Long-term behaviour of *G*-symplectic methods

Yousaf Habib

A thesis submitted in fulfilment of the requirements for the degree of
Doctor of Philosophy in Applied Mathematics,
The University of Auckland, 2010.

Abstract

> *It is not for the sun to overtake the moon, nor does the night outstrip the day. They all float, each in an orbit.* *(Quran 36:40)*

There has been a recent revival of interest in structure preserving numerical methods for ordinary differential equations having quadratic invariants. Much work has been done for Runge–Kutta and multistep methods and there exist excellent symplectic integrators among Runge–Kutta methods. General linear methods provide a unifying framework for these traditional methods but, because of their multivalue nature we cannot hope for true conservation of quadratic invariants. However, not everything is lost and we can still search for *G*-symplectic general linear methods taking account of the underlying invariants.

The multivalue nature of general linear methods exposes them to parasitic solutions. The corruption of the numerical solution is partly due to the parasitic growth parameter and partly due to the differential equation system being susceptible to parasitism. Two control strategies have been employed to contain this situation. One, where the effective parasitic growth parameter of a composition of different *G*-symplectic methods is forced to remain bounded. Several possible composition techniques can be used of which one is employed in this thesis and further reference is provided in the conclusions. The other strategy is to construct methods where parasitic growth parameter is zero by design. The construction of a method with four stages and three output values and a search for a suitable starting method with algebraic analysis using rooted trees constitute an important aspect of this thesis.

These strategies are investigated using various implementations for Hamiltonian and structure preserving systems and compared with a traditional symplectic method. This provides encouraging results for the *G*-symplectic general linear methods. The new methods provide an alternative to the well established symplectic one step methods. The foundation for the search of such methods is laid out in this thesis and it is anticipated that these methods can be implemented for serious real world problems with confidence.

Acknowledgments

First of all, I would like to thank almighty Allah, the creator, for all blessings in my life.

During the course of my PhD, I was fortunate to have the guidance and support of my supervisor Prof John Butcher for which I am deeply grateful. His knowledge and logical way of thinking have been of great value for me. On a personal level, I would especially like to mention his unusually kind and caring nature.

I would like to thank my co-supervisor Dr Robert Chan for the many discussions we have, be it mathematics, cricket or life in general.

The weekly numerical analysis workshops provided a platform to practice my conference talks and listen to the work of fellow researchers. I would like to thank Dr Allison Heard, who was a vital member of these workshops. She always helped me correcting several mistakes in my talks.

I enjoyed the company of several colleagues in the Mathematics department during my doctoral research. I would like to acknowledge people from our research group Dr Shixiao Wang, Angela Tsai, Gulshad, Annie Gorgey and Saghir Ahmad for their support and stimulating discussions. In addition to these people I would also like to mention my fellow research students Muhammad Amer Qureshi, Shafiq ur Rehman, Attique ur Rehman, Wen Duan and Wenjun Zhang with whom I have enjoyed lengthy discussions about my research either in offices or during lunch time.

I am also thankful to my fellow HEC scholars in New Zealand and all my friends back in Pakistan for their support and encouragement throughout my research. In particular, I would like to mention Muhammad Aman Ullah, Faheem Butt, Bashir Hussain and Jibran Walli for making my stay in New Zealand, a memorable one. Here I would also like to thank my long lasting friends Rana Umer Shahzad, Muhammad Ali, Qasim Razi and Zulfiqar Ahmad Noor for always being there whenever I needed them.

Special thanks are due to Higher Education Commission of Pakistan (HEC) and National

University of Science and Technology (NUST), Pakistan for their financial assistance. I also wish to thank University of Auckland for funding me to attend conferences locally and internationally.

I owe my loving thanks to my parents and my aunts. Without their encouragement, understanding and prayers, it would have been impossible for me to finish this work. My special gratitude is due to my brothers Shahzad and Kashif, my sisters and their families for their loving support. I wish to dedicate this thesis to my late uncle Muhammad Saeed Chaudhry, who was like a father to me and whose loving memories are always with me.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Most physical phenomena, like the motion of blood in veins, the behaviour of electric circuits in machines, the movement of stars in galaxies or the dynamics of shares in stock markets, can be understood through their mathematical models. These models often consist of system of ordinary differential equations (ODEs), that have time as the independent variable and the variables of the physical systems as dependent variables. Generally these ODEs are accompanied by an initial condition and thus constitute initial value problems. They take the form

$$y'(x) = f(x, y(x)), \qquad y(x_0) = y_0.$$

In this initial value problem (IVP), $y(x)$ is a vector valued function, $y : \mathbb{R} \to \mathbb{R}^m$ and represents the solution, x denotes time, $f : \mathbb{R} \times \mathbb{R}^m \to \mathbb{R}^m$ and m is the dimension of the problem. Normally we consider autonomous IVP's, in which x is taken as one of the components of the vector $y(x)$ if necessary, and is given by

$$y'(x) = f(y(x)), \qquad y(x_0) = y_0. \tag{1.1}$$

It is often the case that physical systems are modelled by higher order differential equations. An nth order autonomous differential equation system is given as

$$y^{(n)}(x) = f(y, y', y'', \cdots, y^{(n-1)}).$$

To solve such a system we need initial values for $y, y', y'', \cdots, y^{(n-1)}$. The existence and uniqueness of the solution for an initial value problem is guaranteed if the function f satisfies a Lipschitz condition [12].

Definition 1.0.1. *A function $f : \mathbb{R}^m \to \mathbb{R}^m$ satisfies a Lipschitz condition, if for any $Y, Z \in \mathbb{R}^m$, there exists a Lipschitz constant L such that*

$$\|f(Y) - f(Z)\| \leq L\|Y - Z\|.$$

Generally speaking, the world of ordinary differential equations can be divided into stiff and non-stiff problems. Stiff problems are those in which the components of the differential equations have vastly varying time scales and have large Lipschitz constants. All other problems are non-stiff. This thesis is concerned with the numerical solution of differential equations for non-stiff problems.

The solutions of the system of ordinary differential equations (ODEs) exhibit the behaviour of the underlying physical phenomenon. However, the analytical solutions are difficult to find and numerical approximations of the exact solutions are sought. This is achieved by using numerical methods that take an initial condition and move the solution in the direction specified by the ODEs. Numerical methods are categorised as one-step methods, multistep methods and general linear methods. A brief introduction of these numerical methods is given here and a thorough analysis will be carried out in Chapter 2.

One-step methods calculate the solution $y(x)$ at time x_n, using given information from the previous time x_{n-1}. In doing so, these methods may calculate and use the solution values at different points within the interval $[x_{n-1}, x_n]$. The first and the simplest one-step method is the Euler method and is given by the formula

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

The solution y_{n-1} at time x_{n-1} is given, and we proceed along the tangent at this point with slope $f(y_{n-1})$, to a distance of $h = x_n - x_{n-1}$, where h is known as the stepsize. The Euler method is a low order method and requires a smaller stepsize to achieve a given accuracy for certain classes of differential equation systems. Higher order one-step methods can be constructed by approximating the solution at several points in the integration interval. These type of methods are called Runge–Kutta methods and are given as

$$\begin{aligned} Y_i &= y_{n-1} + \sum_{j=1}^{s} a_{ij} hf(Y_j), \qquad i = 1, 2, \cdots, s, \\ y_n &= y_{n-1} + \sum_{i=1}^{s} b_i hf(Y_i). \end{aligned}$$

Here y_{n-1} is the given value of y at time x_{n-1}. The internal stages Y_i are calculated at quadrature nodes c_i within the interval $[x_{n-1}, x_n]$, b_i are the quadrature weights and a_{ij} is the coefficient matrix for the Runge–Kutta method. The coefficients c_i, b_i, a_{ij} completely characterise a Runge–Kutta method.

Linear multistep methods find the solution at time x_n using available information at a number of previous times $x_{n-1}, x_{n-2}, \cdots$. The general form of a k-step linear multistep method is given as

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

If we take $\alpha_1 = 1$, all other $\alpha_i = 0$ and $\beta_0 = 0$ we obtain,

$$y_n = y_{n-1} + h(\beta_1 f(y_{n-1}) + \beta_2 f(y_{n-2}) + \cdots + \beta_k f(y_{n-k})). \qquad (1.2)$$

If instead $\beta_0 \neq 0$, we obtain methods of the form

$$y_n = y_{n-1} + h(\beta_0 f(y_n) + \beta_1 f(y_{n-1}) + \beta_2 f(y_{n-2}) + \cdots + \beta_k f(y_{n-k})), \qquad (1.3)$$

By selecting the β_i suitably in (1.2) we obtain order k. The method is then known as Adams-Bashforth method. Similarly in (1.3), if the β_i are chosen to attain order $k+1$, we obtain Adams-Moulton methods.

Linear multistep methods require values of solution y at more than one point in the past and are implemented in a recursive manner. To start the process, a starting method is generally employed. Usually one-step methods such as Runge–Kutta methods are used as starting methods. Once the data is available to start the procedure, a multistep method is then employed. Here Newton iterations are used for stiff problems and fixed point iteration is used for non-stiff problems. On the other hand, Milne [40] suggested to use Adams-Bashforth as predictor and Adams-Moulton as corrector methods.

General linear methods are multistage and multivalue methods and are natural generalisations of linear multistep and Runge-Kutta methods. A general linear method is given as

$$\begin{aligned} Y &= hAf(Y) + Uy^{[n-1]}, \\ y^{[n]} &= hBf(Y) + Vy^{[n-1]}. \end{aligned}$$

Here A, U, B, V are matrices representing a particular general linear method where

$$Y = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_s \end{bmatrix}, \quad f(Y) = \begin{bmatrix} f(Y_1) \\ f(Y_2) \\ \vdots \\ f(Y_s) \end{bmatrix}, \quad y^{[n-1]} = \begin{bmatrix} y_1^{[n-1]} \\ y_2^{[n-1]} \\ \vdots \\ y_r^{[n-1]} \end{bmatrix}, \quad y^{[n]} = \begin{bmatrix} y_1^{[n]} \\ y_2^{[n]} \\ \vdots \\ y_r^{[n]} \end{bmatrix}$$

At the start of a step n, r quantities $y^{[n-1]}$ are required as input values. However, only one input value is available with the IVP, which is the initial condition. A starting method is therefore required to obtain the r input values. Once these are known, we calculate s stages, Y and finally we calculate the output values $y^{[n]}$.

Numerical methods approximate the exact solutions and hence introduce errors. The error at each time step is referred to as local truncation error or the discretization error. These errors are calculated by comparing the numerical solution at time x_n with the Taylor series of the exact solution around x_n. A related quantity is the order of the method. A method is of order p if the Taylor series of the exact solution matches the numerical solution up to $O(h^p)$, such that the residue has the leading order term with $O(h^{p+1})$. The local error accumulates over the course of the numerical integration and results in the global error. Numerical methods which produce small global errors have always been a preferred choice. However they do not always respect the qualitative features of the problem.

Traditionally, the emphasis has been on getting good quantitative behaviour of the approximate solution. Consistency, stability and convergence has always been the desired and often required goal of these numerical methods and these will be studied in details in Chapter 2 and Chapter 3. A consistent numerical method ensures a correct solution to the simplest IVP, $y'(x) = 1$ with $y(0) = 0$. Stability of a numerical method is a guarantee for bounded numerical solutions for an ODE having bounded exact solutions. Stability in fact plays an important role in the selection of a numerical method for the solution of stiff differential equations. A numerical method is of no use if the numerical solution does not converge to the exact solution during the course of time. All these criteria focus on obtaining the quantitatively correct numerical solutions of the ODEs. However there exist classes of ODEs where qualitative behaviour of the solution is more important than accuracy, and this is the main topic explored in this thesis.

Many physical systems obey certain natural laws which traditional numerical methods do not take into account. We are interested in numerical methods that preserve these laws, or in other words, conserve the geometric properties of the flow of ODEs while having good quantitative properties including stability and convergence. These numerical methods are called geometric integrators or structure preserving numerical methods and they offer a good hope for reliable long time simulations while observing physically natural laws. Although there exist multitude of systems observing several laws, a partial discussion of the qualitative features of some of the systems is summarised here.

The nature of differential equations and the physical laws they maintain, determines an appropriate geometric integrator. If the differential equations evolve on Lie groups which are differentiable manifolds, we use Lie group methods [34]. If the solutions of differential equations possess symmetry, we use symmetry preserving methods. A detailed discussion is available in [30]. However, in this thesis, we explore geometric integrators for the differential equations having quadratic invariants, an important class of which is Hamiltonian systems.

The phase space in which a system evolves provides an understanding of the geometrical properties of its solutions. For example it is well known that the solutions of Hamiltonian systems preserve symplectic structure in its phase space. A detailed analysis of this is given later in this chapter. These integrators are called symplectic integrators and they preserve various quantities including the symplectic structure of the solutions of Hamiltonian systems.

One may ask why is it important to preserve qualitative properties? Since many of the above properties are naturally present in the systems, it makes sense to preserve them numerically. As an example we note that the planets revolve around the sun in fixed orbits. No matter how accurate we mimic the motion of such planets by our traditional numerical methods, if the orbits are not preserved, that would mean the planets will either collide with the sun or go far away which is physically incorrect. As a further example we consider the harmonic oscillator problem which is a Hamiltonian system describing the motion of a unit mass attached to a spring with momentum p and position co-ordinates q given by the ordinary differential equation system

$$q' = p, \qquad p' = -q. \tag{1.4}$$

The total energy of the Hamiltonian system is a conserved quantity given by

$$H = \frac{p^2}{2} + \frac{q^2}{2}.$$

An application of the Euler method to solve (1.4) yields

$$p_{n+1}^2 + q_{n+1}^2 = (1+h^2)(p_n^2 + q_n^2).$$

Since

$$(1+h^2)^n \approx 1 + h(nh),$$

we obtain a linear error growth in the energy of the Hamiltonian system as depicted in Figure 1.1. No matter how small the stepsize h is, the qualitative feature of the Harmonic oscillator is lost.

1.1 Hamiltonian systems

Classical mechanics is a branch of physics which deals with physical laws governing the motion of bodies. Hamiltonian mechanics is a reformulation of classical mechanics in which the equations of motion are based on generalised co-ordinates q_i and generalised momenta p_i. The equations of motion are called Hamiltonian system with Hamiltonian

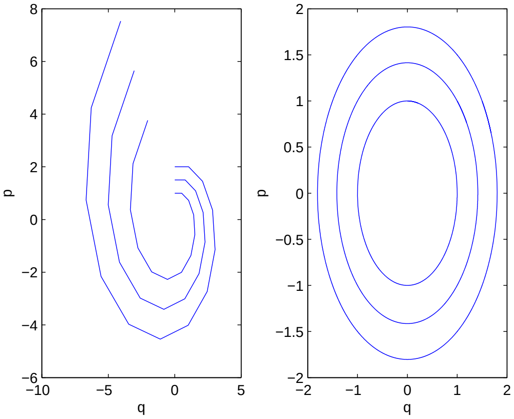


Figure 1.1: The energy of Harmonic oscillator calculated with the Euler method in the left figure and exact Hamiltonian in the figure on the right with different energy levels.

H which is a function of $p = (p_1, p_2, \cdots, p_n)$ and $q = (q_1, q_2, \cdots, q_n)$ and defines the differential equation system

$$\frac{dp_i}{dx} = -\frac{\partial H}{\partial q_i}, \qquad \frac{dq_i}{dx} = \frac{\partial H}{\partial p_i}, \qquad i = 1, \cdots, n, \tag{1.5}$$

having n degrees of freedom. H usually corresponds to the total energy of the underlying mechanical system and is the sum of its kinetic and potential energies. Another reformulation of classical mechanics is Lagrangian mechanics which describes the motion of bodies using configuration space. The Lagrangian L of a mechanical system is the difference of its kinetic energy and potential energy and defines the differential equation system known as Euler Lagrange equations

$$\frac{\partial L}{\partial q_i} = \frac{d}{dx}\left(\frac{\partial L}{\partial q_i'}\right). \tag{1.6}$$

The equations of motion in Lagrangian mechanics can be converted to the equations of motion in Hamiltonian mechanics via the Legendre transformation of the Lagrangian L which is given as

$$H = \sum_i p_i q_i' - L. \tag{1.7}$$

This is visualised by taking the total differential of the Hamiltonian $H(p,q,x)$ and then comparing it with the differential of the equation (1.7) and using the Euler Lagrange equations (1.6).

If we write $y = (p,q)$, the differential equation system (1.5) can be written as

$$y' = J^{-1}\nabla H. \tag{1.8}$$

∇ is a gradient operator and J is a skew symmetric matrix consisting of the zero matrix 0 and $n \times n$ identity matrix I given as

$$J = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix}. \tag{1.9}$$

In practice, we often deal with separable Hamiltonian systems where the Hamiltonian function H has the form

$$H(p,q,x) = T(p) + V(q,x).$$

Here T represents kinetic energy and V represents potential energy.

We have already encountered an example of a Hamiltonian system namely, the Harmonic oscillator in (1.4). We present another example of a Hamiltonian system here, and few more examples are given in Chapter 5.

Example 1.1.1. *The simple pendulum*

Consider a simple pendulum having unit mass of bob attached to a rod of unit length and having acceleration due to gravity as unity. The equations of motion of the simple pendulum defines a Hamiltonian system with generalised momenta p and generalised coordinates q and are given as

$$p' = -\sin(q), \qquad q' = p.$$

The total energy H is given as

$$H = \frac{p^2}{2} - \cos(q).$$

1.1.1 Conservation of energy

For autonomous Hamiltonian systems, their total energy remains conserved. This means that the value of Hamiltonian H remains constant along the solution of the system. Differentiate $H(p,q)$ with respect to time

$$\frac{dH}{dx} = \sum_i \left(\frac{\partial H}{\partial p_i} \frac{dp_i}{dx} + \frac{\partial H}{\partial q_i} \frac{dq_i}{dx} \right) = 0. \tag{1.10}$$

If traditional numerical methods are used to solve Hamiltonian systems, the energy of the system is not conserved. Structure preserving numerical methods provide solutions of Hamiltonian systems that conserve energy approximately.

1.1.2 Symplecticity

An important property of Hamiltonian systems is that their phase flow is symplectic i.e. it preserves oriented area in the case of one degree of freedom. The phase space of the Hamiltonian systems is a $2n$ dimensional space with coordinates (p_i, q_i), $i = 1 \cdots n$. The phase flow is the transformation of the phase space via the solution operator ψ such that

$$\psi : (p(0), q(0)) \longmapsto (p(x), q(x)).$$

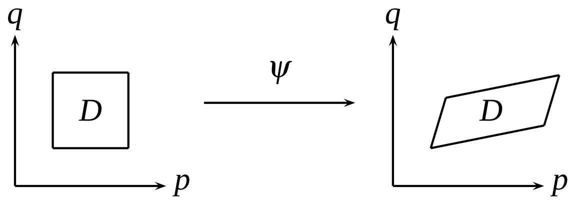
According to Liouville's theorem, the phase flow preserves area in the case of one degree of freedom provided that the vector field f in the phase space has $\operatorname{div} f = 0$.

For Hamiltonian systems, the vector field is given as

$$f = \left[\frac{-\partial H}{\partial q}, \frac{\partial H}{\partial p}\right],$$

$$\operatorname{div} f = \frac{-\partial^2 H}{\partial q \partial p} + \frac{\partial^2 H}{\partial p \partial q} = 0.$$

Hence the solution operator ψ representing the phase flow of the Hamiltonian systems is symplectic. The symplecticity of ψ can be understood from the picture below which represents a sheet of paper having an area D placed in the phase space of a Hamiltonian system and moves along the corresponding phase flow. After transformation through the solution operator ψ, we observe that the paper is stretched but the area of paper remains D.



The notion of symplecticity can be explained in various ways of which three are presented here. The first is via the perturbation in the values of (p, q), the second is via the Jacobian and the third is via the exterior product of the differential two forms. However, these are equivalent in one way or other.

The phase flow of Hamiltonian systems is symplectic. This means that even though the small perturbations in the initial values of momentum p or position q are not conserved by the flow of Hamiltonian systems, the area of a set of possible initial perturbations is preserved. Consider the Hamiltonian system in compact form with $y = (p, q)$ as

$$y' = f(y).$$

If the initial condition $y_0 = (p_0, q_0)$ is perturbed by a small number εz for $\varepsilon \ll 1$ and $z = [z_1, z_2]$, a fixed vector, the solution is modified by $\varepsilon z + O(\varepsilon^2)$. Thus we get

$$y' + \varepsilon z' = f(y) + \varepsilon z f'(y),$$

This implies

$$z' = f'(y)z, \tag{1.11}$$

where

$$f'(y) = \begin{bmatrix} \frac{\partial^2 H}{\partial p \partial q} & \frac{\partial^2 H}{\partial p^2} \\ -\frac{\partial^2 H}{\partial q^2} & -\frac{\partial^2 H}{\partial p \partial q} \end{bmatrix}.$$

We note that

$$\text{trace}(f'(y)) = \frac{\partial^2 H}{\partial p \partial q} - \frac{\partial^2 H}{\partial p \partial q} = 0. \tag{1.12}$$

Let us consider another similar perturbation $\varepsilon \tilde{z}$ for $\varepsilon \ll 1$ and $\tilde{z} = [\tilde{z}_1, \tilde{z}_2]$ such that we obtain

$$\tilde{z}' = f'(y)\tilde{z}, \tag{1.13}$$

Consider the matrix

$$Z = \begin{bmatrix} z_1 & \tilde{z}_1 \\ z_2 & \tilde{z}_2 \end{bmatrix},$$

where the columns of Z are perturbations in the initial condition. We note that

$$\det(Z) = z_1\tilde{z}_2 - \tilde{z}_1 z_2,$$

$$\frac{d}{dx}\det(Z) = z_1'\tilde{z}_2 + z_1\tilde{z}_2' - \tilde{z}_1' z_2 - \tilde{z}_1 z_2', \tag{1.14}$$

From (1.11), (1.12) and (1.14) we get

$$\frac{d}{dx}\det(Z) = \big(z_1\tilde{z}_2 - \tilde{z}_1 z_2\big)\big(\frac{\partial^2 H}{\partial p \partial q} - \frac{\partial^2 H}{\partial p \partial q}\big) = 0.$$

Hence the area formed by the components of Z is conserved.

To explain symplecticity in terms of the Jacobian, again consider the linear transformation $\psi : (p,q) \longmapsto (p^*, q^*)$. ψ is symplectic if, $\psi'^T J \psi' = J$, provided the Jacobian of transformation has a unit determinant. Here J is given by (1.9).

To prove this, let us assume that the Jacobian of the transformation has a unit determinant.

$$\begin{vmatrix} \frac{\partial p^*}{\partial p} & \frac{\partial q^*}{\partial p} \\ \frac{\partial p^*}{\partial q} & \frac{\partial q^*}{\partial q} \end{vmatrix} = \frac{\partial p^* \partial q^*}{\partial p \partial q} - \frac{\partial p^* \partial q^*}{\partial q \partial p} = I.$$

Now

$$\psi' = \begin{bmatrix} \frac{\partial p^*}{\partial p} & \frac{\partial p^*}{\partial q} \\ \frac{\partial q^*}{\partial p} & \frac{\partial q^*}{\partial q} \end{bmatrix}.$$

Thus

$$\begin{aligned} \psi'^T J \psi' &= \begin{bmatrix} \frac{\partial p^* \partial q^*}{\partial p \partial p} - \frac{\partial p^* \partial q^*}{\partial p \partial p} & \frac{\partial p^* \partial q^*}{\partial p \partial q} - \frac{\partial p^* \partial q^*}{\partial q \partial p} \\ \frac{\partial p^* \partial q^*}{\partial q \partial p} - \frac{\partial p^* \partial q^*}{\partial p \partial q} & \frac{\partial p^* \partial q^*}{\partial q \partial q} - \frac{\partial p^* \partial q^*}{\partial q \partial q} \end{bmatrix} \\ &= \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix} = J. \end{aligned}$$

Symplecticity can also be understood in terms of the solution of Hamiltonian systems preserving symplectic structure on a manifold on which the solution exists. We need the following definitions.

Manifold A manifold is a configuration space that looks locally like a Euclidean space. A sphere is an example of a two dimensional manifold.

Differentiable manifold A manifold having a differential structure is called a differentiable manifold. This allows us to do differential calculus on the manifold.

1-form Let $\mathbb{R}^n$ be a $n-$dimensional real vector space. A 1-form is a linear function $w : \mathbb{R}^n \to \mathbb{R}$ such that for $x, y \in \mathbb{R}^n$

$$w(\lambda_1 x, \lambda_2 y) = \lambda_1 w(x) + \lambda_2 w(y).$$

2-form A 2-form is a bilinear, skew symmetric function on pair of vectors

$$w^2 : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R},$$

such that

$$\begin{aligned} w^2(\lambda_1 x + \lambda_2 y, z) &= \lambda_1 w^2(x,z) + \lambda_2 w^2(y,z), \\ w^2(x,y) &= -w^2(y,x). \end{aligned}$$

Exterior product The exterior product of two 1-forms w_1 and w_2 on the pair of vectors $\xi, \eta \in \mathbb{R}^n$ is a 2-form and gives the oriented area of the projection of parallelogram on w_1-w_2 plane. The exterior product is skew-symmetric, distributive and associative.

$$(w_1 \wedge w_2)(\xi, \eta) = \begin{vmatrix} w_1(\xi) & w_2(\xi) \\ w_1(\eta) & w_2(\eta) \end{vmatrix}.$$

Differential 1-form Consider a manifold M. Let TM_x be the space of all tangents to M at point $x \in M$. TM_x is called the tangent space of M at x and the union of all such tangent spaces at all points on M is called the tangent bundle TM. Differential 1-form on manifold M is a smooth map

$$w : TM \to \mathbb{R}^n.$$

Differential 2-form Differential 2-form is obtained by the exterior product of two differential 1-forms.

A symplectic structure on an even dimensional manifold M is a closed, non-degenerate differential 2-form on M. We consider the transformation $\psi : (p,q) \longmapsto (p^*,q^*)$ and two differential 1-forms

$$\begin{aligned} dp^* &= \tfrac{\partial p^*}{\partial p} dp + \tfrac{\partial p^*}{\partial q} dq, \\ dq^* &= \tfrac{\partial q^*}{\partial p} dp + \tfrac{\partial q^*}{\partial q} dq. \end{aligned}$$

Now the exterior product of these differential 1-forms, dp^* and dq^*, will provide a differential 2-form $dp^* \wedge dq^*$, which represents the oriented area of a parallelogram with sides dp^* and dq^*.

$$\begin{aligned} dp^* \wedge dq^* = & \frac{\partial p^*}{\partial p}\frac{\partial q^*}{\partial p} dp \wedge dp + \frac{\partial p^*}{\partial p}\frac{\partial q^*}{\partial q} dp \wedge dq \\ & + \frac{\partial p^*}{\partial q}\frac{\partial q^*}{\partial p} dq \wedge dp + \frac{\partial p^*}{\partial q}\frac{\partial q^*}{\partial q} dq \wedge dq. \end{aligned}$$

The exterior product is skew-symmetric and the Jacobian of the transformation should be unity. Therefore

$$dp^* \wedge dq^* = dp \wedge dq$$

Hence the area of the parallelogram with sides dp and dq is preserved after the transformation through ψ.

1.1.3 Linear and quadratic invariants

Hamiltonian systems belong to an important class of differential equation system where the solution is known to possess invariants. Consider an initial value problem whose solution y has an invariant $I(y)$

$$y'(x) = f(y(x)), \qquad y(x_0) = y_0. \tag{1.15}$$

$I(y)$ is called a first integral of (1.15) if

$$I'(y)f(y) = 0, \qquad \forall y.$$

The energy H of a Hamiltonian system (1.5) is a first integral as shown in (1.10).
Many physical systems have invariants which are quadratic in nature. The quadratic function

$$I(y) = y^T Qy,$$

is an invariant of (1.15) if

$$y^T Qf(y) = 0.$$

where Q is a symmetric square matrix.

Example 1.1.2. *Euler equations for rigid body motion*

Rigid bodies are solid objects such that the distance between any two points on or inside it is constant. The mathematical equations governing the motion of a rigid body were derived by Euler and were given the name Euler equations. Assume the frame of reference is fixed in the rigid body and the center of mass of the rigid body is located at the origin, then the Euler equations are given as

$$\frac{d\omega_x}{dt} = \frac{I_{yy} - I_{zz}}{I_{xx}}\omega_y\omega_z, \tag{1.16}$$

$$\frac{d\omega_y}{dt} = \frac{I_{zz} - I_{xx}}{I_{yy}}\omega_z\omega_x, \tag{1.17}$$

$$\frac{d\omega_z}{dt} = \frac{I_{xx} - I_{yy}}{I_{zz}}\omega_x\omega_y, \tag{1.18}$$

where $\omega_x, \omega_y, \omega_z$ are the components of angular velocity along the principal axis and I_{xx}, I_{yy}, I_{zz} are the principal moments of inertia. The motion of a rigid body has two underlying quadratic invariants namely, the kinetic energy H and the squared norm of angular momentum A, and are given as

$$H = \tfrac{1}{2}\begin{bmatrix} \omega_x & \omega_y & \omega_z \end{bmatrix} \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}, \tag{1.19}$$

$$A = \begin{bmatrix} \omega_x & \omega_y & \omega_z \end{bmatrix} \begin{bmatrix} I_{xx}^2 & 0 & 0 \\ 0 & I_{yy}^2 & 0 \\ 0 & 0 & I_{zz}^2 \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}. \tag{1.20}$$

1.2 Numerical methods for Hamiltonian systems

Hamiltonian systems model a wide variety of applications ranging from celestial mechanics to fluid dynamics and many more. It is vital to preserve the characteristic properties of these systems during the calculation of their approximate solutions via numerical methods. The important properties are the symplectic structure of the phase flow, the conservation of linear and quadratic invariants and in applications such as $N-$body simulations, the time reversible symmetry. An application of a numerical method to solve the Hamiltonian systems with a fixed stepsize approximates its continuous flow map with a discrete flow map. Since the flow of the Hamiltonian systems is symplectic, we want our discrete flow map to be symplectic which gives rise to the concept of symplectic numerical methods. Here it should be mentioned that the selection of variable stepsize can lead to non-symplectic behaviour unless each integration time step is ensured to possess the underlying geometric properties.

Symplectic numerical methods exist for reliable long time integration of Hamiltonian systems. An additional benefit of such methods is their ability to preserve the underlying quadratic invariants effectively. However most of the numerical methods in practice are not symplectic. Multistep methods require more than one initial condition to start with, so they cannot define a map on phase space and hence cannot be symplectic in general.

One-step methods can only be considered as genuine symplectic methods. Many one-step numerical methods have successfully been used in the past without recognising their symplectic behaviour. These include the famous implicit midpoint rule and the Gauss-Legendre methods. Sanz-Serna and Suris systematically developed symplectic Runge-Kutta methods [46]. Their idea is based on features of algebraic stability introduced, in connection with stiff systems, by Burrage and Butcher [4]. General linear methods are multivalue in nature so we cannot expect genuine symplectic behaviour. However a variant of symplectic methods do exist in the class of general linear methods. These methods are called *G*-symplectic methods.

There is a close relation between algebraic stability of numerical methods and symplecticity. Linear stability analysis revolves around the famous Dahlquist test equation $y' = f(y)$, where $f(y) = qy$ is a linear function. However non-linear stability analysis for Runge-Kutta methods by Burrage and Butcher assumes $f(y)$ to be non-linear and dissipative. This gives us a condition on the co-efficients of Runge-Kutta methods $[A, b^T, c]$ that for algebraically stable Runge-Kutta methods we must have

$$M = \operatorname{diag}(b)A + A^T \operatorname{diag}(b) - bb^T,$$

a positive semi-definite matrix [4]. Hamiltonian systems are not dissipative. They in fact conserve the quadratic invariants. This mean that the matrix M should exactly be zero and this is the criteria for symplectic Runge-Kutta methods [46].
Non-linear stability analysis for multistep methods was difficult to grasp. Dahlquist in [23] proposed to use one-leg methods instead of multistep methods to study the non-linear stability of multistep methods for dissipative problems. In multistep methods we use a linear combination of function evaluations f at number of past values while in one-leg methods we evaluate function f only once at the linear combination of past values. One-leg methods were helpful in studying the non-linear stability of dissipative problems and this gives rise to the concept of G-stability. We consider a real, symmetric and positive definite matrix G and the norm

$$\|y\|_G^2 = \sum_{i,j=1}^{r} g_{ij}\langle y_i, y_j\rangle,$$

for

$$y = \begin{bmatrix} y_1 \\ \vdots \\ y_r \end{bmatrix},$$

such that the numerical solution by one-leg methods is contractive under such a G norm which in turn implies nonlinear stability of the numerical methods. The idea of G-stability was extended to study the non-linear stability for general linear methods (A, U, B, V) to solve dissipative problems and it was found that a contractive numerical solution is possible under G norm if the matrix $\tilde{M}$ is positive semi-definite [12] where $\tilde{M}$ is given as

$$\tilde{M} = \begin{bmatrix} DA + A^T D - B^T GB & DU - B^T GV \\ U^T D - V^T GB & G - V^T GV \end{bmatrix}. \tag{1.21}$$

Here D is a positive semi-definite diagonal matrix. Like symplectic Runge-Kutta methods, we can find G-symplectic general linear methods by requiring the matrix $\tilde{M}$ in (1.21) to be exactly equal to zero.

The first G-symplectic general linear method was discovered by Butcher and is given in [11]. This was a two-stage, order four, time reversible method based on Gauss quadrature nodes. Applications of this method on various problems have pointed out that, although it preserves the qualitative features, it has parasitic solutions. This is typical of multivalue methods whereby the initial perturbations in starting approximations are not damped out rather overtakes the actual solution. An analysis of the possible cause of parasitic solutions is carried out and the parasitic growth parameter has surfaced. The parasitic growth parameter is instrumental in the design and implementation of G-symplectic general linear methods avoiding the corruption of numerical solution by parasitic solutions.

The first approach is to use the composition of two G-symplectic general linear methods with parasitic growth parameters having opposite signs. These two methods are implemented side by side in a sequence to control the growth of parasitic solutions. This is due to the addition of the parasitic growth parameters of the two composing methods. Since the composition of two symplectic methods is symplectic, the resulting procedure retains the qualitative features of the underlying Hamiltonian system as expected from a G-symplectic general linear method.

The second approach is to construct G-symplectic general linear methods by ensuring that the parasitic growth parameter is effectively zero. However, it has been analysed that no two-stage, two-step G-symplectic general linear method can be parasitic free. Work has been done on the construction of four stages and three steps G-symplectic general linear method. Since only the implicit methods can be considered, this increases the cost of the method. The cost can be reduced by carefully selecting the structure of the matrix A. As a result, a class of new methods have come out whose performance can be compared to that of traditional Gauss Runge–Kutta methods even though the new methods are multivalue in nature. Time reversal symmetry has also played an important role in the design of this new class of methods.

Chapter 2

Numerical methods for ordinary differential equations

Numerical methods for ordinary differential equations approximate the exact solution of ODE systems. They play a vital role in providing an understanding of the behaviour of the underlying physical system. Many ODEs of practical importance are derived by spatial discretisation of partial differential equations. This leads to large sparse systems and we require numerical methods for their approximate solutions. These methods are categorised as one-step methods, multistep methods or their generalisation, general linear methods.

2.1 Runge–Kutta methods

Runge–Kutta methods are one-step methods for the numerical solutions of initial value problems

$$y'(x) = f(y(x)), \qquad y(x_0) = y_0, \qquad y(x) \in \mathbb{R}^m.$$

The exact solution is $y(x)$ and Runge–Kutta methods provides an approximation at time $x_n = nh$, where $n = 0, 1, \cdots$ and h is the stepsize. The general form of a Runge–Kutta method is

$$\begin{aligned} Y_i &= y_{n-1} + \sum_{j=1}^{s} a_{ij} h f(Y_j), \quad i = 1, 2, \cdots, s, \qquad (2.1) \\ y_n &= y_{n-1} + \sum_{i=1}^{s} b_i h f(Y_i). \end{aligned}$$

where Y_i are s stages calculated during the integration from time x_{n-1} to x_n. The output value y_n is an approximation of the actual solution $y(x_n)$. A Runge–Kutta method is represented by a Butcher tableau

$$\begin{array}{c|cccc} c_1 & a_{11} & a_{12} & \cdots & a_{1s} \\ c_2 & a_{21} & a_{22} & \cdots & a_{2s} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s1} & a_{s2} & \cdots & a_{ss} \\ \hline & b_1 & b_2 & \cdots & b_s \end{array},$$

where b_i are called the weights of the method and $c_i = \sum\limits_{j=1}^{s} a_{ij}$ for $i = 1, \cdots, s$ are the abscissas of the method at which the stages Y_i are evaluated. That is, stage number i provides the following approximation

$$Y_i \approx y(x_{n-1} + hc_i) + O(h^2).$$

Runge–Kutta methods are explicit if $a_{ij} = 0$ for $i \leq j$. This means that the stages can be calculated sequentially. This requires less computation time and hence are a favourite for solving ordinary differential equations. However, explicit methods are less desirable because of their limitation in stability for solving stiff differential equation system. Another reason to avoid explicit Runge–Kutta methods is the inability to solve general Hamiltonian problems which are not separable. The famous Euler method and the midpoint method written in Runge–Kutta formulation respectively are

$$\begin{array}{c|c} 0 & \\ \hline & 1 \end{array}, \qquad \begin{array}{c|cc} 0 & & \\ 1 & 1 & \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}.$$

While higher order explicit Runge–Kutta methods exist, the higher the order, the greater the number of stages required and it is well known that an explicit Runge–Kutta method of order s with only s stages is possible for $s \leq 4$. The classical Runge–Kutta method of order 4 with 4 stages is given as

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

Runge–Kutta methods are implicit if $a_{ij} \neq 0$ for some $i \leq j$. Thus for an s-stage implicit Runge–Kutta method to solve an m dimensional system of ODEs, sm non-linear equations representing the stages need to be solved. This is usually achieved by Newton iterations, which is quite expensive. Hence the general implicit Runge–Kutta methods are at a disadvantage compared to their explicit counterpart when considering the cost of implementation. However, the advantages of using implicit Runge–Kutta methods over explicit methods include the fact that fewer stages are required by implicit Runge–Kutta methods to achieve the same order as that of explicit Runge–Kutta methods. Furthermore, implicit Runge–Kutta methods are the only hope for solving stiff differential equations among the one-step methods, if abnormally small stepsizes need to be avoided. Not only so, certain classes of implicit Runge–Kutta methods are a good candidate for the solution of Hamiltonian and structure preserving ODEs. The most famous implicit Runge–Kutta methods are the Gauss-Legendre Runge–Kutta methods which require s stages to achieve an order $2s$. These methods are based on shifted Legendre polynomials such that the abscissa c_i of the Runge–Kutta methods are the zeros of the shifted Legendre polynomials P_s^* on the interval $[0,1]$ where

$$P_s^*(x) = \frac{s!}{2s}\sum_{k=0}^{s}(-1)^{s-k}\left(\begin{array}{c} s \\ k \end{array}\right)\left(\begin{array}{c} s+k \\ k \end{array}\right)x^k.$$

If we take $s=1$, we get,

$$P_1^*(x) = x - \tfrac{1}{2}.$$

The zero of this polynomial is the abscissa of the 1-stage Runge–Kutta method, i.e. $c_1 = 1/2$ and we get the 1-stage, order 2 implicit midpoint rule

$$\begin{array}{c|c} \frac{1}{2} & \frac{1}{2} \\ \hline & 1 \end{array}.$$

If we take $s=2$, we get

$$P_2^*(x) = x^2 - x + \tfrac{1}{6}.$$

The zeros of this polynomial are the abscissas of the 2-stage Runge–Kutta method, i.e.

$$c_1 = \tfrac{1}{2} - \tfrac{\sqrt{3}}{6}, \qquad c_2 = \tfrac{1}{2} + \tfrac{\sqrt{3}}{6},$$

and we get the 2-stage, order 4 Gauss Runge–Kutta method

$$\begin{array}{c|cc} \frac{1}{2}-\frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4}-\frac{\sqrt{3}}{6} \\ \frac{1}{2}+\frac{\sqrt{3}}{6} & \frac{1}{4}+\frac{\sqrt{3}}{6} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}. \tag{2.2}$$

The values of the coefficients b_i and a_{ij} are calculated from the abscissa c_i in a way to ensure that the order of the method is $2s$.

Implicit Runge–Kutta methods were further developed by Butcher [7] based on Radau and Lobatto quadratures. The rationale behind such methods is to attain L-stability, but it comes at a loss of the order of the method. The idea of stability will be explained in Chapter 3. The first step is to choose the abscissa $c_1 = 0$ or $c_s = 1$ or both of them. The rest of the abscissa are chosen such that, for Radau I methods, the abscissa are the zeros of the quadrature polynomial $P_s^*(x) + P_{s-1}^*(x)$ of order $2s-1$ or, for Radau II methods, the abscissa are the zeros of the quadrature polynomial $P_s^*(x) - P_{s-1}^*(x)$ of order $2s-1$ and, for Lobatto III methods, the abscissas are the zeros of the quadrature polynomial $P_s^*(x) - P_{s-2}^*(x)$ of order $2s-2$.

Although methods based on Radau and Lobatto quadratures have some appealing features, they are not favourable candidates for solving stiff and conservative problems. One reason is the semi-implicit nature of the stages which are expensive for non-stiff problems and less competitive compared to fully implicit Gauss methods for solving stiff problems. The other reason is that most of the methods are not A-stable, a desired property which we will study in detail in Chapter 3. A popular choice for stiff problems is the method Radau IIA. Lobatto methods have been employed to solve separable Hamiltonian problems when used in pairs.

The contributing factor to the cost of implementation of the implicit Runge–Kutta methods is the calculation of the implicit stages using Newton iterations. The cost increases with the dimension of the problem. A cost effective approach is to use diagonally implicit Runge–Kutta methods proposed by Alexander [1], where the matrix a_{ij} is lower triangular which allow us to solve the stages sequentially. All the diagonal entries may be same and the cost to compute them reduces. Nørsett further improved the efficiency of implicit Runge-Kutta methods by choosing matrix a_{ij} to have one point spectrum [42], allowing the stages and output approximation to have the same order. Such methods are very useful for stiff ordinary differential equations because the order achieved by solving them is not the order of the output approximation, rather it is nearly the order of the stages and this is referred to as stage order.

Implicit Runge–Kutta methods are generally not suitable for long time integration of Hamiltonian systems because they introduce non-Hamiltonian perturbations which throw the solution out of Hamiltonian regime. Sanz-Serna [44], Suris [48] independently discovered a condition required for Runge-Kutta methods to be suitable for the long time integration of Hamiltonian problems. Cooper [17] and Lasagni [36] also discovered the same condition but for general quadratic invariants. Such methods are called symplectic Runge–Kutta methods. We will study them in detail in Chapter 3.

2.1.1 Symmetric Runge–Kutta methods

For Hamiltonian systems, the total energy $H(p,q)$ and the symplectic structure of the flow are conserved. This is also true if we reverse the direction of the flow. The Hamiltonian systems are invariant under reflection symmetry, i.e. by changing p with $-p$, the Hamiltonian systems do not change. Thus if we move in phase space of a Hamiltonian system, by reversing the direction of the momentum vector p, we will still be following the solution curves in the same phase space but moving in the opposite direction. This give rise to the concept of time reversal symmetry. Most of the conserved mechanical systems share this property. All second order differential equation systems which can be written as a system of first order differential equations have this property as well.

It is natural to require the numerical method to possess time reversal symmetry to correctly mimic the behaviour of the ordinary differential equation system. A symmetric method when applied to solve an ordinary differential equation takes an initial condition y_0 to the next value y_1 with stepsize h, and when applied to solve the same ordinary differential equation with the stepsize $-h$ and initial condition y_1, yields the output y_0. One important property of symmetric methods lies in the fact that they have an even order. This is particularly helpful in the construction of a method because if for example we construct a symmetric method of order 3, it automatically becomes a 4th order method.

A Runge–Kutta method is symmetric if it is equal to its adjoint method. The adjoint of a Runge–Kutta method is also a Runge–Kutta method such that it undoes the work of original Runge–Kutta method but with the sign of h reversed. Consider the general form of a Runge–Kutta method (2.1) written with $A=[a_{ij}]$, $b^T=[b_1,b_2,\cdots,b_s]$

$$\begin{aligned} Y &= \mathbf{1}y_{n-1}+hAf(Y),\\ y_n &= y_{n-1}+hb^T f(Y). \end{aligned}$$

Here the stages are evaluated at abscissa c_i such that $c_1 < c_2 < \cdots < c_s$. There is a possibility that some of the c_i are equal. The adjoint of the Runge–Kutta method is

$$\begin{aligned} PY &= \mathbf{1}y_n - h(\mathbf{1}b^T P - PAP)(Pf(Y)),\\ y_{n-1} &= y_n - h(b^T P)(Pf(Y)). \end{aligned}$$

Here the permutation matrix P is used given as

$$P = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & \cdots & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{2.3}$$

The stages of the adjoint method are evaluated in reverse direction at abscissa c_i such that $c_s < c_{s-1} < \cdots < c_1$. The permutation matrix ensures the reordering of the stages to align with the actual method such that for s stages

$$(PY)_j = Y_{s+1-j}.$$

Thus a Runge–Kutta method is symmetric if

$$\begin{aligned} b^T &= b^T P, \\ A + PAP &= \mathbf{1} b^T. \end{aligned}$$

The Gauss method of any order is symmetric. For example, when $s = 2$ we obtain the method given in (2.2) and

$$\begin{aligned} b^T P &= [\tfrac{1}{2} \quad \tfrac{1}{2}] \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \\ &= [\tfrac{1}{2} \quad \tfrac{1}{2}], \end{aligned}$$

and,

$$\begin{aligned} A + PAP &= \begin{bmatrix} \frac{1}{4} & \frac{1}{4} - \frac{\sqrt{3}}{6} \\ \frac{1}{4} + \frac{\sqrt{3}}{6} & \frac{1}{4} \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{4} & \frac{1}{4} - \frac{\sqrt{3}}{6} \\ \frac{1}{4} + \frac{\sqrt{3}}{6} & \frac{1}{4} \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{4} & \frac{1}{4} - \frac{\sqrt{3}}{6} \\ \frac{1}{4} + \frac{\sqrt{3}}{6} & \frac{1}{4} \end{bmatrix} + \begin{bmatrix} \frac{1}{4} & \frac{1}{4} + \frac{\sqrt{3}}{6} \\ \frac{1}{4} - \frac{\sqrt{3}}{6} & \frac{1}{4} \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix}. \end{aligned}$$

2.1.2 Error analysis and order

The accuracy of a numerical method is analysed using local and global truncation errors. The error introduced per integration step is termed as local error while the global error is the overall error accumulated during the whole integration process. The round-off errors in the computations are often small compared to the truncation errors and can be neglected. The local truncation error is evaluated by comparing the numerical solution at time x_n with the Taylor series expansion of the exact solution around x_n. The actual solution of a differential equation system is generally not known. An approach due to Richardson is usually employed whereby the given ODE is first solved with initial condition (x_0,y_0) and step size h and obtain two solutions y_1 and y_2 after two steps of the numerical method. Later the same ODE is solved with initial condition (x_0,y_0) and step size $2h$ to get a solution $\tilde{y}$. The local error is estimated by comparing y_2 with $\tilde{y}$. Generally speaking, the higher the order of a numerical method, the lower the error.

The order of a numerical method is a measure to check how close the approximate solution is to the exact solution. By subtracting the numerical solution over one step and the corresponding Taylor series expansion of the exact solution, a numerical method is of order p, if the residue has leading order term with $O(h^{p+1})$. To understand the order of a Runge–Kutta method, we consider the first few terms of an expanded Runge–Kutta method

$$y_n = y_{n-1} + \sum_{i=1}^{s} b_i h y'(x_{n-1}) + \sum_{i=1}^{s} b_i c_i h^2 y''(x_{n-1}) + \cdots .$$

The first few terms of the Taylor series of the exact solution are

$$y(x_{n-1}+h) = y(x_{n-1}) + hy'(x_{n-1}) + \tfrac{h^2}{2!} y''(x_{n-1}) + \cdots .$$

Comparing the two series yields

$$\sum_{i=1}^{s} b_i = 1,$$
$$\sum_{i=1}^{s} b_i c_i = \tfrac{1}{2}.$$

This is a system of algebraic equations involving the coefficients of the Runge–Kutta method. These equations are known as order conditions. For a Runge–Kutta method to have a specified order, its coefficients should satisfy these order conditions. For lower order methods, fewer order conditions are to be satisfied. However as the order increases, the number of order conditions also increases. Thus, for example, there are four order conditions a Runge-Kutta method must satisfy to achieve an order of three and for an order five Runge–Kutta method, the number of order conditions increases to seventeen. It is therefore advantageous to use a systematic approach and this depends on the use of rooted trees as a graphical representation of the order conditions.

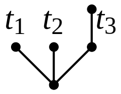
2.1.3 Rooted trees

A tree is a graph having vertices and edges. A rooted tree is a connected non-cyclic graph in which a vertex is assigned as root. Let T denote the set of all rooted trees including the empty tree ϕ. Any $t \in T$ can be defined recursively by removing the root of the tree t and denoting the distinct subtrees as $t_1, t_2, t_3, \cdots, t_m$. The relationship between t and $t_1, t_2, t_3, \cdots, t_m$ is written as $t = [t_1^{n_1}, t_2^{n_2}, t_3^{n_3}, \cdots, t_m^{n_m}]$ where $n_1 \cdots n_m$ are the number of times the tree $t_1 \cdots t_m$ occurs. A single tree with only one vertex is represented by τ and denoted by •. A tree with two vertices is represented by $[\tau]$ and denoted by . Let us consider an example of a tree where $m = 3$.

$$t_1 = t_2 = \tau = \bullet,$$

$$t_3 = [\tau] =$$.

So we can write

$t =$ 

as $t = [t_1 t_2 t_3] = [\tau\tau[\tau]]$.

Order The number of vertices of a tree $t = [t_1^{n_1}, t_2^{n_2}, t_3^{n_3}, \cdots, t_m^{n_m}]$ is called the order of the tree and is denoted by $r(t)$.

$$r(t) = 1 + n_1 r(t_1) + \cdots + n_m r(t_m),$$

$$r(\phi) = 0, \qquad r(\tau) = 1.$$

Density The density $\gamma(t)$ of a tree $t = [t_1^{n_1}, t_2^{n_2}, t_3^{n_3}, \cdots, t_m^{n_m}]$ is a measure of the non-bushiness of the tree. The density of a rooted tree is computed recursively and is a product of the order of the tree and the densities of subtrees when the root is chopped off.

$$\gamma(t) = r(t)\gamma(t_1)^{n_1} \cdots \gamma(t_m)^{n_m},$$

$$\gamma(\phi) = 0, \qquad \gamma(\tau) = 1.$$

Symmetry The symmetry is the order of the automorphism group of t and is denoted by $\sigma(t)$

$$\sigma(t) = (\sigma(t_1)^{n_1} \cdots \sigma(t_m)^{n_m})(n_1! \cdots n_m!).$$

In addition to the functions on the trees defined above, we can associate some combinatorial properties with the trees. A tree can also be uniquely labelled. Let $\alpha(t)$ represent the number of ways of labelling t with an ordered set and $\beta(t)$ is the number of ways of labelling t with an unordered set. It is shown by Butcher in [12] that

$$\alpha(t) = \frac{r(t)!}{\sigma(t)\gamma(t)},$$
$$\beta(t) = \frac{r(t)!}{\sigma(t)}.$$

2.1.4 Elementary differentials

To determine the order of a Runge–Kutta method, the numerical solution is compared with the Taylor series expansion of the exact solution. For the comparison, $f(y(x))$ needs to be differentiated several times.

$$\begin{aligned}
y'(x) &= f(y(x)) \\
&= \mathbf{f}. \\
y''(x) &= f'(y(x))f(y(x)) \\
&= \mathbf{f}'\mathbf{f}. \\
y'''(x) &= f''(y(x))(f(y(x), f(y(x)) + f'(y(x))f'(y(x))f(y(x)) \\
&= \mathbf{f}''(\mathbf{f},\mathbf{f}) + \mathbf{f}'\mathbf{f}'\mathbf{f}.
\end{aligned}$$

and so on. The quantities $\mathbf{f}$, $\mathbf{f}'\mathbf{f}$, $\mathbf{f}''(\mathbf{f},\mathbf{f})$, $\mathbf{f}'\mathbf{f}'\mathbf{f}$ are called elementary differentials $F(t)(y(x))$. As the order of differentiation increases, the number of terms also increases. To handle the situation, these elementary differentials are represented by trees and a recursive formula for their construction is given by

$$F(t)(y) = \begin{cases} y(x), & \text{if} \quad t = \phi, \\ f(y(x)), & \text{if} \quad t = \tau, \\ f^{(m)}(y(x))(F(t_1)(y(x)),\ldots,F(t_m)(y(x))), & \text{if} \quad t = [t_1 t_2 \cdots t_m]. \end{cases}$$

The connection between elementary differentials and trees is given in the Table 2.1.

Theorem 2.1.1. *If $y(x)$ is k times differentiable then*

$$y^{(k)}(x) = \sum_{r(t)=k} \alpha(t) F(t)(y(x)). \tag{2.4}$$

The proof can be found in [12].

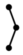
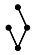
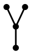
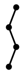
Tree t	Order $r(t)$	Density $\gamma(t)$	Symmetry $\sigma(t)$	$\alpha(t)$	$\beta(t)$	El. differentials $F(t)(y)$	El. weights $\Phi(t)$
	1	1	1	1	1	$\mathbf{f}$	$\sum_{i=1}^{s} b_i$
	2	2	1	1	1	$\mathbf{f}'\mathbf{f}$	$\sum_{i=1}^{s} b_i c_i$
	3	3	2	1	3	$\mathbf{f}''(\mathbf{f},\mathbf{f})$	$\sum_{i=1}^{s} b_i c_i^2$
	3	6	1	1	6	$\mathbf{f}'\mathbf{f}'\mathbf{f}$	$\sum_{i=1}^{s} b_i a_{ij} c_j$
	4	4	6	1	4	$\mathbf{f}'''(\mathbf{f},\mathbf{f},\mathbf{f})$	$\sum_{i=1}^{s} b_i c_i^3$
	4	8	1	3	24	$\mathbf{f}''(\mathbf{f},\mathbf{f}'\mathbf{f})$	$\sum_{i,j=1}^{s} b_i c_i a_{ij} c_j$
	4	12	2	1	12	$\mathbf{f}'\mathbf{f}''(\mathbf{f},\mathbf{f})$	$\sum_{i,j=1}^{s} b_i a_{ij} c_j^2$
	4	24	1	1	24	$\mathbf{f}'\mathbf{f}'\mathbf{f}'\mathbf{f}$	$\sum_{i,j,k=1}^{s} b_i a_{ij} a_{jk} c_k$

Table 2.1: Notation for trees and various functions on trees up to order 4.

The Taylor expansion of the exact solution $y(x_n)$ to order p can be found in terms of elementary differentials as follows:

$$\begin{aligned}
y(x_n) &= y(x_{n-1}) + \sum_{k=1}^{p} \frac{h^k}{k!} y^{(k)}(x_{n-1}) + O(h^{p+1}) \\
&= y(x_{n-1}) + \sum_{k=1}^{p} \frac{h^k}{k!} \sum_{r(t)=k} \alpha(t) F(t)(y(x_{n-1})) + O(h^{p+1}) \\
&= y(x_{n-1}) + \sum_{\substack{t\in T \\ r(t)\leq p}} h^{r(t)} \frac{1}{\sigma(t)\gamma(t)} F(t)(y(x_{n-1})) + O(h^{p+1}). \qquad (2.5)
\end{aligned}$$

2.1.5 Elementary weights

The numerical solution can also be expressed in terms of trees. It has been shown by Butcher in [12] that for an s-stage Runge–Kutta method, its coefficients are related to trees via elementary weights Φ given as

$$\Phi(t) = \begin{cases} \sum_{i=1}^{s} b_i, & \text{if} \quad t = \tau, \\ \sum_{i=1}^{s} b_i \Phi_i(t_1)\Phi_i(t_2)\cdots\Phi_i(t_m), & \text{if} \quad t = [t_1 t_2 \cdots t_m]. \end{cases}$$

where $\Phi_i(t)$ is the elementary stage weight for the i^{th} stage and is defined by

$$\Phi_i(t) = \begin{cases} \sum_{j=1}^{s} a_{ij} = c_i, & \text{if} \quad t = \tau, \\ \sum_{j=1}^{s} a_{ij} \Phi_j(t_1)\Phi_j(t_2), \ldots, \Phi_j(t_m), & \text{if} \quad t = [t_1 t_2 \cdots t_m]. \end{cases}$$

Table 2.1 shows the elementary weights for the trees of order up to 4. There are two special elementary weight functions of particular interest. The first one is the ith derivative operator D_i which maps the solution $y(x)$ to $h^i y^{(i)}(x)$.

$$D_i(t) = \begin{cases} \frac{r(t)!}{\gamma(t)}, & \text{if} \quad r(t) = i, \\ 0, & \text{if} \quad r(t) \neq i. \end{cases} \qquad (2.6)$$

The widely used derivative operator is D_1 or simply D given as

$$D(t) = \begin{cases} 1, & \text{if} \quad t = \tau, \\ 0, & \text{if} \quad t \neq \tau. \end{cases} \qquad (2.7)$$

The second elementary weight function of interest is the exact solution of the differential equation represented by the Picard iterations $E(t)$

$$E^{(n)}(t) = \frac{n^{r(t)}}{\gamma(t)},$$

and the special cases are

$$E(t) = \frac{1}{\gamma(t)}, \qquad E^{-1}(t) = \frac{(-1)^{r(t)}}{\gamma(t)}.$$

It has been shown by Butcher in [12] that the Taylor series expansions for the numerical solution of a Runge–Kutta method is

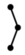
$$y_n = y_{n-1} + \sum_{r(t)\le p} \frac{1}{\sigma(t)} \Phi(t) h^{r(t)} F(t)(y_{n-1}) + O(h^{p+1}). \tag{2.8}$$

For a Runge–Kutta method to have order p, the equations (2.5) and (2.8) should match each other up to order $O(h^{p+1})$. This results in the corresponding order conditions

$$\Phi(t) = \frac{1}{\gamma(t)}, \qquad r(t) \le p.$$

Thus, for a Runge–Kutta method of order three, the following conditions need to be satisfied.

order 3:

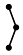
t	$\Phi(t) = \frac{1}{\gamma(t)}$
	$\sum_{i=1}^{s} b_i = 1$
	$\sum_{i=1}^{s} b_i c_i = \frac{1}{2}$
	$\sum_{i=1}^{s} b_i c_i^2 = \frac{1}{3}$
	$\sum_{i,j=1}^{s} b_i a_{ij} c_j = \frac{1}{6}$

2.1.6 Simplifying assumptions

The number of order conditions increases as we seek a higher order Runge–Kutta method. Butcher introduced simplifying assumptions in [6] to reduce the number of order conditions that are required by a Runge–Kutta method to have a particular order.

$$
\begin{aligned}
&B(p): \sum_{i=1}^{s} b_i c_i^{k-1} = \frac{1}{k}, && && k = 1,2,\cdots,p,\\
&C(\eta): \sum_{j=1}^{s} a_{ij} c_j^{k-1} = \frac{c_i^k}{k}, && i = 1,2,\cdots,s, && k = 1,2,\cdots,\eta,\\
&D(\xi): \sum_{i=1}^{s} b_i c_i^{k-1} a_{ij} = \frac{b_j(1-c_j^k)}{k}, && j = 1,2,\cdots,s, && k = 1,2,\cdots,\xi,\\
&E(\eta,\xi): \sum_{j=1}^{s}\sum_{i=1}^{s} b_i c_i^{k-1} a_{ij} c_j^{l-1} = \frac{1}{l(k+l)} && l = 1,2,\cdots,\xi, && k = 1,2,\cdots,\eta.
\end{aligned}
$$

- The $B(p)$ condition ensures that the method has quadrature order p and the order conditions of bushy trees like •, , , $\cdots$ are satisfied for trees up to order p.

- The $C(\eta)$ condition is related to the stage order of a method and ensures that the pair of trees like and  give identical order conditions for $k \leq \eta$. In general, if two trees have elementary weight functions having a factor c_i/k and $\sum a_{ij} c_j^{k-1}$ respectively, with the remaining factors being identical, then their elementary weight functions are equal.

- The $D(\xi)$ condition means that the order conditions of a tree having elementary weight functions involving $b_i c_i^{k-1} a_{ij}$ are satisfied if there are other trees having elementary weight functions involving b_j and $b_j c_j^k$. It is interesting to note that for explicit methods with $s = p = 4$, $D(1)$ must hold.

- The $E(\eta,\xi)$ ensures that the method is at least of order $\eta+\xi$, while automatically satisfying the order conditions for trees $[\tau^{k-1}[\tau^{l-1}]]$.

2.1.7 B-series and composition rules

The Taylor series expansion of the numerical solution (2.8) can be written in terms of a formal series as

$$\begin{aligned} B(\kappa(t), y(x)) &= \sum_{t \in T} \frac{\kappa(t)}{\sigma(t)} h^{r(t)} F(t)(y(x)), \\ &= y + h\kappa(\bullet)\mathbf{f}(y) + h^2\kappa(\overset{\bullet}{\underset{\bullet}{|}})\mathbf{f}'\mathbf{f}(y) + h^2\kappa({}^{\bullet}\!\vee\!{}^{\bullet}_{\!\bullet})\mathbf{f}''(\mathbf{f},\mathbf{f})(y)\cdots. \end{aligned}$$

where $\kappa(t) : T \to \mathbb{R}^m$ are the elementary weight functions defined earlier. Such a series is termed Butcher series by Hairer and Wanner [31] in honour of John Butcher. The above series is a scaled version of Butcher.

The identity mapping corresponding to the elementary weight function $\mathbf{1}(t)$ and the inverse mapping corresponding to the elementary weight function $\kappa^{-1}(t)$ is given as

- $B(\mathbf{1}(t), y_{n-1}) = y_{n-1}$.
- $y_n = B(\kappa(t), y_{n-1}), \iff y_{n-1} = B(\kappa^{-1}(t), y_n)$.

The sum and the composition of two B-series $B(\kappa(t), y)$ and $B(\mu(t), y)$ is given as:

- $B(\kappa(t), y) + B(\mu(t), y) = B((\kappa + \mu)(t), y)$.
- $B(\kappa(t), B(\mu(t), y)) = B(\kappa\mu(t), y)$.

Here $\kappa(\phi) = 1$ and $(\kappa\mu)(t) : T \to \mathbb{R}^m$ is the product of elementary weight functions κ and μ given by the mapping

$$(\kappa\mu)(t) = \mu(\phi)\kappa(t) + \mu(t) + \sum_{u \prec t} \mu(u)\kappa(t \backslash u). \tag{2.9}$$

where u is a subtree of t and $t\backslash u$ is the remaining collection of trees when u is removed from t. Of particular interest is the composition where the second operator is the differentiation operator (2.6) and (2.7), in which case the composition rule (2.9) becomes

$$(\kappa D_i)(t) = \begin{cases} 0, & \text{if} \quad r(t) < i, \\ \frac{i!}{\gamma(t)}, & \text{if} \quad r(t) = i, \\ \displaystyle\sum_{u \prec t, r(u) = i} \frac{i!}{\gamma(u)} \kappa(t \backslash u), & \text{if} \quad r(t) > i. \end{cases}$$

$$(\kappa D)(t) = \begin{cases} 0, & \text{if} \quad t = \phi, \\ 1, & \text{if} \quad t = \tau, \\ \kappa(t_1)\cdots\kappa(t_m), & \text{if} \quad t = [t_1 \cdots t_m]. \end{cases} \tag{2.10}$$

The product $(\kappa\mu)(t)$ refers to the composition of elementary weights of two generalised Runge–Kutta methods $[a, b^T, c]$ and $[A, B^T, C]$ with elementary weights $\kappa(t)$ and $\mu(t)$ respectively.

Let us take a closer look at the composition of two Runge–Kutta methods. Consider 2-stage Runge–Kutta methods having Butcher tableau

$$\begin{array}{c|cc} c_1 & a_{11} & a_{12} \\ c_2 & a_{21} & a_{22} \\ \hline & b_1 & b_2 \end{array}, \qquad\qquad \begin{array}{c|cc} C_1 & A_{11} & A_{12} \\ C_2 & A_{21} & A_{22} \\ \hline & B_1 & B_2 \end{array}.$$

The equations are

$$\begin{aligned} Y_1 &= y_0 + a_{11}hf(Y_1) + a_{12}hf(Y_2), & \check{Y}_1 &= y_1 + A_{11}hf(\check{Y}_1) + A_{12}hf(\check{Y}_2), \\ Y_2 &= y_0 + a_{21}hf(Y_1) + a_{22}hf(Y_2), & \check{Y}_2 &= y_1 + A_{21}hf(\check{Y}_1) + A_{22}hf(\check{Y}_2), \\ y_1 &= y_0 + b_1hf(Y_1) + b_2hf(Y_2), & y_2 &= y_1 + B_1hf(\check{Y}_1) + B_2hf(\check{Y}_2). \end{aligned}$$

The composed method has the form

$$\begin{aligned} Y_1 &= y_0 + a_{11}hf(Y_1) + a_{12}hf(Y_2), \\ Y_2 &= y_0 + a_{21}hf(Y_1) + a_{22}hf(Y_2), \\ \check{Y}_1 &= y_0 + b_1hf(Y_1) + b_2hf(Y_2) + A_{11}hf(\check{Y}_1) + A_{12}hf(\check{Y}_2), \\ \check{Y}_2 &= y_0 + b_1hf(Y_1) + b_2hf(Y_2) + A_{21}hf(\check{Y}_1) + A_{22}hf(\check{Y}_2), \\ y_2 &= y_0 + b_1hf(Y_1) + b_2hf(Y_2) + B_1hf(\check{Y}_1) + B_2hf(\check{Y}_2). \end{aligned}$$

The composition of these Runge–Kutta methods is given by the Butcher tableau as

$$\begin{array}{c|cccc} c_1 & a_{11} & a_{12} & 0 & 0 \\ c_2 & a_{21} & a_{22} & 0 & 0 \\ C_1 + \sum\limits_i b_i & b_1 & b_2 & A_{11} & A_{12} \\ C_2 + \sum\limits_i b_i & b_1 & b_2 & A_{21} & A_{22} \\ \hline & b_1 & b_2 & B_1 & B_2 \end{array}. \tag{2.11}$$

The first order condition corresponding to $t = \bullet$ for the composed method is

$$\begin{aligned}(\kappa\mu)(t) &= b_1 + b_2 + B_1 + B_2,\\ &= \sum_i b_i + \sum_i B_i,\\ &= \kappa(\bullet) + \mu(\bullet).\end{aligned}$$

As the order of the trees increases, so does the complexity. Thus a recursive formula is required for the product of elementary weights. We need following information.

- Consider the Butcher group G and its subgroup G_1 defined as

  $$\begin{aligned}G &= \{\kappa \mid \kappa : T \to \mathbb{R},\ \ \kappa \text{ is a linear functional}\},\\ G_1 &= \{\kappa \mid \kappa \in G,\ \ \kappa(\phi) = 1\}.\end{aligned}$$

  The composition of the two B-series (2.9) is represented by $G_1 \times G \to G$. The product of rooted trees $t, u \in T$ is given by $T \times T \to T$, where tu is formed by joining the root of t and u with the root of t as the root of the product.

- Consider a dual group $\widehat{G}$ defined as

  $$\hat{G} = \{\hat{t} \mid \hat{t} : G \to \mathbb{R},\ \ \hat{t}(\kappa) = \kappa(t),\ \ \kappa \in G,\ \ t \in T\},$$

  where the set of all dual rooted trees $\hat{t}$ is denoted by $\widehat{T}$. The product of dual rooted trees $\hat{t}, \hat{u} \in \widehat{T}$ is given by $\widehat{T} \times \widehat{T} \to \widehat{T}$ such that

  $$\hat{t}.\hat{u} = \widehat{tu}.$$

- Consider a function $\lambda : G_1 \times T \to \widehat{G}$ given by

  $$\lambda(\kappa, t) = \begin{cases}\widehat{\tau}, & \text{if} \quad t = \tau,\\ \lambda(\kappa, t_1)\lambda(\kappa, t_2) + \kappa(t_2)\lambda(\kappa, t_1), & \text{if} \quad t = t_1 t_2.\end{cases}$$

Theorem 2.1.2. *Let $\kappa \in G_1$ and $\mu \in G$ then*

$$\begin{aligned}(\kappa\mu)(\phi) &= \mu(\phi)\\ (\kappa\mu(t)) &= \lambda(\kappa, t)(\mu) + \kappa(t)\mu(\phi)\end{aligned}$$

The proof is given in [12].

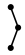
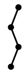
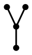
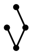
t		$\kappa\mu(t)$
ϕ	t_0	$\mu(t_0)$
•	t_1	$\kappa(t_1)\mu(t_0)+\mu(t_1)$
	t_2	$\kappa(t_2)\mu(t_0)+\kappa(t_1)\mu(t_1)+\mu(t_2)$
	t_3	$\kappa(t_3)\mu(t_0)+\kappa(t_2)\mu(t_1)+\kappa(t_1)\mu(t_2)+\mu(t_3)$
	t_4	$\kappa(t_4)\mu(t_0)+\kappa(t_1)^2\mu(t_1)+2\kappa(t_1)\mu(t_2)+\mu(t_4)$
	t_5	$\kappa(t_5)\mu(t_0)+\kappa(t_3)\mu(t_1)+\kappa(t_2)\mu(t_2)+\kappa(t_1)\mu(t_3)+\mu(t_5)$
	t_6	$\kappa(t_6)\mu(t_0)+\kappa(t_4)\mu(t_1)+\kappa(t_1)^2\mu(t_2)+2\kappa(t_1)\mu(t_3)+\mu(t_6)$
	t_7	$\kappa(t_7)\mu(t_0)+\kappa(t_1)\kappa(t_2)\mu(t_1)+(\kappa(t_1)^2+\kappa(t_2))\mu(t_2)+\kappa(t_1)(\mu(t_3)+\mu(t_4)+\mu(t_7))$
	t_8	$\kappa(t_8)\mu(t_0)+\kappa(t_1)^3\mu(t_1)+3\kappa(t_1)^2\mu(t_2)+3\kappa(t_1)\mu(t_4)+\mu(t_8)$

Table 2.2: Product of elementary weight functions of trees up to order 4.

For the tree with a single vertex, $t = \centerdot$, represented by τ, we have

$$\begin{aligned}
\lambda(\kappa,\tau) &= \widehat{\tau}. \\
(\kappa\mu)(\tau) &= \lambda(\kappa,\tau)\mu+\kappa(\tau)\mu(\phi), \\
&= \hat{\tau}\mu+\kappa(\tau)\mu(\phi), \\
&= \mu(\tau)+\kappa(\tau)\mu(\phi).
\end{aligned}$$

The product $(\kappa\mu)(t)$ for all trees of order up to four is given in Table 2.2.

2.1.8 Effective order

The idea of effective order was first introduced by Butcher in [9] and was later revisited in [10] in an attempt to increase the accuracy of a Runge–Kutta method. This was successfully achieved for Singly Implicit Runge–Kutta methods by Butcher and Chartier [13]. The idea was later used for Diagonally Extended Singly Implicit Runge–Kutta methods by Butcher and Chan [14] and Butcher and Diamantakis [15]. The accuracy of symplectic integrators for Hamiltonian systems was enhanced using effective order by Sanz-Serna et al [38].

A Runge–Kutta method u has an effective order p if there exists another method v such that vuv^{-1} has order p. Sanz-Serna used the term pre-processing for the application of v and post-processing for the application of v^{-1} in [38]. Furthermore

$$(vuv^{-1})^n = vu^nv^{-1}.$$

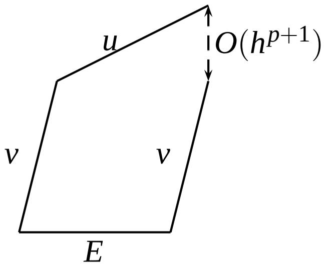


Figure 2.1: Effective order

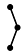
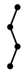
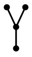
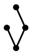
i	t_i	$(\kappa\mu)(t_i)$	$(E\kappa)(t_i)$
1		$\kappa_1+\mu_1$	κ_1+1
2		$\kappa_2+\kappa_1\mu_1+\mu_2$	$\kappa_2+\kappa_1+\frac{1}{2}$
3		$\kappa_3+\kappa_2\mu_1+\kappa_1\mu_2+\mu_3$	$\kappa_3+\kappa_2+\frac{1}{2}\kappa_1+\frac{1}{6}$
4		$\kappa_4+\kappa_1^2\mu_1+2\kappa_1\mu_2+\mu_4$	$\kappa_4+\kappa_1^2+\kappa_1+\frac{1}{3}$
5		$\kappa_5+\kappa_3\mu_1+\kappa_2\mu_2+\kappa_1\mu_3+\mu_5$	$\kappa_5+\kappa_3+\frac{1}{2}\kappa_2+\frac{1}{6}\kappa_1+\frac{1}{24}$
6		$\kappa_6+\kappa_4\mu_1+\kappa_1^2\mu_2+2\kappa_1\mu_3+\mu_6$	$\kappa_6+\kappa_4+\frac{1}{2}\kappa_1^2+\frac{1}{3}\kappa_1+\frac{1}{12}$
7		$\kappa_7+\kappa_1\kappa_2\mu_1+(\kappa_1^2+\kappa_2)\mu_2+\kappa_1(\mu_3+\mu_4)+\mu_7$	$\kappa_7+\kappa_1\kappa_2+\frac{1}{2}(\kappa_1^2+\kappa_2)+\frac{1}{2}\kappa_1+\frac{1}{8}$
8		$\kappa_8+\kappa_1^3\mu_1+3\kappa_1^2\mu_2+3\kappa_1\mu_4+\mu_8$	$\kappa_8+\kappa_1^3+\frac{3}{2}\kappa_1^2+\kappa_1+\frac{1}{4}$

Table 2.3: Expressions for $\kappa\mu$ and $E\kappa$ for trees of order up to 4.

Therefore, the method v is used once only in the beginning and similarly the method v^{-1} is also used once only at the end. Butcher has used an equivalent approach in [10] to get the effective order conditions. The method v is applied followed by the method u. Besides using the composition vu, another composition is also evaluated by first moving along the exact flow E followed by the method v as shown in Figure 2.1. The exact flow E represents the mapping from trees to real numbers as $E=1/\gamma$, where γ is the density of the corresponding tree. The method u is said to have an effective order p, if the elementary weights κ of v and μ of u satisfy the following relation for all trees up to order p.

$$(\kappa\mu)(t)=(E\kappa)(t).$$

Table 2.3 shows the product $(\kappa\mu)(t)$ and $(E\kappa)(t)$ for trees of order up to four with $\kappa(t_i)=\kappa_i$. Here we have assumed that the empty tree ϕ is always mapped to 1.

Comparing the last two columns of Table 2.3 results in effective order conditions given as

$$\mu_1=1,\quad \mu_2=\tfrac{1}{2},$$
$$\mu_3=\tfrac{1}{6},\quad \mu_5=\tfrac{1}{24},$$
$$\mu_4-\mu_8+2\mu_7-\mu_6=\tfrac{1}{4}.$$

We have only five effective order conditions for a fourth order Runge–Kutta method resulting in more choice for the co-efficients of Runge–Kutta method.

Note that the idea of effective order provides a working ground for analysing the order of a general linear method. A starting method is first employed to start the process of a general linear method followed by the implementation of an actual general linear method. This process will be explained in section 2.3.4.

2.2 Linear multistep methods

Linear multistep methods are multivalue numerical methods for the solution of initial value problems

$$y'(x) = f(y(x)), \quad y(0) = y_0.$$

The first multistep methods were introduced by Adams and Bashforth in 1883 [2] and were later given the name Adams-Bashforth methods. The modern theory of linear multistep methods is due to Dahlquist [21]. Other types of multistep methods were developed by Nyström [43] and Milne [39]. The general form of a k-step linear multistep method is

$$y_n = \sum_{i=1}^{k} \alpha_i y_{n-i} + h \sum_{i=0}^{k} \beta_i f(y_{n-i}). \tag{2.12}$$

If we take $\alpha_1 = 1$, all other $\alpha_i = 0$ and $\beta_0 = 0$ we obtain

$$y_n = y_{n-1} + h(\beta_1 f(y_{n-1}) + \beta_2 f(y_{n-2}) + \cdots + \beta_k f(y_{n-k})). \tag{2.13}$$

If instead $\beta_0 \neq 0$, we obtain methods of the form

$$y_n = y_{n-1} + h(\beta_0 f(y_n) + \beta_1 f(y_{n-1}) + \beta_2 f(y_{n-2}) + \cdots + \beta_k f(y_{n-k})), \tag{2.14}$$

By selecting the β_i suitably in (2.13) we obtain order k. The method is then known as Adams-Bashforth method and they are explicit methods. Similarly in (2.14), if the β_i are chosen to attain order $k+1$ we obtain implicit Adams-Moulton methods introduced by Moulton in 1926 [41].

The coefficients of Adams method can easily be found by expanding the terms of the method using truncated Taylor series. Thus for a two step Adams-Bashforth method

$$y_n - y_{n-1} = \beta_1 h y'_{n-1} + \beta_2 h y'_{n-2},$$
$$y_n - [y_n - hy'_n + \tfrac{h^2}{2!} y''_n - \tfrac{h^3}{3!} y'''_n] = h\beta_1 [y'_n - hy''_n + \tfrac{h^2}{2!} y'''_n] + h\beta_2 [y'_n - 2hy''_n].$$

Comparing the terms we get

$$\begin{aligned}\beta_1+\beta_2 &= 1,\\ \beta_1+2\beta_2 &= \tfrac{1}{2}.\end{aligned}$$

Solving these equations result in $\beta_1 = \frac{3}{2}$ and $\beta_2 = -\frac{1}{2}$ and the corresponding Adams-Bashforth method is

$$y_n - y_{n-1} = \tfrac{3}{2}hy'_{n-1} - \tfrac{1}{2}hy'_{n-2}.$$

The coefficients of a linear multistep method can be characterised by the polynomials

$$\begin{aligned}\rho(w) &= w^k - \alpha_1 w^{k-1} - \ldots - \alpha_k, \qquad (2.15)\\ \sigma(w) &= \beta_0 w^k + \beta_1 w^{k-1} + \cdots + \beta_k.\end{aligned}$$

For the solution of stiff differential equations by linear multistep methods, Curtiss and Hirschfelder introduced backward difference formula, aka. BDF, based on numerical differentiation [19]. The characteristic polynomials for BDF methods are

$$\begin{aligned}\rho(w) &= \beta\sum_{i=1}^{k}\frac{1}{i}w^{k-i}(w-1)^i,\\ \sigma(w) &= \beta w^k.\end{aligned}$$

where

$$\beta = \Big(\sum_{i=1}^{k}\frac{1}{i}\Big)^{-1}.$$

Thus for $s = 1$, we get

$$\begin{aligned}\beta &= 1,\\ \rho(w) &= w-1,\\ \sigma(w) &= w.\end{aligned}$$

and we get implicit Euler method

$$y_n - y_{n-1} = hf_n.$$

Similarly for $s = 2$, we get

$$\begin{aligned}\beta &= \frac{1}{1+\frac{1}{2}} = \tfrac{2}{3},\\ \rho(w) &= w^2 - \tfrac{4}{3}w + \tfrac{1}{3},\\ \sigma(w) &= \tfrac{2}{3}w^2.\end{aligned}$$

and we get the following method

$$y_n - \tfrac{4}{3}y_{n-1} + \tfrac{1}{3}y_{n-2} = \tfrac{2}{3}hf_n.$$

BDF methods are considered superior among multistep methods because of their stability properties. However, they are only convergent for $1 \leq s \leq 6$, yet this range of order of BDF methods is sufficient for many practical problems.

Definition 2.2.1. *A linear multistep method* (2.12) *is symmetric if*

$$\alpha_i = -\alpha_{k-i} \qquad \beta_i = \beta_{k-i} \qquad \forall \;\; k$$

2.2.1 Consistency, stability, convergence and order

Linear multistep methods are consistent i.e. producing exact results for a simple differential equation $y' = 1$ if

$$\rho(1) = 0, \qquad \rho'(1) = \sigma(1).$$

Linear multistep methods are stable if they yield bounded results for $y' = 0$. If applied to such a differential equation, the method becomes

$$y_n = \sum_{i=1}^{k} \alpha_i y_{n-i} = \alpha_1 y_{n-1} + \alpha_2 y_{n-2} + \cdots + \alpha_k y_{n-k}.$$

This is a difference equation and bounded solutions are obtained if all zeros of the polynomial $\rho(w)$ lie inside the unit disc and those on the boundary are simple. A stable and consistent linear multistep method is convergent.

A linear multistep method has an order p, if the characteristic polynomials $\rho(w)$ and $\sigma(w)$ satisfy

$$\rho(e^z) - z\sigma(e^z) = O(z^{p+1}).$$

The maximum order of a $k-$step linear multistep method can be $2k$. However for a stable $k-$step method, the order p has the following bounds

$$p = \begin{cases} k+1, & \text{if } k \text{ is odd,} \\ k+2, & \text{if } k \text{ is even.} \end{cases}$$

This is known as the Dahlquist first barrier. It was later proved by Dahlquist that the order of an A-stable linear multistep method can at most be 2. This is known as the Dahlquist second barrier. The concept of A-stability is related to solving stiff differential equations and this will be dealt with in detail in Chapter 3.

2.3 General linear methods

General linear methods are a generalisation of Runge–Kutta methods and multistep methods introduced by Butcher [8] and are used to find numerical solution of initial value problems

$$y'(x) = f(y(x)), \qquad y(0) = y_0.$$

The general form of a general linear method is

$$\begin{aligned} Y &= h(A \otimes I) f(Y) + (U \otimes I) y^{[n-1]}, \\ y^{[n]} &= h(B \otimes I) f(Y) + (V \otimes I) y^{[n-1]}. \end{aligned}$$

$A \otimes I$ represents the Kronecker product of the matrix A and the identity matrix I and h is the stepsize. The $s-$component vector $Y_i \approx y(x_n + c_i h)$ represents the stages and is an approximation at the $i-$th stage evaluated at abscissa c_i. $f(Y)$ is a vector of the stage derivatives. The vector $y^{[n-1]}$ has $r-$components which are provided as input values at the beginning of a step. The application of one step of a general linear method results in an output approximation $y^{[n]}$.

$$Y = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_s \end{bmatrix}, \quad f(Y) = \begin{bmatrix} f(Y_1) \\ f(Y_2) \\ \vdots \\ f(Y_s) \end{bmatrix}, \quad y^{[n-1]} = \begin{bmatrix} y_1^{[n-1]} \\ y_2^{[n-1]} \\ \vdots \\ y_r^{[n-1]} \end{bmatrix}, \quad y^{[n]} = \begin{bmatrix} y_1^{[n]} \\ y_2^{[n]} \\ \vdots \\ y_r^{[n]} \end{bmatrix}.$$

With a slight abuse of notation, general linear methods are written as

$$\begin{aligned} Y &= hAf(Y) + Uy^{[n-1]}, \\ y^{[n]} &= hBf(Y) + Vy^{[n-1]}. \end{aligned} \tag{2.16}$$

The matrices A, U, V and B are representatives of a particular general linear method and are generally given as

$$M = \left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right]. \tag{2.17}$$

The class of general linear methods is large. One-step methods like Runge–Kutta methods and all multistep methods can be written as special cases of general linear methods. Non-traditional methods can also be written in general linear framework such as cyclic composite methods, where several linear multistep methods are used cyclically, the Nordsieck methods, where the information is passed between different steps using the Nordsieck vector $[y_n, hy'_n, \frac{h^2}{2!}y''_n, \ldots, \frac{h^k}{k!}y^k_n]$, the pseudo Runge–Kutta methods where in addition to the stages of the current step, the stages of previous steps is also used and hybrid methods.

A Runge–Kutta method (2.1) has a single input so $r = 1$, $U = \mathbf{1}$, $V = 1$ and the matrix B has only one row. A two-stage Runge-Kutta method written in general linear formulation is

$$\left[\begin{array}{c} Y_1 \\ Y_2 \\ \hline y^{[n]} \end{array}\right] = \left[\begin{array}{cc|c} a_{11} & a_{12} & 1 \\ a_{21} & a_{22} & 1 \\ \hline b_1 & b_2 & 1 \end{array}\right] \left[\begin{array}{c} hf(Y_1) \\ hf(Y_2) \\ \hline y^{[n-1]} \end{array}\right].$$

The linear multistep methods such as Adams-Moulton method (2.14) written in general linear method formulation has $s = 1$ and is given as

$$\left[\begin{array}{c} Y_1 \\ \hline y_n \\ hf(Y_1) \\ hf(y_{n-1}) \\ hf(y_{n-2}) \\ \vdots \\ hf(y_{n-k+1}) \end{array}\right] = \left[\begin{array}{c|cccccc} \beta_0 & 1 & \beta_1 & \beta_2 & \cdots & \beta_{k-1} & \beta_k \\ \hline \beta_0 & 1 & \beta_1 & \beta_2 & \cdots & \beta_{k-1} & \beta_k \\ 1 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & 0 \end{array}\right] \left[\begin{array}{c} hf(Y_1) \\ \hline y_{n-1} \\ hf(y_{n-1}) \\ hf(y_{n-2}) \\ hf(y_{n-3}) \\ \vdots \\ hf(y_{n-k}) \end{array}\right].$$

The linear multistep methods are generally implemented as predictor-corrector pairs with Adams-Bashforth as predictor method and Adams-Moulton as corrector method. This predictor-corrector pair can also be written as a two-stage general linear method [12]. Moreover the BDF methods and one-leg methods can also be written as general linear methods.

2.3.1 Symmetric general linear methods

A general linear method is symmetric if it is equal to its adjoint. The adjoint method is also a general linear method which undoes the work of actual general linear method. Consider the general linear method (2.16)

$$\begin{aligned} Y &= hAf(Y) + Uy^{[n-1]}, \\ y^{[n]} &= hBf(Y) + Vy^{[n-1]}. \end{aligned}$$

Unlike Runge–Kutta methods, it is not straightforward to invert the output approximation because for general linear methods the output approximations involve a matrix V multiplied with the input approximations and V^{-1} might not be equal to V even though, both matrices might have same eigenvalues on the unit circle. Thus we have to introduce an involution L, such that

$$L = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & \cdots & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix},$$

having the properties

$$L^2 = I, \qquad\qquad LV^{-1}L = V.$$

Like Runge–Kutta methods, we introduce a permutation matrix P given in (2.3), so that the stages of the adjoint method is reordered to align with the stages of the actual general linear method. Now considering the fact that the adjoint general linear method takes the step $-h$ we can write its form

$$\begin{aligned} PY &= (-h)(PUBP - PAP)(Pf(Y)) + PUV^{-1}y^{[n]}, \\ Ly^{[n-1]} &= (-h)(LBP)(Pf(Y)) + V(Ly^{[n-1]}), \end{aligned}$$

and time reversal symmetry implies

$$\begin{aligned} B &= LBP, \\ A + PAP &= PUBP, \\ U &= PUV^{-1}. \end{aligned} \tag{2.18}$$

It is important to note that a symmetric method can be constructed by composing a general linear method with its adjoint. We look at the composition of two general linear methods whose representative matrices (2.17) are

$$M_1 = \left[\begin{array}{c|c} A_1 & U_1 \\ \hline B_1 & V_1 \end{array}\right], \qquad\qquad M_2 = \left[\begin{array}{c|c} A_2 & U_2 \\ \hline B_2 & V_2 \end{array}\right].$$

The composition is given as

$$M_1 \circ M_2 = \left[\begin{array}{cc|c} A_1 & 0 & U_1 \\ U_2B_1 & A_2 & U_2V_1 \\ \hline V_2B_1 & B_2 & V_2V_1 \end{array}\right].$$

2.3.2 Pre-consistency, consistency, stability and convergence

The pre-consistency condition of a numerical method is related to its ability to exactly solve the simplest ODE, $y'(x) = 0$ with solution $y(x) = 1$. Thus

$$\begin{aligned} y^{[n-1]} &= uy(x_{n-1}) + O(h), \\ y^{[n]} &= uy(x_n) + O(h). \end{aligned}$$

The vector u is known as the pre-consistency vector and is specific for a specific numerical method. Thus for the Euler method, $y_{n+1} = y_n + hf(y_n)$, the value of u is 1. In the case of a general linear method

$$\begin{aligned} Y &= Uy^{[n-1]} = Uuy(x_{n-1}) + O(h), \\ y^{[n]} &= Uy^{[n-1]} = Vuy(x_{n-1}) + O(h). \end{aligned}$$

Therefore the pre-consistency condition of a general linear method is

$$Uu = \mathbf{1}, \qquad Vu = u.$$

The consistency of a method is determined by its ability to exactly solve the ODE $y'(x) = 1$, with initial condition $y(0) = 0$. Thus

$$\begin{aligned} y^{[n-1]} &= uy(x_{n-1}) + vhy'(x_{n-1}) + O(h^2), \\ y^{[n]} &= uy(x_n) + vhy'(x_n) + O(h^2). \end{aligned}$$

The vector v is known as the consistency vector and is specific for a specific numerical method. Thus for the Euler method, $y_{n+1} = y_n + hf(y_n)$, the value of v is 1. In the case of a general linear method

$$\begin{aligned} Y &= A\mathbf{1}h + Uy^{[n-1]} = A\mathbf{1}h + Uuy(x_{n-1}) + hUvy'(x_{n-1}) + O(h^2), \\ y^{[n]} &= B\mathbf{1}h + Vy^{[n-1]} = B\mathbf{1}h + Vuy(x_{n-1}) + hVvy'(x_{n-1}) + O(h^2). \end{aligned}$$

Therefore the consistency condition of a general linear method is

$$B\mathbf{1} + Vv = u + v.$$

A method is of little or no use if it is not stable. Stability implies that the error introduced by a numerical method in one step does not grow unboundedly in later steps. If a general linear method is solving the ODE $y'(x) = 0$, then

$$y^{[n]} = Vy^{[n-1]} = V^n y^{[0]}.$$

Thus the stability of a general linear method hinges on the matrix V being power bounded.

$$\|V^n\|_\infty \leq C, \qquad \forall \;\; n = 1, 2, \cdots.$$

A general linear method is strictly stable, if all eigenvalues of V are inside the unit disc except one, which is on the boundary.

Butcher [8] generalised the idea of Dahlquist [20] for the case of general linear methods, that the stability and consistency of a general linear method is necessary and sufficient for its convergence. A general linear method is convergent, if there exist a non-zero vector $u \in \mathbb{R}^n$ such that if the starting approximation $y^{[0]}$ converges to $uy(x_0)$, then the final approximation $y^{[n]}$ converges to $uy(x_0 + nh)$ for all n.

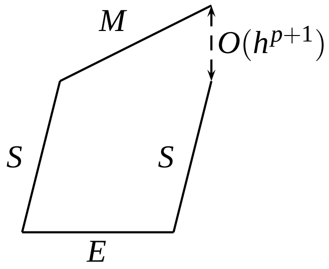


Figure 2.2: Order of accuracy

2.3.3 Order of accuracy

A general linear method requires r approximations $y_i^{[0]}$, $i = 1, 2, \cdots, r$ as input values. However only a single initial condition $y(x_0) = y_0$ is provided with the initial value problem. Thus a starting method is often required to obtain approximations to the initial vector $y^{[0]}$. Consider an $(\bar{s} + r) \times (\bar{s} + \mathbf{1})$ starting method S given as

$$S = \left[\begin{array}{c|c} S_{11} & S_{12} \\ \hline S_{21} & S_{22} \end{array}\right].$$

where r is the number of approximations required for the actual general linear method to start and $\bar{s}$ is the number of stages of the starting method. The pre-consistency conditions for the starting method with u as the pre-consistency vector are

$$S_{22} = u, \qquad S_{12} = \mathbf{1}.$$

Once the starting method S is applied and input vector $y^{[0]}$ is available, the actual general linear method M is then applied resulting in the combined effect as MoS. The exact solution operator E is applied to move the solution from x_{n-1} to x_n. The order of accuracy of the general linear method M is p relative to the starting method S if

$$M \circ S - S \circ E = O(h^{p+1}).$$

This is shown in Figure 2.2.

For a general linear method to be of order p, ideally all components of $y^{[n]}$ should be of order p, however in practice, we content our self with the first component which is approximating the actual solution to be of order p at least. A finishing procedure is often required to undo the effect of starting method S. This is achieved by picking the first component of the output approximation $y^{[n]}$ which approximates the exact solution.

2.3.4 Algebraic analysis of order

The B-series for the starting method is

$$y^{[0]} = B(S(t), y(x_{n-1})).$$

Let $\eta(t)$ be the elementary weight function for the stages Y, then the B-series for the stages is

$$Y = B(\eta(t), y(x_{n-1})).$$

The B-series for the stage derivatives $hf(Y)$ is

$$\begin{aligned} hf(Y) &= B(D(t), Y), \\ &= B(D(t), B(\eta(t), y(x_{n-1}))), \\ &= B(\eta D(t), y(x_{n-1})). \end{aligned}$$

where D is the differentiation operator (2.7) and ηD represents the composition (2.10). General linear method (2.16) can be represented in terms of its B-series. Consider the B-series representation of the internal stages

$$\begin{aligned} B(\eta(t), y(x_{n-1})) &= AB(\eta D(t), y(x_{n-1})) + UB(S(t), y(x_{n-1})), \\ &= B(A\eta D(t) + US(t), y(x_{n-1})). \end{aligned} \tag{2.19}$$

Let $\xi(t)$ be the elementary weight function of for the output approximation $y^{[n]}$ then

$$\begin{aligned} B(\xi(t), y(x_{n-1})) &= BB(\eta D(t), y(x_{n-1})) + VB(S(t), y(x_{n-1})), \\ &= B(B\eta D(t) + VS(t), y(x_{n-1})). \end{aligned} \tag{2.20}$$

The generating functions for the general linear method are obtained from equations (2.19) and (2.20).

$$\eta(t) = A\eta D(t) + US(t), \tag{2.21}$$

$$\xi(t) = B\eta D(t) + VS(t). \tag{2.22}$$

The generating function for output approximations (2.22) is equivalent to the application of the exact solution operator E to the starting method S.

$$ES(t) = B\eta D(t) + VS(t). \tag{2.23}$$

The general linear method is said to be of order p, if at least the first component of (2.23) is equal to $E(t)$, for all tree of order less than or equal to p.

Chapter 3

Stability and symplecticity of numerical methods

Stability of a numerical method plays an important role in the numerical solution of ordinary differential equations. The application of a numerical method yields results having errors, because they are approximations to the exact solution. Stability of a numerical method is concerned with the ability of a numerical method to monitor and control the growth of these errors over an unbounded period of time. A numerical method is said to be stable, if the error introduced by one-step of a numerical method remains bounded throughout the integration process. The concept of stability was introduced in relation to numerical solution of stiff ordinary differential equations. Stiffness is a qualitative property possessed by most of the ordinary differential equation systems modelling the real world phenomena. A stable numerical method is the only choice for the numerical solution of stiff ordinary differential equation system. Stability also plays an important role in the selection of numerical methods for the solution of non-stiff ordinary differential equation system.

Hamiltonian systems are conservative rather than stiff. They conserve energy like properties of the dynamics of mechanical system. Qualitatively accurate numerical solutions of the Hamiltonian systems are best obtained by symplectic numerical methods. Symplecticness is a qualitative property of Hamiltonian systems in terms of their solution and is explained in Chapter 1. The criteria of a numerical method to be symplectic hinges on certain equations, which the coefficients of the numerical methods should satisfy. There is an intricate relation between the criteria of a numerical method to be symplectic and to be stable. This relation is studied in the context of non-linear stability analysis of the numerical solution of non-stiff ordinary differential equations.

3.1 Stability of Runge–Kutta methods: linear case

The linear stability analysis of a Runge–Kutta method makes use of the Dahlquist linear test equation taken from [22]

$$y'(x) = \lambda y(x), \tag{3.1}$$

where λ can be a complex number. An application of an explicit Runge–Kutta method to solve (3.1) yields

$$\begin{aligned} Y_i &= y_{n-1} + \sum_{j=1}^{i-1} a_{ij} h\lambda Y_j, \quad i = 1, 2, \cdots, s, \\ y_n &= y_{n-1} + \sum_{i=1}^{s} b_i h\lambda Y_i. \end{aligned}$$

Calculate the explicit stages Y_i sequentially and then evaluate y_n, we get

$$y_n = R(z) y_{n-1},$$

where $z = h\lambda$ and

$$R(z) = 1 + z\sum_i b_i + z^2 \sum_i b_i a_{ij} + z^3 \sum_i b_i a_{ij} a_{jk} + \cdots.$$

If the explicit Runge–Kutta method is of order p, then the order conditions for trees up to order p should be satisfied. We recall from Chapter 2 that the first few order conditions are

$$\sum_i b_i = 1, \qquad \sum_i b_i a_{ij} = \tfrac{1}{2}, \qquad \sum_i b_i a_{ij} a_{jk} = \tfrac{1}{6}.$$

Therefore an explicit method with order p has the stability function

$$\begin{aligned} R(z) &= 1 + z + \frac{z^2}{2!} + \frac{z^3}{3!} + \cdots + \frac{z^p}{p!} + O(z^{p+1}), \\ &= \exp(z) + O(z^{p+1}). \end{aligned}$$

For explicit Runge–Kutta methods with s stages and order p, if $s = p$ then the stability function is

$$R(z) = 1 + z + \frac{z^2}{2!} + \frac{z^3}{3!} + \cdots + \frac{z^p}{p!}.$$

An $s-$stage implicit Runge–Kutta method applied to solve (3.1) yields

$$\begin{aligned} Y_i &= y_{n-1} + \sum_{j=1}^{s} a_{ij} h\lambda Y_j, \quad i = 1, 2, \cdots, s, \\ y_n &= y_{n-1} + \sum_{i=1}^{s} b_i h\lambda Y_i. \end{aligned}$$

Here the stages Y_i are implicit and represent a system of linear equations. On solving this system and using the values of the stages Y_i in the output value y_n, we have

$$y_n = R(z) y_{n-1}, \tag{3.2}$$

where $z = h\lambda$ and

$$\begin{aligned} R(z) &= 1 + z b^T (I - zA)^{-1} \mathbf{1}, \\ &= \frac{\det(I - zA + z\mathbf{1}b^T)}{\det(I - zA)}. \end{aligned}$$

The stability function of an implicit Runge–Kutta method of order p is a rational function of z and can be written as

$$R(z) = \frac{N(z)}{D(z)},$$

where both $N(z)$ and $D(z)$ are polynomial of degree at most p and $D(0) = 1$.

3.1.1 Padé approximation to the exponential

Consider a polynomial $N_{mn}(z)$of degree $m \geq 0$ and a polynomial $D_{mn}(z)$ of degree $n \geq 0$ and consider a rational function

$$R_{mn}(z) = \frac{N_{mn}(z)}{D_{mn}(z)}.$$

Such a rational function can approximate a function $f(z)$, which is analytic at zero with $f(0) \neq 0$, such that

$$f(z) = \frac{N_{mn}(z)}{D_{mn}(z)} + O(z^{m+n+1}).$$

The rational function $R_{mn}(z)$ is called the (m,n) Padé approximation to the function $f(z)$ and this rational function provides the highest order of approximation to the function $f(z)$ where

$$\begin{aligned} N_{mn}(z) &= \frac{m!}{(m+n)!} \sum_{k=0}^{m} \frac{(m+n-k)!}{k!(m-k)!} z^k, \\ D_{mn}(z) &= \frac{n!}{(m+n)!} \sum_{k=0}^{n} \frac{(m+n-k)!}{k!(n-k)!} (-z)^k. \end{aligned}$$

If $R_{mn}(z)$ approximates the function e^z, we get the Padé approximation to e^z. An interesting feature is that the Padé approximation to e^z are equal to the stability functions of some of the implicit Runge–Kutta methods and provides the stability order

$$e^z D_{mn}(z) = N_{mn}(z) + O(z^{m+n+1}).$$

Let us take a closer look at the simple test equation (3.1) to understand the role of stability function in the selection of a numerical method and the stepsize it can take. The exact solution of (3.1) is

$$y(x) = e^{\lambda x} y_0,$$

it is evident that

$$y(x) \longrightarrow 0 \quad \text{as} \quad x \longrightarrow \infty, \qquad \text{provided} \qquad \mathrm{Re}\lambda < 0.$$

The numerical method should mimic this behaviour and this is possible if

$$y_n \longrightarrow 0 \qquad \text{as} \qquad n \longrightarrow \infty.$$

From equation (3.2)

$$y_n = R(z)^n y_0,$$

and hence

$$y_n \longrightarrow 0 \qquad \text{iff} \qquad |R(z)| < 1.$$

The stability domain is the set

$$D = \{z \in \mathbb{C}; |R(z)| < 1\},$$

We take an example of the explicit Euler method to solve the Dahlquist test equation (3.1). The output is

$$y_n = (1+z)^n y_0.$$

The stability function is

$$R(z) = (1+z)^n,$$

and the bounded solutions require

$$|(1+z)^n| < 1.$$

Since $z = h\lambda$, the stepsize should therefore be chosen to satisfy this equation and hence the selection of stepsize depends on λ, which is a stiffness indicator. The stability function of an implicit Euler method is

$$R(z) = \frac{1}{(1-z)^n},$$

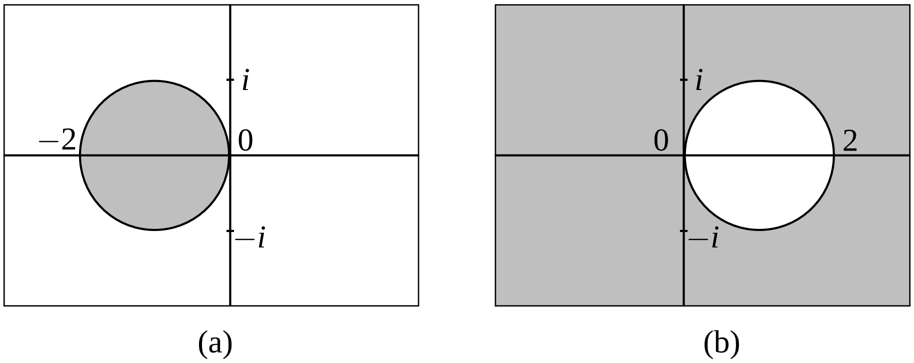


Figure 3.1: The stability regions (shaded) of (a) the explicit Euler method and (b) the implicit Euler method.

The stability regions of the explicit and the implicit Euler methods are plotted in the Figure 3.1. The stability region of the explicit Euler method is a bounded shaded region and this is the case for all explicit Runge–Kutta methods that their stability regions are bounded. The stability region of the implicit Euler method is the unbounded shaded region and this is true for all implicit Runge–Kutta methods that their stability regions are unbounded. The boundedness of the stability region imposes severe restrictions on the stepsize. Dahlquist [22] studied this phenomenon of dependence of stepsize to the stability domain and introduced the concept of A-stability.

Definition 3.1.1. *A method is said to be A-stable if the entire left half of the complex plane* $\mathbb{C}$ *is included in the stability domain D of the numerical method.*

$$\mathbb{C}^- \subseteq D, \quad \textit{where} \quad \mathbb{C}^- = \{z \in \mathbb{C} : Re(z) < 0\}.$$

For a Runge–Kutta method to be A-stable, its stability function must satisfy

$$|R(z)| \leq 1, \qquad \forall \quad Re(z) \leq 0. \tag{3.3}$$

From the Figure 3.1, it is clear that the explicit Euler method is not A-stable because its stability region is only a bounded unit disc centered at $(-1,0)$, while the implicit Euler method is A-stable because its stability region is unbounded and contains the whole left half of the complex plane. In general all practical explicit Runge-Kutta methods are not A-stable while many families of implicit Runge-Kutta methods exist which are A-stable such as Gauss methods. Hence the implicit Runge–Kutta methods are a preferred choice for the solution of stiff differential equations.

Theorem 3.1.2. *A Runge–Kutta method with stability function $R(z) = \frac{N(z)}{D(z)}$ is A-stable, if and only if, all poles of R are in right half plane i.e. they have positive real parts and $|R(iy)| \leq 1$ for $y \in \mathbb{R}$.*

The proof is given in [33].

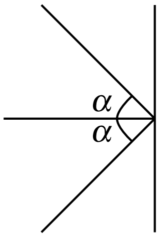


Figure 3.2: $A(\alpha)$ stability region

Definition 3.1.3. *A method is L-stable if in addition to* (3.3)*, it satisfies*

$$R(\infty) = 0.$$

A numerical method for the solution of stiff ODEs should have the property of L-stability as it ensures stable numerical results for z close to real axis with large negative real parts, otherwise $R(-z) \approx 1$ for z very large and the stiff components of the ODEs do not die fast. For some ODEs which have real eigenvalues, like ODEs originating from spatially discretised diffusion equation, we do not require all the negative complex plane to be included in the stability region of the numerical method and the strict A-stability condition can be relaxed. This gives rise to the concept of $A(\alpha)$ stability and we can use methods which are not A-stable.

Definition 3.1.4. *A method is $A(\alpha)$ stable, if a portion of the left half plane*

$$S_\alpha = \{z;\ |\arg(-z)| \leq \alpha,\ z \neq 0\}$$

is included in the stability domain instead of the whole left half of the complex plane as shown in the Figure (3.2)*.*

The linear test equation (3.1) due to Dahlquist assumes λ to be time independent. AN-stability is a generalization of A-stability proposed by Burrage and Butcher [4] in which the test equation to study the stability of numerical method is non-autonomous. Consider the ordinary differential equation

$$y'(x) = \lambda(x)y(x), \tag{3.4}$$

such that for AN-stability

$$Re(\lambda(x)) \leq 0.$$

Apply an implicit Runge–Kutta method to solve (3.4)

$$Y = y_{n-1} + AZY,$$

where

$$Z = \text{diag}[h\lambda(x_{n-1}+hc_1), h\lambda(x_{n-1}+hc_2), \cdots, h\lambda(x_{n-1}+hc_s)].$$

Thus

$$Y = (I - AZ)^{-1}y_{n-1}.$$

The output is

$$\begin{aligned} y_n &= y_{n-1} + b^T ZY \\ &= y_{n_1} + b^T Z(I-AZ)^{-1}y_{n-1} \\ &= R(Z)y_{n-1}. \end{aligned}$$

The function $R(Z)$ is the stability function of the underlying Runge–Kutta method and is given as

$$R(Z) = 1 + b^T Z(I - AZ)^{-1}\mathbf{1}. \tag{3.5}$$

A Runge–Kutta method is AN-stable if

$$|R(Z)| \leq 1, \qquad \forall \quad Re(Z) \leq 0. \tag{3.6}$$

The consequences of AN-stability is summarised in the following theorem by Burrage and Butcher [4] and Crouzeix [18].

Theorem 3.1.5. *An implicit Runge–Kutta method* $[A, b^T, c]$ *is AN-stable if* $b_j \geq 0,\ j = 1, 2, \cdots, s$, *and the matrix*

$$M = \text{diag}(b)A + A^T\,\text{diag}(b) - bb^T,$$

is positive semi-definite.

Proof: We reproduce the proof from [4]. Let $b_i < 0$ for some i and consider the stability function R(Z) in (3.5). Further assume that Z has purely negative real values

$$Z = \begin{bmatrix} -t & 0 & \cdots & 0 \\ 0 & -t & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & -t \end{bmatrix}.$$

Then

$$\begin{aligned} AZ &= A\begin{bmatrix} -t & 0 & \cdots & 0 \\ 0 & -t & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & -t \end{bmatrix} \\ &= -tA. \end{aligned}$$

$$\begin{aligned} Z(I-AZ)^{-1} &= -tI(I+tA)^{-1} \\ &= -tI(I-tA+O(t^2)) \\ &= -tI+O(t^2). \end{aligned}$$

$$\begin{aligned} R(Z) &= 1+b^T Z(I-AZ)^{-1}\mathbf{1} \\ &= 1+b^T(-tI+O(t^2)\mathbf{1} \\ &= 1+b^T(-tI)\mathbf{1}+O(t^2) \\ &> 1. \end{aligned}$$

Thus for AN-stability $b_j \geq 0$. Now consider Z to be purely imaginary.

$$Z = \begin{bmatrix} iv_1 t & 0 & \cdots & 0 \\ 0 & iv_2 t & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & iv_s t \end{bmatrix},$$

where v_i are real numbers and t is a small positive number.

$$\begin{aligned} R(Z) &= 1+b^T Z(I-AZ)^{-1}\mathbf{1} \\ &= 1+itb^T \operatorname{diag}(v)\mathbf{1} - t^2 b^T \operatorname{diag}(v) A \operatorname{diag}(v)\mathbf{1} + O(t^3) \\ &= 1+itb^T v - t^2 v^T \operatorname{diag}(b) A v + O(t^3), \end{aligned}$$

$$|R(z)|^2 = 1 - t^2 v^T M v + O(t^3),$$

where

$$M = \operatorname{diag}(b)A + A^T \operatorname{diag}(b) - bb^T.$$

$|R(z)|^2$ cannot exceed 1 for small t and

$$v^T M v > 0.$$

This implies M is positive semi-definite.

3.2 Stability of Runge–Kutta methods: non-linear case

Burrage and Butcher in [4] introduced the use of non-linear generalization of Dahlquist test equation (3.1) to study the stability of numerical methods. Consider a non-linear differential equation

$$y'(x) = f(y(x)), \tag{3.7}$$

such that f is non-linear and satisfies a contractive condition

$$\langle f(y) - f(z), y - z \rangle \leq 0, \tag{3.8}$$

where y and z are two solutions of (3.7) with different initial conditions and $\langle . \rangle$ is a semi-inner product. The norm induced by such an inner product is given as

$$\|y\|^2 = \langle y, y \rangle.$$

The equation (3.8) ensures that the two solutions do not drift apart and the distance between two solutions is a non-increasing function i.e.

$$\|y(x_1) - z(x_1)\| \leq \|y(x_0) - z(x_0)\|.$$

This is the case because

$$\begin{aligned} \frac{d}{dx}|y(x) - z(x)|^2 &= 2\langle f(y) - f(z), y - z \rangle \\ &\leq 0. \end{aligned}$$

The non-linear stability of a Runge–Kutta method implies

$$\|y_n - z_n\| \leq \|y_{n-1} - z_{n-1}\|.$$

Definition 3.2.1. *A Runge–Kutta method is BN-stable if, when applied to solve a non-linear non-autonomous initial value problem*

$$y'(x) = f(x, y(x)), \qquad y(x_0) = y_0.$$

satisfying the contractivity condition

$$\langle f(x, y), y \rangle \leq 0,$$

yields

$$\|y_n\| \leq \|y_{n-1}\|.$$

Definition 3.2.2. *A Runge–Kutta method is algebraically stable if $b_i > 0$, $i = 1, 2, \cdots, s$, and the matrix*

$$M = \operatorname{diag}(b)A + A^T \operatorname{diag}(b) - bb^T \tag{3.9}$$

is positive semi-definite.

The linear and non-linear stability described above are related to each other as shown in Figure 3.3.

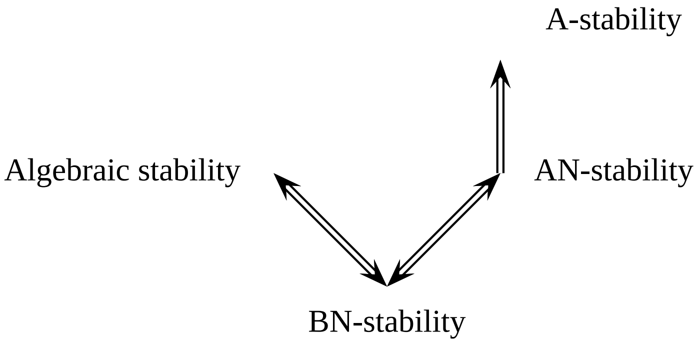


Figure 3.3: Relations between different types of stabilities.

3.3 Canonical and Symplectic Runge–Kutta methods

Consider an initial value problem,

$$y'(x) = f(y(x)), \qquad y(x_0) = y_0. \tag{3.10}$$

whose solution y has a quadratic invariant $I(y) = y^T Qy$, if

$$y^T Qf(y) = 0.$$

where Q is a symmetric square matrix.

A Runge–Kutta method is said to be canonical if it solves the equation (3.10) such that the numerical solution y_n also has the quadratic invariant $I(y_n)$. This property also has wider implications and guarantees that the symplectic property is preserved. Pioneering work in the development of symplectic Runge–Kutta methods is due to Cooper [17], Lasagni [36], Sanz-Serna [44] and Suris [48]. Their idea is based on features of algebraic stability introduced, in connection with stiff systems, by Burrage and Butcher [4] and Crouzeix [18]. The matrix M in (3.9) used for the characterisation of algebraic stability of Runge–Kutta methods play an important role in the symplecticity of Runge–Kutta methods. Stability and symplecticity are both qualitative features and the matrix M in (3.9) is crucial for both of them. Algebraic stability is concerned with solving dissipative systems and the matrix M should be positive semi-definite while the symplecticity is a characteristic property of Hamiltonian systems which are conservative, and requires matrix M to be zero.

An application of a Runge–Kutta method (2.1) to solve (3.10) results in

$$Y_i = y_0 + h \sum_{j=1}^{s} a_{ij} f(Y_j).$$

Since

$$\begin{aligned}\langle Y_i, f(Y_i)\rangle &= 0,\\ \Rightarrow \langle y_0, f(Y_i)\rangle + h\sum_{j=1}^{s} a_{ij}\langle f(Y_j), f(Y_i)\rangle &= 0. \end{aligned} \tag{3.11}$$

The output value is

$$\begin{aligned} y_1 &= y_0 + h\sum_{i=1}^{s} b_i f(Y_i),\\ \langle y_1, y_1\rangle &= \langle y_0, y_0\rangle + h\sum_{i=1}^{s} b_i\langle y_0, f(Y_i)\rangle\\ &\quad + h\sum_{j=1}^{s} b_j\langle f(Y_j), y_0\rangle + h^2\sum_{i,j=1}^{s} b_i b_j\langle f(Y_i), f(Y_j)\rangle. \end{aligned} \tag{3.12}$$

From (3.11) and (3.12), it is evident that

$$\langle y_1, y_1\rangle = \langle y_0, y_0\rangle,$$

provided

$$b_i a_{ij} + b_j a_{ji} - b_i b_j = 0. \tag{3.13}$$

Hamiltonian systems (1.5) belong to an important class of differential equation system with invariants. The solution (p,q) of a Hamiltonian system is symplectic meaning that it preserves the symplectic structure on manifold on which the solution exist which in turn implies that the differential 2-form $dp\wedge dq$ is conserved throughout the flow of the Hamiltonian system. As mentioned in Chapter 1, the differential two form $dp\wedge dq$ is a wedge product and represents the oriented area of a parallelogram with sides dp and dq. It has been shown by Sanz-Serna in [44] that an application of a Runge–Kutta method (2.1) to the Hamiltonian system (1.5) would yield an output (p_n, q_n) satisfying the property

$$dp_n \wedge dq_n = dp_{n-1}\wedge dq_{n-1},$$

provided equation (3.13) is satisfied. We have the following theorem.

Theorem 3.3.1. *A Runge–Kutta method* $[A, b^T, c]$ *is symplectic, if*

$$\operatorname{diag}(b)A + A^T\operatorname{diag}(b) - bb^T = 0. \tag{3.14}$$

The proof is gven in [44].

We take example of three known methods

- The explicit Euler method — non-symplectic

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

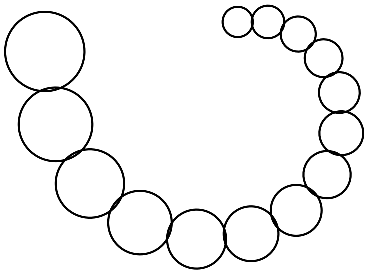


Figure 3.4: The explicit Euler method.

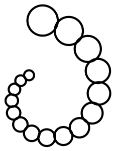


Figure 3.5: The implicit Euler method.

- The implicit Euler method — non-symplectic

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

- The implicit midpoint method — symplectic

$$y_n = y_{n-1} + hf(\frac{y_n + y_{n-1}}{2}).$$

We analyse the behaviour of these methods when solving the harmonic oscillator problem. The initial condition is the locus of a circle. Since harmonic oscillator problem is a Hamiltonian system, we expect the area of the circle to be conserved during the flow of the Hamiltonian system. For the explicit Euler method, the area of the initial circle increases during the integration process given in Figure 3.4 and for the implicit Euler method, the area of the initial circle decreases during the integration process given in Figure 3.5, which clearly shows the non-symplectic behaviour of the explicit Euler and the implicit Euler methods. Not only this, the circles are out of phase as well. However, for the implicit midpoint rule, the area of the circle is conserved given in Figure 3.6, a property shared by symplectic integrators.

Remark 3.3.2. *An s-stage Gauss method is symplectic for* $s = 1, 2\cdots$.

As an example we consider the two stage order four Gauss method

$$\begin{array}{c|cc} \frac{1}{2}-\frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4}-\frac{\sqrt{3}}{6} \\ \frac{1}{2}+\frac{\sqrt{3}}{6} & \frac{1}{4}+\frac{\sqrt{3}}{6} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array} . \tag{3.15}$$

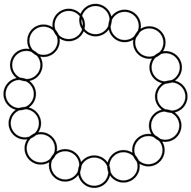


Figure 3.6: The midpoint rule

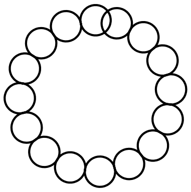


Figure 3.7: The IRK method

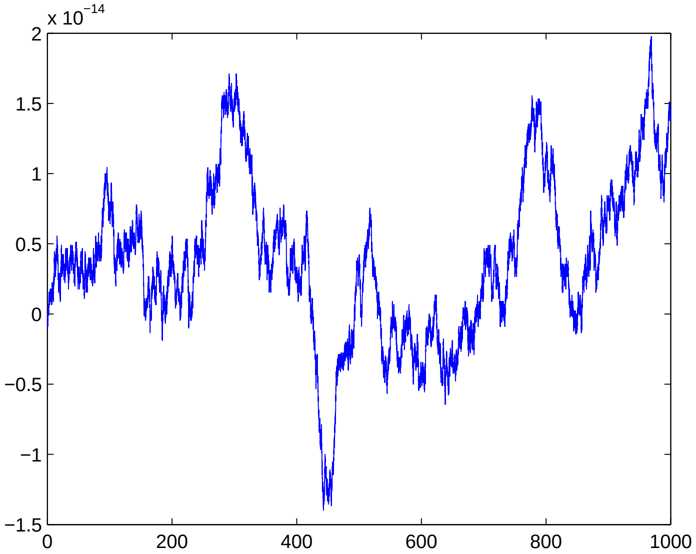


Figure 3.8: The absolute error in the energy of Harmonic oscillator by Gauss IRK (3.15).

The method (3.15) satisfies (3.14) and is therefore symplectic as is shown in Figure 3.7. The method (3.15) approximately conserves the total energy of the Hamiltonian system as well. The Harmonic oscillator problem is solved using the method (3.15) and the absolute error in the energy is depicted in the Figure 3.8 which remains bounded over 100000 steps with stepsize 0.01. Implicit Runge–Kutta methods are a preferred choice for solving stiff systems. However for solving non-stiff problems like the Hamiltonian systems, we can content ourselves to Diagonally implicit Runge–Kutta methods to save the cost of implementation. A class of s-stage diagonally implicit symplectic Runge–

Kutta methods must have $b_i \neq 0$ to avoid reducibility, and has the following structure

$$\begin{array}{c|ccccc} & \frac{b_1}{2} & 0 & 0 & \cdots & 0 \\ & b_1 & \frac{b_2}{2} & 0 & \cdots & 0 \\ & \vdots & \vdots & \vdots & \ddots & \vdots \\ & b_1 & b_2 & 0 & \cdots & \frac{b_s}{2} \\ \hline & b_1 & b_2 & b_3 & \cdots & b_s \end{array} . \tag{3.16}$$

The class of methods (3.16) can be seen as a composition of several implicit midpoint methods and hence by the fact that implicit midpoint method is symplectic and the composition of two symplectic Runge-Kutta methods is symplectic, we can deduce that the class of methods (3.16) is symplectic. A class of methods (3.16) of order 3 with 3 stages were constructed by Suris [49] and Cooper [17].

3.3.1 Composition of symplectic Runge-Kutta methods

The composition of two symplectic Runge–Kutta methods is another symplectic Runge–Kutta method. We recall from (2.11), that the composition of two Runge–Kuta methods $[a, b^T, c]$ and $[A, B^T, C]$ is

$$\begin{array}{c|cc} c & a & 0 \\ C + \mathbf{1}b^T & \mathbf{1}b^T & A \\ \hline & b^T & B^T \end{array} . \tag{3.17}$$

Let both methods be symplectic such that

$$\begin{aligned} \operatorname{diag}(b)a + a^T \operatorname{diag}(b) - bb^T &= 0, \\ \operatorname{diag}(B)A + A^T \operatorname{diag}(B) - BB^T &= 0. \end{aligned}$$

The composed method (3.17) is also a Runge–Kutta method which we denote as $[\overline{a}, \overline{b}^T, \overline{c}]$. The composed method will be symplectic if

$$\operatorname{diag}(\overline{b})\overline{a} + \overline{a}^T \operatorname{diag}(\overline{b}) - \overline{bb}^T = 0.$$

Now

$$\operatorname{diag}(\overline{b})\overline{a} = \begin{bmatrix} \operatorname{diag}(b)a & 0 \\ Bb^T & \operatorname{diag}(B)A \end{bmatrix},$$

and

$$\overline{a}^T \operatorname{diag}(\overline{b}) = \begin{bmatrix} a^T \operatorname{diag}(b) & bB^T \\ 0 & A^T \operatorname{diag}(B) \end{bmatrix},$$

and

$$\overline{bb}^T = \begin{bmatrix} bb^T & bB^T \\ Bb^T & BB^T \end{bmatrix}.$$

Thus we have

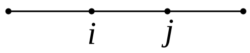
$$\begin{aligned} \operatorname{diag}(\bar{b})\bar{a} + \bar{a}^T \operatorname{diag}(\bar{b}) - \overline{bb}^T &= \begin{bmatrix} \operatorname{diag}(b)a + a^T \operatorname{diag}(b) - bb^T & bB^T - bB^T \\ Bb^T - Bb^T & \operatorname{diag}(B)A + A^T \operatorname{diag}(B) - BB^T \end{bmatrix} \\ &= 0. \end{aligned}$$

3.3.2 Order conditions for symplectic Runge–Kutta methods

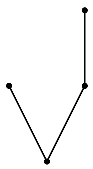
The order conditions for a Runge–Kutta method represent a relation among the coefficients of a Runge–Kutta method such that if these conditions are met, the method achieve a particular order. The order conditions have already been encountered in Chapter 2. The higher the order of a Runge–Kutta method, the higher the number of order conditions it should satisfy. However, the number of order conditions for a symplectic Runge–Kutta method decreases dramatically because the symplectic condition (3.14) acts as a constraint on the coefficients of a Runge–Kutta method. To understand this, we recall that for each order condition there is a rooted tree. Now the fact is that each rooted tree originates from a tree.

For a symplectic Runge–Kutta method, the trees can be divided into two categories, i.e. superfluous trees and non-superfluous trees. For symplectic Runge-Kutta methods, the superfluous trees do not contribute any order condition as the corresponding order condition is already satisfied by the symplectic condition and the non-superfluous trees contribute only one order condition. This results in reduction of order conditions required for a symplectic Runge–Kutta method to achieve a particular order. Before proceeding any further, let us look at the notion of superfluous trees in detail.

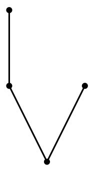
A tree is called superfluous, if it generates identical rooted trees when any two adjacent nodes of the tree are taken as a root. Consider a tree t and name two of its vertices as i and j. Let i is taken as root and $\rho_1 t_i$ is a rooted tree. Again let j is taken as root and $\rho_2 t_j$ is a rooted tree.



t

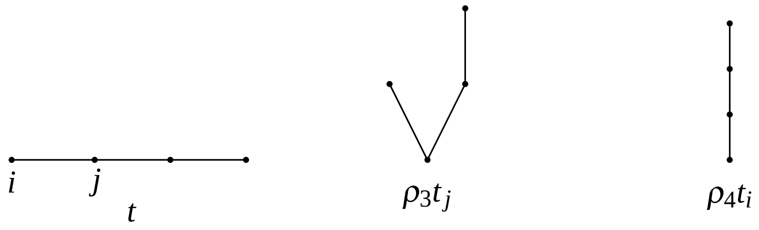


$\rho_1 t_i$



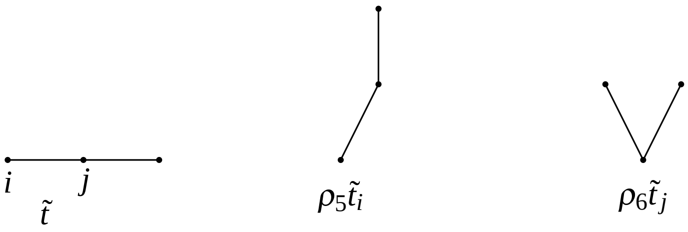
$\rho_2 t_j$

Since $\rho_1 t_i$ and $\rho_2 t_j$ represents similar rooted trees with a difference of orientation, the underlying tree t is superfluous. The important factor being that we can choose any two adjacent vertices as roots. Thus if we choose different vertices as i and j, we get two rooted trees $\rho_3 t_i$ and $\rho_4 t_j$

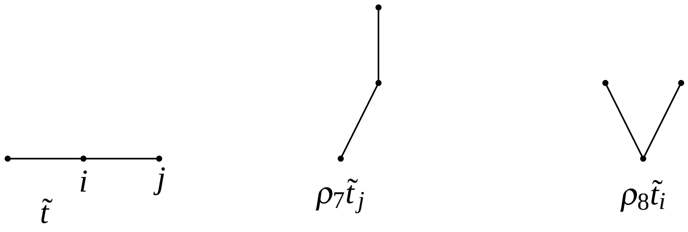


Although the two rooted trees $\rho_4 t_i$ and $\rho_3 t_j$ are not similar in this case, yet the underlying tree t is superfluous as we have already found a scenario in the first case when two middle vertices are selected as roots, yielding similar rooted trees.

Once the superfluous trees are deleted we are left with order conditions associated with non-superfluous trees. We consider another tree $\tilde{t}$ and select two of its vertices as i and j. Let i is taken as root and $\rho_5 \tilde{t}_i$ is a rooted tree. Again let j is taken as root and $\rho_6 \tilde{t}_j$ is a rooted tree.



Both rooted trees are different. Again choose different vertices as i and j to become roots



and we still get different rooted trees $\rho_7 \tilde{t}_i$ and $\rho_8 \tilde{t}_i$. Since we did not get two similar rooted trees for any of the two vertices being taken as root, the tree $\tilde{t}$ is a non-superfluous tree. The effect of superfluous and non-superfluous trees on the order conditions is studied as two separate cases below.

Case 1: This case is related to the tree t which has 4 vertices. We assume that the order conditions for all trees with vertices less than 4 are satisfied. Multiply equation (3.14)

Order	General Runge–Kutta method	Symplectic Runge–Kutta method
1	1	1
2	2	1
3	4	2
4	8	3
5	17	6

Table 3.1: Order conditions for general and symplectic Runge–Kutta methods to order 5.

with c_i and c_j and take summation over i and j

$$\begin{aligned}&\sum_{i,j} b_i c_i a_{ij} c_j + \sum_{i,j} b_j c_j a_{ji} c_i - \sum_{i,j} b_i c_i b_j c_j = 0,\\ \Rightarrow &\sum_{i,j} b_i c_i a_{ij} c_j = \tfrac{1}{8}.\end{aligned} \tag{3.18}$$

This is the order condition related to the rooted tree $\rho_1 t_i$ which originates from a superfluous tree. Thus the order condition related to superfluous trees can always be deduced from the symplectic condition (3.14). Hence, for higher order symplectic Runge–Kutta methods, all order conditions related to superfluous trees are automatically satisfied by the symplectic condition and are therefore not required.

Case 2: This case is related to the tree $\tilde{t}$ which has 3 vertices. We assume that the order conditions for all trees with vertices less than 3 are satisfied. Multiply equation (3.14) with c_j and take summation over i and j

$$\begin{aligned}&\sum_{i,j} b_i a_{ij} c_j + \sum_{i,j} b_j c_j a_{ji} - \sum_{i,j} b_i b_j c_j = 0,\\ \Rightarrow &(\sum_{i,j} b_i a_{ij} c_j - \tfrac{1}{6}) + (\sum_{j} b_j c_j^2 - \tfrac{1}{3}) = 0.\end{aligned}$$

The last equation is a summation of two order conditions for the rooted trees $\rho_6 \tilde{t}_i$ and $\rho_5 \tilde{t}_j$ that originates from a non-superfluous tree $\tilde{t}$. It is evident that if one of them is satisfied, the other is automatically satisfied. So we only require one order condition. Hence for this and higher order symplectic Runge–Kutta methods, the non-superfluous trees only contribute one order condition. Table 3.1 shows the number of order conditions required for a general and symplectic Runge–Kutta method to attain a particular order. We have the following theorem from [45].

Theorem 3.3.3. *A symplectic Runge–Kutta method has an order p, if for each non-superfluous tree $\tilde{t}$ with any vertex as a root*

$$\phi(\rho\tilde{t}) = \frac{1}{\gamma(\rho\tilde{t})},$$

where $\rho\tilde{t}$ represents the rooted tree of $\tilde{t}$ of order up to p.

3.3.3 Symplectic Runge–Kutta methods with transformations

A class of symplectic Runge–Kutta methods can be constructed using Vandermonde transformation. The idea is to pre and post multiply the Vandermonde matrix with the matrix of symplectic condition for Runge–Kutta method (3.14). The idea is best explained with the help of an example whereby the construction of a class of two stage, second order symplectic Runge–Kutta method is considered.

Consider a Runge Kutta–method $[A, b^T, c]$. Consider the matrix for symplectic condition

$$M_{ij} = b_i a_{ij} + b_j a_{ji} - b_i b_j, \qquad i, j = 1, \cdots, s.$$

Consider a Vandermonde matrix V

$$V = c_i^{j-1} = \begin{pmatrix} 1 & c_1 & c_1^2 & \cdots & c_1^{s-1} \\ 1 & c_2 & c_2^2 & \cdots & c_2^{s-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & c_s & c_s^2 & \cdots & c_s^{s-1} \end{pmatrix}.$$

Multiply the matrix M with the matrix V as follows

$$c_i^{k-1} \; (\; b_i a_{ij} + b_j a_{ji} - b_i b_j \;) \; c_j^{l-1} = 0, \qquad \forall \;\; i, j, k, l.$$

For order two put $k, l = 1, 2$ and take summation over i and j from 1 to s

$$\begin{aligned} \sum_{i,j} b_i a_{ij} + \sum_{i,j} b_j a_{ji} - \sum_{i,j} b_i b_j &= 0, \\ \sum_{i,j} b_i a_{ij} c_j + \sum_{i,j} b_j c_j a_{ji} - \sum_{i,j} b_i b_j c_j &= 0, \\ \sum_{i,j} b_i c_i a_{ij} + \sum_{i,j} b_j a_{ji} c_i - \sum_{i,j} b_i c_i b_j &= 0, \\ \sum_{i,j} b_i c_i a_{ij} c_j + \sum_{i,j} b_j c_j a_{ji} c_i - \sum_{i,j} b_i c_i b_j c_j &= 0. \end{aligned}$$

Since we are constructing method of order two, the following order conditions must satisfy.

$$\sum_{i=1}^{s} b_i = 1, \qquad \sum_{i=1}^{s} b_i c_i = \tfrac{1}{2}.$$

Thus we have

$$\begin{aligned}
&\sum_i b_i c_i &&= \tfrac{1}{2},\\
&\sum_{i,j} b_i a_{ij} c_j + \sum_{i,j} b_j c_j a_{ji} &&= \tfrac{1}{2},\\
&\sum_{i,j} b_i c_i a_{ij} + \sum_{i,j} b_j a_{ji} c_i &&= \tfrac{1}{2},\\
&\sum_{i,j} b_i c_i a_{ij} c_j &&= \tfrac{1}{8}.
\end{aligned}$$

Consider the relation

$$b_i(c_i - c_1) = b_i c_i - b_i c_1,$$

Take summation over i from 1 to s, and use previous equations we get

$$\sum_i b_i(c_i - c_1) = \sum_i b_i c_i - \sum_i b_i c_1,$$

$$b_2(c_2 - c_1) = \frac{1}{2} - c_1,$$

$$b_2 = \frac{\frac{1}{2} - c_1}{c_2 - c_1}.$$

Similarly we can get

$$b_1 = \frac{\frac{1}{2} - c_2}{c_1 - c_2}.$$

Now consider the relation

$$b_i(c_i - c_1)a_{ij}(c_j - c_1) = b_i c_i a_{ij} c_j - b_i c_i a_{ij} c_1 - b_i a_{ij} c_j c_1 + b_i a_{ij} c_1 c_1.$$

Take summation over i and j, and use previous equations we get

$$a_{22} = \frac{\frac{1}{8} - \frac{c_1}{3} - \frac{c_1}{6} + \frac{c_1 c_1}{2}}{b_2(c_2 - c_1)(c_2 - c_1)}.$$

Similarly we get

$$a_{11} = \frac{\frac{1}{8} - \frac{c_2}{3} - \frac{c_2}{6} + \frac{c_2 c_2}{2}}{b_1(c_1 - c_2)(c_1 - c_2)},$$

$$a_{21} = \frac{\frac{1}{8} - \frac{c_2}{3} - \frac{c_1}{6} + \frac{c_1 c_2}{2}}{b_2(c_2 - c_1)(c_1 - c_2)},$$

$$a_{12} = \frac{\frac{1}{8} - \frac{c_1}{3} - \frac{c_2}{6} + \frac{c_1 c_2}{2}}{b_1(c_1 - c_2)(c_2 - c_1)}.$$

A class of second order Runge-Kutta methods can be found by choosing c_1 and c_2. If we impose the following extra condition on c_1 and c_2, obtained using quadrature, the method thus obtained is of order 3.

$$\frac{c_1 + c_2}{2} - c_1 c_2 = \tfrac{1}{3}.$$

3.4 Stability and symplecticity of multistep methods

The linear stability analysis of linear multistep methods can be understood by trying to solve the Dahlquist test equation

$$y' = \lambda y,$$

with the linear multistep method

$$y_n = \sum_{i=1}^{k} \alpha_i y_{n-i} + h \sum_{i=0}^{k} \beta_i f(y_{n-i}). \tag{3.19}$$

This result in a linear difference equation given for $z = hq$

$$(1 - z\beta_0)y_n - (\alpha_1 + z\beta_1)y_{n-1} - \ldots - (\alpha_k + z\beta_k)y_{n-k} = 0. \tag{3.20}$$

A linear multistep method is stable if all of its numerical solutions y_n are bounded for $n \to \infty$, while the corresponding set of all $z's$ determines the stability domain of linear multistep methods. An application of Lagrange method to solve the difference equation (3.20) by assuming $y_j = w^j$ yields

$$(1 - z\beta_0)w^n - (\alpha_1 + z\beta_1)w^{n-1} - \ldots - (\alpha_k + z\beta_k)w^{n-k} = 0.$$

Divide throughout by w^{n-k} and rearrange we get

$$(1 - z\beta_0)w^k - (\alpha_1 + z\beta_1)w^{k-1} - \ldots - (\alpha_k + z\beta_k) = 0,$$
$$(w^k - \alpha_1 w^{k-1} - \ldots - \alpha_k) - z(\beta_0 w^k + \beta_1 w^{k-1} + \ldots + \beta_k) = 0.$$

This can be represented in terms of the characteristic polynomials ρ and σ of the linear multistep methods (2.15)

$$\rho(w) - z\sigma(w) = 0. \tag{3.21}$$

The stability domain is the set of all $z \in \mathbb{C}$ such that the roots of (3.21) lies inside the unit disc i.e. $|w(z)| \leq 1$ and for the multiple roots, $|w(z)| < 1$. The boundary locus method is used to plot the stability domain of linear multistep methods. Locus is a term used for a set of points which share similar properties. A unit circle can be represented as a locus of points whose distance from the centre of circle is always one. The procedure to plot the stability domain is as follows. From (3.21)

$$z = \frac{\rho(w)}{\sigma(w)}.$$

Consider all points on the boundary of stability domain

$$|w(z)| = 1, \qquad \Rightarrow \qquad w(z) = e^{i\theta}, \qquad 0 \leq \theta \leq 2\pi.$$

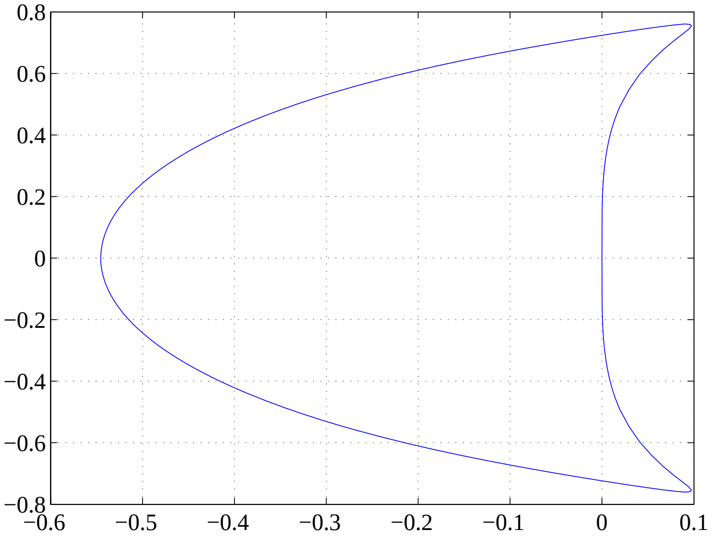


Figure 3.9: The stability domain of Adams-Bashforth method (3.22) bounded by the closed curve.

Therefore the locus of boundary of the stability region is given as

$$z = \frac{\rho(e^{i\theta})}{\sigma(e^{i\theta})}.$$

Consider an example of a 3rd order Adams-Bashforth method

$$y_n = y_{n-1} + h\big(\tfrac{23}{12} f_{n-1} - \tfrac{4}{3} f_{n-2} + \tfrac{5}{12} f_{n-3}\big), \tag{3.22}$$

The characteristic polynomial are

$$\begin{aligned} \rho(w) &= 1 - w, \\ \sigma(w) &= \frac{23}{12}w - \frac{4}{3}w^2 + \frac{5}{12}w^3. \end{aligned}$$

The locus of boundary is

$$z = \frac{1 - e^{i\theta}}{\frac{23}{12}e^{i\theta} - \frac{4}{3}e^{2i\theta} + \frac{5}{12}e^{3i\theta}}.$$

The stability domain of the Adams-Bashforth method (3.22) is a bounded region as shown in Figure 3.9. The concept of A-stability for multistep method is similar to that of Runge–Kutta method. A multistep method is A-stable, if all the negative complex plane is included in the stability domain and therefore explicit multistep methods cannot be A-stable because of having bounded stability domains. Implicit multistep methods can however be A-stable but they are restricted to low order methods because of the Dahlquist second barrier which states that no A-stable multistep method can have order greater than two. This is true for all BDF methods and the predictor-corrector pairs. The somewhat weaker stability in the name of $A(\alpha)$-stability is also applicable to linear multistep methods just like Runge-Kutta methods.

3.4.1 One-leg methods and G-stability

Dahlquist in [23] proposed to use one-leg methods for the stability analysis of multistep methods for non-linear differential equations. The general form of a one-leg method is

$$y_n = \sum_{i=1}^{k} \alpha_i y_{n-i} + hf\Big(\sum_{i=0}^{k} \beta_i x_{n-i}, \sum_{i=0}^{k} \beta_i y_{n-i}\Big), \tag{3.23}$$

provided $\sum_{i=0}^{k} \beta_i = 1$. The one-leg methods are a twin of linear multistep methods (3.19), however, the former has cheaper implementation cost, because linear multistep methods evaluate the function f at a number of past values while the one-leg counterpart evaluates function f only once at the linear combination of the past values. It has also been proposed by Liniger [37], Dahlquist et al [25],[26] and Watanabe and Sheikh [51] to use one-leg methods as independent numerical methods in their own right.

The linear multistep methods and one-leg methods are related to each other via a transformation. Let $\bar{y}$ is the sequence of output values obtained from linear multistep methods (3.19) and let $\hat{y}$ is the sequence of output values obtained from one-leg methods (3.23), then the transformation relating the two methods is

$$\hat{x}_n = \sum_{i=0}^{k} \beta_i \bar{x}_{n-i}, \qquad \hat{y}_n = \sum_{i=0}^{k} \beta_i \bar{y}_{n-i}.$$

therefore the stability analysis of linear multistep methods can be interpreted via the stability analysis of one-leg methods and in fact both methods have same difference equations for linear autonomous problems and hence have same stability regions. The trapezoidal method is a two step method given as

$$y_n = y_{n-1} + \tfrac{1}{2}h\Big(f(x_{n-1}, y_{n-1}) + f(x_n, y_n)\Big).$$

The corresponding one-leg method is the famous mid-point method

$$y_n = y_{n-1} + hf\Big(\frac{x_{n-1}+x_n}{2}, \frac{y_{n-1}+y_n}{2}\Big).$$

The nonlinear stability analysis of linear multistep methods is carried out using one-leg methods as proposed by G. Dahlquist and hence given the name G-stability. Consider and autonomous non-linear differential equation

$$y' = f(y), \tag{3.24}$$

such that the function f is non-linear and satisfies the contractive condition

$$\langle f(y) - f(z), y - z \rangle \leq 0.$$

where y and z are two solutions of (3.24) with different initial conditions and $< . >$ represents an inner product. It is desired from the numerical method to produce results satisfying the contractive condition which in turn implies that two approximately equal numerical approximations at each step of the numerical method should not drift apart. For one-step methods, this is easy to visualise as we have seen for Runge–Kutta methods that the standard norm can be applied given as

$$\|y_n - z_n\| \leq \|y_{n-1} - z_{n-1}\|.$$

However, for a $k-$step linear multistep method, k input values are available at step n, so the norm has to be modified to accommodate the k input values. Suppose

$$Y_n = [y_n, y_{n-1}, y_{n-2}, \ldots, y_{n-k}]^T.$$

Given a positive definite symmetric matrix G of dimension $k \times k$, define a G-norm as

$$\|Y_n\|_G^2 = \sum_{i=0}^{k} \sum_{j=0}^{k} g_{ij} \|Y_{n-i}, Y_{n-j}\|. \tag{3.25}$$

Definition 3.4.1. *The one-leg method* (3.23) *is G-stable, if the two numerical solutions y and z of the contractive nonlinear differential equation* (3.24) *do not drift apart under the G-norm* (3.25), *i.e.*

$$\|y_n - z_n\|_G < \|y_{n-1} - z_{n-1}\|_G.$$

where G is a real symmetric, positive definite matrix.

The stability of linear multistep methods and one-leg methods is related via the following theorem due to Dahlquist [24].

Theorem 3.4.2. *An irreducible linear multistep method is A-stable, if the corresponding one-leg method is G-stable.*

3.4.2 Symplecticity of multistep methods

Long term behaviour of multistep methods for the solution of Hamiltonian systems is unsatisfactory in general. Not only multistep methods are non-symplectic, they also suffer from parasitic solutions. However, there do exist partitioned multistep methods and multistep methods for second order differential equation systems which exhibit acceptable behaviour for long time integration of Hamiltonian systems. This however requires the method to be symmetric and possibly avoiding the double zero of the ρ polynomial on the unit disc as this would lead to exponential error growth [20].

Kirchgraber in [35] showed that every strictly stable linear multistep method is essentially equal to a one-step method. The non-symplectic behaviour of multistep methods is due to non-symplectic nature of their underlying one-step method. Tang proved the following theorem in [50] which was conjectured earlier by Feng Kang.

Theorem 3.4.3. *The underlying one-step method of a consistent linear multistep method cannot be symplectic.*

This negative result extends to one-leg methods and partitioned linear multistep methods. Only implicit midpoint rule is symplectic because its underlying one-step method is symplectic and the explicit and implicit pair of the Euler method results in symplectic Euler method.

Following the concept of G-stability of one-leg methods, we can find G-symplectic one-leg methods with good results for long time integration of Hamiltonian systems. A multistep method is G-symplectic, if it solves conservative equation (3.10) with invariant $I(y) = y^T Qy$, such that the numerical solution satisfies

$$\langle Y_n, Y_n \rangle_{G\otimes Q} = \langle Y_{n-1}, Y_{n-1} \rangle_{G\otimes Q}.$$

The norm induced by such an inner product is given by equation (3.25). We have the following theorem due to Eirola and Sanz-Serna [28]

Theorem 3.4.4. *Every irreducible symmetric one-leg method is G-symplectic for some matrix G.*

3.4.3 Parasitic solutions and backward error analysis

All multistep methods suffer from parasitic solutions, i.e. those numerical solutions which accompany the numerical approximation to the exact solution. The multistep methods require initial approximations to start the procedure and if the perturbations in the initial approximations are not damped out efficiently, the parasitic solutions overtake the actual solution. For some problems, the parasitic solutions decrease as time goes on and dies out and does not affect the numerical solution. However, for other problems, the parasitic solutions increase and become large enough to destroy the actual solution and causes numerical instability. We study the parasitic behaviour of leapfrog method which can be written as

$$y_n - 2hf(y_{n-1}) - y_{n-2} = 0.$$

Let us try to solve $y' = \lambda y$. This would yield for $z = h\lambda$

$$y_n - 2z(y_{n-1}) - y_{n-2} = 0.$$

The characteristic equation is

$$w^2 - 2zw - 1 = 0.$$

This would provide two roots

$$w_1 = z + \sqrt{z^2+1},$$
$$w_2 = z - \sqrt{z^2+1}.$$

The general solution is

$$y_1^n = Aw_1^n + Bw_2^n$$

where the constants A and B can be found from the initial conditions. The root w_1 is approximating the actual solution which is an exponential function, while the root w_2 is the parasitic solution. If $|Re(z)| > 0$ then w_1 dominates w_2 and the parasitic solution dies out eventually. However if $|Re(z)| < 0$, then the parasitic solution w_2 dominates the approximation to the actual solution w_1 and destroys the solution altogether. This is sometimes referred to as weak instability.

Dahlquist in [21] found parasitic growth parameter in connection with stability of the multistep methods which is given as

$$\mu_p = \frac{\sigma(w_p)}{w_p \rho'(w_p)}. \tag{3.26}$$

where μ_p is the parasitic growth parameter for the parasitic root w_p of the $\rho(w)$ polynomial of the linear multistep method and $\sigma(w)$ is its other characteristic polynomial. Hairer et al. in [30] has used the backward error analysis to obtain the same parasitic growth parameter.

In numerical analysis, sometimes backward error analysis gives better understanding than forward error analysis. The forward error analysis is concerned with the error between exact solution and the approximate solution of an ordinary differential equation. The backward error analysis is related to the qualitative behaviour of a numerical method and was introduced by Wilkinson [52]. Thus for a symplectic method to solve a Hamiltonian system, the numerical solution is an exact solution of a nearby Hamiltonian system known as the modified equation. The backward error analysis is concerned with finding that modified equation. The symplectic integrator exactly conserves the total energy of the modified Hamiltonian system. Consider the Hamiltonian system (1.8)

$$y' = J^{-1}\nabla H.$$

Let us solve the Hamiltonian system with a symplectic integrator $\phi_h(y)$. According to the theory of modified equations $\phi_h(y)$ is an exact solution of a modified system which is also a Hamiltonian system [3] and is given as

$$y' = J^{-1}(\nabla H + h\nabla H_2 + h^2\nabla H_3 + \cdots). \tag{3.27}$$

where $H_2, H_3, \cdots$ can be found by comparing the Taylor series expansion of the solution of (3.27) with the symplectic numerical method being used [30]. The analysis is important not only to find the modified equation but because a symplectic integrator cannot simultaneously conserve the exact energy of a Hamiltonian system and symplecticity [46]. Symplecticity is a characteristic property of a Hamiltonian system which should be conserved exactly and by doing so, the symplectic integrator conserve energy of a nearby Hamiltonian system.

Coming back to the role of parasitic growth parameter in multistep methods, Hairer et al. in [30] has considered the parasitic modified equations. The general solution of a multistep method can be written as

$$y_n = y(nh) + \sum_{p\in I^*} w_p^n z_p(nh). \tag{3.28}$$

where $y(x)$ and $z_p(x)$ are smooth solutions representing the actual solution and the parasitic solutions respectively. I^* is the index set of all parasitic roots $w_p = \{w_2, w_3 \cdots, w_k\}$ of the $\rho(w)$ polynomial except the principal root, $w_1 = 1$. The general solution is obtained by truncating the modified equations for every solution of a multistep method and it was found that the truncated modified equation related to parasitic solution z_p is

$$z_p' = \mu_p f'(y) z_p.$$

where μ_p is the parasitic growth parameter given in (3.26). If the parasitic growth parameter is zero then the parasitic solution will not have any effect on the actual solution. There exist long time bounds to limit the effect of the parasitic growth parameter.

3.5 Stability of general linear methods: linear case

The stability analysis of general linear methods follows the same direction as stability analysis of Runge-Kutta methods. The linear stability analysis for stiff ordinary differential equations and non-linear stability analysis for non-autonomous and dissipative non-linear ordinary differential equations is studied here.

Following Dahlquist [22], a general linear method (2.16) is applied to the linear test equation

$$y'(x) = \lambda y(x),$$

The stages become

$$\begin{aligned} Y &= zAY + Uy^{[n-1]}, \\ &= (I - zA)^{-1}Uy^{[n-1]}. \end{aligned}$$

where $z = h\lambda$, and h is the stepsize. The output approximation is

$$\begin{aligned} y^{[n]} &= zBY + Vy^{[n-1]}, \\ &= zB(I - zA)^{-1}Uy^{[n-1]} + Vy^{[n-1]}, \\ &= (V + zB(I - zA)^{-1}U)y^{[n-1]}, \\ &= M(z)y^{[n-1]}. \end{aligned}$$

where $M(z)$ is an $r \times r$ matrix valued function representing stability matrix of the general linear method and is given as

$$M(z) = V + zB(I - zA)^{-1}U,$$

Definition 3.5.1. *The stability function of a general linear method is the polynomial*

$$\phi(w,z) = \det(wI - M(z)),$$

Definition 3.5.2. *A general linear method has a stability order $\tilde{p}$, if*

$$\phi(exp(z),z) = O(z^{\tilde{p}+1}).$$

Definition 3.5.3. *The stability region of a general linear method is the set of all z in the complex plane such that*

$$\sup_{n=1}^{\infty} \|M(z)^n\| < \infty.$$

This means that the eigenvalues of $M(z)$ satisfy $|w| \leq 1$ when w has unit multiplicity and $|w| < 1$ when w has multiplicity greater than 1. The boundary locus method is used to plot the stability region of a general linear methods along the following lines.

- Solve $\phi(w,z) = 0$ to find $z = g(w)$.
- Take values of w on the unit circle, i.e. $w = \exp(i\theta)$ for $\theta \in [0, 2\pi]$.
- The resulting z will provide the boundary of the stability region.

When a general linear method is applied to solve stiff ordinary differential equations, there is a severe restriction on the stepsize either due to stability considerations or due to accuracy concerns. To make sure the stability is not restricting the stepsize, the general linear method has to be A-stable.

Definition 3.5.4. *A general linear method is A-stable, if the stability matrix $M(z)$ is power bounded for all $z \in \mathbb{C}^-$.*

If in addition, the general linear method is to be stable at infinity, then it should be L-stable.

Definition 3.5.5. *A general linear method is L-stable, if it is A-stable and*

$$\rho(M(\infty)) = 0.$$

Definition 3.5.6. *A general linear method is strictly stable, if all eigenvalues of the matrix V lie inside the unit disc except one which is on the boundary.*

The Dahlquist test equation (3.1) is first generalised to include non-autonomous ordinary differential equations of the form

$$y'(x) = \lambda(x)y(x).$$

If a general linear method is applied to solve this ODE, the outcome is

$$\begin{aligned} Y &= (I - AZ)^{-1}Uy^{[n-1]}, \\ y^{[n]} &= (V + BZ(I - AZ)^{-1}U)y^{[n-1]}. \end{aligned}$$

where $Z = \text{diag}(z_1, z_2, \cdots, z_s)$ and $z_i = h\lambda(x_{n-1} + c_i h),\ i = 1, \cdots, s.$ The stability matrix is given as

$$M(Z) = V + BZ(I - AZ)^{-1}U.$$

Definition 3.5.7. *A general linear method is AN-stable, if there exist an inner product norm $\|.\|$ such that $M(Z)$ is power bounded i.e.*

$$sup_n \|M(Z)^n\| < C, \qquad \forall z_i \in \mathbb{C}^-.$$

3.6 Stability of general linear methods: non-linear case

The Dahlquist test equation (3.1) is further generalised to include non-linear ordinary differential equations whose solution is dissipative.

$$y'(x) = f(y(x)),$$

such that

$$\langle f(y), y\rangle \leq 0. \tag{3.29}$$

where $f(y)$ is a non-linear function and the equation (3.29) implies that $\|y(x)\|$ is a non-increasing function. Given a problem of this nature and a general linear method to solve it, the computed solution has the non-increasing nature if,

$$\|y^{[n]}\|_G < \|y^{[n-1]}\|_G.$$

Here a G-norm is used where G is an $r \times r$ matrix. For an s-stage, r-step general linear method, r input values are available at each step, so instead of a standard Euclidean norm, a G-norm is used given as

$$\|y^{[n]}\|_G^2 = \langle y^{[n]}, y^{[n]}\rangle_G,$$

such that

$$\langle y, z\rangle_G = \sum_{i,j=1}^{r} g_{ij}\langle y_i, z_j\rangle.$$

Recall that a general linear method is given as

$$\begin{aligned} Y &= hAf(Y) + Uy^{[n-1]}, \\ y^{[n]} &= hBf(Y) + Vy^{[n-1]}, \end{aligned}$$

where the super vector $Y = [Y_1, \cdots, Y_s]$ represents s-stages and the super vector $y^{[n]} = [y_1^{[n]}, \cdots, y_r^{[n]}]$ represents r-steps. The G-norm defined above is for r-steps, i.e. stable behaviour of a general linear method is guaranteed if

$$\|y^{[n]}\|_G < \|y^{[n-1]}\|_G.$$

For the s-stages Y, we define a D-norm such that

$$\langle U, V\rangle = \sum_{i=1}^{s} d_i\langle U_i, V_i\rangle.$$

where $D = \operatorname{diag}(d_1, \cdots, d_s)$ is an $s \times s$ is a positive semi-definite diagonal matrix. We have the following theorem

Theorem 3.6.1. *For a general linear method (A,U,B,V), to solve dissipative problem* (3.29)*, a contractive numerical solution is possible under G norm such that*

$$\|y^{[n]}\|_G^2 < \|y^{[n-1]}\|_G^2,$$

provided the matrix given below is positive semi-definite.

$$\begin{bmatrix} DA+A^TD-B^TGB & DU-B^TGV \\ U^TD-V^TGB & G-V^TGV \end{bmatrix}. \tag{3.30}$$

The proof is given in [5].

3.7 Symplecticity of general linear methods

General linear methods is a bigger class of methods comprising of Runge–Kutta methods and linear multistep methods as special cases. Because of their multivalue nature, they cannot be symplectic in general, in line with linear multistep methods. The underlying one-step method can shed light on symplectic behaviour of general linear methods.

Following the work of Kirchgraber [35], Stoffer showed in [47] that every strictly stable general linear method is essentially conjugate to a one-step method of the same order. Thus, to proceed further we have to restrict ourselves to the strictly stable general linear methods such that the matrix V has 1 as simple eigenvalue and all other eigenvalues lie inside the unit disc. A transformation T for the general linear method M in (2.17), is therefore required to separate the eigenvalues of matrix V and we consider

$$T^{-1}MT = \left[\begin{array}{c|c} A & UT \\ \hline T^{-1}B & T^{-1}VT \end{array}\right], \tag{3.31}$$

where the transformation is such that

$$T^{-1}VT = \left[\begin{array}{c|c} 1 & 0 \\ \hline 0 & V^* \end{array}\right], \tag{3.32}$$

with spectral radius satisfying $\rho(V*) < 1$. The transformation of the output values at step n is

$$z^{[n]} = T^{-1}y^{[n]}, \tag{3.33}$$

such that the first component of the transformed value $z^{[n]}$ still approximates the exact solution. Considering the matrix V to have the special structure of (3.32), we have the following theorem from [47]

Theorem 3.7.1. *Let M be a strictly stable general linear method of order p. Then there exist a starting method $\overline{S}$ and a one-step method ϕ, such that*

- *M is of order p relative to $\overline{S}$.*
- *ϕ is of order p.*
- *ϕ is conjugate to M.*

The one-step method ϕ is known as the underlying one-step method and the symplectic behaviour of the general linear method hinges on the fact that the underlying one-step method is symplectic. However, it has been proved in [16] that it is not possible for a general linear method to be symplectic unless it is equivalent to a Runge–Kutta method.

The negative result of the non-existence of a truly symplectic general linear method is in line with the behaviour of linear multistep methods. Likewise, it is possible to study the symplectic behaviour of general linear methods under G-norm given in (3.25). The first such method was introduced by Butcher [12]. The idea behind the construction of such methods was taken from the idea of symplectic behaviour of Runge–Kutta methods. For Runge-Kutta methods, the matrix (3.9) which appears in its non-linear stability analysis, if equals to zero, results in the method being symplectic. For general linear methods, the matrix in (3.30) being actually zero results in the general linear method to be G-symplectic. In fact, general linear methods are a generalisation of two different classes of methods, namely the Runge–Kutta methods and the linear multistep methods and justifiably the G-symplecticity of general linear methods inherit the corresponding attributes of both classes of methods. A detailed analysis of G-symplectic behaviour of general linear methods is the subject of Chapter 4.

Chapter 4

General linear methods for ordinary differential equations with invariants

General linear methods are multivalue multistage methods for the solution of ordinary differential equations. A detailed discussion concerning the structure and properties of these methods is given in Chapter 2 and Chapter 3. However in this chapter, we are concerned with the ability of general linear methods to solve Hamiltonian systems so as not only to get accurate approximations to their exact solution, but also to achieve the same qualitative behaviour for the approximate solution as possessed by the exact solution. In general we are interested to know when a general linear method can mimic the dynamics of a differential equation system with quadratic invariants over long time. In particular, this means adhering to symplectic behaviour for Hamiltonian systems.

There are several desirable attributes, a numerical method, in this case a general linear method, should have. Symplecticity and time reversal symmetry play an important role in the choice of the numerical method for Hamiltonian systems. The Hamiltonian systems are reversible and have symplectic behaviour and it is desirable by the numerical method to preserve these properties. Another important factor to be considered when designing a general linear method for the solution of Hamiltonian systems is to avoid the parasitic solution growth which is typical of multivalue methods.

4.1 G-symplectic general linear methods

An s-stage, r-value irreducible general linear method, with $r > 1$, cannot preserve the quadratic invariants in general over long time intervals [16]. However we would like to know how close we can get to conserving them. For $r = 1$, a general linear method reduces to a Runge–Kutta method. Such a method is symplectic provided its coefficients satisfy the symplectic condition (3.14). We would like to know if a similar criterion is available for a general linear method. Consider the method

$$\begin{aligned} Y &= hAf(Y) + Uy^{[n-1]}, \\ y^{[n]} &= hBf(Y) + Vy^{[n-1]}. \end{aligned} \tag{4.1}$$

To study this method for quadratic invariants, the inner product has to be modified to accommodate r-input values $y^{[n-1]}$, and r-output values $y^{[n]}$. We introduce a G-norm similar to that introduced for non-linear stability of general linear methods [4]. Here we consider a symmetric $r \times r$ matrix G with elements g_{ij}, and define the inner product

$$\langle y,z \rangle_G = \sum_{i,j=1}^{r} g_{ij} \langle y_i, z_j \rangle,$$

where

$$y = \begin{bmatrix} y_1 \\ \vdots \\ y_r \end{bmatrix}, \qquad z = \begin{bmatrix} z_1 \\ \vdots \\ z_r \end{bmatrix}.$$

The G-norm introduced by such an inner product is

$$\|y\|_G^2 = \langle y,y \rangle_G.$$

We would like to know if there exist a G matrix for which this property holds. In addition to G, we introduce a diagonal $s \times s$ matrix $D = d_i$ and ask if these can be chosen such that

$$\langle y^{[n]}, y^{[n]} \rangle_G = \langle y^{[n-1]}, y^{[n-1]} \rangle_G + 2h \sum_{i=1}^{s} d_i \langle Y_i, F_i \rangle,$$

where $F_i = f(Y_i)$. If the general linear method is to solve a conservative problem having the property

$$\langle y, f(y) \rangle = 0,$$

this means that

$$\langle y^{[n]}, y^{[n]} \rangle_G = \langle y^{[n-1]}, y^{[n-1]} \rangle_G. \tag{4.2}$$

because the term $2h \sum_{i=1}^{s} d_i \langle Y_i, F_i \rangle$ is zero. Methods satisfying (4.2) are called *G-symplectic general linear methods*.

Theorem 4.1.1. *A general linear method (A,U,B,V) is G-symplectic, if there exist a symmetric $r \times r$ matrix G and a diagonal $s \times s$ matrix D such that*

$$\begin{aligned} G &= V^T GV, \\ DU &= B^T GV, \\ DA + A^T D &= B^T GB. \end{aligned}$$

Proof : Consider the relation

$$\langle y^{[n]}, y^{[n]} \rangle_G - \langle y^{[n-1]}, y^{[n-1]} \rangle_G - 2h \sum_{i=1}^{s} d_i \langle Y_i, F_i \rangle$$

Using the general linear method (4.1)

$$\begin{aligned} = &\sum_{i,j=1}^{r} g_{ij} \langle BhF_i + V y_i^{n-1}, BhF_j + V y_j^{n-1} \rangle \\ &- \sum_{i,j=1}^{r} g_{ij} \langle y_i^{[n-1]}, y_j^{[n-1]} \rangle \\ &- 2h \sum_{i=1}^{s} d_i \langle AhF_i + U y_i^{[n-1]}, F_i \rangle. \end{aligned}$$

Expanding the inner product and combining like terms yield

$$\begin{aligned} = &- [G - V^T GV] \langle y_i^{[n-1]}, y_j^{[n-1]} \rangle \\ &- 2[DU - B^T GV] \langle y_i^{[n-1]}, hF_j \rangle \\ &- [DA + A^T D - B^T GB] \langle hF_i, hF_j \rangle, \end{aligned}$$

and hence, for problems which satisfy

$$\langle y, f(y) \rangle = 0,$$

we get

$$\langle y^{[n]} y^{[n]} \rangle_G = \langle y^{[n-1]}, y^{[n-1]} \rangle_G,$$

provided

$$\begin{aligned} G &= V^T GV, \\ DU &= B^T GV, \\ DA + A^T D &= B^T GB. \end{aligned}$$

4.1.1 Examples

Example 1 : Consider the following general linear method

$$\left[\begin{array}{c} Y \\ \hline y_1^{[n]} \\ y_2^{[n]} \end{array}\right] = \left[\begin{array}{c|cc} 0 & 1 & 0 \\ \hline 2 & 0 & 1 \\ 0 & 1 & 0 \end{array}\right] \left[\begin{array}{c} hf(Y) \\ \hline y_1^{[n-1]} \\ y_2^{[n-1]} \end{array}\right]. \tag{4.3}$$

This is the leap-frog method written in general linear formulation. Although we cannot hope to have true conservation of quadratic invariants, i.e. we cannot get

$$\langle y_1^{[n]}, y_1^{[n]} \rangle_G = \langle y_1^{[n-1]}, y_1^{[n-1]} \rangle,$$
$$\langle y_2^{[n]}, y_2^{[n]} \rangle_G = \langle y_2^{[n-1]}, y_2^{[n-1]} \rangle,$$

we can preserve the quadratic invariants under a G-norm and by making use of Theorem 4.1.1 we can find the matrices G and D as

$$G = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \qquad D = \begin{bmatrix} 2 \end{bmatrix}.$$

We get

$$\langle y^{[n]}, y^{[n]} \rangle_G = \langle y^{[n-1]}, y^{[n-1]} \rangle,$$
$$\sum_{i,j=1}^{2} g_{ij} \langle y_i^{[n]}, y_j^{[n]} \rangle = \sum_{i,j=1}^{2} g_{ij} \langle y_i^{[n-1]}, y_j^{[n-1]} \rangle,$$
$$\langle y_1^{[n]}, y_2^{[n]} \rangle = \langle y_1^{[n-1]}, y_2^{[n-1]} \rangle.$$

Example 2 : Another example of a G-symplectic general linear method, which was presented by Butcher in [12] has coefficient matrix

$$\left[\begin{array}{cc|cc} \frac{3+\sqrt{3}}{6} & 0 & 1 & \frac{3+2\sqrt{3}}{3} \\ -\frac{\sqrt{3}}{3} & \frac{3+\sqrt{3}}{6} & 1 & -\frac{3+2\sqrt{3}}{3} \\ \hline \frac{1}{2} & \frac{1}{2} & 1 & 0 \\ -\frac{1}{2} & \frac{1}{2} & 0 & -1 \end{array}\right]. \tag{4.4}$$

This method satisfies the conditions of Theorem 4.1.1 with

$$G = \begin{bmatrix} 1 & 0 \\ 0 & \frac{3+2\sqrt{3}}{3} \end{bmatrix}, \qquad D = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix}.$$

Associated with this method is a similar method in which the sign of $\sqrt{3}$ is changed and we will present this method in section 4.3.

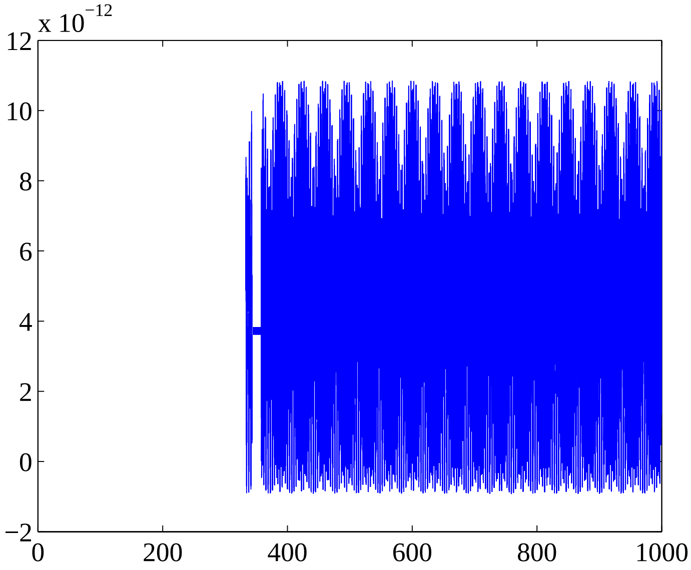


Figure 4.1: The error in energy of the simple pendulum problem with initial value of $q = 1.2$.

4.1.2 Experiment

We recall from Chapter 1, the equations of motion of the simple pendulum

$$p' = -\sin(q), \qquad q' = p.$$

The total energy H is a conserved quantity and is given as

$$H = \frac{p^2}{2} - \cos(q).$$

The general linear method (4.4) has been applied to solve the simple pendulum problem. The initial values are chosen to be $p = 0$, $q = 1.2$. The error in energy is plotted for 100,000 steps with stepsize 0.01 and is given in the Figure 4.1. The results are completely consistent with our belief that the method is G-symplectic since it is clearly conserving the total energy of the simple pendulum problem with very small errors. However if we take the initial value of q to be 2.3, we get the error in energy of the simple pendulum as plotted in the Figure 4.2. In the second case we have taken only 10,000 steps with the same stepsize of 0.01 but obtain large error in energy. Although the method (4.4) is G-symplectic and is supposed to conserve the energy for all initial values, we see a large energy error and this is due to the corruption by the parasitic solutions.

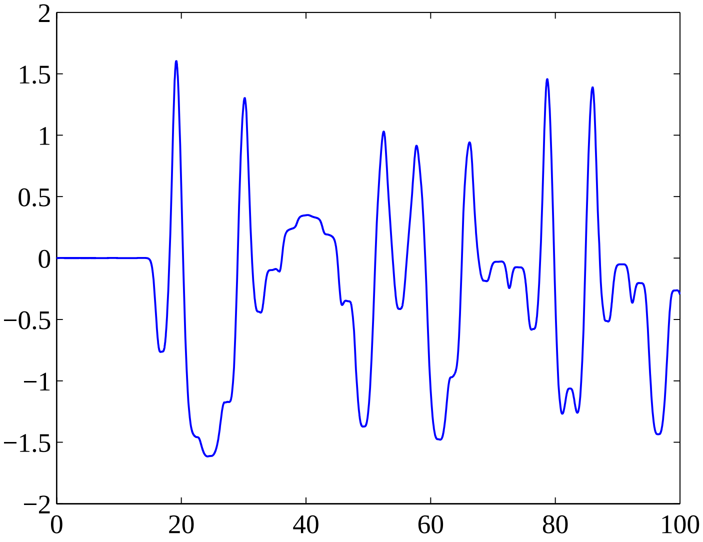


Figure 4.2: The error in energy of the simple pendulum problem with initial value of $q = 2.3$.

4.2 Parasitic solutions

It is typical of multivalue methods to suffer from parasitic solutions. Parasitic solutions are those numerical solutions which are obtained in addition to the numerical approximation of the exact solution. We have already encountered them in section 3.4.3 in relation to multistep methods. An analysis is done to study the parasitic solutions using general linear methods having structure

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \hline y_1^{[n]} \\ y_2^{[n]} \end{bmatrix} = \left[\begin{array}{cc|cc} a_{11} & a_{11} & u_{11} & u_{12} \\ a_{21} & a_{22} & u_{21} & u_{22} \\ \hline b_{11} & b_{12} & 1 & 0 \\ b_{21} & b_{22} & 0 & -1 \end{array}\right] \begin{bmatrix} hF_1 \\ hF_2 \\ \hline y_1^{[n-1]} \\ y_2^{[n-1]} \end{bmatrix},$$

where $F_i = f(Y_i)$. Note that V has eigenvalues 1 and -1. The stages and output values are

$$\begin{aligned} Y_i &= h\sum_{j=1}^{2} a_{ij}F_j + u_{i1}y_1^{[n-1]} + u_{i2}y_2^{[n-1]}, \qquad\qquad i = 1,2, \\ y_1^{[n]} &= h\sum_{i=1}^{2} b_{1i}F_i + y_1^{[n-1]}, \\ y_2^{[n]} &= h\sum_{i=1}^{2} b_{2i}F_i - y_2^{[n-1]}. \end{aligned}$$

The first component $y_1^{[n]}$ approximates the exact solution and the second component $y_2^{[n]}$ approximates a related quantity e.g. scaled second derivative in the case of (4.4) . The second component $y_2^{[n]}$ is the parasitic solution. We perturb the second component and study the rate of growth of the parasitic solution at the start of step n

$$y_2^{[n-1]} \longrightarrow y_2^{[n-1]} + (-1)^{n-1}z_{n-1}.$$

This perturbation will affect the stages Y_i approximately as follows

$$\begin{aligned} Y_i + \delta Y_i &= h\sum_{j=1}^{2} a_{ij}F_j + u_{i1}y_1^{[n-1]} + u_{i2}(y_2^{[n-1]} + (-1)^{n-1}z_{n-1}), \\ \Rightarrow \delta Y_i &= (-1)^{n-1}u_{i2}z_{n-1}. \end{aligned}$$

The stage derivatives F_i will be perturbed approximately as

$$\begin{aligned} F_i + \delta F_i &= f(Y_i + \delta Y_i) \\ &= f(Y_i) + \delta Y_i \frac{\partial f}{\partial y}, \\ \Rightarrow \delta F_i &= \delta Y_i \frac{\partial f}{\partial y} \\ &= (-1)^{n-1}\frac{\partial f}{\partial y}u_{i2}z_{n-1}. \end{aligned}$$

Combining these equations, we see that the perturbation in the second output value is

$$\begin{aligned} y_2^{[n]} + (-1)^n z_n &= h\sum_{i=1}^{2} b_{2i}(F_i + \delta F_i) - (y_2^{[n-1]} + (-1)^{n-1}z_{n-1}) \\ &= h\sum_{i=1}^{2} b_{2i}F_i - y_2^{[n-1]} + (-1)^{n-1}h\sum_{i=1}^{2}\frac{\partial f}{\partial y}b_{2i}u_{i2}z_{n-1} - (-1)^{n-1}z_{n-1}, \\ \Rightarrow z_n &= \big(1 - h\sum_{i=1}^{2}\frac{\partial f}{\partial y}b_{2i}u_{i2}\big)z_{n-1}. \end{aligned}$$

This represents Euler method solving the differential equation

$$z' = \mu \frac{\partial f}{\partial y} z, \tag{4.5}$$

where $\mu = -\sum_{i=1}^{2} b_{2i} u_{i2}$ is responsible for the growth of parasitic solution and hence termed as parasitic growth parameter. The term μ can be found from the matrix product

$$BU = \begin{bmatrix} 1 & 0 \\ 0 & -\mu \end{bmatrix}.$$

It appears that $\mu = 0$ is required to have parasitic free solutions for particular problems. We will consider parasitic growth parameter for the methods in which $r \geq 3$ in section 4.5.

4.3 Construction of two stages, two input value G-symplectic methods

We discuss the construction of a two stage, two input value G-symplectic general linear method which was presented by Butcher in [12] and is given in (4.4). We will observe that the parasitic growth is intrinsic to such methods. We will construct parasitic free methods with $s, r > 2$ values later in this chapter. The general linear method is desired to be G-symplectic and symmetric having the following structure

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{cc|cc} a_{11} & 0 & 1 & u_{12} \\ a_{21} & a_{22} & 1 & u_{22} \\ \hline b_{11} & b_{12} & v_{11} & v_{12} \\ b_{21} & b_{22} & v_{21} & v_{22} \end{array}\right],$$

with,

$$G = \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix}, \qquad D = \begin{bmatrix} d_1 & 0 \\ 0 & d_2 \end{bmatrix}.$$

The matrix A is lower triangular for cheap implementation. The structure of the matrix U is such that the first column is a vector of ones because during the calculation of the stages, the first column of the matrix U is multiplied with the input value representing the actual solution and we want stages to at least approximate the actual solution exactly. Consider the G-symplectic condition

$$G = V^T G V,$$

$$\begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix} = \begin{bmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix} \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix} \begin{bmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix},$$

$$\begin{aligned} g_{11} &= v_{11}^2 g_{11} + 2v_{21} g_{12} v_{11} + v_{21}^2 g_{22}, \\ g_{12} &= v_{11} g_{11} v_{12} + v_{12} g_{21} v_{21} + v_{11} g_{12} v_{22} + v_{21} g_{22} v_{22}, \\ g_{22} &= v_{12}^2 g_{11} + 2v_{12} g_{12} v_{22} + v_{22}^2 g_{22}. \end{aligned}$$

By comparing both sides of the equations we get

$$G = \begin{bmatrix} 1 & 0 \\ 0 & g \end{bmatrix}, \qquad V = \begin{bmatrix} 1 & 0 \\ 0 & v \end{bmatrix}.$$

where v can either be $+1$ or -1. Let us consider the case where

$$V = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

Since we are trying to construct methods which are symmetric, we will see how the symmetric properties of a method shape the stages and output values.

$$\begin{aligned} Y_1 &= ha_{11}F_1 + y_1^{[0]} + u_{12} y_2^{[0]}, \\ Y_2 &= ha_{21}F_1 + ha_{22}F_2 + y_1^{[0]} + u_{22} y_2^{[0]}, \\ y_1^{[1]} &= hb_{11}F_1 + hb_{12}F_2 + y_1^{[0]}, \\ y_2^{[1]} &= hb_{21}F_1 + hb_{22}F_2 - y_2^{[0]}. \end{aligned}$$

Here the first stage Y_1 is calculated at c_1 and the second stage Y_2 is calculated at c_2. We refer to the c values as stage abscissas. Now if we take a step backward with stepsize $-h$, then the first stage Y_1 will be calculated at c_2 and second stage Y_2 will be calculated at c_1. The input values will now be $[y_1^{[1]}, y_2^{[1]}]$ and the output is

$$\begin{aligned} y_1^{[0]} &= -hb_{11}F_2 - hb_{12}F_1 + y_1^{[1]}, \\ y_2^{[0]} &= -hb_{21}F_2 - hb_{22}F_1 - y_2^{[1]}. \end{aligned}$$

This implies,

$$\begin{aligned} y_1^{[1]} &= +hb_{11}F_2 + hb_{12}F_1 + y_1^{[0]}, \\ y_2^{[1]} &= -hb_{21}F_2 - hb_{22}F_1 - y_2^{[0]}. \end{aligned}$$

For a method to be symmetric, the output of the adjoint method with stepsize $-h$ should be equal to the input of the actual method with stepsize h and comparing them yields

$$b_{11} = b_{12}, \qquad b_{22} = -b_{21}.$$

Now consider the G-symplectic condition

$$\begin{gathered} B^T GV = DU, \\ \begin{bmatrix} b_{11} & -gb_{21} \\ b_{11} & gb_{21} \end{bmatrix} = \begin{bmatrix} d_1 & d_1 u_{12} \\ d_2 & d_2 u_{22} \end{bmatrix}. \end{gathered}$$

By comparing we get,

$$D = \begin{bmatrix} b_{11} & 0 \\ 0 & b_{11} \end{bmatrix}, \qquad U = \begin{bmatrix} 1 & -g\frac{b_{21}}{b_{11}} \\ 1 & g\frac{b_{21}}{b_{11}} \end{bmatrix}.$$

Let $x = b_{21}/b_{11}$, then the structure of the general linear matrix becomes

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{cc|cc} a_{11} & 0 & 1 & -gx \\ a_{21} & a_{22} & 1 & gx \\ \hline b_{11} & b_{11} & 1 & 0 \\ b_{11}x & -b_{11}x & 0 & -1 \end{array}\right],$$

with,

$$G = \begin{bmatrix} 1 & 0 \\ 0 & g \end{bmatrix}, \qquad D = \begin{bmatrix} b_{11} & 0 \\ 0 & b_{11} \end{bmatrix}.$$

Now consider the matrix product

$$\begin{aligned} BU &= \begin{bmatrix} b_{11} & b_{11} \\ b_{11}x & -b_{11}x \end{bmatrix} \begin{bmatrix} 1 & -gx \\ 1 & gx \end{bmatrix} \\ &= \begin{bmatrix} 2b_{11} & 0 \\ 0 & -2gb_{11}x^2 \end{bmatrix}. \end{aligned}$$

To avoid parasitism we must have

$$-2gb_{11}x^2 = 0.$$

Since, $g \neq 0$, $b_{11} \neq 0$, so the only option is $x^2 = 0$, which is not possible because this makes the second input component redundant and the general linear method reduces to a Runge–Kutta method. Hence it is not possible to have parasitic free general linear methods with two stages and two input values. The matrix B can be written as

$$\begin{aligned} B &= \begin{bmatrix} b_{11} & b_{11} \\ b_{11}x & -b_{11}x \end{bmatrix} \\ &= \begin{bmatrix} 1 & 1 \\ x & -x \end{bmatrix} \begin{bmatrix} b_{11} & 0 \\ 0 & b_{11} \end{bmatrix} \\ &= XD. \end{aligned}$$

Now consider the G-symplectic condition

$$\begin{aligned} DA + A^T D &= B^T G B, \\ \begin{bmatrix} 2a_{11} & a_{21} \\ a_{21} & 2a_{22} \end{bmatrix} &= X^T G X D, \\ \begin{bmatrix} 2a_{11} & a_{21} \\ a_{21} & 2a_{22} \end{bmatrix} &= \begin{bmatrix} 1 & x \\ 1 & -x \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & g \end{bmatrix} \begin{bmatrix} 1 & 1 \\ x & -x \end{bmatrix} \begin{bmatrix} b_{11} & 0 \\ 0 & b_{11} \end{bmatrix} \\ &= \begin{bmatrix} b_{11}(1+gx^2) & b_{11}(1-gx^2) \\ b_{11}(1-gx^2) & b_{11}(1+gx^2) \end{bmatrix}. \end{aligned}$$

This implies,

$$\begin{aligned} a_{11} &= \frac{b_{11}}{2}(1+gx^2), \\ a_{21} &= b_{11}(1-gx^2), \\ a_{22} &= a_{11}. \end{aligned}$$

We are trying to construct a method of order 4 and the method satisfies the relative order conditions. The first order condition is

$$\begin{aligned} b_{11} + b_{11} &= 1, \\ \Rightarrow b_{11} &= \tfrac{1}{2}. \end{aligned}$$

The second order condition is

$$\begin{aligned} b_{11}(c_1 + c_2) &= \tfrac{1}{2}, \\ \Rightarrow c_1 &= 1 - c_2. \end{aligned}$$

The third order condition is

$$\begin{aligned} b_{11}(c_1^2 + c_2^2) &= \tfrac{1}{3}, \\ c_1^2 - c_1 + \tfrac{1}{3} &= 0, \\ \Rightarrow c_1 &= \tfrac{3 \pm \sqrt{3}}{6}. \end{aligned}$$

Let us take $c_1 = \frac{3+\sqrt{3}}{6}$. Since $a_{11} = c_1$, we get,

$$gx^2 = \frac{3+2\sqrt{3}}{3}.$$

For convenience we choose $x = 1$, so that

$$b_{11} = b_{21} = \frac{1}{2}.$$

The general linear method is

$$P: \qquad \left[\begin{array}{cc|cc} \frac{3+\sqrt{3}}{6} & 0 & 1 & -\frac{3+2\sqrt{3}}{3} \\ -\frac{\sqrt{3}}{3} & \frac{3+\sqrt{3}}{6} & 1 & \frac{3+2\sqrt{3}}{3} \\ \hline \frac{1}{2} & \frac{1}{2} & 1 & 0 \\ \frac{1}{2} & -\frac{1}{2} & 0 & -1 \end{array}\right], \tag{4.6}$$

$$G = \begin{bmatrix} 1 & 0 \\ 0 & \frac{3+2\sqrt{3}}{3} \end{bmatrix}, \qquad D = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix}.$$

The general linear method (4.6) suffers from parasitic solution. The parasitic growth parameter, which is the $(2,2)$ component of the matrix product BU is given as

$$\mu_1 = 1 + \frac{2}{\sqrt{3}}.$$

The general linear method (4.6) requires a starting method and one such starting method was presented by Butcher in [11] to ensure that the method has order 4. The starting method is

$$\left[\begin{array}{cc|c} \frac{3+\sqrt{3}}{6} & 0 & 1 \\ -\frac{3+\sqrt{3}}{3} & \frac{3+\sqrt{3}}{6} & 1 \\ \hline 0 & 0 & 1 \\ \frac{\sqrt{3}-1}{8} & -\frac{\sqrt{3}-1}{8} & 0 \end{array}\right]. \tag{4.7}$$

If we take $c_1 = \frac{3-\sqrt{3}}{6}$, we get the following method

$$N: \qquad \left[\begin{array}{cc|cc} \frac{3-\sqrt{3}}{6} & 0 & 1 & -\frac{3-2\sqrt{3}}{3} \\ \frac{\sqrt{3}}{3} & \frac{3-\sqrt{3}}{6} & 1 & \frac{3-2\sqrt{3}}{3} \\ \hline \frac{1}{2} & \frac{1}{2} & 1 & 0 \\ \frac{1}{2} & -\frac{1}{2} & 0 & -1 \end{array}\right], \tag{4.8}$$

$$G = \begin{bmatrix} 1 & 0 \\ 0 & \frac{3-2\sqrt{3}}{3} \end{bmatrix}, \qquad D = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix}.$$

The general linear method (4.8) also suffers from parasitic solution and the parasitic growth parameter is given as

$$\mu_2 = 1 - \tfrac{2}{\sqrt{3}}.$$

The starting method for the general linear method (4.8) is

$$\left[\begin{array}{cc|c} \frac{3-\sqrt{3}}{6} & 0 & 1 \\ -\frac{3-\sqrt{3}}{3} & \frac{3-\sqrt{3}}{6} & 1 \\ \hline 0 & 0 & 1 \\ -\frac{\sqrt{3}+1}{8} & \frac{\sqrt{3}+1}{8} & 0 \end{array}\right]. \tag{4.9}$$

4.4 Avoiding parasitism via composition

The parasitic corruption of the numerical solution by the general linear methods can be controlled by the composition of general linear methods. We consider two G-symplectic general linear methods P and N given by (4.6) and (4.8) with a slight change in method P by changing the signs in the second row of matrix B and second column of matrix U. This is done to ensure that the starting method for both of these methods is same which is the method (4.9).

The general linear methods P and N are implemented in a sequence in such a way that the cumulative value of their parasitic growth parameters does not lie outside the interval $[-\frac{2}{\sqrt{3}}, \frac{2}{\sqrt{3}}]$. This is achievable since the parasitic growth parameters of the two different general linear methods add up when the two methods are used in composition. This fact was pointed out in [32]. However it depends on the magnitude of the parasitic growth parameters as to which sequence of methods is used. For the methods (4.6) and (4.8), a sequence suggested by Butcher is very effective,

$$N^7\ P\ N^{14}\ P\ N^{14}\ P\ N^{14}\ P\ N^{14}\ P\ N^{14}\ P\ N^{14}\ P\ N^{13}\ P \cdots$$

This sequence is shown in Figure 4.3. It simply says that we start the numerical integration by first taking seven steps of method N. In doing so, the parasitic growth parameter μ_2 of method N adds up to a total amount which is slightly below -1. Afterwards, we take one step of method P and the parasitic growth parameter μ_1 will add to already accumulated values of μ_2. The sequence is maintained such that the parasitic growth parameter

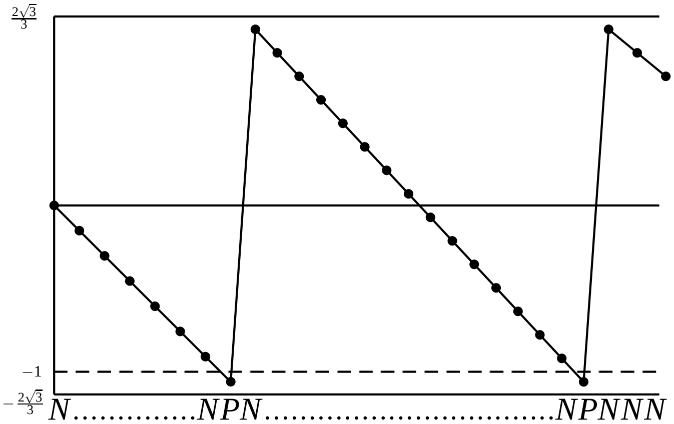


Figure 4.3: Composition of methods P and N.

does not go out of the interval $[-\frac{2}{\sqrt{3}}, \frac{2}{\sqrt{3}}]$. Note that at step number 112, we have used method P instead of N to compensate for the fact that the ratio of $|\mu_1|$ and $|\mu_2|$ is slightly less than 14. The pendulum problem is solved with this sequence of methods and with the initial conditions of $p = 0$ and $q = 2.3$ to study the effect of parasitism. However, the effect of parasitism is controlled by the sequence of methods and the error in the energy of the simple pendulum problem is presented in Figure 4.4. Here we have taken $100,000$ steps with stepsize 0.01.

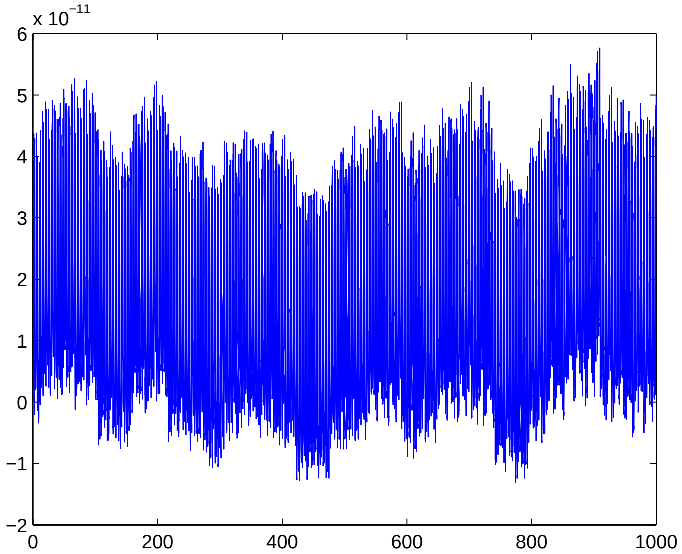


Figure 4.4: The error in energy of the simple pendulum problem with initial values of $p = 0, q = 2.3$.

4.5 Parasitic free G-symplectic methods

Although it is not possible for two stage, two input value general linear methods to avoid parasitism, it is possible to construct methods with more stages and input values, which do not suffer from parasitic solutions.

4.5.1 Four stages, three input value method

A G-symplectic general linear method is constructed with four stages $(s = 4)$ and three input values $(r = 3)$. This method is further required to be symmetric and having no parasitism. We will assume that the structure of the method is

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{cccc|ccc} a_{11} & 0 & 0 & 0 & 1 & u_{12} & u_{13} \\ a_{21} & a_{22} & 0 & 0 & 1 & u_{22} & u_{23} \\ a_{31} & a_{32} & a_{33} & 0 & 1 & u_{32} & u_{33} \\ a_{41} & a_{42} & a_{43} & a_{44} & 1 & u_{42} & u_{43} \\ \hline b_{11} & b_{12} & b_{13} & b_{14} & 1 & 0 & 0 \\ b_{21} & b_{22} & b_{23} & b_{24} & 0 & z & 0 \\ \overline{b}_{21} & \overline{b}_{22} & \overline{b}_{23} & \overline{b}_{24} & 0 & 0 & \overline{z} \end{array}\right],$$

with,

$$G = \begin{bmatrix} 1 & 0 & 0 \\ 0 & g & 0 \\ 0 & 0 & g \end{bmatrix}, \qquad D = \begin{bmatrix} d_1 & 0 & 0 & 0 \\ 0 & d_2 & 0 & 0 \\ 0 & 0 & d_3 & 0 \\ 0 & 0 & 0 & d_4 \end{bmatrix}.$$

Note that A has been chosen to be lower triangular for cheap implementation. The structure of the matrix U is such that the first column is a vector of ones because during the calculation of the stages, the first column of the matrix U is multiplied with the input value representing the actual solution and we want stages to at least approximate the actual solution exactly. This is to make sure that for the pre-consistency vector u, we have

$$Uu = \mathbf{1}, \qquad Vu = u.$$

The second and third row of the matrix B are complex conjugates, so does the second and third row of the matrix V. The complex numbers z and $\overline{z}$ in the matrix V are chosen such that $|z| = 1$, but not $+1$ or -1 and we have taken them to be the cube roots of unity

$$z = e^{\frac{2\pi i}{3}}, \qquad\qquad \overline{z} = e^{\frac{-2\pi i}{3}}.$$

Complex numbers are used here keeping in mind that they can easily be converted to real numbers via a transformation aiming at a careful construction of U. Since we are constructing a symmetric method, to see the effects of time reversal symmetry, we consider the general linear method in the form

$$\begin{aligned}
Y_1 &= ha_{11}F_1 + y_1^{[0]} + u_{12}y_2^{[0]} + u_{13}y_3^{[0]},\\
Y_2 &= ha_{21}F_1 + ha_{22}F_2 + y_1^{[0]} + u_{22}y_2^{[0]} + u_{23}y_3^{[0]},\\
Y_3 &= ha_{31}F_1 + ha_{32}F_2 + ha_{33}F_3 + y_1^{[0]} + u_{32}y_2^{[0]} + u_{33}y_3^{[0]},\\
Y_4 &= ha_{41}F_1 + ha_{42}F_2 + ha_{43}F_3 + ha_{44}F_4 + y_1^{[0]} + u_{42}y_2^{[0]} + u_{43}y_3^{[0]},
\end{aligned}$$

$$\begin{aligned}
y_1^{[1]} &= hb_{11}F_1 + hb_{12}F_2 + hb_{13}F_3 + hb_{14}F_4 + y_1^{[0]},\\
y_2^{[1]} &= hb_{21}F_1 + hb_{22}F_2 + hb_{23}F_3 + hb_{24}F_4 + zy_2^{[0]},\\
y_3^{[1]} &= h\overline{b}_{21}F_1 + h\overline{b}_{22}F_2 + h\overline{b}_{23}F_3 + h\overline{b}_{24}F_4 + \overline{z}y_3^{[0]}.
\end{aligned}$$

where, $F_i = f(Y_i)$. If the method has time reversal symmetry, then we can take a step backward with stepsize $-h$ and we should get the output values same as the input values provided to the original method with step size h. The stages are such that

$$\begin{bmatrix} \widetilde{Y}_1 \\ \widetilde{Y}_2 \\ \widetilde{Y}_3 \\ \widetilde{Y}_4 \end{bmatrix} = P \begin{bmatrix} Y_1 \\ Y_2 \\ Y_3 \\ Y_4 \end{bmatrix},$$

where

$$P = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

is an $s \times s$ permutation matrix as given in (2.3). The first stage of the adjoint method with stepsize $-h$ is

$$\begin{aligned}
\widetilde{Y}_1 &= -ha_{11}F_4 + y_1^{[1]} + u_{12}y_2^{[1]} + u_{13}y_3^{[1]}\\
&= h(b_{11} + u_{12}b_{21} + u_{13}b_{31})F_1 + h(b_{12} + u_{12}b_{22} + u_{13}b_{32})F_2\\
&\quad + h(b_{13} + u_{12}b_{23} + u_{13}b_{33})F_3 + h(-a_{11} + b_{14} + u_{12}b_{24} + u_{13}b_{34})F_4\\
&\quad + y_1^{[0]} + u_{12}zy_2^{[0]} + u_{13}\overline{z}y_3^{[0]}.
\end{aligned}$$

Similarly the fourth stage of the adjoint method is

$$
\begin{aligned}
\widetilde{Y}_4 &= -ha_{41}F_4 - ha_{42}F_3 - ha_{43}F_2 - ha_{44}F_1 + y_1^{[1]} + u_{42}y_2^{[1]} + u_{43}y_3^{[1]} \\
&= h(-a_{44} + b_{11} + u_{42}b_{21} + u_{43}b_{31})F_1 + h(-a_{43} + b_{12} + u_{42}b_{22} + u_{43}b_{32})F_2 \\
&\quad + h(-a_{42} + b_{13} + u_{42}b_{23} + u_{43}b_{33})F_3 + h(-a_{41} + b_{14} + u_{42}b_{24} + u_{43}b_{34})F_4 \\
&\quad + y_1^{[0]} + u_{42}zy_2^{[0]} + u_{43}\bar{z}y_3^{[0]}.
\end{aligned}
$$

Comparing Y_1 with $\widetilde{Y}_4$ and Y_4 with $\widetilde{Y}_1$, we get

$$u_{43} = -u_{13}, \qquad u_{42} = -u_{12}. \tag{4.10}$$

Similarly it can be shown that

$$u_{33} = -u_{23}, \qquad u_{32} = -u_{22}. \tag{4.11}$$

This is possible only if the method satisfies all the conditions of time reversal symmetry given in (2.18). The G-symplectic conditions for a general linear method with complex entries is

$$
\begin{aligned}
G &= V^*GV, \\
DU &= B^*GV, \\
DA + A^*D &= B^*GB.
\end{aligned}
$$

where $*$ represents conjugate transpose. Now consider the condition

$$B^*GV = DU,$$

$$
\begin{bmatrix}
b_{11} & gz\bar{b}_{21} & g\bar{z}b_{21} \\
b_{12} & gz\bar{b}_{22} & g\bar{z}b_{22} \\
b_{12} & gz\bar{b}_{23} & g\bar{z}b_{23} \\
b_{11} & gz\bar{b}_{24} & g\bar{z}b_{24}
\end{bmatrix}
=
\begin{bmatrix}
d_1 & d_1u_{12} & d_1u_{13} \\
d_2 & d_2u_{22} & d_2u_{23} \\
d_3 & -d_3u_{22} & -d_3u_{23} \\
d_4 & -d_4u_{12} & -d_3u_{13}
\end{bmatrix}.
$$

Comparing the terms in the two matrices and using the equations (4.10) and (4.11), we get the following structure of the matrices D, B and U.

$$
D = \begin{bmatrix}
b_{11} & 0 & 0 & 0 \\
0 & b_{12} & 0 & 0 \\
0 & 0 & b_{12} & 0 \\
0 & 0 & 0 & b_{11}
\end{bmatrix},
$$

$$
B = \begin{bmatrix}
b_{11} & b_{12} & b_{12} & b_{11} \\
b_{21} & b_{22} & -b_{22} & -b_{21} \\
\bar{b}_{21} & \bar{b}_{22} & -\bar{b}_{22} & -\bar{b}_{21}
\end{bmatrix}.
$$

Let

$$\begin{aligned} B_1 &= \frac{b_{21}}{b_{11}}, \\ B_2 &= \frac{b_{22}}{b_{12}}. \end{aligned}$$

This implies,

$$\begin{aligned} B &= \begin{bmatrix} b_{11} & b_{12} & b_{12} & b_{11} \\ b_{11}B_1 & b_{12}B_2 & -b_{12}B_2 & -b_{11}B_1 \\ \overline{b_{11}B_1} & \overline{b_{12}B_2} & -\overline{b_{12}B_2} & -\overline{b_{11}B_1} \end{bmatrix} \\ &= \begin{bmatrix} 1 & 1 & 1 & 1 \\ B_1 & B_2 & -B_2 & -B_1 \\ \bar{B}_1 & \bar{B}_2 & -\bar{B}_2 & -\bar{B}_1 \end{bmatrix} \begin{bmatrix} b_{11} & 0 & 0 & 0 \\ 0 & b_{12} & 0 & 0 \\ 0 & 0 & b_{12} & 0 \\ 0 & 0 & 0 & b_{11} \end{bmatrix} \\ &= XD, \end{aligned}$$

and

$$U = \begin{bmatrix} 1 & gz\overline{B_1} & g\bar{z}B_1 \\ 1 & gz\overline{B_2} & g\bar{z}B_2 \\ 1 & -gz\overline{B_2} & -g\bar{z}B_2 \\ 1 & -gz\overline{B_1} & -g\bar{z}B_1 \end{bmatrix}.$$

Now consider the condition

$$DA + A^*D = B^*GB.$$

The lower triangular part $L(.)$ of the matrix on the left hand side is equal to the lower triangular part of the matrix on the right hand side.

$$L(DA + A^*D) = L(B^*GB),$$

where $L(.)$ is a linear operator and

$$\begin{aligned} L(DA + A^*D) &= \begin{bmatrix} 2a_{11}b_{11} & 0 & 0 & 0 \\ a_{21}b_{12} & 2a_{22}b_{12} & 0 & 0 \\ a_{31}b_{12} & a_{32}b_{12} & 2a_{33}b_{12} & 0 \\ a_{41}b_{11} & a_{42}b_{11} & a_{43}b_{11} & 2a_{44}b_{11} \end{bmatrix} \\ &= \begin{bmatrix} b_{11} & 0 & 0 & 0 \\ 0 & b_{12} & 0 & 0 \\ 0 & 0 & b_{12} & 0 \\ 0 & 0 & 0 & b_{11} \end{bmatrix} \begin{bmatrix} 2a_{11} & 0 & 0 & 0 \\ a_{21} & 2a_{22} & 0 & 0 \\ a_{31} & a_{32} & 2a_{33} & 0 \\ a_{41} & a_{42} & a_{43} & 2a_{44} \end{bmatrix} \\ &= DY. \end{aligned}$$

Thus

$$\begin{aligned} DY &= L(B^*GB) = L(DX^*GXD), \\ Y &= L(X^*GXD). \end{aligned} \tag{4.12}$$

From the relation (4.12) we get

$$\begin{aligned}
a_{11} &= \frac{b_{11}}{2}(1+2g|B_1|^2), \\
a_{21} &= b_{11}(1+2g\text{Re}(B_1\overline{B}_2)), \\
a_{22} &= \frac{b_{12}}{2}(1+2g|B_2|^2), \\
a_{31} &= b_{11}(1-2g\text{Re}(B_1\overline{B}_2)), \\
a_{32} &= b_{12}(1-2g|B_2|^2), \\
a_{33} &= a_{22}, \\
a_{41} &= b_{11}(1-2g|B_1|^2), \\
a_{42} &= b_{12}(1-2g\text{Re}(B_1\overline{B}_2), \\
a_{43} &= b_{12}(1+2g\text{Re}(B_1\overline{B}_2), \\
a_{44} &= a_{11}.
\end{aligned}$$

where $\text{Re}(.)$ represents the real part of a complex number. Due to symmetry

$$c_3 = 1 - c_2, \qquad\qquad c_4 = 1 - c_1.$$

Let us suppose that $c_1 = 0$ and $B_1 = 1$. This implies

$$\begin{aligned} a_{11} &= 0, \\ \Rightarrow g &= -\tfrac{1}{2}. \end{aligned}$$

Moreover

$$\begin{aligned} a_{11} + a_{22} &= \tfrac{1}{4}, \\ \Rightarrow a_{22} &= \tfrac{1}{4}. \end{aligned}$$

We are trying to construct a method of order 4 and the method satisfies the relative order conditions. The first order condition is

$$b_{11} + b_{12} = \tfrac{1}{2}. \tag{4.13}$$

The third order condition is

$$b_{11}(c_1^2 - c_1 + \tfrac{1}{6}) + b_{12}(c_2^2 - c_2 + \tfrac{1}{6}) = 0.$$

Since $c_1 = 0$, we get

$$2r^2 - 3s^2 + 1 = 0, \tag{4.14}$$

where

$$r^2 = -\frac{b_{11}}{b_{12}}, \tag{4.15}$$
$$s = (2c_2^2 - 1)^2.$$

All solutions of the Diophantine equation (4.14) are given by

$$s = \frac{6t^2 + 4t + 1}{6t^2 - 1}, \tag{4.16}$$

$$r = \frac{6t^2 + 6t + 1}{6t^2 - 1}. \tag{4.17}$$

From the equation (4.13) and (4.15), we get

$$b_{11} = -\frac{r^2}{2(1 - r^2)}, \qquad b_{12} = \frac{1}{2(1 - r^2)}.$$

We are constructing a method without parasitism and this is possible if $(2,2)$ and $(3,3)$ component of the matrix product BU is zero. This implies

$$b_{11}|B_1|^2 + b_{12}|B_2|^2 = 0.$$

Since we have assumed $B_1 = 1$, we get

$$|B_2|^2 = r^2.$$

Before proceeding any further, let us convert the complex numbers into real numbers, using the transformation

$$T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{i}{2} \\ 0 & \frac{1}{2} & -\frac{i}{2} \end{bmatrix}. \tag{4.18}$$

The coefficient matrices of the general linear method will transform as

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] \longrightarrow \left[\begin{array}{c|c} A & UT \\ \hline T^{-1}B & T^{-1}VT \end{array}\right],$$

with

$$G \longrightarrow T^{*}GT,$$

where T^* is the conjugate transpose of T such that

$$T^* = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & -i & i \end{bmatrix}.$$

The general linear method thus become

$$\left[\begin{array}{cccc|ccc} a_{11} & 0 & 0 & 0 & 1 & g\mathrm{Re}(z\overline{B_1}) & -g\mathrm{Im}(z\overline{B_1}) \\ a_{21} & a_{22} & 0 & 0 & 1 & g\mathrm{Re}(z\overline{B_2}) & -g\mathrm{Im}(z\overline{B_2}) \\ a_{31} & a_{32} & a_{33} & 0 & 1 & -g\mathrm{Re}(z\overline{B_2}) & g\mathrm{Im}(z\overline{B_2}) \\ a_{41} & a_{42} & a_{43} & a_{44} & 1 & -g\mathrm{Re}(z\overline{B_1}) & g\mathrm{Im}(z\overline{B_1}) \\ \hline b_{11} & b_{12} & b_{12} & b_{11} & 1 & 0 & 0 \\ 2b_{11}\mathrm{Re}(B_1) & 2b_{12}\mathrm{Re}(B_2) & -2b_{12}\mathrm{Re}(B_2) & -2b_{11}\mathrm{Re}(B_1) & 0 & \mathrm{Re}(z) & -\mathrm{Im}(z) \\ 2b_{11}\mathrm{Im}(B_1) & 2b_{12}\mathrm{Im}(B_2) & -2b_{12}\mathrm{Im}(B_2) & -2b_{11}\mathrm{Im}(B_1) & 0 & \mathrm{Im}(z) & \mathrm{Re}(z) \end{array}\right],$$

with

$$G = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -\frac{1}{4} & 0 \\ 0 & 0 & -\frac{1}{4} \end{bmatrix}, \qquad D = \begin{bmatrix} b_{11} & 0 & 0 & 0 \\ 0 & b_{12} & 0 & 0 \\ 0 & 0 & b_{12} & 0 \\ 0 & 0 & 0 & b_{11} \end{bmatrix}.$$

where $\mathrm{Re}(.)$ represents the real part of a complex number and $\mathrm{Im}(.)$ represents the imaginary part and

$$\begin{aligned} \mathrm{Re}(z\overline{B_2}) &= \mathrm{Re}(z)\mathrm{Re}(\overline{B_2}) + \mathrm{Im}(z)\mathrm{Im}(\overline{B_2}), \\ \mathrm{Im}(z\overline{B_2}) &= \mathrm{Re}(z)\mathrm{Im}(\overline{B_2}) - \mathrm{Im}(z)\mathrm{Re}(\overline{B_2}). \end{aligned}$$

Since

$$B_1 = 1 \qquad \Rightarrow \qquad \mathrm{Re}(B_1) = 1, \qquad \mathrm{Im}(B_1) = 0.$$

Also

$$|B_2|^2 = r^2 \qquad \Rightarrow \qquad \mathrm{Im}(B_2) = \sqrt{r^2 - \mathrm{Re}(B_2)^2},$$

where $\mathrm{Re}(B_2)$ is calculated as follows. Consider the equation

$$\begin{aligned} a_{21} &= b_{11}(1 + 2g\mathrm{Re}(B_1\overline{B_2})) \\ &= b_{11}(1 - \mathrm{Re}(B_2)). \end{aligned} \tag{4.19}$$

and the fact that

$$\begin{aligned} c_2 &= a_{21} + a_{22}, \\ a_{21} &= c_2 - \tfrac{1}{4}. \end{aligned} \tag{4.20}$$

From (4.19) and (4.20), we get

$$\begin{aligned} \mathrm{Re}(B_2) &= 1 - \frac{c_2 - \frac{1}{4}}{b_{11}} \\ &= 1 - \frac{(4c_2-1)(2(1-r^2))}{4r^2} \\ &= 1 - \frac{(2(2c_2-1)+1)(2(1-r^2))}{4r^2} \\ &= 1 - \frac{(2s+1)(2(1-r^2))}{4r^2}. \end{aligned}$$

Everything depends on parameter t in (4.16) and (4.17). Numerical searches have found several acceptable values of t provided

1. $\mathrm{Re}(B_2)$ satisfies,

$$-|B_2| < \mathrm{Re}(B_2) < |B_2|.$$

2. The method has a stability order 4, i.e.

$$\phi(exp(z),z) = O(z^5),$$

where

$$\phi(w,z) = det(wI - M(z)),$$

and $M(z)$ is the stability matrix given by

$$\begin{aligned} M(z) &= V + zB(I-zA)^{-1}U \\ &= V + zBU + z^2BAU + z^3BA^2U. \end{aligned}$$

A choice of $t = -\frac{1}{7}$ leads to the following general linear method

$$\left[\begin{array}{cccc|ccc}
0 & 0 & 0 & 0 & 1 & \frac{1}{4} & \frac{\sqrt{3}}{4} \\
-\frac{11}{72} & \frac{1}{4} & 0 & 0 & 1 & -\frac{1973}{29068}+\frac{2\sqrt{3}\sqrt{14595}}{7267} & -\frac{1973\sqrt{3}}{29068}-\frac{2\sqrt{14595}}{7267} \\
-\frac{2647}{72240} & \frac{1009}{1680} & \frac{1}{4} & 0 & 1 & \frac{1973}{29068}-\frac{2\sqrt{3}\sqrt{14595}}{7267} & \frac{1973\sqrt{3}}{29068}+\frac{2\sqrt{14595}}{7267} \\
-\frac{169}{1680} & \frac{113821}{283920} & \frac{473}{676} & 0 & 1 & -\frac{1}{4} & -\frac{\sqrt{3}}{4} \\
\hline
-\frac{169}{3360} & \frac{1849}{3360} & \frac{1849}{3360} & -\frac{169}{3360} & 1 & 0 & 0 \\
-\frac{169}{1680} & -\frac{84839}{283920} & \frac{84839}{283920} & \frac{169}{1680} & 0 & -\frac{1}{2} & -\frac{\sqrt{3}}{2} \\
0 & -\frac{43\sqrt{14595}}{35490} & \frac{43\sqrt{14595}}{35490} & 0 & 0 & \frac{\sqrt{3}}{2} & -\frac{1}{2}
\end{array}\right], \tag{4.21}$$

with,

$$G=\begin{bmatrix} 1 & 0 & 0 \\ 0 & -\frac{1}{4} & 0 \\ 0 & 0 & -\frac{1}{4} \end{bmatrix}, \qquad D=\begin{bmatrix} -\frac{169}{3360} & 0 & 0 & 0 \\ 0 & \frac{1849}{3360} & 0 & 0 \\ 0 & 0 & \frac{1849}{3360} & 0 \\ 0 & 0 & 0 & -\frac{169}{3360} \end{bmatrix}.$$

The method (4.21) is a G-symplectic symmetric method and does not suffer from parasitic solution. Moreover the method has matrix A which is lower triangular and has cheap implementation. The cost of implementation is further decreased by the fact that the first and last stage of method is explicit and only the second and the fourth stage is implicit for which we may need Newton iterations to iteratively solve them.

4.5.2 Algebraic analysis of the order and the starting method

The general linear method (4.21) has an order 4. We employ the algebraic analysis of the order to the general linear method (4.21), which we will refer to as method M. It requires three input values to start the procedure. However only one initial condition is provided with the initial value problem. The rest of the initial input values are calculated using a starting method say S. As explained in section 2.3.3, the order of accuracy of the general linear method M relative to the starting method S is p if

$$M \circ S - S \circ E = O(h^{p+1}),$$

where E is the shift operator representing the exact solution. The general linear method (4.21) is symmetric and therefore has an even order. The analysis of the order of the general linear method (4.21) is carried out only for trees of order up to three because the symmetry will ensure that the method actually has order 4. To start the procedure, the general linear method (4.21) is transformed into its complex formulation using the same transformation T as in (4.18) but in the reverse direction. The coefficient matrices of the general linear method will transform from real to complex formulation as

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] \longrightarrow \left[\begin{array}{c|c} A & UT^* \\ \hline TB & TVT^* \end{array}\right],$$

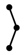
and the matrices of the method are

$$\left[\begin{array}{cccc|ccc} 0 & 0 & 0 & 0 & 1 & \frac{1}{4}-i\frac{\sqrt{3}}{4} & \frac{1}{4}+i\frac{\sqrt{3}}{4} \\ -\frac{11}{72} & \frac{1}{4} & 0 & 0 & 1 & u_{22} & \overline{u}_{22} \\ -\frac{2647}{72240} & \frac{1009}{1680} & \frac{1}{4} & 0 & 1 & -u_{22} & -\overline{u}_{22} \\ -\frac{169}{1680} & \frac{113821}{283920} & \frac{473}{676} & 0 & 1 & -\frac{1}{4}+i\frac{\sqrt{3}}{4} & -\frac{1}{4}-i\frac{\sqrt{3}}{4} \\ \hline -\frac{169}{3360} & \frac{1849}{3360} & \frac{1849}{3360} & -\frac{169}{3360} & 1 & 0 & 0 \\ -\frac{169}{3360} & b_{22} & -b_{22} & \frac{169}{3360} & 0 & -\frac{1}{2}+i\frac{\sqrt{3}}{2} & 0 \\ -\frac{169}{3360} & \overline{b}_{22} & -\overline{b}_{22} & \frac{169}{3360} & 0 & 0 & -\frac{1}{2}-i\frac{\sqrt{3}}{2} \end{array}\right], \qquad (4.22)$$

where

$$\begin{aligned} u_{22} &= \tfrac{1973}{29068}(-1+i\sqrt{3})+\tfrac{2\sqrt{4865}}{7267}(\sqrt{3}+i), \\ b_{22} &= -\tfrac{84839}{567840}-i\tfrac{43\sqrt{14595}}{70980}. \end{aligned}$$

Out of the three output components of the general linear method (4.22), the first one is an approximation to the actual solution and therefore the input to the first component, which is provided by the starting method is taken as the identity method. The second and third components of the method (4.22) which are complex conjugate of each other, approximate some nearby quantities and we take starting approximations to be $\theta : T \longrightarrow \mathbb{R}$ and $\overline{\theta} : T \longrightarrow \mathbb{R}$ where $\theta(t)$ and $\overline{\theta}(t)$ for all trees up to order 3 are

	ϕ	•			
θ	0	0	θ_1	θ_2	θ_3
$\overline{\theta}$	0	0	$\overline{\theta}_1$	$\overline{\theta}_2$	$\overline{\theta}_3$

Note that in B-series notation with a scaled version of Butcher, we have

$$B(\theta,y_0) = \theta_1 h^2 \mathbf{f}'\mathbf{f} + \frac{1}{2!}\theta_2 h^3 \mathbf{f}''(\mathbf{f},\mathbf{f}) + \theta_3 h^3 \mathbf{f}'\mathbf{f}'\mathbf{f} + \cdots,$$

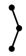
where $\mathbf{f}'\mathbf{f}$, $\mathbf{f}''(\mathbf{f},\mathbf{f})$, and $\mathbf{f}'\mathbf{f}'\mathbf{f}$ are Elementary differentials given in Table 2.1.

The algebraic analysis of order of general linear method (4.22) is done on the same lines as explained in section (2.3.4). Let ξ be the generating function for the input approximations and let η be the generating function for the internal stages. Then from (2.21), (2.22) and (2.23)

$$\eta = A\eta D + U\xi, \tag{4.23}$$

$$E\xi = B\eta D + V\xi. \tag{4.24}$$

To start the procedure we assume $\xi_1 = 1$, $\xi_2 = \theta$ and $\xi_3 = \overline{\theta}$, such that

	ϕ	•			
ξ_1	1	0	0	0	0
ξ_2	0	0	θ_1	θ_2	θ_3
ξ_3	0	0	$\overline{\theta}_1$	$\overline{\theta}_2$	$\overline{\theta}_3$

Now we calculate the generating functions for the internal stages η, their stage derivatives ηD and the output values $\hat{\xi}$ as follows.

For the empty tree ϕ, we take

	η_1	$\eta_1 D$	η_2	$\eta_2 D$	η_3	$\eta_3 D$	η_4	$\eta_4 D$	$\hat{\xi}_1$	$\hat{\xi}_2$	$\hat{\xi}_3$
ϕ	1	0	1	0	1	0	1	0	1	0	0

For the tree with one vertex $\bullet$,

$$\begin{aligned}
\eta_1 D(\bullet) &= \eta_1(\phi) \\
&= 1, \\
\eta_1(\bullet) &= 1\xi_1(\bullet) + u_{12}\xi_2(\bullet) + \bar{u}_{12}\xi_3(\bullet) \\
&= 0, \\
\eta_2 D(\bullet) &= \eta_2(\phi) \\
&= 1, \\
\eta_2(\bullet) &= a_{21}\eta_1 D(\bullet) + a_{22}\eta_2 D(\bullet) + 1\xi_1(\bullet) + u_{22}\xi_2(\bullet) + \bar{u}_{22}\xi_3(\bullet) \\
&= a_{21} + a_{22} \\
&= \tfrac{8}{43}, \\
\eta_3 D(\bullet) &= \eta_3(\phi) \\
&= 1, \\
\eta_3(\bullet) &= a_{31}\eta_1 D(\bullet) + a_{32}\eta_2 D(\bullet) + a_{33}\eta_3 D(\bullet) + 1\xi_1(\bullet) + u_{32}\xi_2(\bullet) + \bar{u}_{32}\xi_3(\bullet) \\
&= a_{31} + a_{32} + a_{33} \\
&= \tfrac{35}{43}, \\
\eta_4 D(\bullet) &= \eta_4(\phi) \\
&= 1, \\
\eta_4(\bullet) &= a_{41}\eta_1 D(\bullet) + a_{42}\eta_2 D(\bullet) + a_{43}\eta_3 D(\bullet) + 1\xi_1(\bullet) + u_{42}\xi_2(\bullet) + \bar{u}_{42}\xi_3(\bullet) \\
&= a_{41} + a_{42} + a_{43} \\
&= 1, \\
\hat{\xi}_1(\bullet) &= b_{11}\eta_1 D(\bullet) + b_{12}\eta_2 D(\bullet) + b_{12}\eta_3 D(\bullet) + b_{11}\eta_4 D(\bullet) + 1\xi_1(\bullet) \\
&= b_{11} + b_{12} + b_{12} + b_{11} \\
&= 1, \\
\hat{\xi}_2(\bullet) &= b_{21}\eta_1 D(\bullet) + b_{22}\eta_2 D(\bullet) - b_{22}\eta_3 D(\bullet) - b_{21}\eta_4 D(\bullet) + v_{22}\xi_2(\bullet) \\
&= b_{21} + b_{22} - b_{22} - b_{21} \\
&= 0, \\
\hat{\xi}_3(\bullet) &= b_{31}\eta_1 D(\bullet) + b_{32}\eta_2 D(\bullet) - b_{32}\eta_3 D(\bullet) - b_{31}\eta_4 D(\bullet) + v_{33}\xi_3(\bullet) \\
&= b_{31} + b_{32} - b_{32} - b_{31} \\
&= 0.
\end{aligned}$$

For the tree $\overset{\bullet}{\underset{\bullet}{|}}$, to calculate $\eta D(\overset{\bullet}{\underset{\bullet}{|}})$ using the formula (2.10), we chop the root of the tree $\overset{\bullet}{\underset{\bullet}{|}}$ and apply η on the rest of the tree which is $\bullet$, such that

$$\begin{aligned}
\eta_1 D(\overset{\bullet}{\underset{\bullet}{|}}) &= \eta_1(\bullet) \\
&= 0.
\end{aligned}$$

Thus

$$\begin{aligned}
\eta_1(\overset{\bullet}{\underset{\bullet}{|}}) &= 1\xi_1(\overset{\bullet}{\underset{\bullet}{|}}) + u_{12}\xi_2(\overset{\bullet}{\underset{\bullet}{|}}) + \bar{u}_{12}\xi_3(\overset{\bullet}{\underset{\bullet}{|}}) \\
&= u_{12}\theta_1 + \bar{u}_{12}\bar{\theta}_1 \\
&= \theta_1(\tfrac{1-i\sqrt{3}}{4}) + \bar{\theta}_1(\tfrac{1+i\sqrt{3}}{4}), \\
\eta_2 D(\overset{\bullet}{\underset{\bullet}{|}}) &= \eta_2(\bullet) \\
&= \tfrac{8}{43}, \\
\eta_2(\overset{\bullet}{\underset{\bullet}{|}}) &= a_{21}\eta_1 D(\overset{\bullet}{\underset{\bullet}{|}}) + a_{22}\eta_2 D(\overset{\bullet}{\underset{\bullet}{|}}) + 1\xi_1(\overset{\bullet}{\underset{\bullet}{|}}) + u_{22}\xi_2(\overset{\bullet}{\underset{\bullet}{|}}) + \bar{u}_{22}\xi_3(\overset{\bullet}{\underset{\bullet}{|}}) \\
&= \tfrac{2}{43} + \tfrac{-1973+24\sqrt{4865}}{29068}(\theta_1 + \bar{\theta}_1) + \tfrac{1973\sqrt{3}+8\sqrt{3}\sqrt{4865}}{29068}(i\theta_1 - i\bar{\theta}_1), \\
\eta_3 D(\overset{\bullet}{\underset{\bullet}{|}}) &= \eta_3(\bullet) \\
&= \tfrac{35}{43}, \\
\eta_3(\overset{\bullet}{\underset{\bullet}{|}}) &= a_{31}\eta_1 D(\overset{\bullet}{\underset{\bullet}{|}}) + a_{32}\eta_2 D(\overset{\bullet}{\underset{\bullet}{|}}) + a_{33}\eta_3 D(\overset{\bullet}{\underset{\bullet}{|}}) + 1\xi_1(\overset{\bullet}{\underset{\bullet}{|}}) + u_{32}\xi_2(\overset{\bullet}{\underset{\bullet}{|}}) + \bar{u}_{32}\xi_3(\overset{\bullet}{\underset{\bullet}{|}}) \\
&= \tfrac{5693}{18060} + \tfrac{1973-24\sqrt{4865}}{29068}(\theta_1 + \bar{\theta}_1) + \tfrac{-1973\sqrt{3}-8\sqrt{3}\sqrt{4865}}{29068}(i\theta_1 - i\bar{\theta}_1), \\
\eta_4 D(\overset{\bullet}{\underset{\bullet}{|}}) &= \eta_4(\bullet) \\
&= 1, \\
\eta_4(\overset{\bullet}{\underset{\bullet}{|}}) &= a_{41}\eta_1 D(\overset{\bullet}{\underset{\bullet}{|}}) + a_{42}\eta_2 D(\overset{\bullet}{\underset{\bullet}{|}}) + a_{43}\eta_3 D(\overset{\bullet}{\underset{\bullet}{|}}) + 1\xi_1(\overset{\bullet}{\underset{\bullet}{|}}) + u_{42}\xi_2(\overset{\bullet}{\underset{\bullet}{|}}) + \bar{u}_{42}\xi_3(\overset{\bullet}{\underset{\bullet}{|}}) \\
&= \tfrac{45719}{70980} - \tfrac{1}{4}(\theta_1 + \bar{\theta}_1) - \tfrac{\sqrt{3}}{4}(i\theta_1 - i\bar{\theta}_1), \\
\hat{\xi}_1(\overset{\bullet}{\underset{\bullet}{|}}) &= b_{11}\eta_1 D(\overset{\bullet}{\underset{\bullet}{|}}) + b_{12}\eta_2 D(\overset{\bullet}{\underset{\bullet}{|}}) + b_{12}\eta_3 D(\overset{\bullet}{\underset{\bullet}{|}}) + b_{11}\eta_4 D(\overset{\bullet}{\underset{\bullet}{|}}) + 1\xi_1(\overset{\bullet}{\underset{\bullet}{|}}) \\
&= \tfrac{1}{2}, \\
\hat{\xi}_2(\overset{\bullet}{\underset{\bullet}{|}}) &= b_{21}\eta_1 D(\overset{\bullet}{\underset{\bullet}{|}}) + b_{22}\eta_2 D(\overset{\bullet}{\underset{\bullet}{|}}) - b_{22}\eta_3 D(\overset{\bullet}{\underset{\bullet}{|}}) - b_{21}\eta_4 D(\overset{\bullet}{\underset{\bullet}{|}}) + v_{22}\xi_2(\overset{\bullet}{\underset{\bullet}{|}}) \\
&= \tfrac{10229}{70980} + i\tfrac{9\sqrt{3}\sqrt{4865}}{2360} - \tfrac{\theta_1}{2} + i\tfrac{\sqrt{3}\theta_1}{2}, \\
\hat{\xi}_3(\overset{\bullet}{\underset{\bullet}{|}}) &= b_{31}\eta_1 D(\overset{\bullet}{\underset{\bullet}{|}}) + b_{32}\eta_2 D(\overset{\bullet}{\underset{\bullet}{|}}) - b_{32}\eta_3 D(\overset{\bullet}{\underset{\bullet}{|}}) - b_{31}\eta_4 D(\overset{\bullet}{\underset{\bullet}{|}}) + v_{33}\xi_3(\overset{\bullet}{\underset{\bullet}{|}}) \\
&= \tfrac{10229}{70980} - i\tfrac{9\sqrt{3}\sqrt{4865}}{2360} - \tfrac{\bar{\theta}_1}{2} - i\tfrac{\sqrt{3}\bar{\theta}_1}{2}.
\end{aligned}$$

For the tree $\bigvee$, to calculate $\eta D(\bigvee)$ using the formula (2.10), we chop the root of the tree $\bigvee$, to obtain two similar trees $\bullet$ and $\bullet$ and apply η on them such that

$$\begin{aligned}\eta_1 D(\bigvee) &= \eta_1^2(\bullet)\\ &= 0.\end{aligned}$$

Thus

$$\begin{aligned}
\eta_1(\bigvee) &= 1\xi_1(\bigvee) + u_{12}\xi_2(\bigvee) + \bar{u}_{12}\xi_3(\bigvee)\\
&= u_{12}\theta_2 + \bar{u}_{12}\bar{\theta}_2\\
&= \theta_2(\tfrac{1-i\sqrt{3}}{4}) + \bar{\theta}_2(\tfrac{1+i\sqrt{3}}{4}),\\
\eta_2 D(\bigvee) &= \eta_2^2(\bullet)\\
&= \tfrac{64}{1849},\\
\eta_2(\bigvee) &= a_{21}\eta_1 D(\bigvee) + a_{22}\eta_2 D(\bigvee) + 1\xi_1(\bigvee) + u_{22}\xi_2(\bigvee) + \bar{u}_{22}\xi_3(\bigvee)\\
&= \tfrac{16}{1849} + \tfrac{-1973+24\sqrt{4865}}{29068}(\theta_2 + \bar{\theta}_2) + \tfrac{1973\sqrt{3}+8\sqrt{3}\sqrt{4865}}{29068}(i\theta_2 - i\bar{\theta}_2),\\
\eta_3 D(\bigvee) &= \eta_3^2(\bullet)\\
&= \tfrac{1225}{1849},\\
\eta_3(\bigvee) &= a_{31}\eta_1 D(\bigvee) + a_{32}\eta_2 D(\bigvee) + a_{33}\eta_3 D(\bigvee) + 1\xi_1(\bigvee) + u_{32}\xi_2(\bigvee) + \bar{u}_{32}\xi_3(\bigvee)\\
&= \tfrac{144769}{776580} + \tfrac{1973-24\sqrt{4865}}{29068}(\theta_2 + \bar{\theta}_2) + \tfrac{-1973\sqrt{3}-8\sqrt{3}\sqrt{4865}}{29068}(i\theta_2 - i\bar{\theta}_2),\\
\eta_4 D(\bigvee) &= \eta_4^2(\bullet)\\
&= 1,\\
\eta_4(\bigvee) &= a_{41}\eta_1 D(\bigvee) + a_{42}\eta_2 D(\bigvee) + a_{43}\eta_3 D(\bigvee) + 1\xi_1(\bigvee) + u_{42}\xi_2(\bigvee) + \bar{u}_{42}\xi_3(\bigvee)\\
&= \tfrac{33889}{70980} - \tfrac{1}{4}(\theta_2 + \bar{\theta}_2) - \tfrac{\sqrt{3}}{4}(i\theta_2 - i\bar{\theta}_2),\\
\hat{\xi}_1(\bigvee) &= b_{11}\eta_1 D(\bigvee) + b_{12}\eta_2 D(\bigvee) + b_{12}\eta_3 D(\bigvee) + b_{11}\eta_4 D(\bigvee) + 1\xi_1(\bigvee)\\
&= \tfrac{1}{3},\\
\hat{\xi}_2(\bigvee) &= b_{21}\eta_1 D(\bigvee) + b_{22}\eta_2 D(\bigvee) - b_{22}\eta_3 D(\bigvee) - b_{21}\eta_4 D(\bigvee) + v_{22}\xi_2(\bigvee)\\
&= \tfrac{10229}{70980} + i\tfrac{9\sqrt{3}\sqrt{4865}}{2360} - \tfrac{\theta_2}{2} + i\tfrac{\sqrt{3}\theta_2}{2},\\
\hat{\xi}_3(\bigvee) &= b_{31}\eta_1 D(\bigvee) + b_{32}\eta_2 D(\bigvee) - b_{32}\eta_3 D(\bigvee) - b_{31}\eta_4 D(\bigvee) + v_{33}\xi_3(\bigvee)\\
&= \tfrac{10229}{70980} - i\tfrac{9\sqrt{3}\sqrt{4865}}{2360} - \tfrac{\bar{\theta}_2}{2} - i\tfrac{\sqrt{3}\bar{\theta}_2}{2}.
\end{aligned}$$

For the tree $\wr$, to calculate $\eta D(\wr)$ using the formula (2.10), we chop the root of the tree $\wr$, to obtain $\vert$ and apply η on it such that

$$\begin{aligned}\eta_1 D(\wr) &= \eta_1(\vert)\\ &= \theta_1(\tfrac{1-i\sqrt{3}}{4}) + \bar{\theta}_1(\tfrac{1+i\sqrt{3}}{4}).\end{aligned}$$

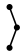
Thus

$$\begin{aligned}
\eta_1(\wr) &= 1\xi_1(\wr) + u_{12}\xi_2(\wr) + \bar{u}_{12}\xi_3(\wr) \\
&= u_{12}\theta_3 + \bar{u}_{12}\bar{\theta}_3 \\
&= \theta_3(\tfrac{1-i\sqrt{3}}{4}) + \bar{\theta}_3(\tfrac{1+i\sqrt{3}}{4}), \\
\eta_2 D(\wr) &= \eta_2(\vert) \\
&= \tfrac{2}{43} + \tfrac{-1973+24\sqrt{4865}}{29068}(\theta_1+\bar{\theta}_1) + \tfrac{1973\sqrt{3}+8\sqrt{3}\sqrt{4865}}{29068}(i\theta_1 - i\bar{\theta}_1), \\
\eta_2(\wr) &= a_{21}\eta_1 D(\wr) + a_{22}\eta_2 D(\wr) + 1\xi_1(\wr) + u_{22}\xi_2(\wr) + \bar{u}_{22}\xi_3(\wr) \\
&= \tfrac{-479+3\sqrt{4865}}{14534}(\theta_1+\bar{\theta}_1) - \tfrac{\sqrt{3}(479+\sqrt{4865})}{14534}(i\theta_1 - i\bar{\theta}_1) \\
&\quad - \tfrac{\sqrt{3}(1973+8\sqrt{4865})}{29068}(i\theta_3 - i\bar{\theta}_3) - \tfrac{1973-24\sqrt{4865}}{29068}(\theta_3+\bar{\theta}_3), \\
\eta_3 D(\wr) &= \eta_3(\vert) \\
&= \tfrac{5693}{18060} + \tfrac{1973-24\sqrt{4865}}{29068}(\theta_1+\bar{\theta}_1) + \tfrac{-1973\sqrt{3}-8\sqrt{3}\sqrt{4865}}{29068}(i\theta_1 - i\bar{\theta}_1), \\
\eta_3(\wr) &= a_{31}\eta_1 D(\wr) + a_{32}\eta_2 D(\wr) + a_{33}\eta_3 D(\wr) + 1\xi_1(\wr) + u_{32}\xi_2(\wr) + \bar{u}_{32}\xi_3(\wr) \\
&= \tfrac{7711}{72240} - \tfrac{67060-589\sqrt{4865}}{2034760}(\theta_1+\bar{\theta}_1) + \tfrac{\sqrt{3}(201180+589\sqrt{4865})}{6104280}(i\theta_1 - i\bar{\theta}_1) \\
&\quad + \tfrac{\sqrt{3}(1973+8\sqrt{4865})}{7267}(i\theta_3 - i\bar{\theta}_3) + \tfrac{1973-24\sqrt{4865}}{29068}(\theta_3+\bar{\theta}_3), \\
\eta_4 D(\wr) &= \eta_4(\vert) \\
&= \tfrac{45719}{70980} - \tfrac{1}{4}(\theta_1+\bar{\theta}_1) - \tfrac{\sqrt{3}}{4}(i\theta_1 - i\bar{\theta}_1), \\
\eta_4(\wr) &= a_{41}\eta_1 D(\wr) + a_{42}\eta_2 D(\wr) + a_{43}\eta_3 D(\wr) + 1\xi_1(\wr) + u_{42}\xi_2(\wr) + \bar{u}_{42}\xi_3(\wr) \\
&= \tfrac{22639}{94640} - \tfrac{38920+1973\sqrt{4865}}{7997080}(\theta_1+\bar{\theta}_1) + \tfrac{\sqrt{3}(116760-1973\sqrt{4865})}{23991240}(i\theta_1 - i\bar{\theta}_1) \\
&\quad + \tfrac{\sqrt{3}}{4}(i\theta_3 - i\bar{\theta}_3) + \tfrac{1}{4}(\theta_3+\bar{\theta}_3), \\
\hat{\xi}_1(\wr) &= b_{11}\eta_1 D(\wr) + b_{12}\eta_2 D(\wr) + b_{12}\eta_3 D(\wr) + b_{11}\eta_4 D(\wr) + 1\xi_1(\vee) \\
&= \tfrac{1}{6}, \\
\hat{\xi}_2(\wr) &= b_{21}\eta_1 D(\wr) + b_{22}\eta_2 D(\wr) - b_{22}\eta_3 D(\wr) - b_{21}\eta_4 D(\wr) + v_{22}\xi_2(\wr) \\
&= \tfrac{20597}{283920} + i\tfrac{4853\sqrt{3}\sqrt{4865}}{29811600} - \tfrac{1973\sqrt{4865}\bar{\theta}_1}{3998540} - i\tfrac{278\sqrt{3}\bar{\theta}_1}{28561} \\
&\quad - \tfrac{278\bar{\theta}_1}{28561} + i\tfrac{1973\sqrt{4865}\sqrt{3}\bar{\theta}_1}{11995620} - \tfrac{\theta_3}{2} + i\tfrac{\sqrt{3}\theta_3}{2}, \\
\hat{\xi}_3(\wr) &= b_{31}\eta_1 D(\wr) + b_{32}\eta_2 D(\wr) - b_{32}\eta_3 D(\wr) - b_{31}\eta_4 D(\wr) + v_{33}\xi_3(\wr) \\
&= \tfrac{20597}{283920} + i\tfrac{4853\sqrt{3}\sqrt{4865}}{29811600} - \tfrac{1973\sqrt{4865}\theta_1}{3998540} - i\tfrac{278\sqrt{3}\theta_1}{28561} \\
&\quad - \tfrac{278\theta_1}{28561} + i\tfrac{1973\sqrt{4865}\sqrt{3}\theta_1}{11995620} - \tfrac{\bar{\theta}_3}{2} + i\tfrac{\sqrt{3}\bar{\theta}_3}{2}.
\end{aligned}$$

The general linear method (4.22) will be of order 4, if

$$
\begin{aligned}
E\xi_1(t) &= \hat{\xi}_1(t),\\
E\xi_2(t) &= \hat{\xi}_2(t),\\
E\xi_3(t) &= \hat{\xi}_3(t),
\end{aligned}
$$

where

	ϕ	•			
$E\xi_1$	1	1	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{6}$
$E\xi_2$	0	0	θ_1	$2\theta_1+\theta_2$	$\theta_1+\theta_3$
$E\xi_3$	0	0	$\overline{\theta}_1$	$2\overline{\theta}_1+\overline{\theta}_2$	$\overline{\theta}_1+\overline{\theta}_3$

Thus we get unique values of θ

$$
\begin{aligned}
\theta_1 &= 0.05878953294 + i0.064578611121,\\
\theta_2 &= 0.03728447850 - i0.033942152651,\\
\theta_3 &= 0.01776807645 - i0.01766421180.
\end{aligned}
$$

Using the values of θ, we can obtain a starting method S to make sure that the general linear method (4.22) is of order 4 relative to the starting method. We take the abscissa $\tilde{c}$ and matrix $\widetilde{A}$ of the starting method to be of classical order 4 Runge–Kutta method given as

$$
\tilde{c} = [0 \quad \tfrac{1}{2} \quad \tfrac{1}{2} \quad 1],
$$

$$
\widetilde{A} = \begin{bmatrix} 0 & 0 & 0 & 0\\ \frac{1}{2} & 0 & 0 & 0\\ 0 & \frac{1}{2} & 0 & 0\\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{4.25}
$$

The starting method will get one input value and three output values so the structure of the starting method is

$$
\left[\begin{array}{cccc|c}
0 & 0 & 0 & 0 & 1\\
\frac{1}{2} & 0 & 0 & 0 & 1\\
0 & \frac{1}{2} & 0 & 0 & 1\\
0 & 0 & 1 & 0 & 1\\
\hline
0 & 0 & 0 & 0 & 1\\
\tilde{b}_1 & \tilde{b}_2 & \tilde{b}_3 & \tilde{b}_4 & 0\\
\overline{\tilde{b}}_1 & \overline{\tilde{b}}_2 & \overline{\tilde{b}}_3 & \overline{\tilde{b}}_4 & 0
\end{array}\right].
$$

The matrix $\tilde{B}$ has the second and third row as complex conjugate of each other in line with the general linear method (4.22). The value of $\tilde{B} = [\tilde{b}_1, \tilde{b}_2, \tilde{b}_3, \tilde{b}_4]$ can be found using the order conditions

$$\begin{aligned} \sum_i \tilde{b}_i &= 0, \\ \sum_i \tilde{b}_i \tilde{c}_i &= \theta_1, \\ \sum_i \tilde{b}_i \tilde{c}_i^2 &= \theta_2, \\ \sum_{ij} \tilde{b}_i \tilde{a}_{ij} \tilde{c}_j &= \theta_3. \end{aligned}$$

This implies

$$\begin{bmatrix} \tilde{b}_1 \\ \tilde{b}_2 \\ \tilde{b}_3 \\ \tilde{b}_4 \end{bmatrix} = \begin{bmatrix} -0.101799641805663 + i0.261620138702276 \\ 0.046506760094787 - i0.199814069408086 \\ 0.039513457627372 - i0.194268985756399 \\ 0.015779424083504 + i0.132462916462209 \end{bmatrix}.$$

Transforming the complex numbers into real using the transformation matrix T in (4.18), such that $B \longrightarrow T^*B$ yields the starting method

$$\left[\begin{array}{cccc|c} 0 & 0 & 0 & 0 & 1 \\ \frac{1}{2} & 0 & 0 & 0 & 1 \\ 0 & \frac{1}{2} & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ \hline 0 & 0 & 0 & 0 & 1 \\ \tilde{b}_1 & \tilde{b}_2 & \tilde{b}_3 & \tilde{b}_4 & 0 \\ \tilde{b}_5 & \tilde{b}_6 & \tilde{b}_7 & \tilde{b}_8 & 0 \end{array}\right], \tag{4.26}$$

where

$$\begin{bmatrix} \tilde{b}_1 \\ \tilde{b}_2 \\ \tilde{b}_3 \\ \tilde{b}_4 \\ \tilde{b}_5 \\ \tilde{b}_6 \\ \tilde{b}_7 \\ \tilde{b}_8 \end{bmatrix} = \begin{bmatrix} -0.203599283611326 \\ 0.093013520189574 \\ 0.079026915254744 \\ 0.031558848167008 \\ -0.523240277404552 \\ 0.399628138816171 \\ 0.388537971512798 \\ -0.264925832924417 \end{bmatrix}.$$

4.5.3 Verification of order

In order to verify that the general linear method (4.21) is of order 4 relative to the starting method (4.26) we proceed as follows. Suppose we solve an ordinary differential equation system and take one integration step using the starting method (4.26). Afterwards, we take the second integration step with the actual general linear method (4.21). This procedure is equivalent to proceeding with a bigger step taken by a method originated by composing the starting method and the actual general linear method. If we only consider the component of output approximation from the general linear method which is approximating the actual solution then the structure of the composed method will be

$$\begin{array}{c|cc} \tilde{c} & \widetilde{A} & 0 \\ c & U\times\widetilde{B} & A \\ \hline & \widetilde{B}(1,:) & B(1,:) \end{array}\,, \tag{4.27}$$

where $\tilde{c}$, $\widetilde{B}$ and $\widetilde{A}$ each represents the corresponding entries in the starting method (4.26) and rest of the entries are from actual general linear method (4.21). The structure of the composed method (4.27) represents a Runge–Kutta method with abscissas

$$C_{\mathbf{com}} = [0 \quad \tfrac{1}{2} \quad \tfrac{1}{2} \quad 1 \quad 0 \quad 0.186046511627907 \quad 0.813953488372093 \quad 1].$$

The vector $B_{\mathbf{com}}$ has structure

$$B_{\mathbf{com}} = [0 \;\; 0 \;\; 0 \;\; 0 \;\; -0.05029761904 \;\; 0.55029761904 \;\; 0.55029761904 \;\; -0.05029761904].$$

The matrix $A_{\mathbf{com}}$ has structure

$$A_{\mathbf{com}} = \begin{bmatrix} \widetilde{A} & 0 \\ U\times\widetilde{B} & A \end{bmatrix},$$

where the matrix $\widetilde{A}$ is given by (4.25), A is the same matrix given in (4.21), and

$$U\times\widetilde{B} = \begin{bmatrix} -0.27746950716 & 0.19629744018 & 0.18799860564 & -0.10682653867 \\ 0.08100545721 & -0.06122566487 & -0.05940925575 & 0.03962946341 \\ -0.08100545721 & 0.06122566487 & 0.05940925575 & -0.03962946341 \\ 0.27746950716 & -0.19629744018 & -0.18799860564 & 0.10682653867 \end{bmatrix}.$$

Now it is easy to see that the method (4.27) is of order 4, because it satisfies order conditions for all trees of order up to 4. Hence the general linear method (4.21) is of order 4 relative to the starting method (4.26).

Chapter 5

Numerical experiments

This chapter presents the results of numerical methods constructed in this thesis to two types of problems, the Hamiltonian problems and the problems with quadratic invariants. The aim is to observe the ability of the methods to provide qualitatively correct numerical results over long time. Moreover, we would like to observe the accuracy of the methods for short time integration and the efficiency of methods in terms of minimum cost of implementation for which the behaviour of global error and the number of function evaluations are very important. The reference method is taken to be the famous fourth order Gauss symplectic Runge–Kutta method given in (3.15). Although the methods studied in this thesis are general linear methods with multiple input values, we aim to achieve comparable performance to the one-step method.

It is widely believed that a qualitatively correct numerical result is obtained with a fixed stepsize implementation of the numerical method for solving, for example, Hamiltonian problems with symplectic methods, see for example [46]. We therefore use constant stepsize in all numerical experiments. The numerical methods employed in this chapter have implicit stages to evaluate. We use modified Newton iterations and aim for convergence within machine accuracy. The experiments carried out are of preliminary nature but pave a way for more practical calculations using the methods and techniques described in this thesis.

The first section of this chapter discusses numerical methods used for the experiments. A construction of these methods was the subject of Chapter 4. The later sections introduce problems for later experiments. We have taken three problems from among Hamiltonian systems and two problems with quadratic invariants. Each problem is accompanied by the numerical results and a discussion.

5.1 Methods

- P :

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{cc|cc} \frac{3+\sqrt{3}}{6} & 0 & 1 & -\frac{3+2\sqrt{3}}{3} \\ -\frac{\sqrt{3}}{3} & \frac{3+\sqrt{3}}{6} & 1 & \frac{3+2\sqrt{3}}{3} \\ \hline \frac{1}{2} & \frac{1}{2} & 1 & 0 \\ \frac{1}{2} & -\frac{1}{2} & 0 & -1 \end{array}\right].$$

  The starting method is

$$\left[\begin{array}{cc|c} \frac{3+\sqrt{3}}{6} & 0 & 1 \\ -\frac{3+\sqrt{3}}{3} & \frac{3+\sqrt{3}}{6} & 1 \\ \hline 0 & 0 & 1 \\ \frac{\sqrt{3}-1}{8} & -\frac{\sqrt{3}-1}{8} & 0 \end{array}\right].$$

- N :

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{cc|cc} \frac{3-\sqrt{3}}{6} & 0 & 1 & -\frac{3-2\sqrt{3}}{3} \\ \frac{\sqrt{3}}{3} & \frac{3-\sqrt{3}}{6} & 1 & \frac{3-2\sqrt{3}}{3} \\ \hline \frac{1}{2} & \frac{1}{2} & 1 & 0 \\ \frac{1}{2} & -\frac{1}{2} & 0 & -1 \end{array}\right].$$

  The starting method is

$$\left[\begin{array}{cc|c} \frac{3-\sqrt{3}}{6} & 0 & 1 \\ -\frac{3-\sqrt{3}}{3} & \frac{3-\sqrt{3}}{6} & 1 \\ \hline 0 & 0 & 1 \\ -\frac{\sqrt{3}+1}{8} & \frac{\sqrt{3}+1}{8} & 0 \end{array}\right].$$

- *COM* : We compose the methods N and P in the sequence

$$N^7\ P\ N^{14}\ P\ N^{14}\ P\ N^{14}\ P\ N^{14}\ P\ N^{14}\ P\ N^{14}\ P\ N^{13}\ P\cdots$$

  provided the second row of matrix B and second column of matrix U both multiply with -1 for the method P. The starting method for the method N is used as the starting method for *COM* as discussed in section 4.4.

- GLM43:

$$\left[\begin{array}{cccc|ccc}
0 & 0 & 0 & 0 & 1 & \frac{1}{4} & \frac{\sqrt{3}}{4} \\
-\frac{11}{72} & \frac{1}{4} & 0 & 0 & 1 & -\frac{1973}{29068}+\frac{2\sqrt{3}\sqrt{14595}}{7267} & -\frac{1973\sqrt{3}}{29068}-\frac{2\sqrt{14595}}{7267} \\
-\frac{2647}{72240} & \frac{1009}{1680} & \frac{1}{4} & 0 & 1 & \frac{1973}{29068}-\frac{2\sqrt{3}\sqrt{14595}}{7267} & \frac{1973\sqrt{3}}{29068}+\frac{2\sqrt{14595}}{7267} \\
-\frac{169}{1680} & \frac{113821}{283920} & \frac{473}{676} & 0 & 1 & -\frac{1}{4} & -\frac{\sqrt{3}}{4} \\
\hline
-\frac{169}{3360} & \frac{1849}{3360} & \frac{1849}{3360} & -\frac{169}{3360} & 1 & 0 & 0 \\
-\frac{169}{1680} & -\frac{84839}{283920} & \frac{84839}{283920} & \frac{169}{1680} & 0 & -\frac{1}{2} & -\frac{\sqrt{3}}{2} \\
0 & -\frac{43\sqrt{14595}}{35490} & \frac{43\sqrt{14595}}{35490} & 0 & 0 & \frac{\sqrt{3}}{2} & -\frac{1}{2}
\end{array}\right].$$

The starting method is

$$\left[\begin{array}{cccc|c}
0 & 0 & 0 & 0 & 1 \\
\frac{1}{2} & 0 & 0 & 0 & 1 \\
0 & \frac{1}{2} & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 1 \\
\hline
0 & 0 & 0 & 0 & 1 \\
\tilde{b}_1 & \tilde{b}_2 & \tilde{b}_3 & \tilde{b}_4 & 0 \\
\tilde{b}_5 & \tilde{b}_6 & \tilde{b}_7 & \tilde{b}_8 & 0
\end{array}\right],$$

where

$$\begin{bmatrix} \tilde{b}_1 \\ \tilde{b}_2 \\ \tilde{b}_3 \\ \tilde{b}_4 \\ \tilde{b}_5 \\ \tilde{b}_6 \\ \tilde{b}_7 \\ \tilde{b}_8 \end{bmatrix} = \begin{bmatrix} -0.203599283611326 \\ 0.093013520189574 \\ 0.079026915254744 \\ 0.031558848167008 \\ -0.523240277404552 \\ 0.399628138816171 \\ 0.388537971512798 \\ -0.264925832924417 \end{bmatrix}.$$

- $IRK4$:

$$\begin{array}{c|cc}
\frac{1}{2}-\frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4}-\frac{\sqrt{3}}{6} \\
\frac{1}{2}+\frac{\sqrt{3}}{6} & \frac{1}{4}+\frac{\sqrt{3}}{6} & \frac{1}{4} \\
\hline
 & \frac{1}{2} & \frac{1}{2}
\end{array}.$$

5.2 The Kepler problem

The Kepler problem describes the motion of a planet revolving around sun which is considered to be fixed at origin. The equations of motion defines a separable Hamiltonian system

$$\begin{aligned} q_1' &= p_1, \\ q_2' &= p_2, \\ p_1' &= \frac{-q_1}{(q_1^2+q_2^2)^{\frac{3}{2}}}, \\ p_2' &= \frac{-q_2}{(q_1^2+q_2^2)^{\frac{3}{2}}}, \end{aligned}$$

where (q_1, q_2) are the generalised position coordinates of the body and (p_1, p_2) are the generalised momenta. The total energy of the system is

$$H = \tfrac{1}{2}(p_1^2+p_2^2) - \frac{1}{\sqrt{q_1^2+q_2^2}}.$$

The initial conditions are taken to be

$$(\,q_1,\ q_2,\ p_1,\ p_2\,) = \left(1-e,\ 0,\ 0,\ \sqrt{\tfrac{1+e}{1-e}}\,\right),$$

where $0 \le e < 1$ is the eccentricity of the elliptic orbits which are formed by the motion of one body around the other. The exact solution of the Kepler problem is given in [27] which is

$$\begin{aligned} q_1(x) &= \cos(E) - e, \\ q_2(x) &= \sqrt{1-e^2}\sin(E), \end{aligned}$$

where E is the eccentric anomaly given by the Kepler formula

$$x = E - \sin(E).$$

The first experiment studies the conservation of energy of the Kepler problem with $e = 0$ and stepsize $\frac{2\pi}{1000}$. We see the corruption of numerical solution by parasitism in the method P in Figure 5.1, where only after 10000 steps, we observe an energy drift. For the method N in Figure 5.2, a similar behaviour is observed but the numerical method is corrupted after longer time. This is because the parasitic parameter of the method P is $1+2/\sqrt{3}$ which is greater than the parasitic parameter of the method N which is

$1-2/\sqrt{3}$. However, for the method COM, which is a composition of the methods P and N, we get long time energy conservation in Figure 5.3 for one million steps. The method $GLM43$, a G-symplectic methods without parasitism, preserves the energy well, as shown in Figure 5.4 for one million steps which is comparable to the energy conservation of the method $IRK4$ as shown in Figure 5.5 over same integration time.

Our second experiment studies the conservation of energy of the Kepler problem in the case when $e = 0.5$. As expected, we get energy conservation over one million steps and the results are given in Figure 5.6 for $IRK4$, Figure 5.7 for COM and Figure 5.8 for $GLM43$.

The third experiment is to compare the global error of the methods by comparing the numerical solution with the exact solution of the Kepler problem. To study the accuracy for a fixed short time interval we consider the global error over half period π with $e = 0$. The results are shown in the Table 5.1. The results clearly show excellent short term behaviour of the methods $GLM43$ and COM and they are comparable to the behaviour of the method $IRK4$. The long term behaviour of the methods in terms of global errors is shown in Figure 5.9. Here we have taken a stepsize of $2\pi/100$ and $e = 0$. We notice that method $IRK4$ performs better than the method COM and $GLM43$. We get similar results for $e = 0.5$ with a smaller stepsize of $2\pi/1000$ as shown in Figure 5.10. Figure 5.11 and Figure 5.12 show a comparison of global error of methods P, N and COM for $e = 0$ and $e = 0.5$ respectively and they reveal parasitic corruption of numerical solution by the methods P and N.

The fourth experiment is to compare the efficiency of the methods by calculating the number of function evaluations. Because of the implicit nature of the stages, the Jacobian is evaluated for each Newton iteration. Figure 5.13 and Figure 5.14 show a plot between number of function evaluations and the global error as the stepsize decreases for Kepler problem with $e = 0$ and $e = 0.5$ respectively. In this context, both graphs clearly show the superiority of the method COM over $IRK4$ and $GLM43$. This is because the method COM has diagonally implicit stages whereas the method $IRK4$ has fully implicit stages and the method $GLM43$ has double the number of stages of the method COM. The method $IRK4$ initially performs better than the method $GLM43$, but the method $GLM43$ outperforms $IRK4$ later. The Table 5.2 contains the data for the global error as stepsize is decreased and shows that the $RK4$ method and $GLM43$ method both have $O(h^5)$ behaviour. This should have been $O(h^4)$ behaviour because both methods are of order 4. The reason is that, due to the symmetry of the Kepler problem, some error constants might have cancelled each other. The method COM shows $O(h^5)$ behaviour as the stepsize is decreased and becomes smaller.

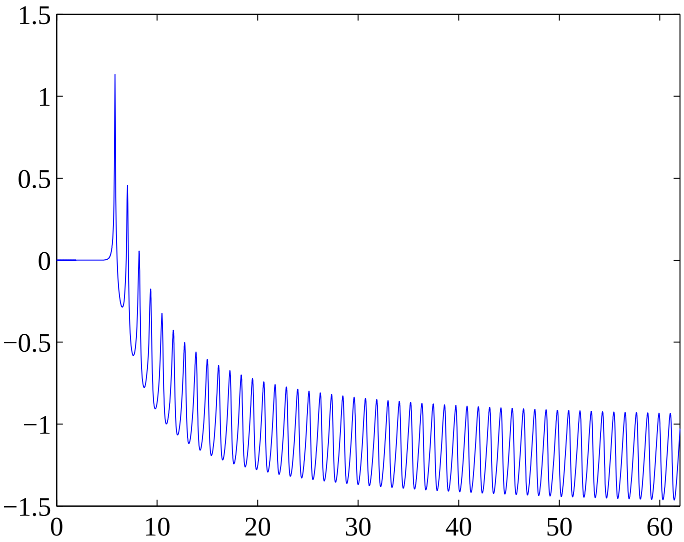


Figure 5.1: The error in energy of the Kepler problem (e = 0) with the method *P*.

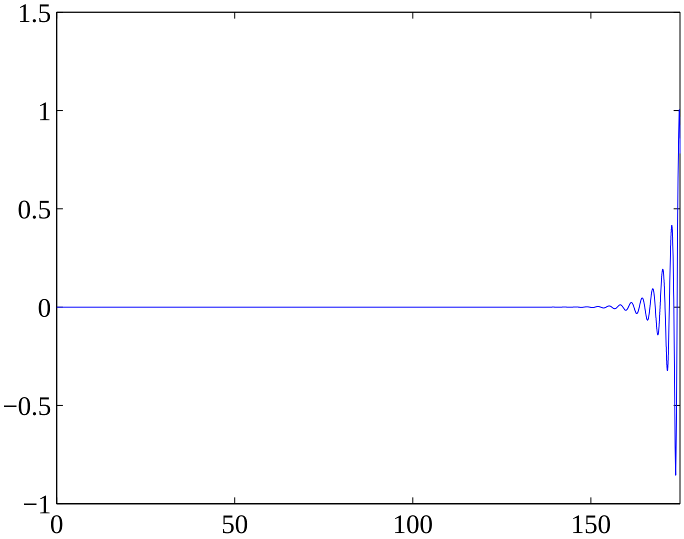


Figure 5.2: The error in energy of the Kepler problem (e = 0) with the method *N*.

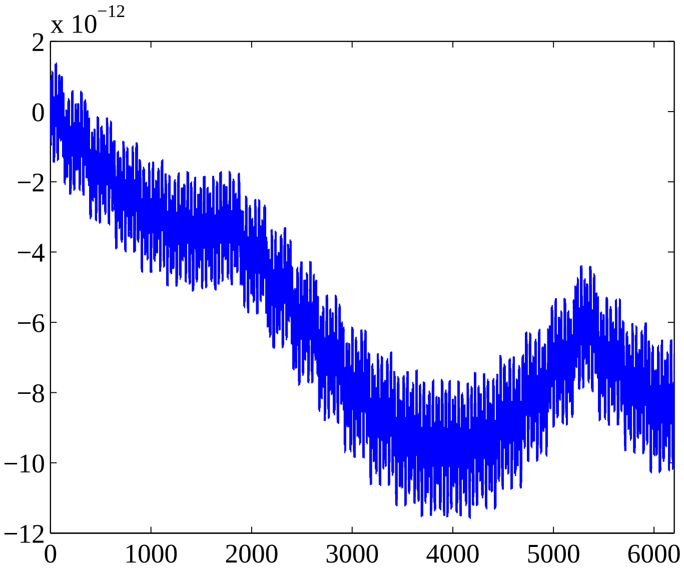


Figure 5.3: The error in energy of the Kepler problem (e = 0) with the method *COM*.

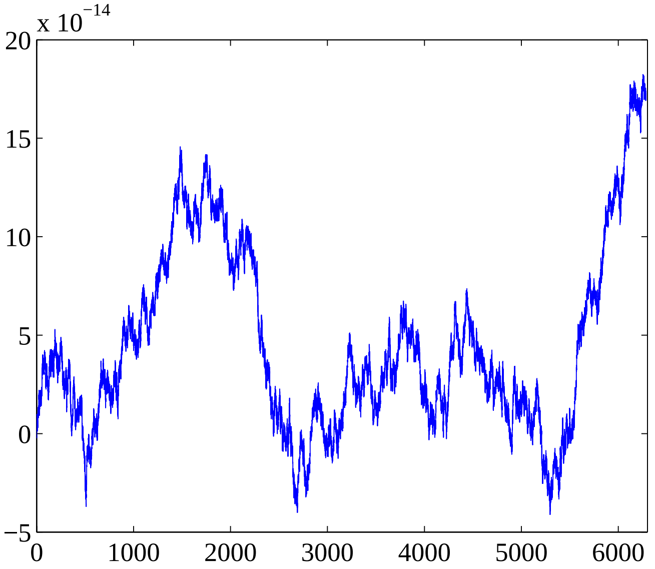


Figure 5.4: The error in energy of the Kepler problem (e = 0) with the method *GLM*43.

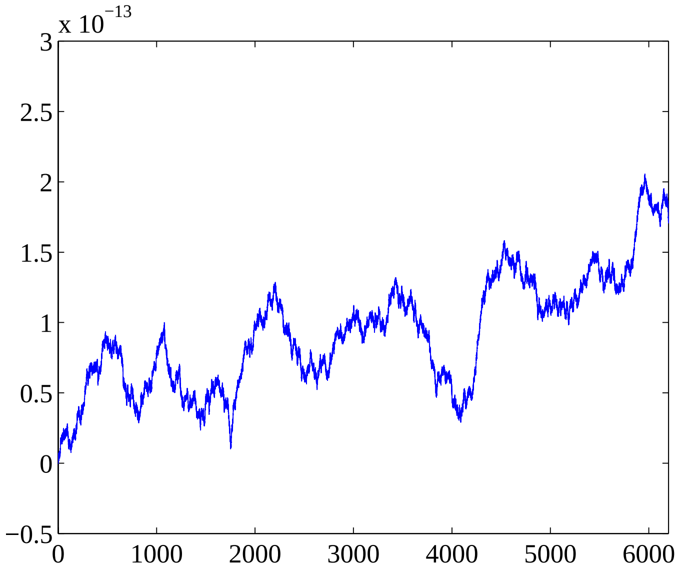


Figure 5.5: The error in energy of the Kepler problem (e = 0) with the method *IRK*4.

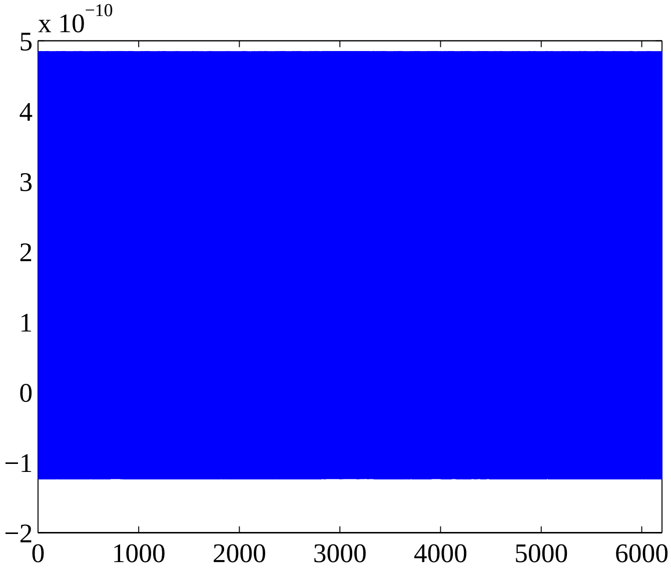


Figure 5.6: The error in energy of the Kepler problem (e = 0.5) with the method *IRK*4.

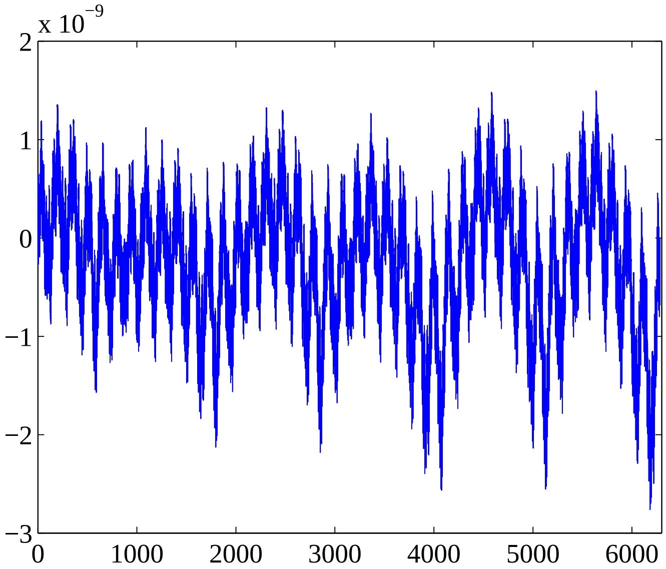


Figure 5.7: The error in energy of the Kepler problem (e = 0.5) with the method *COM*.

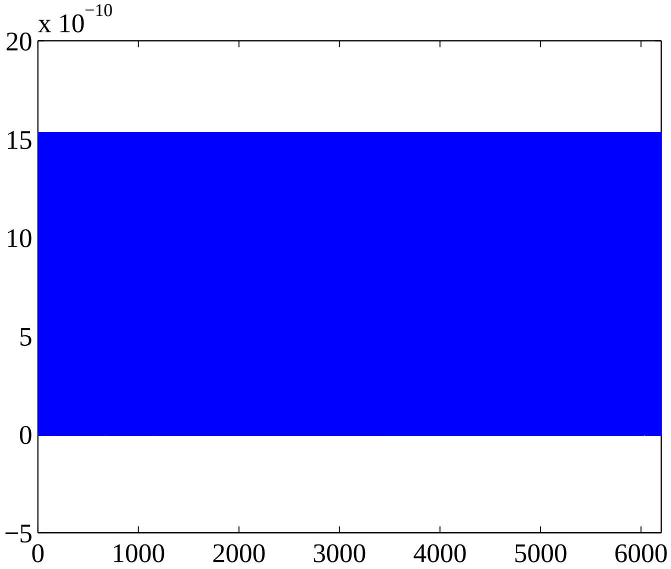


Figure 5.8: The error in energy of the Kepler problem (e = 0.5) with the method *GLM*43.

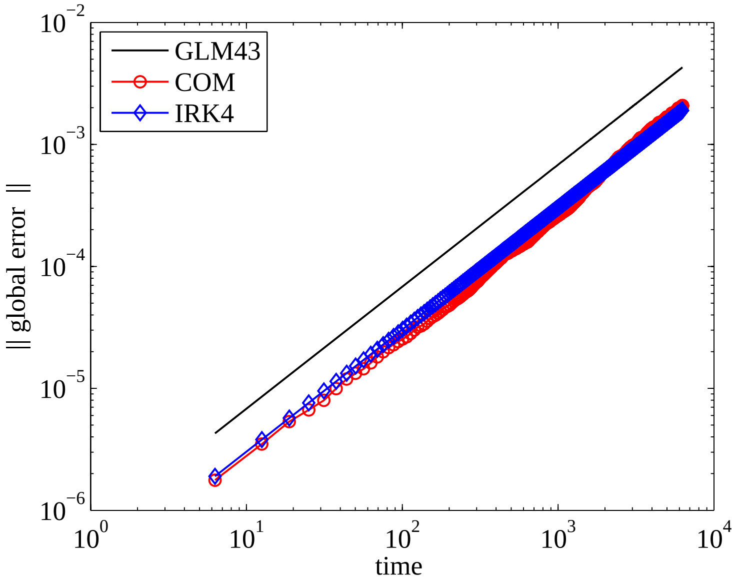


Figure 5.9: Global error for the Kepler problem (e = 0).

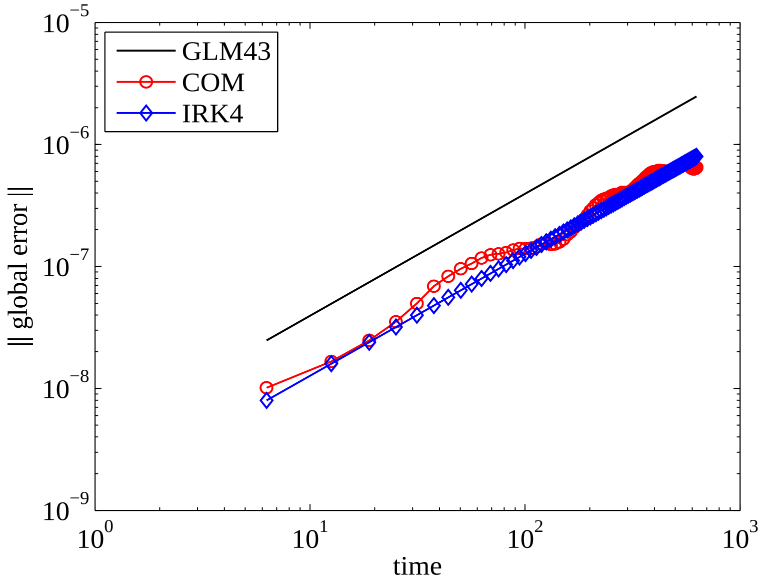


Figure 5.10: Global error for the Kepler problem (e = 0.5).

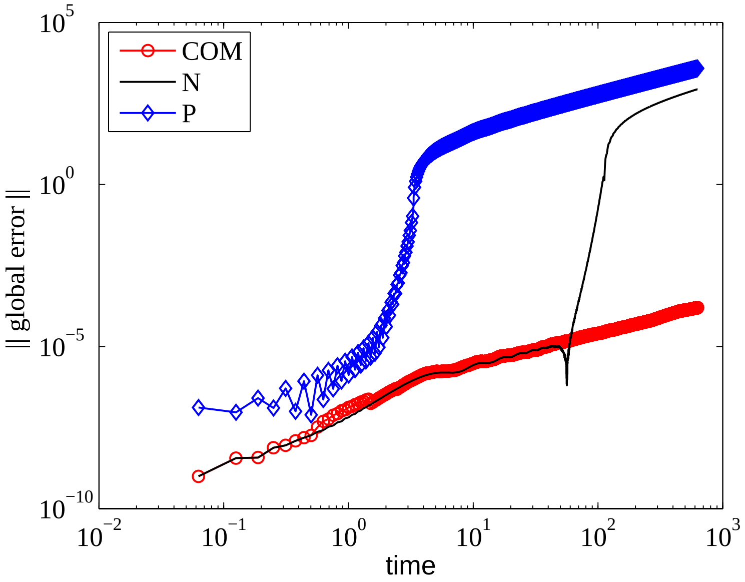


Figure 5.11: Global error for the Kepler problem (e = 0).

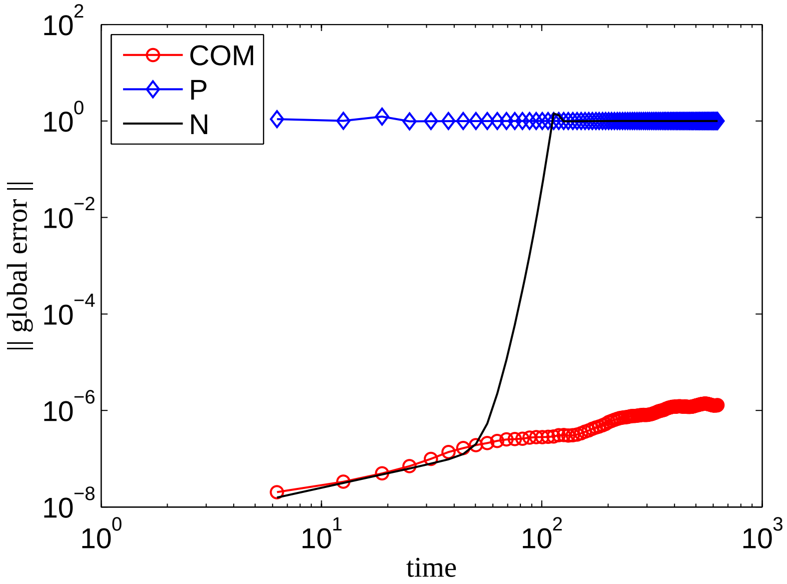


Figure 5.12: Global error for the Kepler problem (e = 0.5).

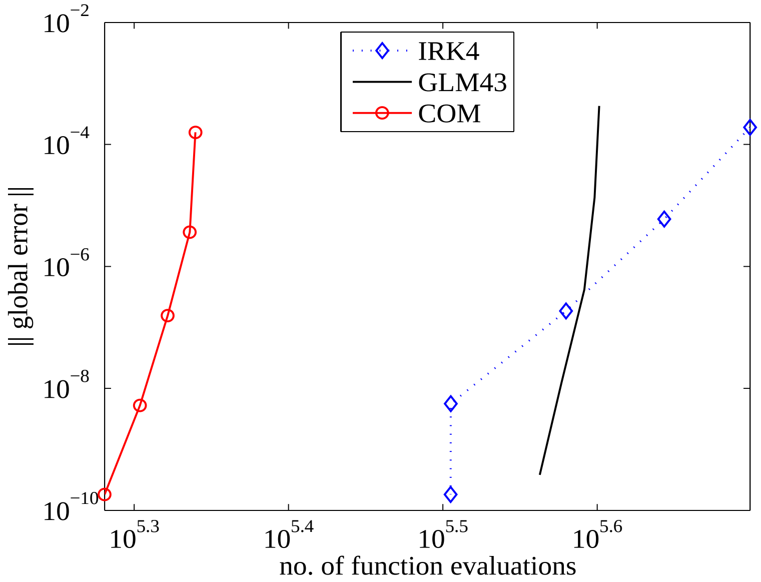


Figure 5.13: No. of function evaluations vs. global error for the Kepler problem (e = 0).

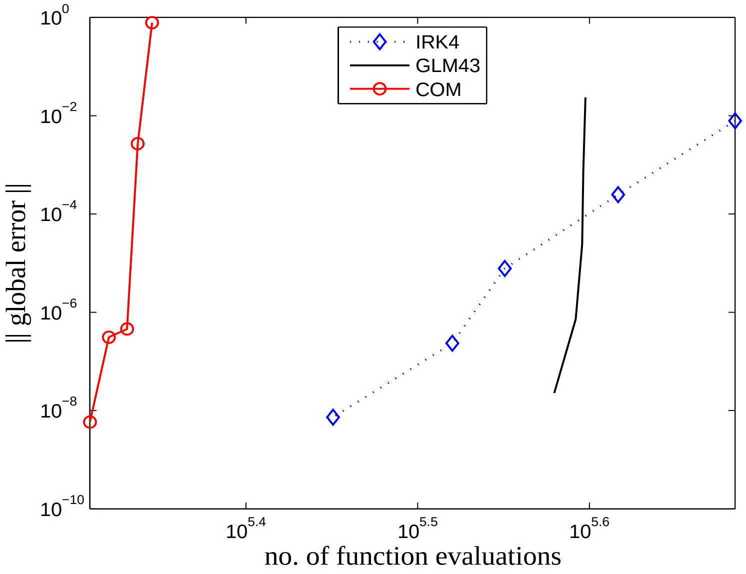


Figure 5.14: No. of function evaluations vs. global error for the Kepler problem (e = 0.5).

h	$IRK4$	COM	$GLM43$
	$\times 10^{-8}$	$\times 10^{-8}$	$\times 10^{-8}$
$\frac{2\pi}{250}$	2.5749841826202	2.4814200779159	5.7479511003845
$\frac{2\pi}{500}$	0.1609427202781	0.1668628110285	0.3591377951417
$\frac{2\pi}{1000}$	0.0100604075192	0.0100796753011	0.0224437601322

Table 5.1: Global error for the Kepler problem (e = 0) over half period.

h	$IRK4$	COM	$GLM43$
	$\times 10^{-3}$	$\times 10^{-3}$	$\times 10^{-3}$
$\frac{2\pi}{100}$	0.190348790675350	0.157793341150169	0.428141240916403
$\frac{2\pi}{200}$	0.005949952292952	0.003630915274365	0.013354383403031
$\frac{2\pi}{400}$	0.000185979799382	0.000156258671458	0.000417095524816
$\frac{2\pi}{800}$	0.000005594508212	0.000005244850396	0.000012497546977
$\frac{2\pi}{1600}$	0.000000181963291	0.000000182358518	0.000000383102910

Table 5.2: Global error for the Kepler problem (e = 0) for 10000 steps.

5.3 Harmonic oscillator

The motion of a unit mass attached to a spring with momentum p and position coordinates q defines a Hamiltonian system

$$q' = p, \qquad p' = -q.$$

The energy is given by

$$H = \frac{p^2}{2} + \frac{q^2}{2}.$$

The exact solution is

$$\begin{bmatrix} p(x) \\ q(x) \end{bmatrix} = \begin{bmatrix} \cos(x) & -\sin(x) \\ \sin(x) & \cos(x) \end{bmatrix} \begin{bmatrix} p(0) \\ q(0) \end{bmatrix}.$$

For the numerical solution of the Harmonic oscillator, a stepsize of 0.01 is taken and the problem is integrated for one million steps. The parasitic solution does not overtake the actual solution and we get energy conservation for methods P and N as shown in Figure 5.15 and Figure 5.16 respectively. Not surprisingly, we get excellent energy conservation for the method $IRK4$ is Figure 5.17, for the method COM in Figure 5.18 and for the method $GLM43$ in Figure 5.19

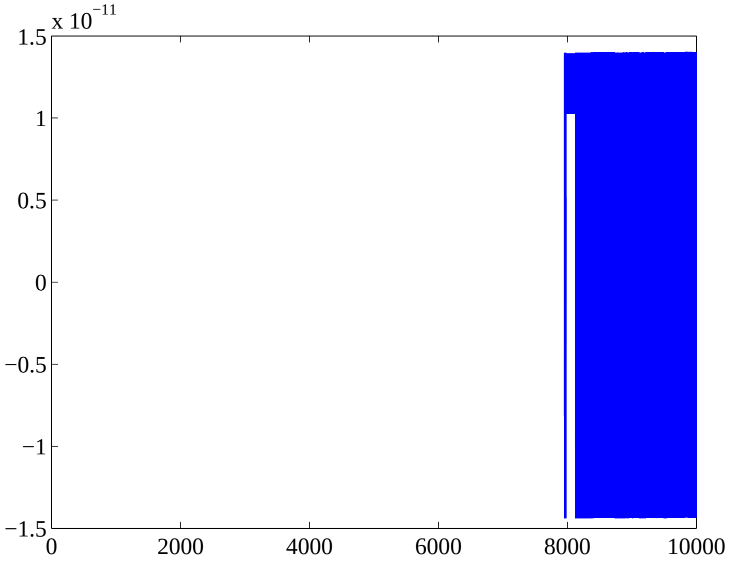


Figure 5.15: Conservation of energy of Harmonic Oscillator by P.

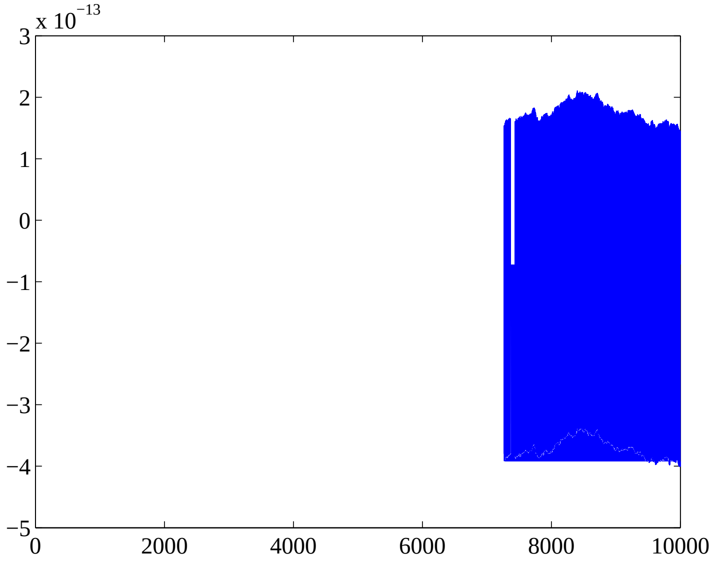


Figure 5.16: Conservation of energy of Harmonic Oscillator by *N*.

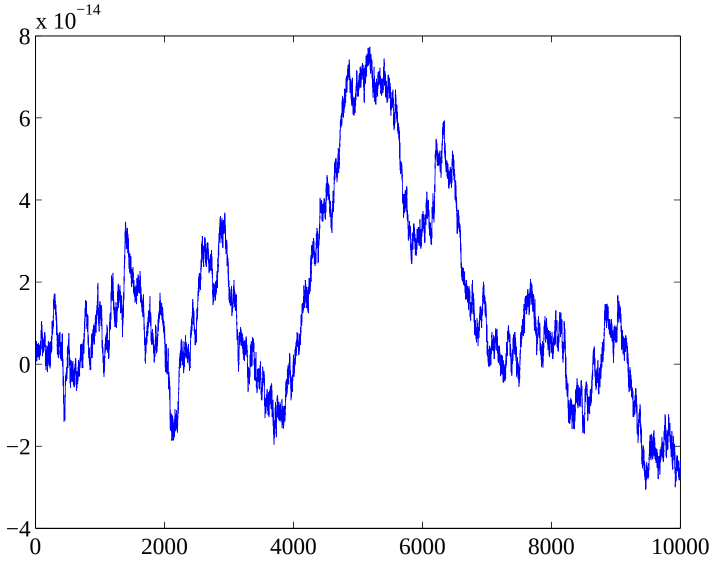


Figure 5.17: Conservation of energy of Harmonic Oscillator by *IRK*4.

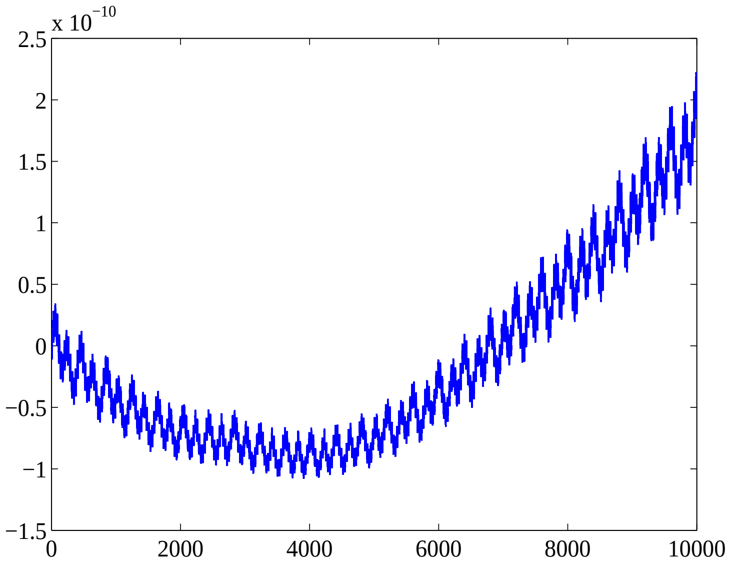


Figure 5.18: Conservation of energy of Harmonic Oscillator by *COM*.

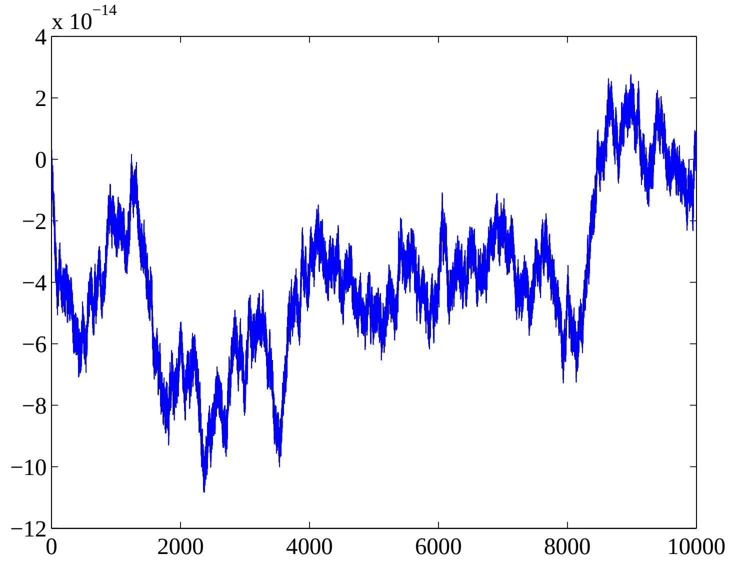


Figure 5.19: Conservation of energy of Harmonic Oscillator by *GLM*43.

5.4 The simple pendulum

We recall from Chapter 1, the equations of motion of the simple pendulum,

$$p' = -\sin(q), \qquad q' = p.$$

The total energy H is a conserved quantity and is given as

$$H = \frac{p^2}{2} - \cos(q).$$

We have seen in Chapter 4, that the simple pendulum problem is capable of having parasitic solutions for initial conditions chosen as $p = 0$, $q = 2.3$. This can be observed from Figure 5.20 and Figure 5.21 where we have applied the method P and the method N with stepsize 0.05, to the simple pendulum problem respectively. Here we have taken 1000 steps only for method P and 2000 steps for method N. However the methods $IRK4$, COM and $GLM43$ have excellent energy conservation for one million steps with the same stepsize, as shown in Figure 5.22, Figure 5.23 and Figure 5.24 respectively.

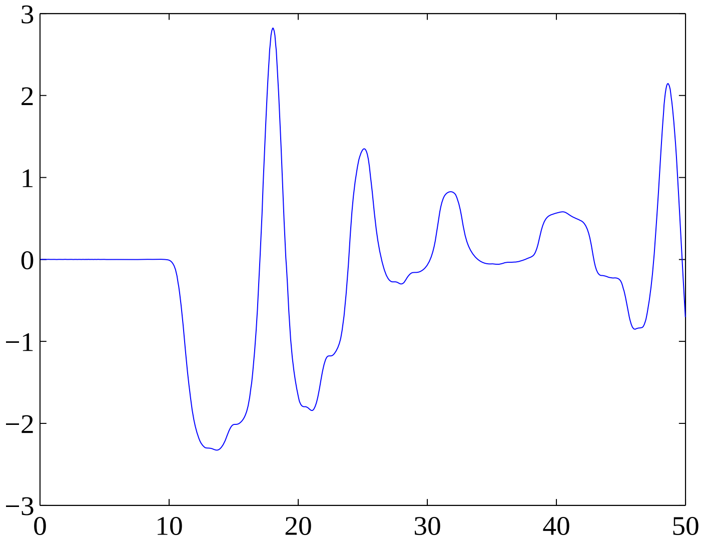


Figure 5.20: Conservation of energy of the simple pendulum by P.

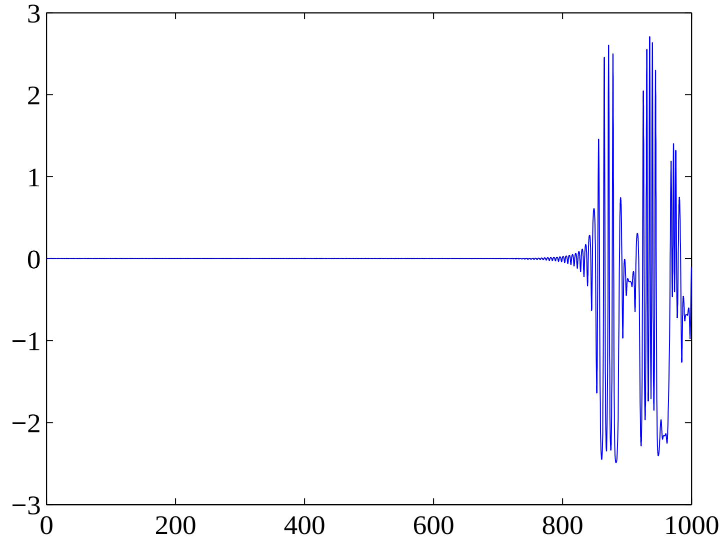


Figure 5.21: Conservation of energy of the simple pendulum by *N*.

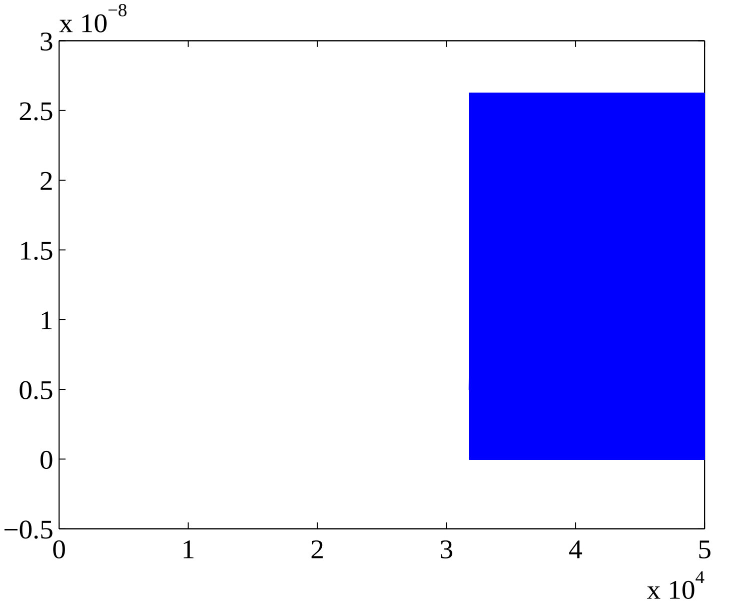


Figure 5.22: Conservation of energy of the simple pendulum by *IRK*4.

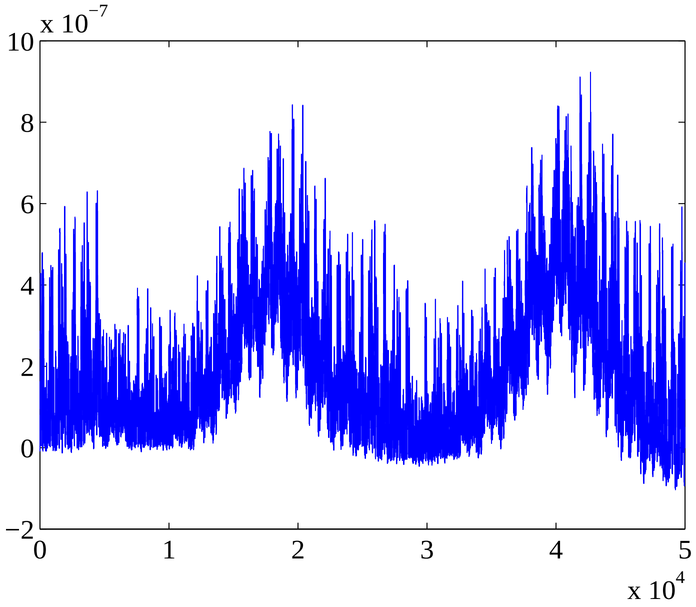


Figure 5.23: Conservation of energy of the simple pendulum by *COM*.

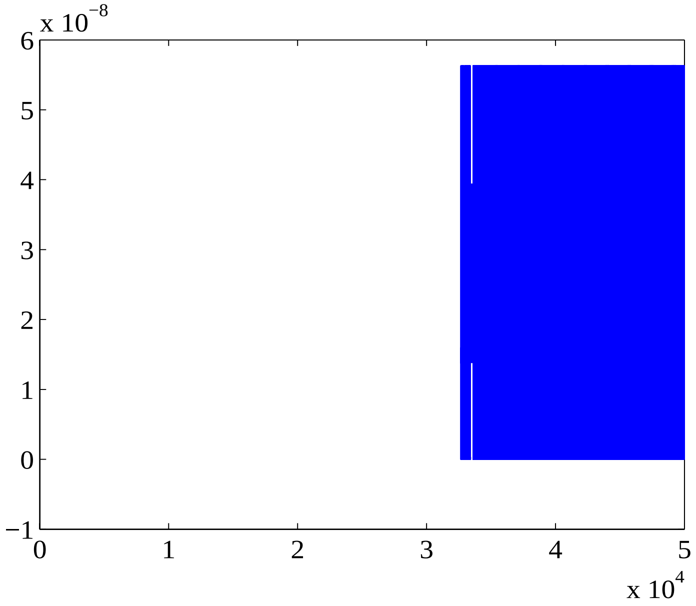


Figure 5.24: Conservation of energy of the simple pendulum by *GLM*43.

5.5 Euler equations for rigid body motion

The mathematical equations governing the motion of a rigid body are recalled from (1.16)

$$\begin{aligned}
\frac{d\omega_x}{dt} &= \frac{I_{yy}-I_{zz}}{I_{xx}}\omega_y\omega_z,\\
\frac{d\omega_y}{dt} &= \frac{I_{zz}-I_{xx}}{I_{yy}}\omega_z\omega_x,\\
\frac{d\omega_z}{dt} &= \frac{I_{xx}-I_{yy}}{I_{zz}}\omega_x\omega_y,
\end{aligned}$$

where $\omega_x, \omega_y, \omega_z$ are the components of angular velocity around the principal axis and I_{xx}, I_{yy}, I_{zz} are principal moment of inertia. The motion of rigid body has the following two underlying quadratic invariants namely, the kinetic energy H and the squared norm of angular momentum A given in (1.19) and (1.20).

We apply the numerical methods for one million steps of stepsize 0.01 to see whether these invariants are preserved by the numerical solution. For the method $IRK4$, the Figure 5.25 shows excellent preservation of these invariants. A similar result is obtained for the method COM as shown in the Figure 5.26 and the method $GLM43$ as shown in Figure 5.27. For the methods P and N, the numerical results is not corrupted by the parasitic solution and we get conservation of invariants as given in Figure 5.28 and Figure 5.29 respectively.

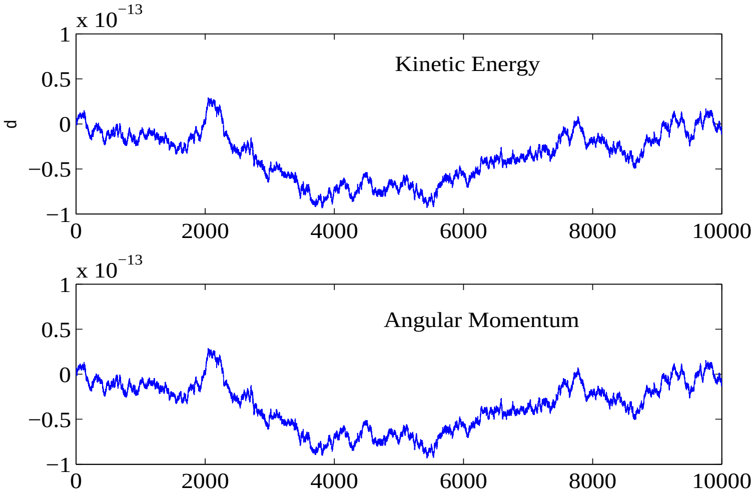


Figure 5.25: Conservation of invariants of Euler rigid body motion by $IRK4$.

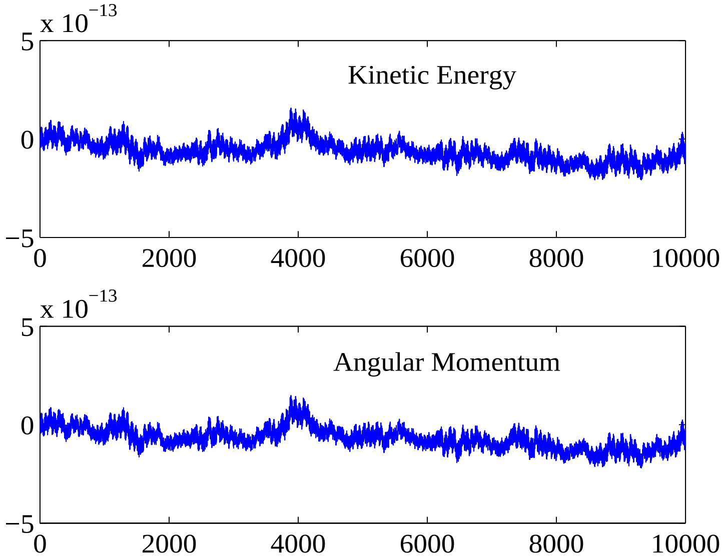


Figure 5.26: Conservation of invariants of Euler rigid body motion by *COM*.

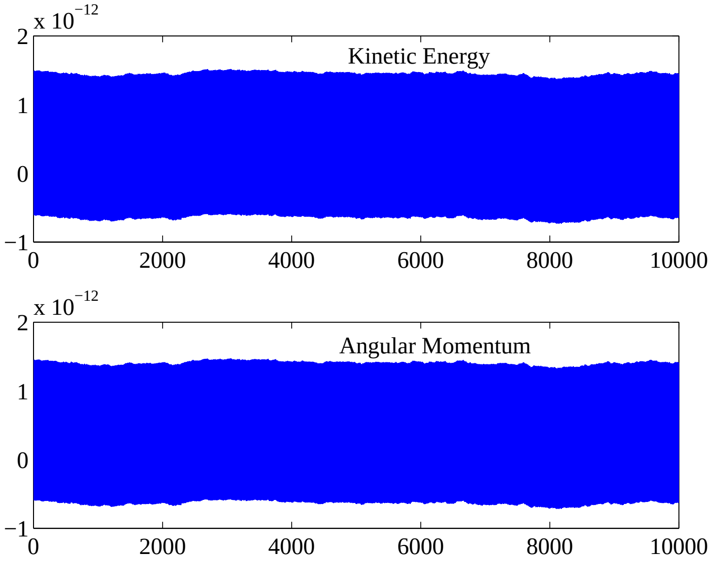


Figure 5.27: Conservation of invariants of Euler rigid body motion by *GLM*43.

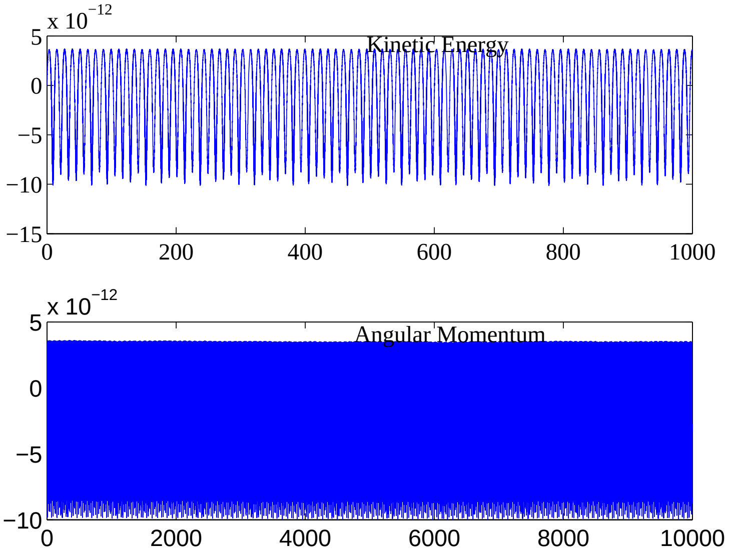


Figure 5.28: Conservation of invariants of Euler rigid body motion by P.

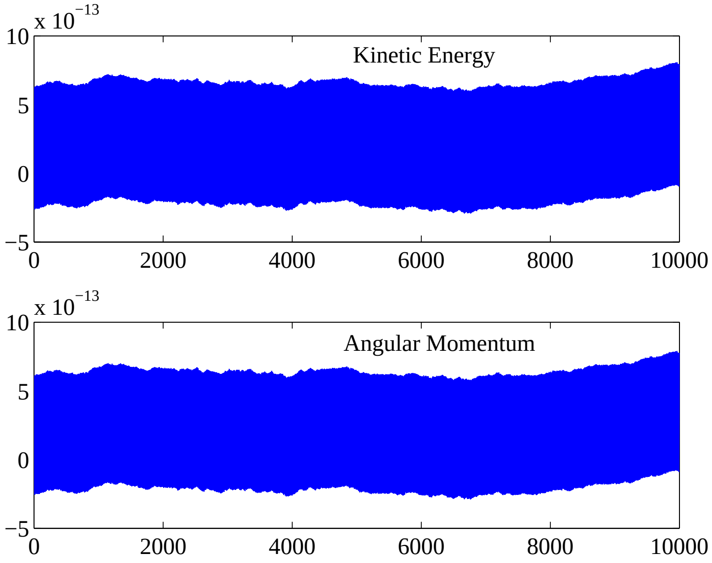


Figure 5.29: Conservation of invariants of Euler rigid body motion by N.

5.6 Differential equations on sphere

We consider a system of ordinary differential equations

$$y' = f(x,y),$$

such that $\|y\|^2$ is constant. The solution of such a system evolves on a sphere. We consider a particular example similar to the one given in Diffman [29], where the solution evolves on a unit sphere. The equations of motion are

$$y' = \begin{bmatrix} 0 & 0.1\sin(x) & -0.2\cos(x) \\ -0.1\sin(x) & 0 & 0.3\sin(2x) \\ 0.2\cos(x) & -0.3\sin(2x) & 0 \end{bmatrix} y = Ay, \qquad y(0) = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad (5.1)$$

where A is a skew-symmetric matrix. Equations of type (5.1) are at the centre of numerical methods for differential equations on Lie groups. We have done certain experiments using the methods outlined at the start of this chapter, to see how close the numerical solution is to the manifold which is a unit sphere. We have always taken 10^5 steps with stepsize of 0.001.

The first experiment is with the method $IRK4$. Figure 5.30 shows the drift of the numerical solution from the unit sphere. The result is consistent with the fact that $IRK4$ is a symplectic method. The second experiment is with the method $GLM43$ and even though it is a G-symplectic method we get excellent results in terms of adherence of the numerical solution to the unit sphere as shown in Figure 5.31. The method COM also performed very well as shown in Figure 5.32. Similar good results are obtained for the method P as shown in Figure 5.33 and for the method N as shown in Figure 5.34.

We study the G-symplectic condition of the method $GLM43$ such that

$$\langle y^{[n]}, y^{[n]}\rangle_G = \langle y^{[n-1]}, y^{[n-1]}\rangle,$$
$$\sum_{i,j=1}^{3} g_{ij}\langle y_i^{[n]}, y_j^{[n]}\rangle = \sum_{i,j=1}^{3} g_{ij}\langle y_i^{[n-1]}, y_j^{[n-1]}\rangle,$$
$$(y_1^{[n]})^2 - (\tfrac{1}{4})(y_2^{[n]})^2 + (\tfrac{1}{4})(y_3^{[n]})^2 = (y_1^{[n-1]})^2 - (\tfrac{1}{4})(y_2^{[n-1]})^2 + (\tfrac{1}{4})(y_3^{[n-1]})^2$$

and we can clearly see that this condition is satisfied in Figure 5.35. Similar results are obtained for the method P in Figure 5.36, whose G-symplectic condition is

$$(y_1^{[n]})^2 + (\tfrac{3+2\sqrt{3}}{3})(y_3^{[n]})^2 = ((y_1^{[n-1]})^2 + (\tfrac{3+2\sqrt{3}}{3})(y_3^{[n-1]})^2$$

For the method N, again similar results are obtained as shown in Figure 5.37 with the same G-symplectic condition as of P but with $-\sqrt{3}$ instead of $\sqrt{3}$.

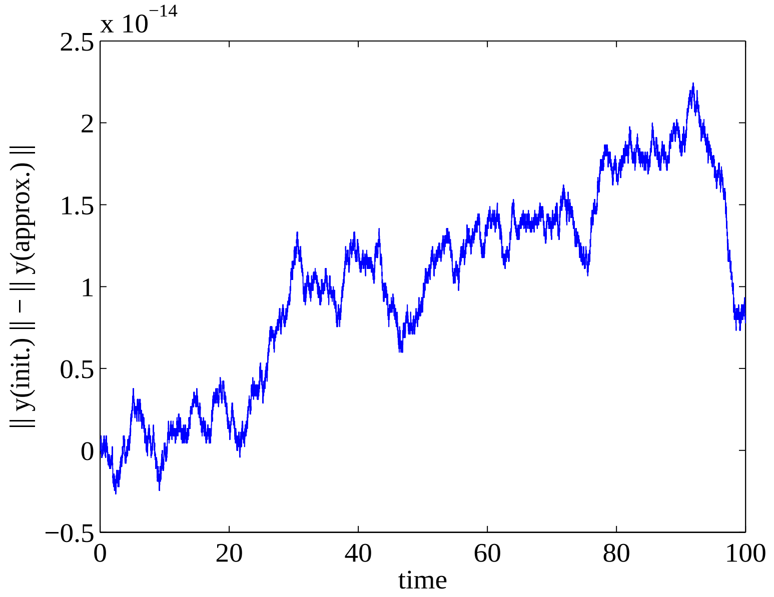


Figure 5.30: The drift from unit sphere of solution of the method *IRK*4.

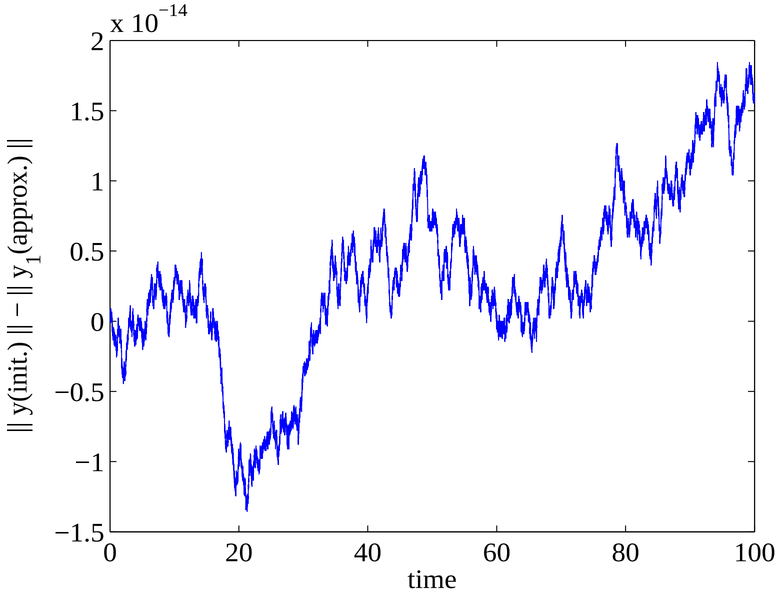


Figure 5.31: The drift from unit sphere of first component of the method *GLM*43.

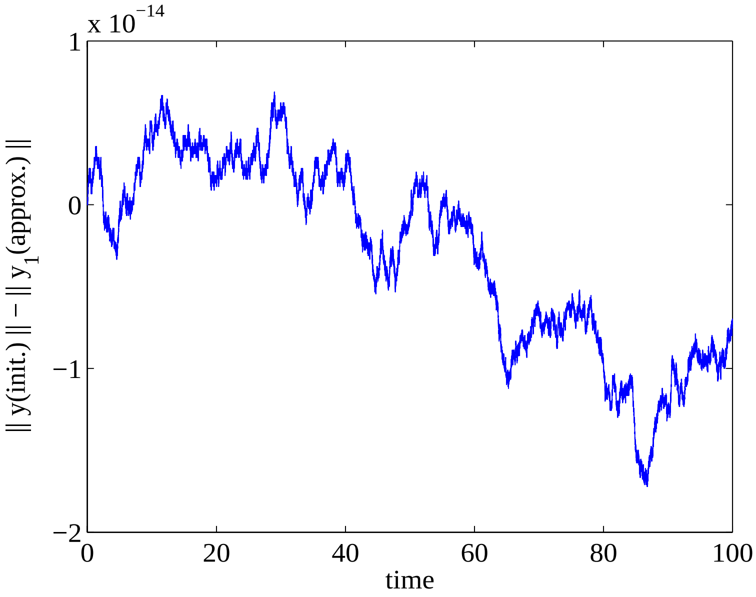


Figure 5.32: The drift from unit sphere of first component of the method *COM*.

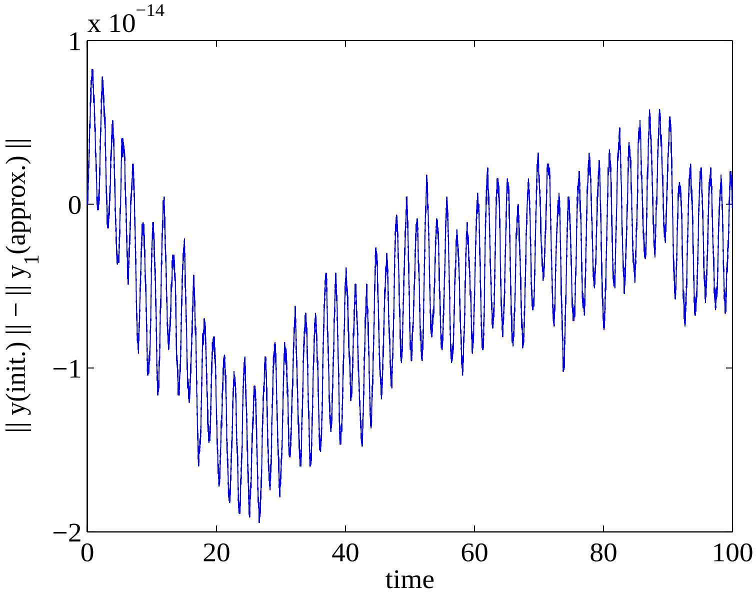


Figure 5.33: The drift from unit sphere of first component of the method *P*.

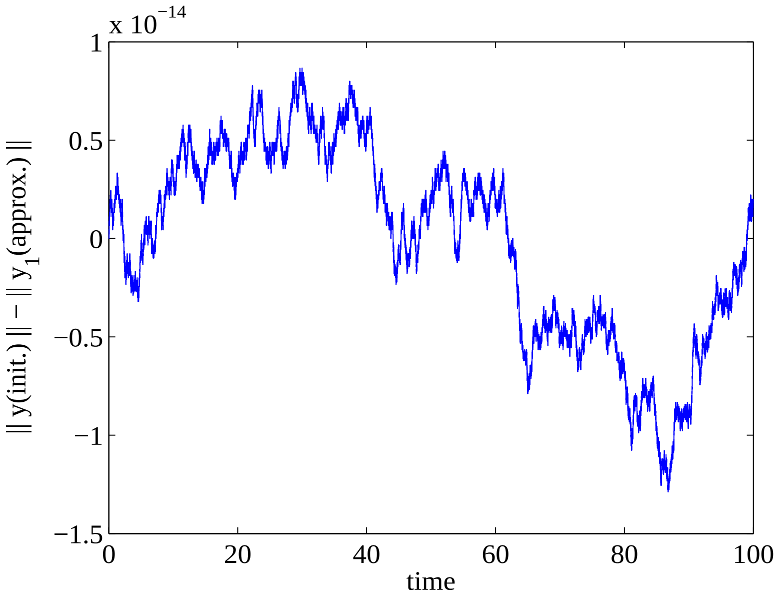


Figure 5.34: The drift from unit sphere of first component of the method N.

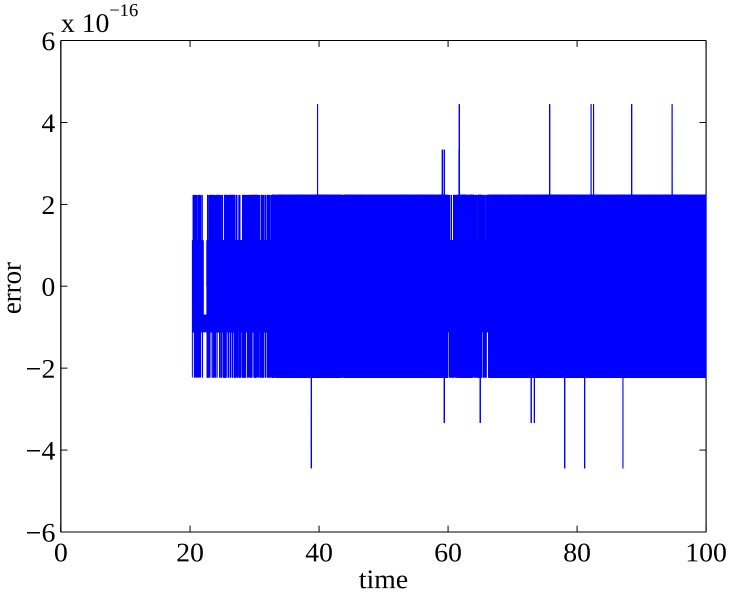


Figure 5.35: G-symplectic error of the method $GLM43$.

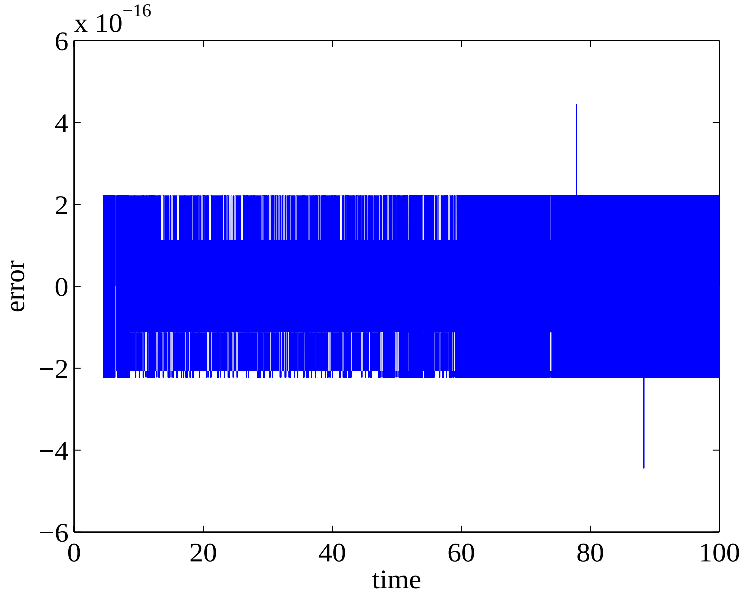


Figure 5.36: *G*-symplectic error of the method *P*.

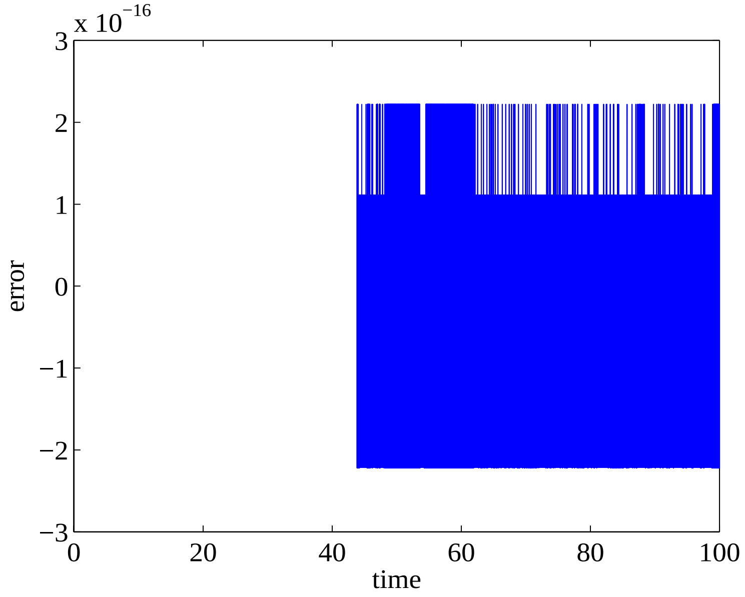


Figure 5.37: *G*-symplectic error of the method *N*.

The results of these experiments are encouraging. The *G*-symplectic methods performed well compared to the symplectic Runge–Kutta method and even better for analysing certain aspects of some problems despite that they are multivalue methods. These methods have a potential for being used as geometric integrators for long time integration of conservative systems without losing many qualitative features of the underlying system.

Chapter 6

Conclusions and future work

Numerical integration of ordinary differential equation systems with quadratic invariants is the main topic explored in this thesis with an emphasis on *G*-symplectic general linear methods. The multivalue nature of these methods contributes to the corruption of numerical solution by the parasitic solution component. Two approaches have been employed in order to contain this effect. The first is to compose methods with parasitic growth parameters having opposite signs. The second is to construct methods for which the parasitic growth parameter is zero by design.

The structure of the thesis is devised with an objective to provide a coherent and concise step by step understanding of the main topic. In Chapter 1, an introduction of Hamiltonian systems and related conservative problems was provided. A review of the traditional methods for numerically solving ordinary differential equation systems and conservative problems was given. Chapter 2 deals with a detailed study of the numerical methods viz. Runge–Kutta methods, multistep methods and general linear methods with an aim to get an insight of their working and interconnectivity. This was further extended to Chapter 3, where stability and symplecticity of the methods were discussed in particular. Although symplectic Runge–Kutta methods were developed independently, there is an intricate relation between their non-linear stability and symplecticity. This also extends to *G*-symplectic general linear methods and is explored in Chapter 4.

The ability of general linear methods to solve practical problems is hampered by the parasitic solutions ingrained in such methods. This also extends to *G*-symplectic general liner methods for the solution of those conservative problems which are capable of having parasitism. The growth of parasitic solution, therefore depends on the numerical method and the problem it is trying to solve. Their relation was studied in this thesis. As explained earlier, a successful attempt was made to combine two G-symplectic methods of order four in a sequence with parasitic growth parameter of one method having an opposite sign to that of other. Such a sequence was implemented to ensure that the accumulated parasitic growth parameter does not grow out of a specified bound. Excellent energy conservation is observed by the composition method for solving Hamiltonian problems. More general quadratic invariants were also well preserved, including especially the case where the solution lies on a manifold of unit sphere. The composition proved to be useful not only in containing the effect of parasitism but also to be more efficient than the Runge–Kutta method in terms of number of function evaluations. This is supported by the experiments. Moreover the global error of the composition method is comparable to the Runge–Kutta method, although the former is a multistep multivalue method and the later is a one-step method.

A fourth order symmetric G-symplectic general linear method was constructed with no parasitism. This necessarily had four stages and three output values, since we have observed that a G-symplectic method cannot avoid parasitic corruption with only two stages and two output values by design. Complex numbers were chosen for matrices B and V such that the second and third rows of matrices B and V were complex conjugates. Because the eigenvalues of the matrix V should lie on a unit disc, we took V to be a diagonal matrix with entries 1, z and $\overline{z}$ such that $|z| = 1$. We took a particular value of $z = \exp(2\pi i/3)$, whereas methods with $z = i$ are also known to exist. A transformation was applied to obtain the method with real entries. The starting method had four explicit stages for cheap implementation and three output values with only one input value given by the initial condition of the differential equation system. The matrix B for the starting method also had second and third rows to be complex conjugate and this was also transformed to real numbers via the transformation. The coefficients of matrix B were found from the algebraic analysis of the order with Butcher series. The starting method and the actual method are implemented in a way equivalent to effective order approach and we used its tools in our search for the starting method.

The G-symplectic general linear method was constructed by making sure that the quadrature order and the stability order was four, and the experiments confirmed that the method itself turned out to be of order four. The parasitic growth parameter was essentially taken to be zero and we saw energy conservation for Hamiltonian systems over long time integration by the numerical solution without being distorted by the parasitic solutions. This method was more efficient than the two stage, fourth order implicit Gauss Runge–Kutta method in the long run considering the number of function evaluations for a particular problem.

There are several open questions we would like to explore. In the course of finding a remedy for the corruption of numerical solution by parasitism, the composition of methods turned out to be an effective approach. Whether the sequence of methods used in composition, we have considered and implemented, provides an optimal solution, is still unknown. The error constants of the methods P and N are different and we have taken same stepsize for both methods. An option that may lead to better performance and more control is to use a block of sequence of methods with a combined greater stepsize, while internally the block may have used different stepsizes. Also, the consequence of having different G matrices for the composed methods is also open to investigation. We can also do an analysis of the order of the composed methods with many N methods and an occasional P method.

For the construction of a G-symplectic method without parasitism, we took several choices for convenience. For example, the value of z in matrix V is taken to be $\exp(2\pi i/3)$. These methods belong to a larger class, where different values of z yields different methods. It is not known what is the optimal choice of z. The choice of z will have an effect on the starting method and we may be able to attain higher accuracy.

Hamiltonian systems are often separable as is the case with the examples in our experiments. However our methods are geared towards general Hamiltonian problems. The advantage of solving different variables of the system with different numerical methods and still avoiding the parasitic growth is also open to investigation. The comparison of general G-symplectic general linear methods and the partitioned G-symplectic methods can shed light on their suitability and effectiveness.

The methods and techniques derived in this thesis are novel ideas. They have desired features of G-symplecticity, time reversal symmetry and no parasitism. Their lucid construction provided in this thesis can be manipulated for higher order methods. The experimental evidence points out that they may be suitable to form the kernel of an efficient solver for conservative systems.

Bibliography

[1] R. Alexander. Diagonally implicit Runge–Kutta methods for stiff ODEs. *SIAM J. Numer. Anal.*, 14:1006–1021, 1977.

[2] F. Bashforth and J. C. Adams. An attempt to test the theories of capillary action by comparing the theoretical and measured forms of drops of fluid, with an explanation of method of integration employed in constructing the tables which give the theoretical forms of such drops. *Cambridge University*, 1883.

[3] G. Benettin and A. Giorgilli. On the Hamiltonian interpolation of near to identity symplectic mappings with application to symplectic integration algorithms. *J. Statis. Phys.*, 74:1117–1143, 1994.

[4] K. Burrage and J. C. Butcher. Stability criteria for Implicit Runge–Kutta Methods. *SIAM J. Numer. Anal.*, 16:46–57, 1979.

[5] K. Burrage and J. C. Butcher. Non-linear stability of general class of differential equation methods. *BIT*, 20:185–203, 1980.

[6] J. C. Butcher. Coefficients for the study of Runge–Kutta integration processes. *J. Aust. Math. Soc.*, 3:185–201, 1963.

[7] J. C. Butcher. Implicit Runge–Kutta processes. *Math. Comp.*, 18:50–64, 1964.

[8] J. C. Butcher. On the convergence of numerical solution to ordinary differential equations. *Math. Comp.*, 20:1–10, 1966.

[9] J. C. Butcher. The effective order of Runge–Kutta methods. *Lecture Notes in Math.*, 109:133–139, 1969.

[10] J. C. Butcher. Order and effective order. *Applied Numerical Mathematics*, 28:179–191, 1998.

[11] J. C. Butcher. General linear methods. *Acta Numerica*, 15:157–256, 2006.

[12] J. C. Butcher. *Numerical Methods for Ordinary Differential Equations*. Wiley, second edition, 2008.

[13] J. C. Butcher and P. Chartier. A generalization of singly-implicit Runge–Kutta methods. *Applied Numerical Mathematics*, 24:343–350, 1997.

[14] J. C. Butcher and D. J. L. Chen. Effective order Diagonally Extended Singly-implicit Runge–Kutta methods for stiff differential equations. *unpublished work*.

[15] J. C. Butcher and M. T. Diamantakis. DESIRE: diagonally extended singly implicit Runge–Kutta effective order methods. *Numer. Algorithms*, 17:121–145, 1998.

[16] J. C. Butcher and L. L. Hewitt. The existence of symplectic general linear methods. *Numer. Algorithms*, 51:77–84, 2009.

[17] G. J. Cooper. Stability of Runge–Kutta methods for trajectory problems. *IMA J. Numer. Anal.*, 7:1–13, 1987.

[18] M. Crouzeix. Sur la B-stabilité des méthodes de Runge–Kutta. *Numer. Math.*, 32:75–82, 1979.

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

[20] G. Dahlquist. Convergence and stability in the numerical integration of ordinary differential equations. *Math. Scand.*, 4:33–53, 1956.

[21] G. Dahlquist. Stability and error bounds in the numerical integration of ordinary differential equations. *Trans. of the Royal Inst. of Techn. Stockholm, Sweden.*, 130, 1959.

[22] G. Dahlquist. A special stability property for linear multistep methods. *BIT*, 3:27–43, 1963.

[23] G. Dahlquist. Error analysis of a class of methods for stiff nonlinear initial value problems. *Numerical Analysis, Dundee, Lecture Notes in Mathematics*, 506:60–74, 1976.

[24] G. Dahlquist. G-stability is equivalent to A-stability. *BIT*, 18:384–401, 1978.

[25] G. Dahlquist. On one-leg multistep methods. *SIAM J. Numer. Anal.*, 20(6):1130–1138, 1983.

[26] G. Dahlquist, W. Liniger, and O. Nevanlinna. Stability of two-step methods for variable integration steps. *SIAM J. Numer. Anal.*, 20:1071–1085, 1983.

[27] Hans Van de Vyver. An embedded exponentially fitted Runge-Kutta-Nyström method for the numerical solution of orbital problems. *New Astronomy*, 11:577–587, 2006.

[28] T. Eirola and J. M. Sanz-Serna. Conservation of integrals and symplectic structure in the integration of differential equations by multistep methods. *Numer. Math.*, 61:281–290, 1992.

[29] K. Engø, A. Marthinsen, and H. Z. Munthe-Kaas. Diffman User's Guide Version 2.0. *Department of Informatics, University of Bergen, Norway.*, 2000.

[30] E. Hairer, C. Lubich, and G. Wanner. *Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations*. Springer, second edition, 2005.

[31] E. Hairer and G. Wanner. On the Butcher group and general multivalue methods. *Computing (Arch. Elektron. Rechnen)*, 13:1–15, 1974.

[32] A. T. Hill. (private communication).

[33] A. Iserles. *A First Course in the Numerical Analysis of Differential Equations*. Cambridge University Press, second edition, 2008.

[34] A. Iserles, H. Z. Munthe-Kaas, S. P. Nørsett, and A. Zanna. Lie-group methods. *Acta Numerica*, 9:215–365, 2000.

[35] U. Kirchgraber. Multistep methods are essentially one-step methods. *Numer. Math.*, 48:85–90, 1986.

[36] F. M. Lasagni. Canonical Runge–Kutta methods. *ZAMP*, 39:952–953, 1988.

[37] W. Liniger. A criterion for A-Stability of Linear Multistep Integration Formulae. *J-Computing*, 3(4):280–285, 1968.

[38] M. Lopez-Marcos, J. M. Sanz-Serna, and R. D. Skeel. Cheap Enhancement of Symplectic Integrators. *D. F. Griffiths and G. A. Watson editors, Numerical Analysis 1995*, pages 107–122, 1996.

[39] W. E. Milne. Numerical integration of ordinary differential equations. *Amer. Math. Monthly*, 33:455–460, 1926.

[40] W. E. Milne. A note on the numerical integration of differential equations. *J. Research Nat. Bur. Standards*, 43:537–542, 1949.

[41] F. R. Moulton. *New methods in Exterior Ballistics*. University of Chicago, 1926.

[42] S. P. Nørsett. Runge–Kutta methods with a multiple real eigen value only. *BIT*, 16:388–393, 1976.

[43] E. J. Nyström. Über die numerische Integration von Differentialgleichungen. *Acta Soc. Sci. Fennicae*, 50:1–54, 1925.

[44] J. M. Sanz-Serna. Runge–Kutta schemes for Hamiltonian systems. *BIT*, 28:877–883, 1988.

[45] J. M. Sanz-Serna and L. Abia. Order Conditions for Canonical Runge-Kutta Schemes. *SIAM Journal on Numerical Analysis*, 28:1081–1096, 1991.

[46] J. M. Sanz-Serna and M. P. Calvo. *Numerical Hamiltonian Problems*. Chapman and Hal, first edition, 1994.

[47] D. Stoffer. General linear methods: connection to one step methods and invariant curves. *Numer. Math.*, 64:395–408, 1993.

[48] Y. B. Suris. Preservation of symplectic structure in the numerical solution of Hamiltonian systems, Numerical Solution of Differential Equations (S. S. Filippov, ed.). *Akad. Nauk. USSR, (In Russian)*, pages 148–160, 1988.

[49] Y. B. Suris. Canonical transformations generated by methods of Runge–Kutta type for the numerical integration of the system $x'' = \frac{\partial u}{\partial x}$. *Zh. Vychisl. Mat. Fiz. (In Russian)*, 29:202–211, 1989.

[50] Y. F. Tang. The symplecticity of multistep methods. *Coputers Math. Applic.*, 25:83–90, 1993.

[51] Daniel S. Watanabe and Qasim M. Sheikh. One-leg Formulas for Stiff Ordinary Differential Equations. *SIAM J. Sci. and Stat. Comput.*, 5(2):489–496, 1984.

[52] J. H. Wilkinson. *The Algebraic Eigenvalue Problem*. Oxford University Press, first edition, 1965.