.00×10^{-5}
	10^{-7}	1811	2615	181	9.44×10^{-7}
	10^{-8}	2576	3511	207	1.68×10^{-7}
RADAU5	10^{-5}	476	3473	294	1.20×10^{-6}
	10^{-6}	834	5451	377	1.02×10^{-8}
	10^{-7}	1475	9168	475	6.23×10^{-9}

Summary

By introducing a set of simplifying conditions, we constructed the contractive function of VCM methods as the Padé approximants of the exponential function $\exp(z)$. To determine the coefficients, we first separate the order condition to three parts, one of the three parts can be satisfied because the relations between the coefficients. For the remain two parts, transfer them to a linear system by index change. Through elemantary matrix operation, we expressed the coefficient matrix of the linear system in the powers of a Vandomone matrix. So we can determine all the coefficients of a specific method by solving a linear system.

The convergence properties of VCM methods are important since the stability properties based on the linear test equation can not guantee convergence for VCM methods on nonlinear problems. By using the contractive function and the recursion relations of the solution, we showed the stiff-independent convergence for VCM methods on general nonlinear dissipative problems. By selecting the main part of the Jacobian, we proved the convergence being independent of the perturbation parameter for singular perturbation problems.

The numerical test results on a set of test problems indicate that the VCM codes work well on stiff problems. This confirms our stability and convergence analysis in this monograph.

Appendix: coefficients of several VCM methods

We list here the coefficients of the linearly implicit VCM methods. For fixed stepsize, the list is from order 3 to 6, for variable stepsize, we only list the fourth order VCM method. The free parameters α, β, γ can, in specific instances, be chosen so as to minimize the truncation errors according to theorem 2.4.1 or 2.4.3.

Table A.1: Coefficients of (3,3,2) algorithm

$a_0^{(0)} = 0$	$a_1^{(0)} = 0$	$a_2^{(0)} = -1$	$a_3^{(0)} = 1$
$a_0^{(1)} = \frac{5}{12}$	$a_1^{(1)} = -\frac{4}{3}$	$a_2^{(1)} = \frac{19}{12}$	$a_3^{(1)} = -\frac{2}{3}$
$a_0^{(2)} = \frac{1}{3} + \alpha$	$a_1^{(2)} = -\frac{1}{2} - 2\alpha$	$a_2^{(2)} = \alpha$	$a_3^{(2)} = \frac{1}{6}$
$b_0^{(0)} = \frac{5}{12}$	$b_1^{(0)} = -\frac{4}{3}$	$b_2^{(0)} = \frac{23}{12}$	
$b_0^{(1)} = \frac{1}{3} + \alpha$	$b_1^{(1)} = -\frac{1}{2} - 2\alpha$	$b_2^{(1)} = \alpha$	

Table A.2: Coefficients of (4,4,2) algorithm

$a_0^{(0)} = 0$	$a_1^{(0)} = 0$	$a_2^{(0)} = 0$	$a_3^{(0)} = -1$	$a_4^{(0)} = 1$
$a_0^{(1)} = -\frac{3}{8}$	$a_1^{(1)} = \frac{37}{24}$	$a_2^{(1)} = -\frac{59}{24}$	$a_3^{(1)} = \frac{43}{24}$	$a_4^{(1)} = -\frac{1}{2}$
$a_0^{(2)} = -\frac{1}{6} - \alpha$	$a_1^{(2)} = \frac{5}{12} + 3\alpha$	$a_2^{(2)} = -\frac{1}{4} - 3\alpha$	$a_3^{(2)} = -\frac{1}{12} + \alpha$	$a_4^{(2)} = \frac{1}{12}$
$b_0^{(0)} = -\frac{3}{8}$	$b_1^{(0)} = \frac{37}{24}$	$b_2^{(0)} = -\frac{59}{24}$	$b_3^{(0)} = \frac{55}{24}$	
$b_0^{(1)} = -\frac{1}{6} - \alpha$	$b_1^{(1)} = \frac{5}{12} + 3\alpha$	$b_2^{(1)} = -\frac{1}{4} - 3\alpha$	$b_3^{(1)} = \alpha$	

Table A.3: Coefficients of (5,5,3) algorithm

$a_0^{(0)} = 0$	$a_1^{(0)} = 0$	$a_2^{(0)} = 0$ $a_4^{(0)} = -1$	$a_3^{(0)} = 0$ $a_5^{(0)} = 1$
$a_0^{(1)} = \frac{251}{720}$	$a_1^{(1)} = -\frac{637}{360}$	$a_2^{(1)} = \frac{109}{30}$ $a_4^{(1)} = \frac{1613}{720}$	$a_3^{(1)} = -\frac{1387}{360}$ $a_5^{(1)} = -\frac{3}{5}$
$a_0^{(2)} = \frac{43}{90} + \alpha$	$a_1^{(2)} = -\frac{43}{24} - 4\alpha$	$a_2^{(2)} = \frac{143}{60} + 6\alpha$ $a_4^{(2)} = -\frac{1}{20} + \alpha$	$a_3^{(2)} = -\frac{121}{360} - 4\alpha$ $a_5^{(2)} = \frac{3}{20}$
$a_0^{(3)} = \frac{1}{10} - \beta - 3\gamma$	$a_1^{(3)} = -\frac{1}{4} + 3\beta + 8\gamma$	$a_2^{(3)} = \frac{1}{6} - 3\beta - 6\gamma$ $a_4^{(3)} = \gamma$	$a_3^{(3)} = \beta$ $a_5^{(3)} = -\frac{1}{60}$
$b_0^{(0)} = \frac{251}{720}$	$b_1^{(0)} = -\frac{637}{360}$	$b_2^{(0)} = \frac{109}{30}$ $b_4^{(0)} = \frac{1901}{720}$	$b_3^{(0)} = -\frac{1387}{360}$
$b_0^{(1)} = \frac{43}{90} + \alpha$	$b_1^{(1)} = -\frac{43}{24} - 4\alpha$	$b_2^{(1)} = \frac{143}{60} + 6\alpha$ $b_4^{(1)} = \alpha$	$b_3^{(1)} = -\frac{421}{360} - 4\alpha$
$b_0^{(2)} = \frac{1}{10} - \beta - 3\gamma$	$b_1^{(2)} = -\frac{1}{4} + 3\beta + 8\gamma$	$b_2^{(2)} = \frac{1}{6} - 3\beta - 6\gamma$ $b_4^{(2)} = \gamma$	$b_3^{(2)} = \beta$

Table A.4: Coefficients of (6,6,3) algorithm

$a_0^{(0)} = 0$	$a_1^{(0)} = 0$ $a_4^{(0)} = 0$	$a_2^{(0)} = 0$ $a_5^{(0)} = -1$	$a_3^{0} = 0$ $a_6^{(0)} = 1$
$a_0^{(1)} = -\frac{95}{288}$	$a_1^{(1)} = \frac{959}{480}$ $a_4^{(1)} = -\frac{2641}{480}$	$a_2^{(1)} = -\frac{3649}{480}$ $a_5^{(1)} = \frac{3557}{1440}$	$a_3^{(1)} = \frac{4991}{720}$ $a_6^{(1)} = -\frac{1}{2}$
$a_0^{(2)} = -\frac{79}{240} - \alpha$	$a_1^{(2)} = \frac{1127}{720} + 5\alpha$ $a_4^{(2)} = -\frac{149}{180} - 5\alpha$	$a_2^{(2)} = -\frac{231}{80} - 10\alpha$ $a_5^{(2)} = -\frac{1}{10} + \alpha$	$a_3^{(2)} = \frac{119}{48} + 10\alpha$ $a_6^{(2)} = \frac{1}{10}$
$a_0^{(3)} = -\frac{7}{60} + \beta + 4\gamma$	$a_1^{(3)} = \frac{17}{40} - 4\beta - 15\gamma$ $a_4^{(3)} = \beta$	$a_2^{(3)} = -\frac{13}{24} + 6\beta + 20\gamma$ $a_5^{(3)} = -\frac{1}{120} + \gamma$	$a_3^{(3)} = \frac{1}{4} - 4\beta - 10\gamma$ $a_6^{(3)} = -\frac{1}{120}$
$b_0^{(0)} = -\frac{95}{288}$	$b_1^{(0)} = \frac{959}{480}$ $b_4^{(0)} = -\frac{2641}{480}$	$b_2^{(0)} = -\frac{3649}{480}$ $b_5^{(0)} = \frac{4277}{1440}$	$b_3^{(0)} = \frac{4991}{720}$
$b_0^{(1)} = -\frac{79}{240} - \alpha$	$b_1^{(1)} = \frac{1127}{720} + 5\alpha$ $b_4^{(1)} = -\frac{149}{180} - 5\alpha$	$b_2^{(1)} = -\frac{231}{80} - 10\alpha$ $b_5^{(1)} = \alpha$	$b_3^{(1)} = \frac{119}{48} + 10\alpha$
$b_0^{(2)} = -\frac{7}{60} + \beta + 4\gamma$	$b_1^{(2)} = \frac{17}{40} - 4\beta - 15\gamma$ $b_4^{(2)} = \beta$	$b_2^{(2)} = -\frac{13}{24} + 6\beta + 20\gamma$ $b_5^{(2)} = \gamma$	$b_3^{(2)} = \frac{1}{4} - 4\beta - 10\gamma$

Table A.5: Coefficients of variable stepsize (4,4,2) algorithm, $u = 1$

$a_0^{(0)} = 0$	$a_1^{(0)} = 0$	$a_2^{(0)} = 0$	$a_3^{(0)} = -1$ $a_4^{(0)} =$ [illegible]
$a_0^{(1)} = -\frac{r(2+r)^2}{24}$	$a_1^{(1)} = \frac{r(18+16r+3r^2)}{24}$	$a_2^{(1)} = -\frac{r(36+20r+3r^2)}{24}$	$a_3^{(1)} = -\frac{1}{2} + \frac{(4+r)(6+4r+r^2)}{24}$ $a_4^{(1)} = -\frac{1}{[illegible]}$
$a_0^{(2)} = -\frac{3r+r^2+24\alpha}{24}$	$a_1^{(2)} = \frac{4r+r^2+36\alpha}{12}$	$a_2^{(2)} = -\frac{5r+r^2+72\alpha}{24}$	$a_3^{(2)} = -\frac{1}{12} + \alpha$ $a_4^{(2)} = \frac{1}{12}$
$b_0^{(0)} = -\frac{r(2+r)^2}{24}$	$b_1^{(0)} = \frac{r(18+16r+3r^2)}{24}$	$b_2^{(0)} = -\frac{r(36+20r+3r^2)}{24}$	$b_3^{(0)} = \frac{(4+r)(6+4r+r^2)}{24}$ $b_4^{(0)} = 0$
$b_0^{(1)} = -\frac{3r+r^2+24\alpha}{24}$	$b_1^{(1)} = \frac{4r+r^2+36\alpha}{12}$	$b_2^{(1)} = -\frac{5r+r^2+72\alpha}{24}$	$b_3^{(1)} = \alpha$ $b_4^{(1)} = 0$

Table A.6: Coefficients of variable stepsize (4,4,2) algorithm, $a = 2$

$a_0^{(0)} = 0$	$a_1^{(0)} = 0$	$a_2^{(0)} = 0$	$a_3^{(0)} = -1$ $a_4^{(0)} = 1$
$a_0^{(1)} = -\frac{r^2(10+17r)}{24(2+r)}$	$a_1^{(1)} = \frac{r^2(20+17r)}{12(1+r)}$	$a_2^{(1)} = -\frac{12+30r+17r^2}{24}$	$a_3^{(1)} = -\frac{1}{2} + \frac{12+28r+15r^2}{8+12r+4r^2}$ $a_4^{(1)} = -\frac{1}{2}$
$a_0^{(2)} = -\frac{r(1+3r+12\alpha+12r\alpha)}{24}$	$a_1^{(2)} = \frac{r(2+3r+24\alpha+12r\alpha)}{12}$	$a_2^{(2)} = -\frac{(1+r)(r+8\alpha+4r\alpha)}{8}$	$a_3^{(2)} = -\frac{1}{12} + \alpha$ $a_4^{(2)} = \frac{1}{12}$
$b_0^{(0)} = -\frac{r^2(10+17r)}{24(2+r)}$	$b_1^{(0)} = \frac{r^2(20+17r)}{12(1+r)}$	$b_2^{(0)} = -\frac{12+30r+17r^2}{24}$	$b_3^{(0)} = \frac{12+28r+15r^2}{8+12r+4r^2}$ $b_4^{(0)} = 0$
$b_0^{(1)} = -\frac{r(1+3r+12\alpha+12r\alpha)}{24}$	$b_1^{(1)} = \frac{r(2+3r+24\alpha+12r\alpha)}{12}$	$b_2^{(1)} = -\frac{(1+r)(r+8\alpha+4r\alpha)}{8}$	$b_3^{(1)} = \alpha$ $b_4^{(1)} = 0$

Table A.7: Coefficients of variable stepsize (4,4,2) algorithm, $u = 3$

$a_0^{(0)} = 0$	$a_1^{(0)} = 0$	$a_2^{(0)} = 0$	$a_3^{(0)} = -1$ $a_4^{(0)} = 1$
$a_0^{(1)} = -\frac{9r^3}{4+12r+8r^2}$	$a_1^{(1)} = \frac{10+27r}{24}$	$a_2^{(1)} = -\frac{16+43r}{12+12r}$	$a_3^{(1)} = -\frac{1}{2} + \frac{46+119r}{24+48r}$ $a_4^{(1)} = -\frac{1}{2}$
$a_0^{(2)} = -\frac{r^2(1+6\alpha)}{3(1+r)}$	$a_1^{(2)} = \frac{1+4r+12\alpha+24r\alpha}{12}$	$a_2^{(2)} = -\frac{1+5r+24\alpha+48r\alpha}{12+12r}$	$a_3^{(2)} = -\frac{1}{12} + \alpha$ $a_4^{(2)} = \frac{1}{12}$
$b_0^{(0)} = -\frac{9r^3}{4+12r+8r^2}$	$b_1^{(0)} = \frac{10+27r}{24}$	$b_2^{(0)} = -\frac{16+43r}{12+12r}$	$b_3^{(0)} = \frac{46+119r}{24+48r}$ $b_4^{(0)} = 0$
$b_0^{(1)} = -\frac{r^2(1+6\alpha)}{3(1+r)}$	$b_1^{(1)} = \frac{1+4r+12\alpha+24r\alpha}{12}$	$b_2^{(1)} = -\frac{1+5r+24\alpha+48r\alpha}{12+12r}$	$b_3^{(1)} = \alpha$ $b_4^{(1)} = 0$

References

Atkinson, K.E. (1989): *An introduction to numerical analysis*, Wiley.

Baker, G. A., Jr. and Graves-Morris, P. R. (1981): *Padé approximants. Part I: Basic theory*, Addison-Wesley.

Bjurel, G., Dahlquist, G., Lindberg, B., Linde, S. and Odén, L. (1970): *Survey of stiff ordinary differential equations*, Report NA 70.11, Dept. of Information Processing, Royal Inst. of Tech., Stockholm.

Brunner, H. (1967): Stabilization of optimal difference operators, *Z. Angew. Math. Phys.*, 18, 438–444.

Butcher, J.C. (1987): *The numerical analysis of ordinary differential equations*, J. Wiley & Sons.

Byrne, G. D. and Hindmarsh, A. C. (1987): Stiff ODE Solvers: A review of current and coming attractions, *J. Comp. Phys.*, 70, 1–62.

Calvo, M., Lisbona, F. and Montjano, J. (1987): On the stability of variable-stepsize Nordsieck BDF methods, *SIAM J. Numer. Anal.*, 24, 844–854.

Charron, R.J. (1993): A-contractivity of variable stepsize, variable matrix coefficient multistep methods, submitted.

Curtiss, C.F. and Hirschfelder, J.O. (1952): Integration of stiff equations, *Proc. Nat. Acad. Sci.*, 38, 235–243.

Dahlquist, G. (1956): Convergence and stability in the numerical integration of ordinary differential equations, *Math. Scand.*, 4, 33-53.

Dahlquist, G. (1959): Stability and error bounds in the numerical solution of ordinary differential equations, Thesis, in *Trans. Royal Inst. of Technology, No 130*, Stockholm.

Dahlquist, G. (1963): A special stability problem for linear multistep methods, *BIT*, 3, 27–43.

Dekker, K. and Verwer, J.G. (1984): *Stability of Runge-Kutta methods for stiff nonlinear differential equations*, North-Holland, Amsterdam.

Dorondicyn, A.A. (1947): Asymptotic solution of the van der Pol equation. *Prikl. Mat. i Mch.*, 11, 313–328; Translations AMS, Ser. 1, 4, 1–23.

Enright, W. H. (1973): Optimal second derivative methods for stiff systems, In *Stiff Differential Systems*, R. A. Willoughby, Ed., Plenum Press, New York, 1973, 95–111.

Enright, W. H. (1978): Improving the efficiency of matrix operations in the numerical solution of stiff ordinary differential equations, *ACM Transactions on Mathematics Software*, 4 127–136.

Enright, W. H. and Hull, T. E. (1976): Comparing numerical methods for the solution of stiff systems of ODEs arising in chemistry, In *Numerical Methods for Differential Systems*, L. Lapidus and W. E. Schiesser, Ed., Academic Press, New York, 1976, 45–66.

Enright, W. H., Hull, T. E. and Lindberg, B. (1975): Comparing numerical methods for stiff systems of O.D.E.'s, *BIT*, **15**, 10–48.

Field, R. J. and Noyes, R. M. (1974): Oscillations in chemical systems. IV. Limit cycle behavior in a model of a real chemical reaction, *Journal of Chem. Phys.*, 60, 1877–1884.

Frank, R., Schneid, J. and Ueberhuber, C.W. (1981): The concept of B-convergence, *SIAM J. Numer. Anal.*, **18**, 753–780.

Frank, R., Schneid, J. and Ueberhuber, C.W. (1985a): Stability of properties of implicit Runge-Kutta methods, *SIAM J. Numer. Anal.*, **22**, 497–514.

Frank, R., Schneid, J. and Ueberhuber, C.W. (1985b): Order results for implicit Runge-Kutta methods applied to stiff systems, *SIAM J. Numer. Anal.*, **22**, 515–534.

Garfinkel, D., Ching, D. W., Adelman, M. and Clark, P. (1966): Techniques and problems in the construction of computer models of biochemical systems including real enzymes, *Annals New York Academy of Sciences*, 1054–1068.

Gear, C. W. (1969): The automatic integration of stiff ordinary differential equations, in *Information Processing*, 68, ed. A.J.H. Morrel. North Holland Publishing Company, Amsterdam, 187–193.

Gear, C. W. and Tu, K. W. (1974): The effect of variable mesh size on the stability of multistep methods, *SIAM J. of Numer Anal.*, **11**, 1025–1043.

Gear, C. W. and Watanabe, D. S. (1974): Stability and convergence of variable order multistep methods, *SIAM J. of Numer Anal.*, **11**, 1044–1058.

Grigorieff, R. D. (1983): Stability of multistep methods on variable grids, *Numer. Math.*, **42**, 359–377.

Hairer, E., Nørsett, S. P. and Wanner, G. (1987): *Solving ordinary differential equations I*, SCM vol. 8, Springer-Verlag, Berlin.

Hairer, E. and Wanner, G. (1991): *Solving ordinary differential equations II*, SCM vol. 14, Springer-Verlag, Berlin.

Hundsdorfer, W.H. (1981): *Nonlinear stability analysis for a simple Rosenbrock method*, Report No 81/31, Inst. of Appl. Math. and Comp. Sc., University of Leiden, Netherlands.

Hundsdorfer, W. H. (1984): *The numerical solution of nonlinear stiff initial value problems*, Ph. D. thesis, Centrum voor Wiskunde en Informatica, Amsterdam.

Hundsdorfer, W. H. and Steininger, B.I. (1991): Convergence of linear multistep and one-leg methods for stiff nonlinear initial value problems, *BIT*, **31**, 124–143.

Kaps, P. and Wanner, G. (1981): A study of Rosenbrock-type methods of high order, Numer. Math., 38, 279–298.

Lambert, J. D. (1970): Linear multistep methods with mildly varying coefficients, *Math. Comput.*, **24**, 81–94.

Lambert, J. D. (1991): *Numerical methods for ordinary differential systems*, John Wiley & Sons Ltd.

Lambert, J. D. and Sigurdsson, S. T. (1972): Multistep methods with variable matrix coefficients, *SIAM J. Numer. Anal.*, **9**, 715–733.

Lapidus, L., Aiken, R. C. and Liu, Y. A. (1974): The occurrence and numerical solution of physical and chemical systems having widely varying time constants, In *Stiff Differential Systems*, Willoughby, R. A. ed., Plenum Press, 187–200.

Liniger, W. and Willoughby, R. A. (1967): *Efficient numerical integration of stiff systems of ordinary differential equations*, Technical Report RC-1970, Thomas J. Watson Research Center, Yorktown Heights, N. Y.

Lubich, Ch. (1991): On the convergence of multistep methods for nonlinear stiff differential equations, *Numer. Math.*, **58**, 839–853.

Luss, D. and Amundson, N. R. (1968): Stability of batch catalytic fluidized beds, *AIChE Journal*, **14**, 211–221.

Neumann, J. von (1951): Eine spektraltheorie für allgemine operatoren eines unitären raumes, *Math. Nachrichten*, **4**, 258–281.

Nevanlinna, O. and Liniger, W. (1978): Contractive methods for stiff differential equations: Part I, *BIT*, **18**, 457–474.

Nevanlinna, O. and Liniger, W. (1979): Contractive methods for stiff differential equations: Part II, *BIT*, **19**, 53–72.

Perron, O. (1913): *Die Lehre von den Kettenbrüchen*, 3rd. ed., B.G. Teubner, Stuttgart, 1977.

Prothero, A. and Robibson, A. (1974): On the stability and accuracy of one-step methods for solving stiff systems of ordinary differential equations, *Math. Comput.*, **28**, 145–162.

Robertson, H. H. (1962): The solution of a set of reaction rate equations, In *Numerical Analysis, An introduction*, J. Walsh ed., Academic Press, London, 178–182.

Rockswold, G. K. (1988): Implementation of α−type multistep methods for stiff differential equations, *J. Comp. Applied Math.*, **22**, 63–69.

Rosenbrock, H.H. (1963): Some general implicit processes for the numerical solution of differential equations, *Computer J.*, **5**, 329–330.

Sanz-Serna, J. M. (1981): Linearly implicit variable coefficient methods of Lambert-Sigurdsson type, *IMA J. Numer. Anal.*, **1**, 39–45.

Shampine, L.F. and Gear, C.W. (1979): A user's view of solving stiff ordinary differential equations, *SIAM. Rev.*, **21**, 1–17.

Skeel, R. D. and Jackson, L. W. (1983): The stability of variable stepsize Nordsieck methods, *SIAM J. Numer. Anal.*, **20**, 840–853.

Skeel, R. D. and Kong, A. K. (1977): Blended linear multistep methods, *ACM Trans. Math. Software*, **3**, 326–343.

Spijker, M. N. (1983): Contractivity in the numerical solution of initial value problems, *Numer. Math.*, **42**, 271-290.

Spijker, M. N. (1985): Feasibility and contractivity in implicit Runge-Kutta methods. *J. Comp. Appl. Math.*, **12**, 563-578.

Spijker, M. N. (1987): A note on contractivity in the numerical solution of initial value problems, *BIT*, **27**, 424-437.

Van der Pol, B. (1926): On "Relaxation Oscillations", *Phil. Mag.*, **2**, 978–992; Reproduced in: B. Van der Pol, Selected Scientific Papers, 1, North-Holland Publ. Comp. Amsterdam (1960).

Wanner, G., Hairer, E. and Nørsett, S. P. (1978): Order stars and stability theorems, *BIT*, **18**, 475–489.

Widlund, O. (1967): A note on unconditionally stable linear multistep methods, *BIT*, **7**, 65–70.