# Control of Parasitism in G-Symplectic General Linear Methods via Symmetric Projection

**Amber Masood**


A dissertation submitted in partial fulfillment of the requirements

for the degree of Master of Philosophy in Mathematics

Supervised by

**Dr. Yousaf Habib**

**School of Natural Sciences**

National University of Sciences and Technology

Islamabad, Pakistan

March 2015

FORM TH-4

# National University of Sciences & Technology

## M.Phil THESIS WORK

We hereby recommend that the dissertation prepared under our supervision by: AMBER MASOOD, Regn No. NUST201260291MCAMP78012F Titled: Control of Parasitism in G-Symplectic General Linear Methods via Symmetric Projection be accepted in partial fulfillment of the requirements for the award of **M.Phil** degree.

**Examination Committee Members**

1. Name: DR. MUHAMMAD ASIF FAROOQ Signature: _____
2. Name: DR. MERAJ MUSTAFA HASHMI Signature: _____
3. Name: DR. MUHAMMAD SAFDAR Signature: _____
4. Name: DR. MUHAMMAD AMER QURESHI Signature: _____

Supervisor's Name: DR. YOUSAF HABIB Signature: _____

_____ 24-03-2015
Head of Department Date

**COUNTERSINGED**

Date: 24/03/15 _____
**Dean/Principal**

**Dedicated to**

***My Loving Parents***

# Acknowledgement

*Only those fear Allah, from among His servants, who have knowledge. Indeed, Allah is Exalted in Might and Forgiving. (Quran 35:28)*

Alhamdulillah, all praises to Allah Almighty for the strengths and His blessings in completing this work successfully. Special appreciation goes to my supervisor, Dr. Yousaf Habib, for his supervision and constant support. His invaluable help of constructive comments and suggestions throughout the experimental and thesis work has contributed to the success of this research. This thesis would not have been possible without his supervision and persistent help. I would like to thank him for giving me an opportunity to work with him on an interesting topic. My acknowledgement also goes to all faculty members and staff of SNS for providing me a peaceful working environment. Finally, my deepest gratitude goes to my beloved parents and also to my siblings for their endless love, prayers and encouragement.

# Abstract

Numerical methods applied to the solution of differential equations on manifolds based on projection methods are considered to be very effective in many practical problems. It is however observed that geometric properties (such as symplecticity or reversibility) are usually destroyed by numerical discretization, even when the underlying method is symplectic or symmetric. Therefore symmetric projection methods are introduced, which allows the preservation of symplectic invariants for Hamiltonian systems over long time interval. In this thesis we are using multi-value multi-derivative methods. Since these methods can introduce the parasitic components in the numerical solution, symmetric projection methods illustrates excellent results in eliminating the effect of parasitism and in projecting the numerical solution on the invariant manifold. Numerical results using MATLAB verify these claims.

# Contents

# Chapter 1

# Introduction

Numerical methods like the Euler's method, the most extensively used Runge-Kutta methods (RK), the multi-step methods (Adam's and predictor-corrector methods) and the wider class of methods termed as 'General linear methods' are used for numerical integration of ordinary differential equations. However, when solving dynamical systems it is desirable to use such numerical methods which are suitable for the long time integration. Structure preserving algorithms are methods of choice for integrating Hamiltonian systems.

Hamiltonian systems, in addition to being energy preserving (i.e. remains constant along solutions of the system) accompanies another remarkable property which is known to be as the symplectic property. Symplecticity means that the variational equation conserves quadratic invariants [2]. More specifically symplectic map is area and volume preserving. It is well understood that symplectic one-step methods, which when applied to Hamiltonian systems, preserves energy over long time interval and exactly preserves the quadratic first integrals. Symplectic one-step methods including Runge-Kutta methods provides a good example in this regard. Traditional one-step numerical methods for Hamiltonian problems are symplectic RK methods [2, 11, 13]. Symplectic RK methods allows exact conservation of such quadratic invariants. Recently, G-symplectic general linear methods (GLMs) have been considered for approximately preserving underlying invariants [12, 15].

Certain classes of multi-value methods also preserves the energy of Hamiltonian sys-

tems over long times. Much effort is put on the construction of multi-step methods by Feng Kang [8] with the aim of achieving long time integration. Conservative properties of these linear multi-step methods were studied by Eirola and Sanz-Serna [3], who proved that all time-symmetric methods posses the G-symplectic property. Now here the question arises whether G-symplecticity also plays an effective role in long time integration of numerical results? The answer to this question is focused in [19] by Butcher. However, one should be careful because the long-time behavior of multi-step methods is determined by their underlying one-step methods.

Inspired by the work done on multi-step methods (see [3]), the term symplecticity for general linear methods has been broaden to as G-symplecticity, which was proposed by Hairer in [14]. G-symplectic general linear methods, a class of multi-stage and multi-value methods, are constructed to solve Hamiltonian differential equations numerically and considered to be an efficient technique in integrating the solutions of general separable and non-separable Hamiltonian problems. Such methods are regarded to be an effective approach in approximately preserving energy and symplectic invariants over long time intervals. If a general linear method is to solve a conservative problem like Hamiltonian system having the property,

$$\langle y, f(y)\rangle = 0. \tag{1.0.1}$$

then we would require

$$\langle y^{[n]}, y^{[n]}\rangle_G = \langle y^{[n-1]}, y^{[n-1]}\rangle_G. \tag{1.0.2}$$

Then $G$-norm defined by such an inner product is,

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

The condition (1.0.2) is similar to the symplectic condition for one-step method but accounts for $r$ input and output vectors and G is a symmetric $r \times r$ matrix.

Besides long-time integration of multi-value methods, an additional complication related to these methods is that they also suffer from parasitism which inevitably corrupts the numerical solutions. For this purpose parasitism control strategies have been suggested in [16, 20]. We present another strategy to control parasitism in general linear methods by using projection methods [8].

Standard projection methods can achieve excellent results in conserving invariants but can also destroy long time behaviour of the numerical solution. The basic idea of standard projection method is that we first apply the numerical method $\Phi$ taking the initial value $y_n$ to $\tilde{y}_{n+1}$ and then project it back onto the invariant manifold $I(y_n) = 0$ to get $y_{n+1}$. The algorithm then reads as,

- Numerical integration step: $\tilde{y}_{n+1}$=$\Phi(y_n)$.
- Projection: $y_{n+1}$=$\tilde{y}_{n+1}$+$I'(y_{n+1})^T\lambda$.

Where $\lambda = \frac{I(y_0)-I(\tilde{y}_n)}{I'(\tilde{y}_n)\cdot I'(\tilde{y}_n)}$. In order to evaluate $\lambda$, one may need to have a numerical method other than the base method $\Phi$. It is known that even in the case where the underlying method is symplectic or symmetric, numerical discretization with the above projection algorithm destroys the geometric properties and makes it inappropriate for long-time integrations therefore symmetric projection methods are introduced.

Symmetric projection methods are considered to be an efficient approach in conserving invariants and ensuring symmetry of the algorithm by applying perturbation step at the start of each integration step. It also bounds the parasitic effects of the numerical solutions. The idea of one step of symmetric projection algorithm from $y_n$ to $y_{n+1}$ is to perturb $y_n$ to get $\tilde{y}_n$, followed by one step of numerical method $\Phi$ taking $\tilde{y}_n$ to $\tilde{y}_{n+1}$ and finally projecting the numerical solution $\tilde{y}_{n+1}$ onto the invariant manifold $I(y_n) = 0$ to get $y_{n+1}$. Under certain conditions this procedure ensures that overall algorithm is symmetric. Thus for preserving the invariant $I(y_n) = 0$, the algorithm reads,

- Perturbation: $\tilde{y}_n$=$y_n$+$I'(y_n)^T\lambda$.
- Numerical integration step: $\tilde{y}_{n+1}$=$\Phi(\tilde{y}_n)$.
- Projection: $y_{n+1}$=$\tilde{y}_{n+1}$+$I'(y_{n+1})^T\lambda$.

To ensure symmetry, the method $\Phi$ should be symmetric together with same $\lambda$ in the perturbation and projection steps. We have used Newton method to evaluate the impicit equation in the projection step. General linear methods being multi-value

in nature have several input and output values. To apply the symmetric projection algorithm, we perturb the first component of input vector as this component represents the actual solution and rest as parasitic solutions. This is then followed by one step of the general linear method and finally the first component of the output vector is projected onto the invariant manifold.
In this thesis we have considered two different general linear methods from [12] having parasitic growth factors with opposite signs. We have used one general linear method to calculate $\lambda$ in the perturbation step. We call it the perturbation method. We have used other general linear method as the base method and result is then projected onto the manifold. This procedure provides good invariant preservation. Furthermore, we have also used same general linear method as perturbation method and as base method and this leads to better results for preserving underlying invariants.

## 1.1 Numerical methods

There are number of ordinary differential equations (ODEs) that are difficult to solve analytically, therefore solving these ordinary differential equations numerically is effective. Classical numerical methods which are commonly used are one-step methods. The most simple one-step method is Euler's method which is formulated as:

$$y_n = y_{n-1} + hf(y_{n-1}) \tag{1.1.1}$$

where $h$ is step-size and $f$ represents the slope. The idea of Euler's method is that it approximates the solution $y_i$ at time $t_i$ dividing the interval $[t_{n-1}, t_n]$ (which is to be integrated over) into subintervals of size $h_i$, where $i$ denotes the number of steps. This method uses linear approximation along with tangent line to move from one point on the solution curve to the next. Since Euler's method is only accurate upto first order, therefore, to achieve higher order accuracy, Runge-Kutta method are used. The general form of an $s$-stage Runge-Kutta methods defined by three

sets of parameters $a_{ij}$, $b_i$ and $c_i$ is:

$$\begin{aligned} Y_i &= y_{n-1} + h\sum_{j=1}^{s} a_{ij} f(Y_j), \quad i = 1, 2, \cdots, s, \\ y_n &= y_{n-1} + h\sum_{i=1}^{s} b_i f(Y_i). \end{aligned} \tag{1.1.2}$$

where $Y_i$ are stages and $y_n$ are the output values of the actual solution $y(t_n)$. Here $A = (a_{ij})$ denotes $s \times s$ matrix and the quadrature weights $b_i = (b_l, ..., b_s)^T$ is $s \times 1$ column vector. The idea of Runge-Kutta methods is that we first evaluate $s$-stage values $Y_1, Y_2, \cdots, Y_s$ and then calculate output values using linear combination of stage derivatives. The Runge-Kutta methods can be represented by Butcher tableau as:

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

where $c_i = \sum_{j=1}^{s} a_{ij}$ are the nodes or abscissa of the method at which the stages $Y_i$ are evaluated. The coefficients $a_{ij}$ and $b_i$ are calculated from abscissa $c_i$ in a way to ensure order of method. Runge-Kutta methods are explicit if $a_{ij} = 0$ for $i \leq j$ otherwise they are implicit.

Linear multi-step methods constitute an important class of numerical integrators for ordinary differential equations and particular methods are well suited for solving stiff and non-stiff equations. A linear multi-step method uses more than one information from previous time-steps while approximating the solution of $y' = f(y(t))$ with $y(t_0) = y_0$ at current time-steps. The standard form of k-step linear multi-step methods is:

$$y_n = \sum_{i=1}^{k} \alpha_i y_{n-i} + h\sum_{i=0}^{k} \beta_i f(y_{n-i}). \tag{1.1.3}$$

The most popular families of linear multi-step methods are Adams Bashforth methods and Adams Moulton methods. It is important to note that in the above equation

(1.1.3) if we take $\alpha_1 = 1$, all other $\alpha_i^{'}s = 0$ and $\beta_0 = 0$ such that:

$$y_n = y_{n-1} + h[\beta_1 f(y_{n-1}) + \beta_2 f(y_{n-2}) + \cdots + \beta_k f(y_{n-k})]. \tag{1.1.4}$$

If instead $\beta_0 \neq 0$, we get

$$y_n = y_{n-1} + h[\beta_0 f(y_n) + \beta_1 f(y_{n-1}) + \beta_2 f(y_{n-2}) + \cdots + \beta_k f(y_{n-k})]. \tag{1.1.5}$$

then the (1.1.4) is explicit method of order $k$ (where $k$ is the number of steps) and known as Adams Bashforth method. Similarly (1.1.5) is implicit method of order $k+1$, known as Adams Moulton method. On the other hand these two methods can be implemented as predictor-corrector pair (by first using Adams-Bashforth as predictor and then Adams-Moulton as corrector method). Since multi-step methods need values of solution at more than one point, therefore a starting method is employed to start the procedure. Usually one-step methods are used as starting method.

General linear methods are the generalization of multi-stage (Runge-Kutta) and multi-value (linear multi-step) methods. The basic idea behind general linear methods is that a number of input quantities are imported at the beginning of any particular step. Then a number of stage derivatives are computed along with their stage values where each of the stage values is linear combination of the input quantities and the stage derivatives. Finally, output quantities which are also linear combination of the input quantities and the stage derivatives are computed corresponding to the input quantities imported at the beginning of the step. General linear methods are used to find numerical solution of initial value problems (IVPs)

$$y'(t) = f(y(t)), \qquad y(t_0) = y_0. \tag{1.1.6}$$

The general formula of general linear methods can then be written as:

$$\begin{aligned} Y_i^{[n]} &= \sum_{j=1}^{s} hAf(Y_j^{[n]}) + \sum_{j=1}^{r} Uy_j^{[n-1]}, && i = 1, 2, \cdots, s, \\ y_i^{[n]} &= \sum_{j=1}^{s} hBf(Y_j^{[n]}) + \sum_{j=1}^{r} Vy_j^{[n-1]}, && i = 1, 2, \cdots, r, \end{aligned} \tag{1.1.7}$$

where $r$ and $s$ denotes the number of input quantities and number of stage values respectively. The stage values and stage derivatives calculated at steps $n$ are denoted by $Y_i^{[n]} \approx y(t_n + c_i h)$ evaluated at abscissa $c_i$ and $F_i = f(Y_i^{[n]})$ respectively where $i = 1, 2, \cdots, s$ . Similarly $y^{[n-1]}$ represents the input quantities and $y^{[n]}$, the output quantities computed at the beginning of step $n$. Let $[A, U, B, V]$ represents the coefficient matrix and $h$ the step size. This formulation of coefficient matrix for general linear methods was first introduced by Burrage and Butcher [1] in 1980. This coefficient matrix determines the implementation cost of this method. General linear methods can be represented in matrix notation as:

$$\begin{bmatrix} Y_i^{[n]} \\ \\ y_i^{[n]} \end{bmatrix} = \begin{bmatrix} A & U \\ \\ B & V \end{bmatrix} \begin{bmatrix} hf(Y_i^{[n]}) \\ \\ y_i^{[n-1]} \end{bmatrix}. \tag{1.1.8}$$

For all multi-value methods, some starting procedure is required before carrying out the first integration step. The difference between general linear methods and linear multi-step methods is the generality of the quantities that general linear methods pass from one step to the next. Hence, to carry out the first step of integration a starting method computes a variety of possible input quantities $y_1^{[0]}, y_2^{[0]}, \cdots, y_r^{[0]}$. Runge-Kutta and linear multi-step methods can be written in the form of general linear methods. Consider the example of coefficient matrix of the fourth order implicit Gauss Runge-Kutta method [6]:

$$\begin{array}{c|c} c & A \\ \hline & b^T \end{array} = \begin{array}{c|cc} \frac{1}{2} - \frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4} - \frac{\sqrt{3}}{6} \\ \frac{1}{2} + \frac{\sqrt{3}}{6} & \frac{1}{4} + \frac{\sqrt{3}}{6} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}. \tag{1.1.9}$$

The abscissa $c_i$'s are chosen as the roots of the shifted Legendre polynomial on the interval $[0, 1]$ of degree $s$. The coefficients $A = a_{ij}$ and $b_i$ are calculated from abscissa $c_i$ in a way to ensure order of method.

$$P_s(x) = \frac{s!}{2s} \sum_{m=0}^{s} (-1)^{s-m} \begin{pmatrix} s \\ m \end{pmatrix} \begin{pmatrix} s+m \\ m \end{pmatrix} x^m.$$

Then this method can be represented in general linear form (1.1.8) as:

$$\left[\begin{array}{c} Y_1 \\ Y_2 \\ \hline y_1^{[n]} \end{array}\right] = \left[\begin{array}{cc|c} \frac{1}{4} & \frac{1}{4}-\frac{\sqrt{3}}{6} & 1 \\ \frac{1}{4}+\frac{\sqrt{3}}{6} & \frac{1}{4} & 1 \\ \hline \frac{1}{2} & \frac{1}{2} & 1 \end{array}\right] \left[\begin{array}{c} hf(Y_1) \\ hf(Y_2) \\ \hline y_1^{[n-1]} \end{array}\right].$$

The $A$ matrix of the general linear method is the same as the $A$ matrix of the Runge-Kutta method. The $B$ matrix is $b^T$ where $b$ is the vector of weights of the Runge-Kutta method. Assuming the input vector $y^{[n-1]}$ is an approximation to $y(t_{n-1})$, the $U$ matrix is simply a vector of 1s. The $V$ matrix consists only of the number 1. Similarly, a second order Adams-Bashforth method (1.1.4), derived from Taylor series is as follows

$$y_n - y_{n-1} = h\beta_1 f(y_{n-1}) + h\beta_2 f(y_{n-2}),$$
$$y_n - [y_n - hf(y_n) + \frac{h^2}{2} f^{'}(y_n)] = h\beta_1[f(y_n) - hf^{'}(y_n)] + h\beta_2[f(y_n) - 2hf^{'}(y_n)],$$

comparing coefficients we will get system of equations

$$\beta_1 + \beta_2 = 1,$$
$$\beta_1 + 2\beta_2 = \frac{1}{2}.$$

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

$$y_n = y_{n-1} + h\Big[\frac{3}{2} f(y_{n-1}) - \frac{1}{2} f(y_{n-2})\Big].$$

This can be written in general linear formulation as:

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

Note that in Runge-Kutta methods $r = 1$, since only one input quantity is passed to the next step. On the other hand the number of stage values for linear multi-step methods is $s = 1$ because in each step the function is evaluated only once.

## 1.2 Hamiltonian systems

Hamiltonian systems are an important class of dynamical systems which were first studied in mechanics. However, our interest lies in the numerical solution of Hamiltonian systems. If $H = H(q, p)$ is a sufficiently smooth real function in a $2n$-dimensional Euclidean space then the dynamical system defined in [11] as:

$$\frac{dp_i}{dt} = -\frac{\partial H}{\partial q_i} \quad , \quad \frac{dq_i}{dt} = \frac{\partial H}{\partial p_i}, \qquad i = 1, 2, ..., n. \tag{1.2.1}$$

having n degrees of freedom is called a Hamiltonian system. Here the Hamiltonian $H$ represents the total energy (i.e sum of kinetic and potential energies) of the system and $p_i = (p_1, p_2, \cdots, p_n)$ and $q_i = (q_1, q_2, \cdots, q_n)$ are generalised momenta and position vectors respectively. We emphasize here two main properties of Hamiltonian systems:

- Energy conservation,
- Symplecticity.

If the Hamiltonian $H$ is autonomous then,

$$\frac{dH}{dt} = \sum_i \left( \frac{\partial H}{\partial p_i} \frac{dp_i}{dt} + \frac{\partial H}{\partial q_i} \frac{dq_i}{dt} \right) = 0. \tag{1.2.2}$$

which means $H$ is a conserved quantity. This shows the characteristic property of Hamiltonian system that its solution preserves energy $H$. In other words $H(y(t)) = H(y(0))$, that is, it remains constant along solutions of the system.

### 1.2.1 Symplecticity

In addition to being energy preserving, Hamiltonian systems possess a noteworthy property that their phase flow is symplectic. The Hamiltonian systems possess a $2n$-dimensional phase space with coordinates $(p_i, q_i)$, $i = 1, 2, \cdots, n$. Then the transformation of phase space therefore can be expressed in terms of flow of the differential equation via solution operator $\psi$ as

$$\psi : (p(0), q(0)) \to (p(t), q(t)), \tag{1.2.3}$$

where the vector field $f$ is defined as:

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

such that

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

which shows that the solution operator $\psi$ representing phase flow is symplectic. Consider a sheet having an area A positioned in the phase space of a Hamiltonian system and moving along the corresponding phase flow. We observe that the sheet is stretched after transformation through the solution operator $\psi$ but its area remains same as shown in the Figure 1.1.

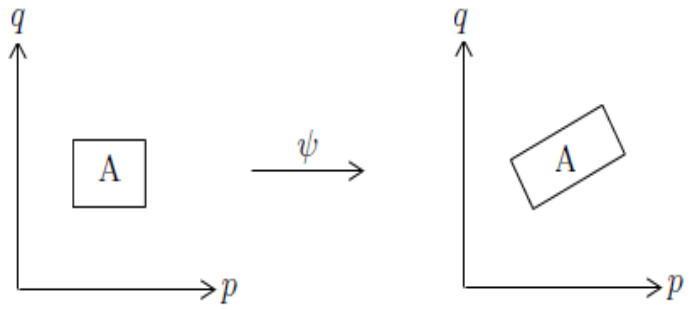


Figure 1.1: Symplectic behaviour of a sheet.

### 1.2.2 Simple pendulum

The simple pendulum consists of a bob of mass $m$ attached to one end of a massless string and whose other end is fixed. When the bob is slightly moved about the equilibrium position and then released, it executes simple harmonic motion. Let us assume that bob moves in a vertical plane then the angle between the string and vertical axis be denoted by $q$ and distance of the bob from the point of suspension by $l$.

Considering the horizontal plane that passes through the point of suspension as a

reference level for zero potential energy, the set of differential equations defines the Hamiltonian system as

$$\begin{aligned} p' &= -\sin(q), \\ q' &= p. \end{aligned} \tag{1.2.4}$$

Where $p' = \frac{dp}{dt}$ and $q' = \frac{dq}{dt}$. Then the function

$$H = \frac{1}{2}p^2 - \cos(q).$$

is the Hamiltonian or total energy of this mechanical system. The initial conditions chosen to be as $p = 0$ and $q = 2.3$ with step-size of 0.01. The position vector $q$ is taken along x-axis and momentum $p$ along y-axis. The symplectic behaviour of the simple pendulum can be observed by applying the symplectic Gauss Runge-Kutta method on (1.2.4). The result is shown in the Figure 1.2. It preserves the area and hence is symplectic.

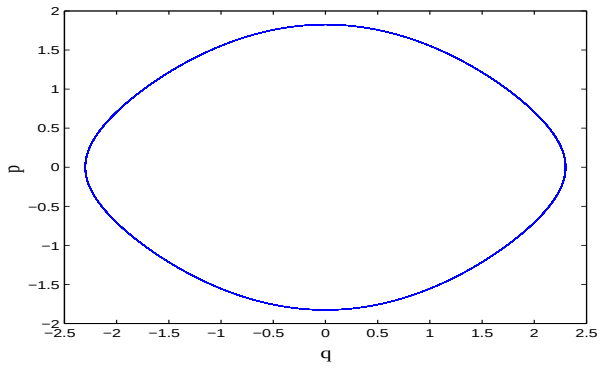


Figure 1.2: Symplecticity of implicit Gauss method for simple pendulum.

# Chapter 2

# G-symplecticity and control of parasitism

The concept of G-symplecticity to a wider class of multi-value numerical methods, i.e. the general linear methods is introduced by Butcher in [11]. G-symplectic general linear methods are designed to conserve the symplectic invariants but it does not bound the parasitism effect that arises due to multi-value nature of general linear methods. Such methods can integrate the solutions of Hamiltonian problems efficiently and preserves the energy and symplectic invariants over approximately long time intervals. In particular, we address here the question of controlling parasitic effects, which corrupts the numerical solution.

## 2.1 G-symplectic behaviour of general linear methods

General linear methods provide a unifying structure for traditional numerical methods, but because of their multi-value nature we cannot achieve true conservation of quadratic invariants. One-step methods are said to conserve quadratic invariants and symplectic behaviour. It is, however, believed that multi-value multi-derivative methods cannot posses the symplectic property unless they get just one value of the current step from the preceding one, means the general linear method is reduced to

one-step method [14].

By the symplectic behaviour, we mean that the inner product of the values at the initial point is same as the inner product of the values at the later point. In general, an s-stage and r-value ($r > 1$) irreducible general linear method cannot preserve the quadratic invariants over long time intervals [14]. Consider a general linear method having $r$ input quantities and $s$ stage values:

$$\begin{aligned} Y_i &= h\sum_{j=1}^{s} a_{ij}F_j + \sum_{j=1}^{r} u_{ij}y_j^{[n-1]}, \qquad && i = 1, 2, \cdots, s, \\ y_i^{[n]} &= h\sum_{j=1}^{s} b_{ij}F_j + \sum_{j=1}^{r} v_{ij}y_j^{[n-1]}, \qquad && i = 1, 2, \cdots, r. \end{aligned} \tag{2.1.1}$$

where the stage values $Y_i$ are defined by two matrices $A = a_{ij}$ and $U = u_{ij}$. Similarly the output quantities $y_i^{[n]}$ are defined by the matrices $B = b_{ij}$ and $V = v_{ij}$. To accommodate the multiple inputs with multiple outputs of general linear methods, we bring together a G-matrix with a G-norm as

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

where $g_{ij}$ represent $i, j^{th}$ entry of symmetric $r \times r$ matrix G and

$$y = \begin{bmatrix} y_1 \\ \vdots \\ y_r \end{bmatrix}, \qquad w = \begin{bmatrix} w_1 \\ \vdots \\ w_r \end{bmatrix}.$$

The $G$-norm presented by such an inner product is,

$$\|y\|_G^2 = \langle y, y\rangle_G. \tag{2.1.2}$$

In addition to the matrix $G$, a diagonal $s \times s$ matrix $D = d_i$ 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 $F_i = f(Y_i)$. Then to solve a conservative problem, we must have

$$\langle y, f(y)\rangle = 0, \tag{2.1.3}$$

which means that

$$\langle y^{[n]}, y^{[n]} \rangle_G = \langle y^{[n-1]}, y^{[n-1]} \rangle_G, \tag{2.1.4}$$

such that the term $2h \sum_{i=1}^{s} d_i \langle Y_i, F_i \rangle$ is zero. Methods satisfying equation (2.1.4) are known as G-symplectic general linear methods.

**Theorem 2.1.1** ([14]). *A general linear method is said to be G-symplectic, if there exist a diagonal $s \times s$ matrix $D$ and a symmetric $r \times r$ matrix $G$ such that*

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

It should be noted here that general linear methods cannot possess true quadratic invariants and hence cannot be symplectic. However, general linear methods can preserve quadratic behaviour of invariants under a G-norm and such a general linear method is called a G-symplectic general linear method.

**Example:** Consider the following general linear method

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{cc|cc} \frac{3-\sqrt{3}}{6} & 0 & 1 & -\frac{3+2\sqrt{3}}{3} \\ -\frac{\sqrt{3}}{3} & \frac{3+\sqrt{3}}{6} & 1 & \frac{3+2\sqrt{3}}{3} \\ \hline \frac{1}{2} & \frac{1}{2} & 1 & 0 \\ \frac{1}{2} & -\frac{1}{2} & 0 & -1 \end{array}\right]. \tag{2.1.5}$$

The G-symplectic conditions of this method can be proved using the matrices $G$ and $D$. Consider the matrices $G$ and $D$ of the form

$$G = \begin{bmatrix} 1 & g_{12} \\ g_{21} & g_{22} \end{bmatrix}, \qquad D = \begin{bmatrix} d_{11} & 0 \\ 0 & d_{22} \end{bmatrix}.$$

We assume $g_{11} = 1$, this choice is due to the fact that the matrix $V$ in (2.1.5) makes the first condition in Theorem (2.1.1) a linear system of two equations in three unknowns. Thus we choose $g_{11}$ as parameter and fix its value to 1. The remaining

entries of matrices $D$ and $G$ can be found using conditions of Theorem (2.1.1). We consider the second condition of Theorem (2.1.1) by comparing the first column of the matrix $DU$ with that of the matrix $B^TGV$. We obtain the following values for the entries of the matrix $D$

$$d_{11} = \frac{b_{11} + b_{21}v_{12} - b_{11}v_{22}}{1 - v_{22}} = \frac{1}{2},$$
$$d_{22} = \frac{b_{12} + b_{22}v_{12} - b_{12}v_{22}}{1 - v_{22}} = \frac{1}{2}.$$

Similarly for matrix $G$ we consider the first and third conditions of Theorem (2.1.1), we obtain

$$g_{12} = \frac{v_{12}}{1 - v_{22}} = 0,$$
$$g_{21} = \frac{d_{11} - b_{11}g_{11}}{b_{21}} = 0$$
$$g_{22} = \frac{d_{22}u_{22}}{b_{22}v_{22}} = \tfrac{3+2\sqrt{3}}{3}.$$

The method (2.1.5) is then G-symplectic with

$$G = \begin{bmatrix} 1 & 0 \\ 0 & \frac{3+2\sqrt{3}}{3} \end{bmatrix}, \qquad D = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix}, \tag{2.1.6}$$

if we have

$$\langle y^{[n]}, y^{[n]} \rangle_G = \langle y^{[n-1]}, y^{[n-1]} \rangle_G, \tag{2.1.7}$$

$$\sum_{i,j=1}^{2} g_{ij} \langle y^{[n]}, y^{[n]} \rangle_G = \sum_{i,j=1}^{2} g_{ij} \langle y^{[n-1]}, y^{[n-1]} \rangle_G, \tag{2.1.8}$$

$$g_{11}\langle y_1^{[n]}, y_1^{[n]} \rangle + g_{22}\langle y_2^{[n]}, y_2^{[n]} \rangle = g_{11}\langle y_1^{[n-1]}, y_1^{[n-1]} \rangle + g_{22}\langle y_2^{[n-1]}, y_2^{[n-1]} \rangle. \tag{2.1.9}$$

where, $g_{11} = 1$ and $g_{22} = \frac{3+2\sqrt{3}}{3}$. Therefore, instead of conserving the true quadratic invariants, i.e.

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

The linear combination

$$\langle y_1^{[n]}, y_1^{[n]} \rangle + \tfrac{3+2\sqrt{3}}{3} \langle y_2^{[n]}, y_2^{[n]} \rangle = \langle y_1^{[n-1]}, y_1^{[n-1]} \rangle + \tfrac{3+2\sqrt{3}}{3} \langle y_2^{[n-1]}, y_2^{[n-1]} \rangle.$$

is conserved under $G$-norm.

### 2.1.1 Some examples of G-symplectic methods

Ordinary differential equations (ODEs), in general, are given with an initial condition from where we get an initial value. In dealing with general linear methods which consists of multiple input and output approximations, one value is obtained from the given initial condition and a starting method is applied to obtain the remaining initial values. Here we consider the methods namely GLM43, N method and P method along with their starting methods [15].

- **N Method**

  It is implicit fourth order symmetric general linear method with two stages and two output values. The general representation (2.1.1) of this method is

$$\begin{aligned}
Y_1^{[n]} &= h(\tfrac{3-\sqrt{3}}{6})f(Y_1^{[n]}) + y_1^{[n-1]} - (\tfrac{3-2\sqrt{3}}{3})y_2^{[n-1]},\\
Y_2^{[n]} &= h(\tfrac{\sqrt{3}}{3})f(Y_1^{[n]}) + h(\tfrac{3-\sqrt{3}}{6})f(Y_2^{[n]}) + y_1^{[n-1]} + (\tfrac{3-2\sqrt{3}}{3})y_2^{[n-1]},\\
y_1^{[n]} &= (\tfrac{h}{2})f(Y_1^{[n]}) + (\tfrac{h}{2})f(Y_2^{[n]}) + y_1^{[n-1]},\\
y_2^{[n]} &= (\tfrac{h}{2})f(Y_1^{[n]}) - (\tfrac{h}{2})f(Y_2^{[n]}) - y_2^{[n-1]},
\end{aligned}$$

  where its coefficient matrix is

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{cc|cc} \frac{3-\sqrt{3}}{6} & 0 & 1 & -\frac{3-2\sqrt{3}}{3} \\ \frac{\sqrt{3}}{3} & \frac{3-\sqrt{3}}{6} & 1 & \frac{3-2\sqrt{3}}{3} \\ \hline \frac{1}{2} & \frac{1}{2} & 1 & 0 \\ \frac{1}{2} & -\frac{1}{2} & 0 & -1 \end{array}\right]. \tag{2.1.10}$$

  The matrix $A$ is a lower triangular matrix. For the approximation of actual solution exactly, the first column of matrix $U$ is taken as a vector of ones because during the calculation of the stages, the input values representing the actual solution are multiplied with the first column of the matrix $U$. It requires two input values to start the procedure. However only one initial value is provided with the initial value problem. Other initial value is calculated using a starting method. The starting method introduced by Butcher [12] is given

as

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{cc|c} \frac{3-\sqrt{3}}{6} & 0 & 1 \\ -\frac{3-\sqrt{3}}{3} & \frac{3-\sqrt{3}}{6} & 1 \\ \hline 0 & 0 & 1 \\ -\frac{\sqrt{3}+1}{8} & \frac{\sqrt{3}+1}{8} & 0 \end{array}\right]. \tag{2.1.11}$$

The stage values of starting method are explicit in nature. The vector $U$ of starting method is always taken as vector of ones. The first element of vector $V$ is chosen as 1 and all other elements are taken as zero. The coefficient matrices $A$ and $B$ of starting method can be obtained to make sure that the general linear method (2.1.10) satisfies the order of the method [16] and for this algebraic analysis of order using rooted trees is employed as given in [13].

- **P Method**

  It is implicit fourth order symmetric general linear method with two stages and two output values. The general representation (2.1.1) of this method is

$$\begin{aligned} Y_1^{[n]} &= h(\tfrac{3+\sqrt{3}}{6})f(Y_1^{[n]}) + y_1^{[n-1]} - (\tfrac{3+2\sqrt{3}}{3})y_2^{[n-1]}, \\ Y_2^{[n]} &= -h(\tfrac{\sqrt{3}}{3})f(Y_1^{[n]}) + h(\tfrac{3+\sqrt{3}}{6})f(Y_2^{[n]}) + y_1^{[n-1]} + (\tfrac{3+2\sqrt{3}}{3})y_2^{[n-1]}, \\ y_1^{[n]} &= (\tfrac{h}{2})f(Y_1^{[n]}) + (\tfrac{h}{2})f(Y_2^{[n]}) + y_1^{[n-1]}, \\ y_2^{[n]} &= (\tfrac{h}{2})f(Y_1^{[n]}) - (\tfrac{h}{2})f(Y_2^{[n]}) - y_2^{[n-1]}, \end{aligned}$$

  where its coefficient matrix is

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{cc|cc} \frac{3+\sqrt{3}}{6} & 0 & 1 & -\frac{3+2\sqrt{3}}{3} \\ -\frac{\sqrt{3}}{3} & \frac{3+\sqrt{3}}{6} & 1 & \frac{3+2\sqrt{3}}{3} \\ \hline \frac{1}{2} & \frac{1}{2} & 1 & 0 \\ \frac{1}{2} & -\frac{1}{2} & 0 & -1 \end{array}\right]. \tag{2.1.12}$$

  The matrix $A$ is a lower triangular matrix. For the approximation of actual solution exactly, the first column of matrix $U$ is taken as a vector of ones because during the calculation of the stages, the input values representing the

actual solution are multiplied with the first column of the matrix $U$. It requires two input values to start the procedure. However only one initial value is provided with the initial value problem. Other initial value is calculated using a starting method. The starting method introduced by Butcher [12] is given as

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{cc|c} \frac{3+\sqrt{3}}{6} & 0 & 1 \\ -\frac{3+\sqrt{3}}{3} & \frac{3+\sqrt{3}}{6} & 1 \\ \hline 0 & 0 & 1 \\ \frac{\sqrt{3}-1}{8} & -\frac{\sqrt{3}-1}{8} & 0 \end{array}\right]. \tag{2.1.13}$$

The stage values of starting method are explicit in nature. The vector $U$ of starting method is always taken as vector of ones. The first element of vector $V$ is chosen as 1 and all other elements are taken as zero. The coefficient matrices $A$ and $B$ of starting method can be obtained to make sure that the general linear method (2.1.12) satisfies the order of the method [16] and for this algebraic analysis of order using rooted trees is employed as given in [13].

- **GLM**43

  It is fourth order symmetric general linear method with four stages and three output values. The general representation (2.1.1) of this method is

$$\left[\begin{array}{c} Y_1^{[n]} \\ Y_2^{[n]} \\ Y_3^{[n]} \\ Y_4^{[n]} \\ \hline y_1^{[n]} \\ y_2^{[n]} \\ y_3^{[n]} \end{array}\right] = \left[\begin{array}{cccc|ccc} 0 & 0 & 0 & 0 & 1 & \frac{1}{4} & \frac{\sqrt{3}}{4} \\ -\frac{11}{172} & \frac{1}{4} & 0 & 0 & 1 & u_{22} & u_{23} \\ -\frac{2647}{72240} & \frac{1009}{1680} & \frac{1}{4} & 0 & 1 & -u_{22} & -u_{23} \\ -\frac{169}{1680} & \frac{113821}{283920} & \frac{473}{676} & 0 & 1 & -\frac{1}{4} & -\frac{\sqrt{3}}{4} \\ \hline -\frac{169}{3360} & \frac{1849}{3360} & \frac{1849}{3360} & -\frac{169}{3360} & 1 & 0 & 0 \\ -\frac{169}{1680} & -\frac{84839}{283920} & \frac{84839}{283920} & \frac{169}{1680} & 0 & -\frac{1}{2} & -\frac{\sqrt{3}}{2} \\ 0 & -\frac{43\sqrt{14595}}{35490} & \frac{43\sqrt{14595}}{35490} & 0 & 0 & \frac{\sqrt{3}}{2} & -\frac{1}{2} \end{array}\right] \cdot \left[\begin{array}{c} hf(Y_1^{[n]}) \\ hf(Y_2^{[n]}) \\ hf(Y_3^{[n]}) \\ hf(Y_4^{[n]}) \\ \hline y_1^{[n-1]} \\ y_2^{[n-1]} \\ y_3^{[n-1]} \end{array}\right],$$

  where $u_{22} = -\frac{1973}{29068} + \frac{2\sqrt{3}\sqrt{14595}}{7267}$ and $u_{23} = -\frac{1973\sqrt{3}}{29068} - \frac{2\sqrt{14595}}{7267}$. The first and

last row of matrix $A$ are explicit in nature while second and third rows are implicit. For the approximation of actual solution exactly, the first column of matrix $U$ is taken as a vector of ones because during the calculation of the stages, the input values representing the actual solution are multiplied with the first column of the matrix $U$. It requires three input values to start the procedure. However only one initial value is provided with the initial value problem. Other initial values are calculated using a starting method. The matrix $A$ and abscissa $c$ of starting method are taken same as of classical order 4 Runge-Kutta method [6].The starting method will get one input and three output values. The coefficient matrix $B$ of starting method can be obtained to make sure that the general linear method (2.1.12) satisfies the order of the method [16] and for this algebraic analysis of order using rooted trees is employed as given in [13]. The starting method introduced by Butcher [12] is given as

$$\left[\begin{array}{c|c} A & U \\ \hline B & V \end{array}\right] = \left[\begin{array}{cccc|c} 0 & 0 & 0 & 0 & 1 \\ \frac{1}{2} & 0 & 0 & 0 & 1 \\ 0 & \frac{1}{2} & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ \hline 0 & 0 & 0 & 0 & 1 \\ \tilde{b}_1 & \tilde{b}_2 & \tilde{b}_3 & \tilde{b}_4 & 0 \\ \tilde{b}_5 & \tilde{b}_6 & \tilde{b}_7 & \tilde{b}_8 & 0 \end{array}\right]. \tag{2.1.14}$$

The second and third row of matrix $\tilde{B}$ are complex conjugates of each other. The values of $\tilde{B} = [\tilde{b}_1, \tilde{b}_2, \tilde{b}_3, \tilde{b}_4]$ can be found using order conditions upto third order [16]. However due to symmetry, it will ensure that the method is of

fourth order.

$$\begin{bmatrix} \tilde{b}_1 \\ \tilde{b}_2 \\ \tilde{b}_3 \\ \tilde{b}_4 \\ \tilde{b}_5 \\ \tilde{b}_6 \\ \tilde{b}_7 \\ \tilde{b}_8 \end{bmatrix} = \begin{bmatrix} -0.203599283611326 \\ 0.093013520189574 \\ 0.079026915254744 \\ 0.031558848167008 \\ -0.523240277404552 \\ 0.399628138816171 \\ 0.388537971512798 \\ -0.264925832924417 \end{bmatrix}. \tag{2.1.15}$$

It should be noted here that the methods we mentioned above are G-symplectic methods.

Consider the method (2.1.10). This method is G-symplectic if it satisfies the conditions of Theorem (2.1.1) with matrices $G$ and $D$ given as [16]

$$G = \begin{bmatrix} 1 & 0 \\ 0 & \frac{3-2\sqrt{3}}{3} \end{bmatrix}, \qquad D = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix}. \tag{2.1.16}$$

Since

$$A = \begin{bmatrix} \frac{3-\sqrt{3}}{6} & 0 \\ \frac{\sqrt{3}}{3} & \frac{3-\sqrt{3}}{6} \end{bmatrix}, \qquad B = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} \end{bmatrix}. \tag{2.1.17}$$

$$U = \begin{bmatrix} 1 & -\frac{3-2\sqrt{3}}{3} \\ 1 & \frac{3-2\sqrt{3}}{3} \end{bmatrix}, \qquad V = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \tag{2.1.18}$$

Then for this method to be G-symplectic it should satisfy $G = V^T G V$, $DU = B^T G V$ and $DA + A^T D = B^T G B$.

$$\begin{aligned} V^T G V &= \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}^T \begin{bmatrix} 1 & 0 \\ 0 & \frac{3-2\sqrt{3}}{3} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \\ &= \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & -\frac{3-2\sqrt{3}}{3} \end{bmatrix}, \\ &= \begin{bmatrix} 1 & 0 \\ 0 & -\frac{3-2\sqrt{3}}{3} \end{bmatrix} = G. \end{aligned}$$

So first condition is satisfied.

Now consider

$$DU = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix} \begin{bmatrix} 1 & -\frac{3-2\sqrt{3}}{3} \\ 1 & \frac{3-2\sqrt{3}}{3} \end{bmatrix},$$
$$= \begin{bmatrix} \frac{1}{2} & -\frac{3-2\sqrt{3}}{6} \\ \frac{1}{2} & \frac{3-2\sqrt{3}}{6} \end{bmatrix},$$

and

$$B^T GV = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} \end{bmatrix}^T \begin{bmatrix} 1 & 0 \\ 0 & \frac{3-2\sqrt{3}}{3} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix},$$
$$= \begin{bmatrix} \frac{1}{2} & -\frac{3-2\sqrt{3}}{6} \\ \frac{1}{2} & \frac{3-2\sqrt{3}}{6} \end{bmatrix} = DU.$$

which satisfies the second condition $DU = B^T GV$.

Now considering,

$$B^T GB = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} \end{bmatrix}^T \begin{bmatrix} 1 & 0 \\ 0 & \frac{3-2\sqrt{3}}{3} \end{bmatrix} \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} \end{bmatrix},$$
$$= \begin{bmatrix} \frac{3-\sqrt{3}}{6} & \frac{\sqrt{3}}{6} \\ \frac{\sqrt{3}}{6} & \frac{3-\sqrt{3}}{6} \end{bmatrix}.$$

and

$$DA + A^T D = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix} \begin{bmatrix} \frac{3-\sqrt{3}}{6} & 0 \\ \frac{\sqrt{3}}{3} & \frac{3-\sqrt{3}}{6} \end{bmatrix} + \begin{bmatrix} \frac{3-\sqrt{3}}{6} & 0 \\ \frac{\sqrt{3}}{3} & \frac{3-\sqrt{3}}{6} \end{bmatrix}^T \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix},$$
$$= \begin{bmatrix} \frac{3-\sqrt{3}}{6} & \frac{\sqrt{3}}{6} \\ \frac{\sqrt{3}}{6} & \frac{3-\sqrt{3}}{6} \end{bmatrix} = B^T GB.$$

This satisfies the third condition proving that the method is G-symplectic. Similarly P-method and GLM43 also satisfies G-symplectic conditions.

### 2.1.2 Experiment

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

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

where the total energy function is

$$H = \frac{1}{2}p^2 - \cos(q).$$

To solve this simple pendulum problem, N method (2.1.10) has been applied. The initial values are chosen as $p = 0$, $q = 1.2$. The error in energy is plotted in the Figure 2.1 for 1 million steps with step-size 0.01. The graph is plotted between error and time using MATLAB. Time is taken along x-axis and error in energy along y-axis. The energy error is calculated by subtracting the energy calculated at actual value from the energy calculated at approximated value, i.e. energy error= $H_{approx.} - H_{exact}$. However $H_{approx.}$ is evaluated at first output value. The result shows very small energy errors with the certainty that the method conserves the total energy of the simple pendulum problem. However, on increasing the amplitude of the simple pendulum, i.e. by changing the value of $q$ from 1.2 to 2.3, we get the error in energy as plotted in the Figure 2.2. In the second case we have taken only $100,000$ steps with the same step-size of 0.01 but large error in energy is observed. Although this method is G-symplectic and meant to conserve the energy for all initial values, we observed a large error in energy and this is because of the corruption by the parasitic solutions.

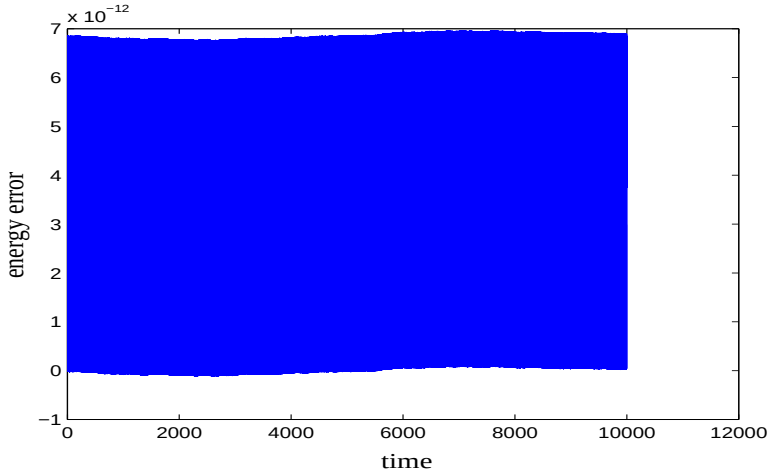


Figure 2.1: The error in energy conservation of the simple pendulum problem with initial value $q = 1.2$.

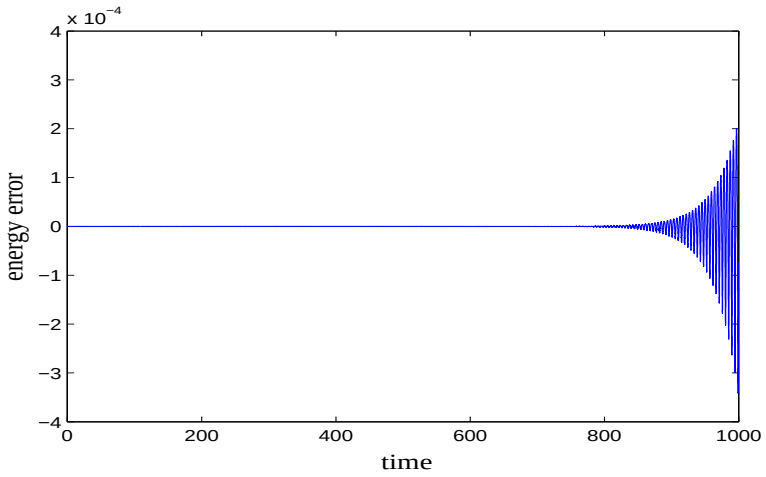


Figure 2.2: The error in energy conservation of the simple pendulum problem with initial value $q = 2.3$.

## 2.2 Parasitic solutions of GLMs

The multi-value nature of general linear methods subjects them to suffer from parasitic solutions. The numerical solutions which are attained in addition to the numerical approximation of the exact solution are known as parasitic solutions. To study the effect of parasitism, a typical step of the general linear method is considered.

$$\left[\begin{array}{c} Y_1 \\ Y_2 \\ \hline y_1^{[n]} \\ y_2^{[n]} \end{array}\right] = \left[\begin{array}{cc|cc} a_{11} & a_{12} & u_{11} & u_{12} \\ a_{21} & a_{22} & u_{21} & u_{22} \\ \hline b_{11} & b_{12} & 1 & 0 \\ b_{21} & b_{22} & 0 & -1 \end{array}\right] \left[\begin{array}{c} hf(Y_1) \\ hf(Y_2) \\ \hline y_1^{[n-1]} \\ y_2^{[n-1]} \end{array}\right].$$

The stage and output values can be represented as

$$\begin{aligned} Y_i &= \sum_{j=1}^{2} ha_{ij} f(Y_j) + u_{i1} y_1^{[n-1]} + u_{i2} y_2^{[n-1]}, \qquad i = 1, 2. \\ y_1^{[n]} &= \sum_{i=1}^{2} hb_{1i} f(Y_i) + y_1^{[n-1]}, \\ y_2^{[n]} &= \sum_{i=1}^{2} hb_{2i} f(Y_i) - y_2^{[n-1]}, \end{aligned} \tag{2.2.1}$$

while $y_1^{[n]}$, the first component approximates the exact solution and $y_2^{[n]}$, the second component approximates the scaled derivative. To see how parasitism occurs in the value of $y_2^{[n]}$, we have introduced a perturbation at the start of step n:

$$y_2^{[n-1]} \to y_2^{[n-1]} + (-1)^{n-1} w_{n-1}.$$

Then the perturbation in the stage values $Y_i$ will be given as

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

and the approximate perturbation in the values of stage derivatives is

$$F_i \to F_i + \delta F_i = f(Y_i + \delta Y_i), \tag{2.2.2}$$

$$= f(Y_i) + \delta Y_i \frac{\partial f}{\partial y}, \tag{2.2.3}$$

$$\delta F_i = (-1)^{n-1} u_{i2} w_{n-1} \frac{\partial f}{\partial y}.$$

Now the change at step $n$ in second output value should be

$$y_2^{[n]} + (-1)^n w_n = h \sum_{i=1}^{2} b_{2i}(F_i + \delta F_i) - [y_2^{[n-1]} + (-1)^{n-1} w_{n-1}],$$

$$= h \sum_{i=1}^{2} b_{2i} F_i - y_2^{[n-1]} + h \sum_{i=1}^{2} b_{2i} \delta F_i - (-1)^{n-1} w_{n-1},$$

$$w_n = \left[1 - h \sum_{i=1}^{2} b_{2i} u_{i2} \frac{\partial f}{\partial y}\right] w_{n-1},$$

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

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

The particular problem has parasitic free solution only if $\mu = 0$. The parasitic component of N method can then be found by taking product of B and U matrices (2.1.10)

$$BU = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} \end{bmatrix} \begin{bmatrix} 1 & -\frac{3-2\sqrt{3}}{3} \\ 1 & \frac{3-2\sqrt{3}}{3} \end{bmatrix},$$

$$= \begin{bmatrix} 1 & 0 \\ 0 & -\frac{3-2\sqrt{3}}{3} \end{bmatrix}.$$

Then

$$\mu_N = 1 - \tfrac{2}{\sqrt{3}}.$$

Similarly the parasitic component of P method (2.1.12) is

$$\mu_P = 1 + \tfrac{2}{\sqrt{3}}.$$

## 2.3 Projection methods

Consider a differential equation $y' = f(y)$ whose solution lies on manifold, where a manifold is a configuration space that resembles Euclidean space locally (i.e. near every point).

$$\mathcal{M} = \{y;\ I(y) = 0\}, \tag{2.3.1}$$

where $I : \mathbb{R}^n \to \mathbb{R}^m$ and the differential equation satisfying the property that

$$y_0 \in \mathcal{M}\ implies\ y(t) \in \mathcal{M}, \quad \forall t. \tag{2.3.2}$$

In other words, the equation (2.3.2) is equivalent to $I'(y)f(y) = 0$ for $y \in \mathcal{M}$. However for conservation of invariants we require that $I'(y)f(y) = 0$ for all $y \in \mathbb{R}^n$. Therefore, $I(\mathrm{y})$ is known to be as a weak invariant and $y' = f(y(t))$, a differential equation on the manifold $\mathcal{M}$.
There are certain numerical methods which conserve the invariants automatically. If any numerical method fails to conserve it, we apply projection method manually. The idea in projection method is that we try to eliminate the small integral error by projecting the numerical solution on the integral manifold.

### 2.3.1 Standard projection method

An approach for the numerical solution of differential equations on manifolds is projection method [7]. The basic idea of standard projection method is that we first apply the numerical method $\Phi$ taking the initial value $y_n$ to $\tilde{y}_{n+1}$ and then project it back onto the invariant manifold $H(y) = 0$ to get $y_{n+1}$. Then the procedure for one step of projecting the numerical solution onto the integral manifold reads as,

- Using $y_0$ as initial condition, apply a general linear method as base method to get $[\tilde{y}_1^{[1]}, \tilde{y}_2^{[1]}]$.
- Perform the projection onto the invariant manifold as,

$$y_1 = \tilde{y}_1^{[1]} + \nabla H(y_1) \times \frac{H(y_0) - H(\tilde{y}_1^{[1]})}{\nabla H(\tilde{y}_1^{[1]}) \cdot \nabla H(\tilde{y}_1^{[1]})}. \tag{2.3.3}$$

Let us assume

$$\lambda = \frac{H(y_0) - H(\tilde{y}_1^{[1]})}{\nabla H(\tilde{y}_1^{[1]}) \cdot \nabla H(\tilde{y}_1^{[1]})}.$$

$H(y_0)$ is the initial value of $H$ and will not update. $\nabla H(y)$ denotes the Jacobian of $H(y)$. The first output value $y_1$ is prone to parasitic solution, therefore we applied projection step only on first output value. Since the equation (2.3.3) is implicit, we have used Newton iterations to solve this equation. Equation (2.3.3) can then be written as,

$$\begin{aligned} y_1 - \tilde{y}_1^{[1]} - \nabla H(y_1)\lambda = 0, \\ g(y_1) = 0. \end{aligned}$$

We replace $y_1$ with $\tilde{y}_1$ in the argument of $\nabla H(y_1)$. Then applying Newton iterations, we get

$$y_1 = \tilde{y}_1 + (I - \nabla^2 H(\tilde{y}_1)\lambda)^{-1}(\nabla H(\tilde{y}_1)\lambda). \tag{2.3.4}$$

Here $I$ is $2n \times 2n$ identity matrix and $J = \nabla^2 H(\tilde{y_1})$ is the Jacobian matrix. The Jacobian matrix $J$ is updated at each new step. The initial estimate $y_1$ is taken same as $\tilde{y}_1$. We have used the following stopping criteria for the convergence of Newton iteration,

$$\|y_1 - \tilde{y}_1\| < TOL = 10^{-15}.$$

It is observed that the code converges for 8 maximum iterations.
Standard projection methods can achieve excellent results in conserving invariants but can also destroy long time behaviour of the numerical solution. In standard projection method we apply the projection at the end of the integration step, therefore the method is not symmetric. In order to make the procedure symmetric, symmetric projection method is introduced. In order to retain symmetry, both the numerical method and the problem should be symmetric.

### 2.3.2 Symmetry of method

Symmetry of numerical methods can be explained on basis of symplecticity and reversibility. The one-step Runge-Kutta method (1.1.2) is said to be linearly symplectic if it satisfies the condition $R(z)R(-z) = 1$, where the stability fuction $R(z)$

can be found by considering the Dahlquist test problem:

$$y' = \lambda y, \qquad \lambda \in \mathbb{C}$$

then the general Runge-Kutta method (1.1.2) takes the form

$$\begin{aligned} Y_i &= y_{n-1} + h\sum_{j=1}^{s} a_{ij}\lambda f(Y_j), \quad i = 1, 2, \cdots, s \\ y_n &= y_{n-1} + h\sum_{i=1}^{s} b_i \lambda f(Y_i) \end{aligned} \tag{2.3.5}$$

Introducing the vectors $e = (1, 1, \cdots, 1)^T$, $Y = (Y_1, Y_2, \cdots, Y_s)^T$ and assuming $z = h\lambda$ we can write (2.3.5) as:

$$\begin{aligned} Y &= y_{n-1}e + zAY, \qquad A = (a_{ij}) \\ Y &= y_{n-1}(I - zA)^{-1}e, \\ y_n &= y_{n-1} + zb^T Y, \\ y_n &= (1 + zb^T(I - zA)^{-1}e)y_{n-1}. \end{aligned}$$

then the function $R(z)$ is defined as

$$R(z) = 1 + zb^T(I - zA)^{-1}e.$$

where

$$(I - zA)^{-1} = \frac{adj(I - zA)}{det(I - zA)},$$

with $det(I - zA) \in P_s$ and $adj(I - zA) \in P_{s-1}$, where $P_s$ and $P_{s-1}$ is the space of polynomials of degree $s$ and $s-1$ respectively. In general $R(z)$ is a rational function i.e, $R(z) = \frac{P(z)}{Q(z)}$.

Symplectic Runge-kutta method is reversible for any linear system with constant coefficients where the scheme $\Phi$ is said to be reversible if $\Phi_{-h} = \Phi_h^{-1}$ [10]. Runge-kutta method is then said to be symmetric if it is reversible, when applied to any arbitrary ordinary differential equation [9].

### 2.3.3 Symmetric projection method

Symmetric projection method was first applied by Hairer in 2000 [9]. The basic idea of symmetric projection method is that we first perturb the solution $y_n$ to get $\tilde{y}_n$ , followed by one step of symmetric numerical method $\Phi$ taking $\tilde{y}_n$ to $\tilde{y}_{n+1}$ and finally projecting the numerical solution $\tilde{y}_{n+1}$ onto the invariant manifold $H(y) = 0$. The procedure for symmetric projection is as follow,

- Using the initial condition $y_0$, apply a general linear method as perturbation method for one step to get $[y_1^{[1]}, y_2^{[1]}]$ and calculate the perturbation in initial condition as,

$$\tilde{y}_0 = y_1^{[1]} + \nabla H(y_1^{[1]}) \times \frac{H(y_0) - H(y_1^{[1]})}{\nabla H(y_1^{[1]}) \cdot \nabla H(y_1^{[1]})}. \tag{2.3.6}$$

- Using $\tilde{y}_0$ as initial condition, apply a general linear method (base method) to get $[\tilde{y}_1^{[1]}, \tilde{y}_2^{[1]}]$.

- Perform the projection onto the invariant manifold as,

$$y_1 = \tilde{y}_1^{[1]} + \nabla H(y_1) \times \frac{H(y_0) - H(y_1^{[1]})}{\nabla H(y_1^{[1]}) \cdot \nabla H(y_1^{[1]})}. \tag{2.3.7}$$

Let

$$\lambda = \frac{H(y_0) - H(y_1^{[1]})}{\nabla H(y_1^{[1]}) \cdot \nabla H(y_1^{[1]})}.$$

It is however important to take same vector $\lambda$ in perturbation and projection step in order to make the overall algorithm symmetric. To evaluate $\lambda$, a numerical method other than base method is required.

### 2.3.4 Implementation using Newton iterations

The equation (2.3.7) is implicit and we have used Newton iteration scheme to solve this equation. Equation (2.3.7) can then be reformulated as,

$$\begin{aligned} y_1 - \tilde{y}_1^{[1]} - \nabla H(y_1)\lambda = 0, \\ g(y_1) = 0. \end{aligned}$$

We replace $y_1$ with $\tilde{y}_1$ in the argument of $\nabla H(y_1)$, in order to save some evaluations of $\nabla H(y_1)$. Then using Newton iterations, we get

$$y_1 = \tilde{y}_1 + (I - \nabla^2 H(\tilde{y}_1)\lambda)^{-1}(\nabla H(\tilde{y}_1)\lambda). \tag{2.3.8}$$

Here $I$ is $2n \times 2n$ identity matrix and $J = \nabla^2 H(\tilde{y_1})$ is the Jacobian matrix. The Jacobian matrix $J$ is updated at each new step. The initial estimate $y_1$ is taken same as $\tilde{y}_1$. In order to make the code more efficient relatively high number of iterations are allowed. Therefore, we have used the following stopping criteria for the convergence of Newton iteration,

$$\|y_1 - \tilde{y}_1\| < TOL = 10^{-15}.$$

It is however observed that the code converges for 8 maximum iterations.

**Example:**

The qualitative superiorty of symmetric projection methods in controlling parasitism can be illustrated by a representative example of simple pendulum problem (1.2.4) whose total energy is $H = \frac{1}{2}p^2 - \cos(q)$. This problem is capable of parasitic corruption for the initial conditions chosen to be as $p = 0$ and $q = 2.3$ with a step-size of 0.01. The energy error is calculated by subtracting the energy calculated at actual value given with the simple pendulum problem from the energy calculated at approximated value, i.e. energy error= $H_{approx.} - H_{exact}$. However $H_{approx.}$ is evaluated at first output value. We have applied standard projection procedure using method P. However it can be observed from Figure 2.3 that the numerical method is corrupted only after $10,000$ steps. We then applied symmetric projection method using method P as perturbation as well as base method. It is evident from Figure 2.4 that energy is conserved over one million steps.

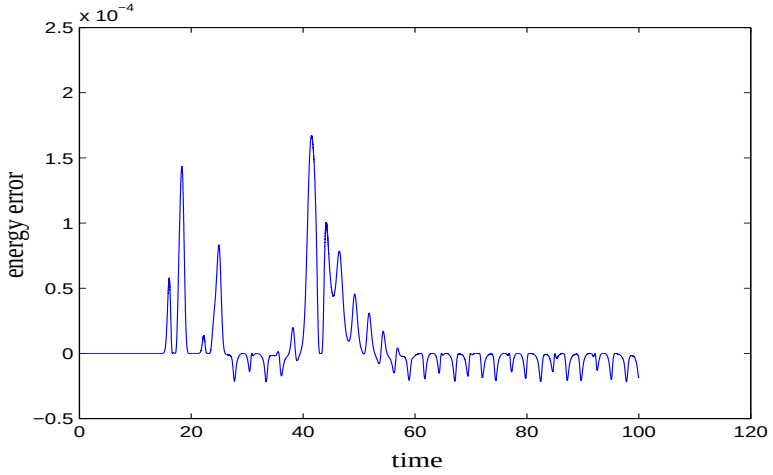


Figure 2.3: Energy conservation of simple pendulum with standard projection for methods P.

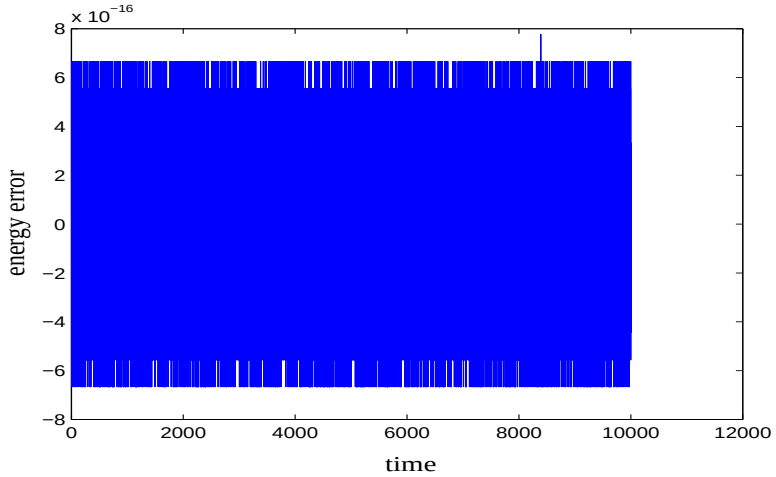


Figure 2.4: Energy conservation of simple pendulum with symmetric projection for methods P.

# Chapter 3

# Numerical experiments

The numerical methods presented in this thesis have been constructed for solving Hamiltonian problems. However, our interest lies in those numerical methods which preserves the qualitative behaviour of the problems over exponentially long time. The methods chosen for implementation include the G-symplectic general linear methods represented as N method, P method and symmetric projection method which uses composition of N and P methods as discussed in Chapter 2. It should be noted here that we have used a fixed step-size in all numerical methods. The numerical methods employed here have implicit stages to evaluate. We have used modified Newton iterations. Each problem is explained by the numerical results along with discussion.

## 3.1 Choice of numerical methods

The numerical solution of the general linear methods prone to parasitism can be controlled by using the symmetric projection methods. We have noted that the numerical solution of general linear method using P method (2.1.12) and N method (2.1.10) suffer from the parasitic solution. The parasitic components of these methods are:

$$\mu_N = 1 - \tfrac{2}{\sqrt{3}}, \qquad \mu_P = 1 + \tfrac{2}{\sqrt{3}}.$$

We observe that in symmetric projection methods, the perturbation at the start of integration step and then the projection of the solution back onto the manifold at the end of integration step minimizes the growth of parameter $\mu$ which is responsible for parasitic solution. We have used two approaches considering symmetric projection method using P method as perturbation method and N method as the base method. This procedure provides good invariant preservation. Similar results hold when we applied symmetric projection method using N method as perturbation method and P method as the base method.

The second approach is to use only one G-symplectic general linear method as perturbation and base method. Here we have used method P and method N as a perturbation method as well as base method. Although good results are achieved but in first approach the adherence of the solution to the invariant manifold is ensured by cancellation of parasitic growth parameters of P and N method being opposite in sign.

The use of different G-symplectic general linear methods with parasitic growth parameters having opposite signs together with symmetric projection works very well as compared to just simply using symmetric projection for only one parasitism infected G-symplectic general linear method. The good behaviour of first approach is due to the fact that parasitic growth parameters of different general linear methods implemented in pairs adds up and hence cancel each other being opposite in sign [17].

Beside these two approaches we have also applied parasitic free G-symplectic general linear method GLM43. It provides qualitatively correct numerical results over long time.

## 3.2 Simple pendulum

The equations of motion of the simple pendulum defines a Hamiltonian system with generalized momenta $p$ and coordinates $q$ and are given as,

$$p' = -\sin(q), \qquad q' = p. \tag{3.2.1}$$

The total energy $H$ which is a conserved quantity is given as,

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

The simple pendulum problem prone to parasitic solutions for initial conditions chosen to be $p = 0$ and $q = 2.3$ using step-size of $h = 0.01$. The energy error is calculated by subtracting the energy calculated at actual value given with the simple pendulum problem from the energy calculated at approximated value, i.e. energy error$= H_{approx.} - H_{exact}$. However $H_{approx.}$ is evaluated at first output value. This is evident from Figure 3.1 where simple pendulum is solved using P method alone for $10,000$ steps only and error in energy is presented. However it can be seen in Figure 3.2 that for the N method the numerical method is corrupted after $100,000$ steps. This is due to the fact that the parasitic parameter of the P method is 2.1547 which is greater than the parasitic parameter of the N method which is $-0.1547$.
We have applied symmetric projection procedure with P method as perturbation method and N method as a base method on the simple pendulum. We have taken 1 million steps and the error in energy is plotted in Figure 3.3 using step-size. The results show good energy conservation. Similar result holds if we swap P and N methods. We have then used N method as perturbation method as well as base method and the error in energy is plotted in Figure 3.4. Although this seems to be good result, however the graph shows that in first case, the adherence of solution to the invariant manifold is ensured by both the symmetric projection as well as cancellation of parasitic growth parameters of P and N methods being opposite in sign whereas in later case we only have symmetric projection to project the solution back onto the manifold. Figure 3.5 also shows good energy conservation for parasitic free method GLM43.

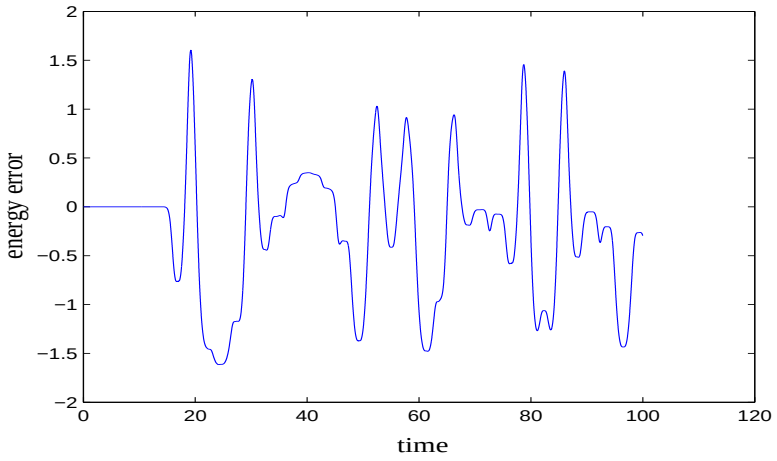


Figure 3.1: Energy conservation of simple pendulum for method P using $h = 0.01$.

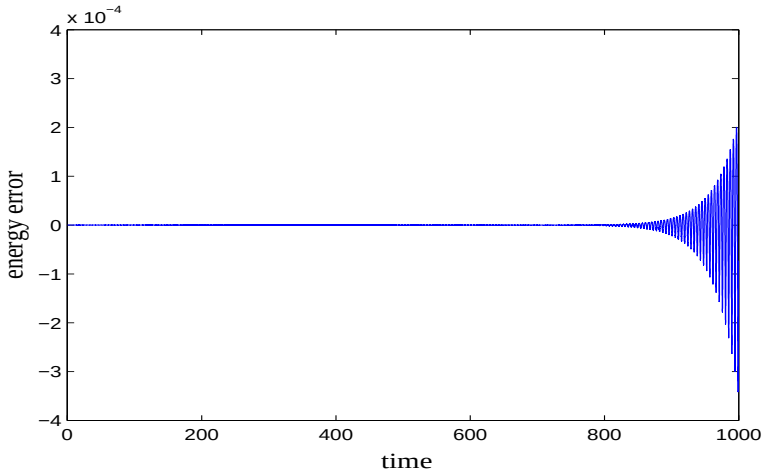


Figure 3.2: Energy conservation of simple pendulum for method N using $h = 0.01$.

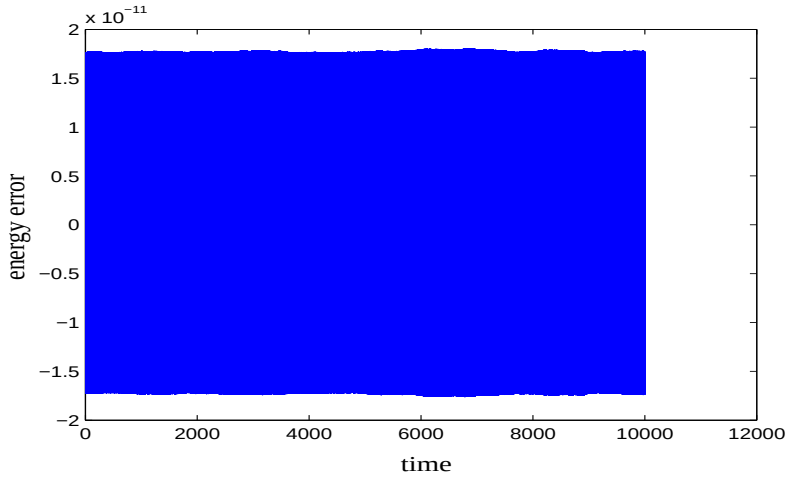


Figure 3.3: Energy conservation of simple pendulum with symmetric projection for methods P and N using $h = 0.01$.

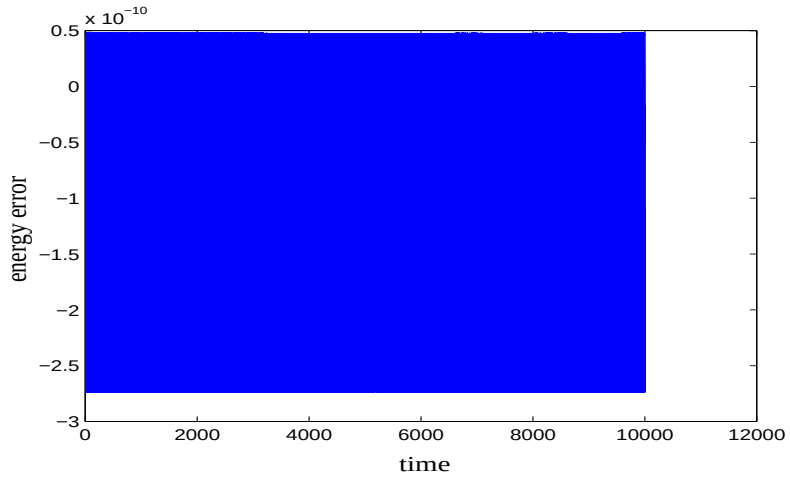


Figure 3.4: Energy conservation of simple pendulum with symmetric projection for method N using $h = 0.01$.

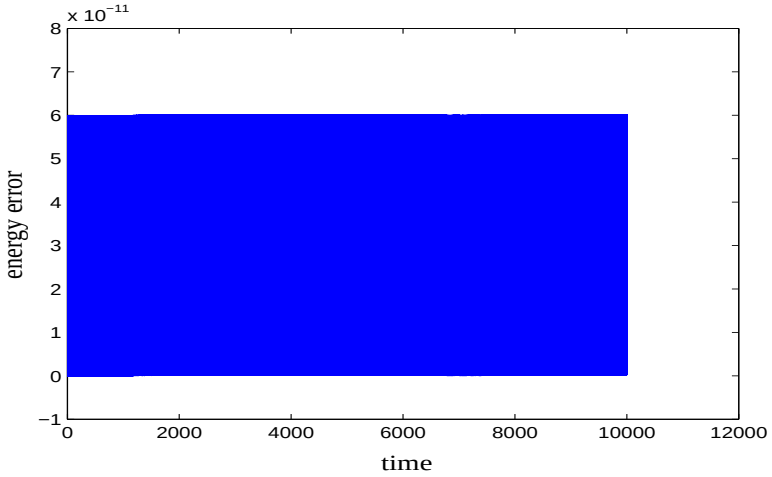


Figure 3.5: Energy conservation of simple pendulum for GLM43 using $h = 0.01$.

## 3.3 The perturbed Kepler problem

Consider the following equations of motion for the perturbed Kepler problem,

$$\begin{aligned} q_1' &= p_1, \\ q_2' &= p_2, \\ p_1' &= -\frac{q_1}{\sqrt{(q_1^2+q_2^2)^3}} - \frac{(0.0075)q_1}{\sqrt{(q_1^2+q_2^2)^5}}, \\ p_2' &= -\frac{q_2}{\sqrt{(q_1^2+q_2^2)^3}} - \frac{(0.0075)q_2}{\sqrt{(q_1^2+q_2^2)^5}}, \end{aligned} \tag{3.3.1}$$

where $(q_1, q_2)$ are the generalized coordinates and $(p_1, p_2)$ are the generalized momenta. The total energy of the system is

$$H = \frac{1}{2}(p_1^2 + p_2^2) - \frac{1}{\sqrt{(q_1^2+q_2^2)}} - \frac{0.005}{2\sqrt{(q_1^2+q_2^2)^3}}.$$

The initial conditions are

$$(q_1, q_2, p_1, p_2) = (1-e, 0, 0, \sqrt{\frac{1+e}{1-e}}).$$

where $0 < e < 1$ is the eccentricity of the elliptic orbits which are formed by the

motion of one body around the other.

In this experiment we examined the energy conservation of perturbed Kepler problem (3.3.1) with $e = 0$ and step-size $\frac{2\pi}{600}$. The energy error is calculated by subtracting the energy calculated at actual value given with the perturbed Kepler problem from the energy calculated at approximated value, i.e. energy error= $H_{approx.} - H_{exact}$. However $H_{approx.}$ is evaluated at first output value. We observed the corruption of numerical solution due to parasitism in the P method shown in Figure 3.6, where an energy drift is observed only after 200 steps. However for the N method in Figure 3.7 it can be seen that the numerical method is corrupted after longer time.

We have applied symmetric projection procedure with P method as perturbation method and N method as a base method using $e = 0$ with same step-size. The error in energy is shown in Figure 3.8 for 1 million steps. We have then used P method as perturbation method as well as base method and the error in energy is plotted in Figure 3.9. Similarly the energy error for GLM43 is shown in Figure 3.10.

It can be followed from graphs that error in energy using symmetric projections with different G-symplectic general linear methods and GLM43 is very small and ensures long time energy conservation. The use of different G-symplectic general linear methods with parasitic growth parameters having opposite signs together with symmetric projection works very well as compared to just simply using symmetric projection for only one parasitism infected G-symplectic general linear method. The good behaviour of earlier case is down to the fact that parasitic growth parameters of different general linear methods implemented in pairs adds up and hence cancel each other being opposite in sign.

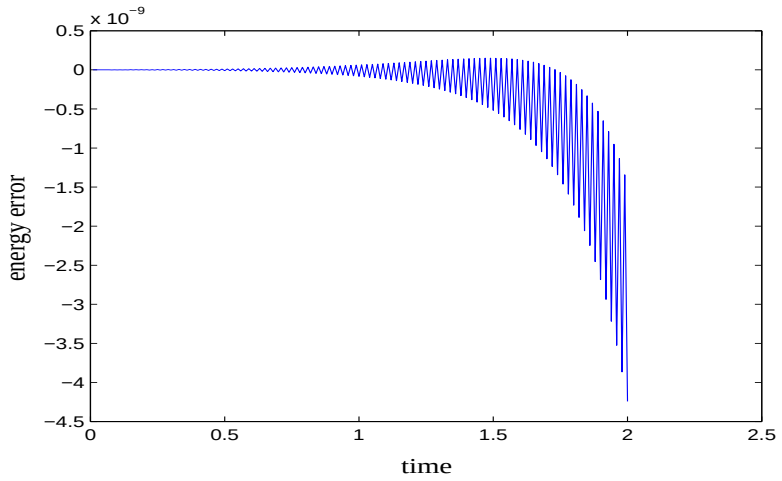


Figure 3.6: Energy conservation of perturbed Kepler for method P, $e = 0$.

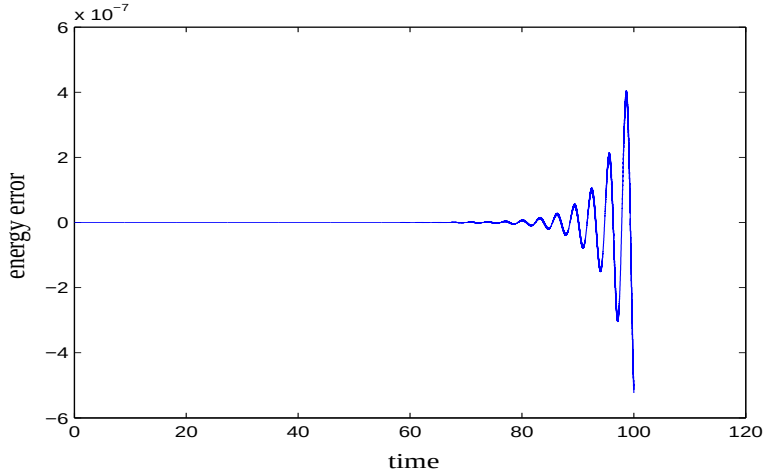


Figure 3.7: Energy conservation of perturbed Kepler for method N, $e = 0$.

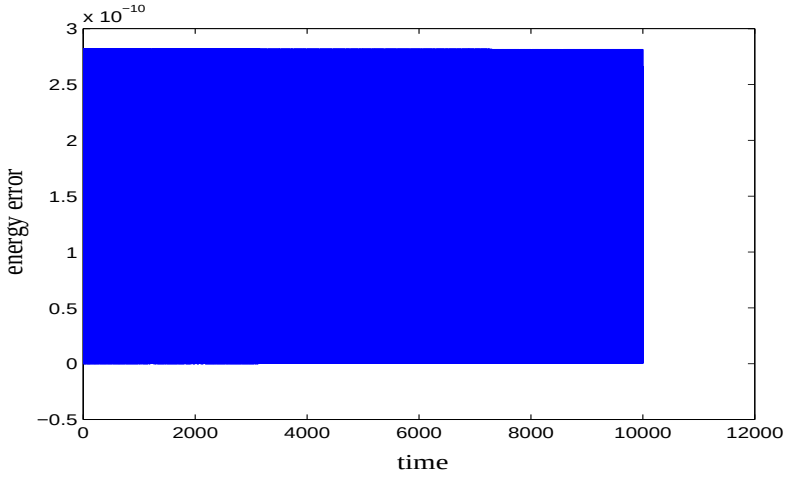


Figure 3.8: Energy conservation of perturbed Kepler with symmetric projection for methods P and N, $e = 0$.

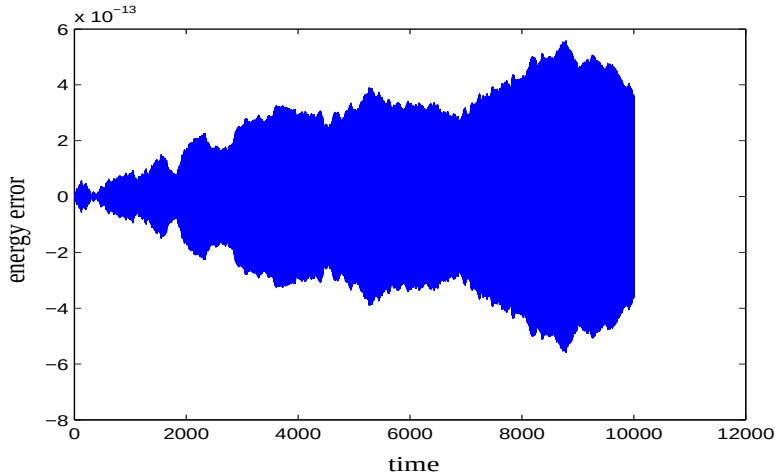


Figure 3.9: Energy conservation of perturbed Kepler with symmetric projection for method P, $e = 0$.

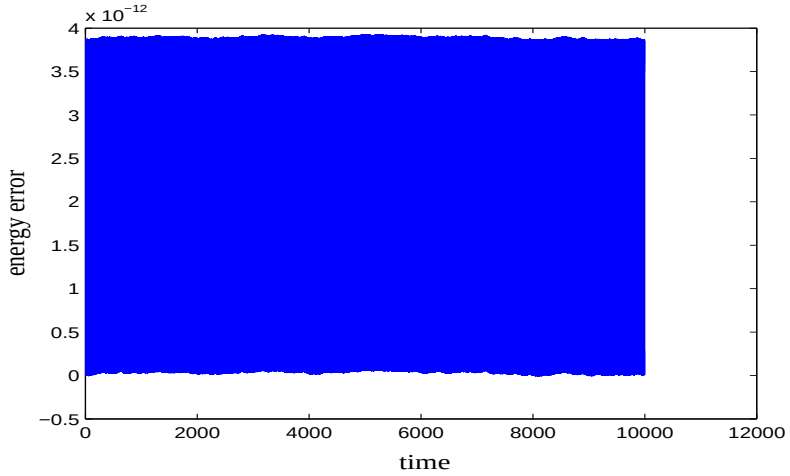


Figure 3.10: Energy conservation of perturbed Kepler for GLM43, $e = 0$.

## 3.4 Harmonic oscillator

The Hamiltonian system defined by the motion of a unit mass attached to a spring with momentum $p$ and position coordinates $q$ is given as,

$$p' = -q, \qquad q' = p, \tag{3.4.1}$$

and the total energy $H$,

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

For numerical solution of harmonic oscillator problem, a step-size of 0.01 is taken with $p = 0$ and $q = 1.2$ for one million steps. The energy error is calculated by subtracting the energy calculated at actual value given with the harmonic oscillator problem from the energy calculated at approximated value, i.e. energy error$= H_{approx.} - H_{exact}$. However $H_{approx.}$ is evaluated at first output value. We observed good energy conservation for P and N methods as shown in Figures 3.11 and 3.12 respectively. This is due to the fact that the parasitic solution is not overtaken by the actual solution. Not surprisingly, we obtain excellent energy conservation for symmetric projection method with P method as perturbation method and N method as a base method in Figure 3.13, for symmetric projection method with P

method as perturbation as well as base method in Figure 3.14 and for the method GLM43 in Figure 3.15.

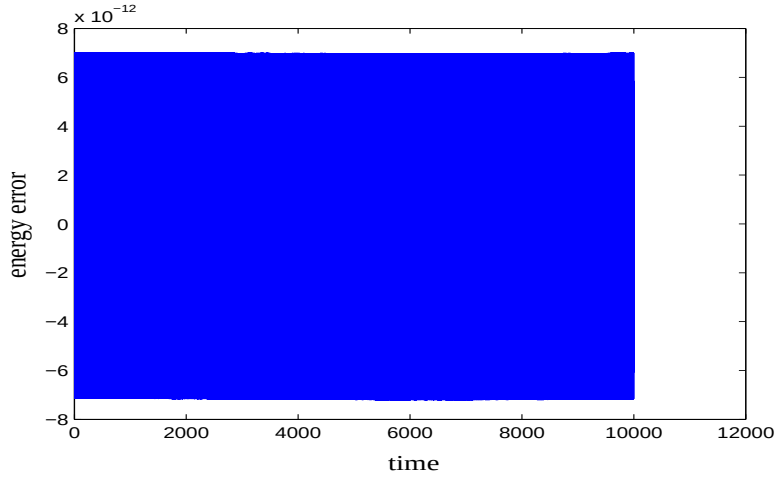


Figure 3.11: Energy conservation of harmonic oscillator for method P.

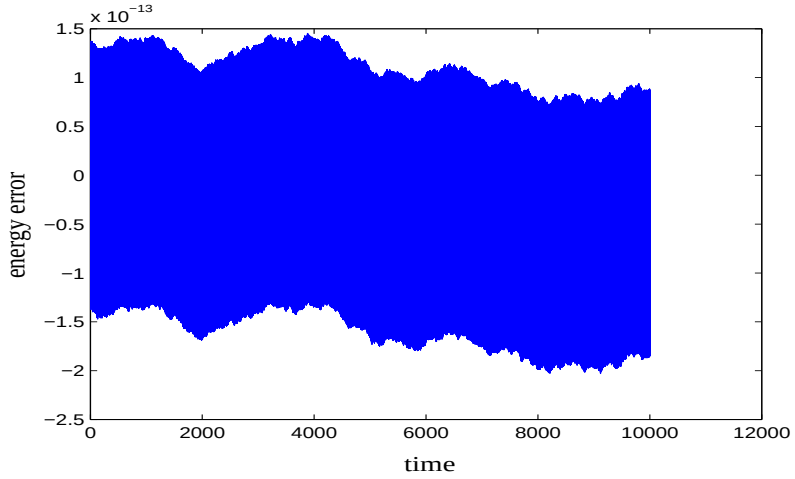


Figure 3.12: Energy conservation of harmonic oscillator for method N.

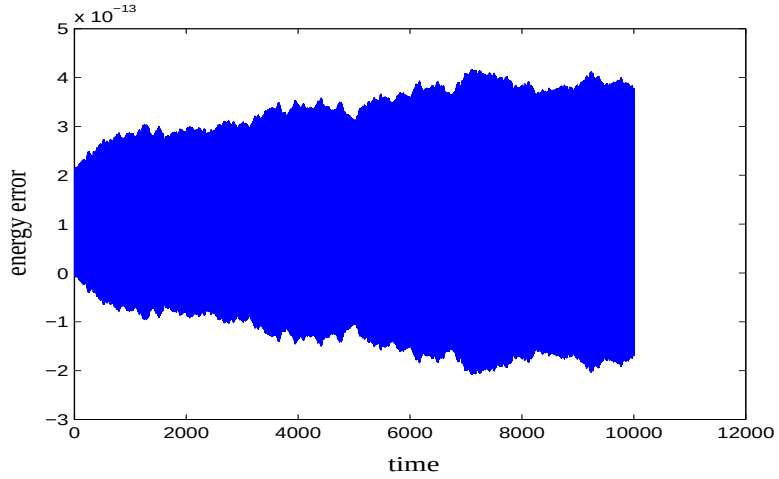


Figure 3.13: Energy conservation of harmonic oscillator with symmetric projection for methods P and N.

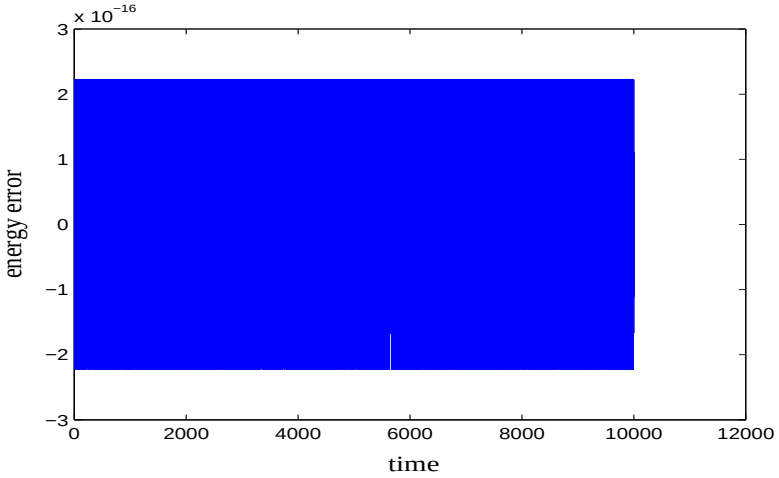


Figure 3.14: Energy conservation of harmonic oscillator with symmetric projection for method P.

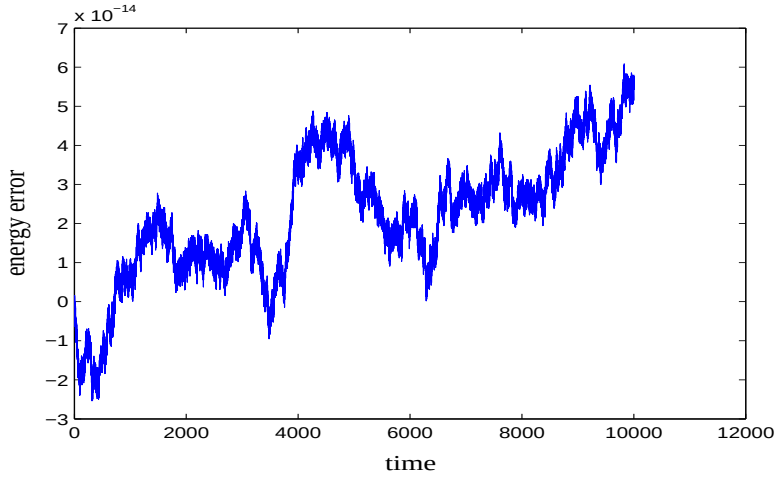


Figure 3.15: Energy conservation of harmonic oscillator for GLM43.

## 3.5 Rigid body motion

The motion of a rigid body, with centre of mass at the origin, can be illustrated by the Euler equations as,

$$\begin{aligned}
\frac{d\omega_x}{dt} &= \frac{(I_y - I_z)}{I_y I_z}\omega_y\omega_z, \\
\frac{d\omega_y}{dt} &= \frac{(I_z - I_x)}{I_z I_x}\omega_z\omega_x, \\
\frac{d\omega_z}{dt} &= \frac{(I_x - I_y)}{I_x I_y}\omega_x\omega_y.
\end{aligned}$$

Here the vector $\omega = [\omega_x, \omega_y, \omega_z]^T$ denotes the angular momentum and $I_x$,$I_y$,$I_z$ are the principal moment of inertia. The two underlying quadratic invariants of the rigid body motion namely, the kinetic energy $H$ and the angular momentum $A$ are represented as,

$$\begin{aligned}
H &= \frac{1}{2}(\frac{\omega_x^2}{I_x} + \frac{\omega_y^2}{I_y} + \frac{\omega_z^2}{I_z}), \\
A &= \omega_x^2 + \omega_y^2 + \omega_z^2.
\end{aligned}$$

We apply the numerical methods with a step-size of 0.01 and initial condition $\omega_0 = [\cos(0.9), 0, \sin(0.9)]^T$ for one million steps. The energy error is calculated

by subtracting the energy calculated at actual value given with the harmonic oscillator problem from the energy calculated at approximated value, i.e. energy error= $H_{approx.} - H_{exact}$. However $H_{approx.}$ is evaluated at first output value. Similarly the momentum error is calculated as momentum error= $A_{approx.} - A_{exact}$. The principal moments of inertia are chosen to be as $I_x = 2$, $I_y = 1$, $I_z = 2/3$. For P and N methods, excellent preservation of these quadratic invariants are observed as shown in the Figures 3.16 and 3.17 respectively. As expected, we get good conservation of invariants for symmetric projection method with N method as perturbation method and P method as a base method in Figure 3.18, for symmetric projection method with N method as perturbation as well as base method in Figure 3.19 and for the method GLM43 in Figure 3.20.

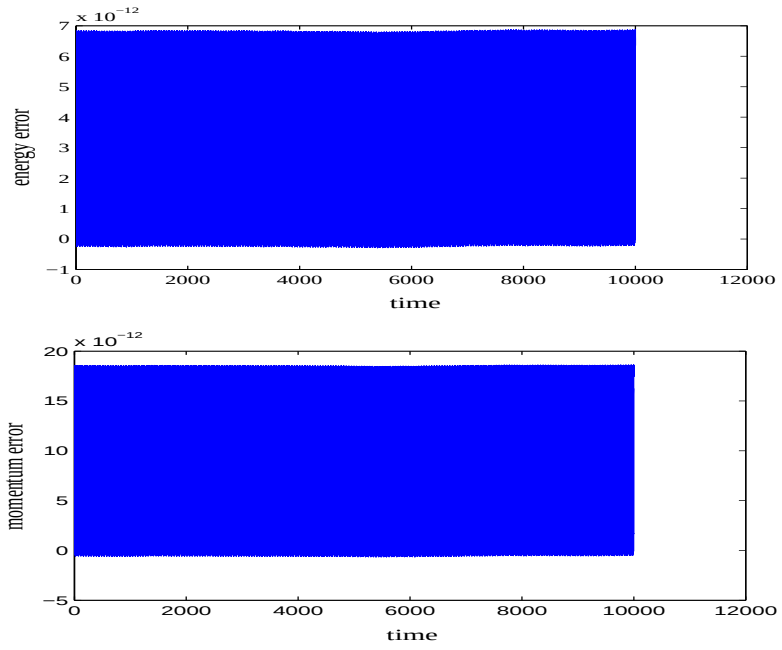


Figure 3.16: Conservation of invariants of rigid body motion for method P.

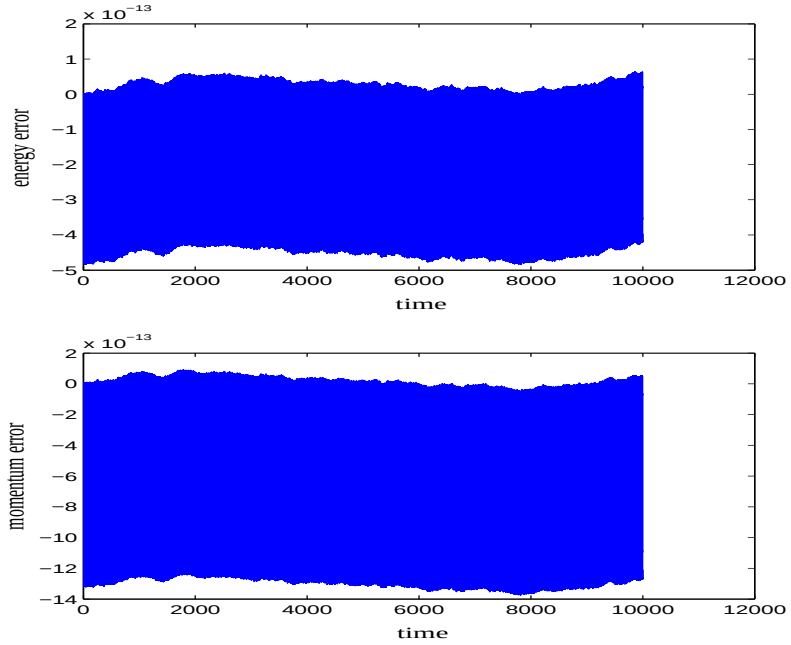


Figure 3.17: Conservation of invariants of rigid body motion for method N.

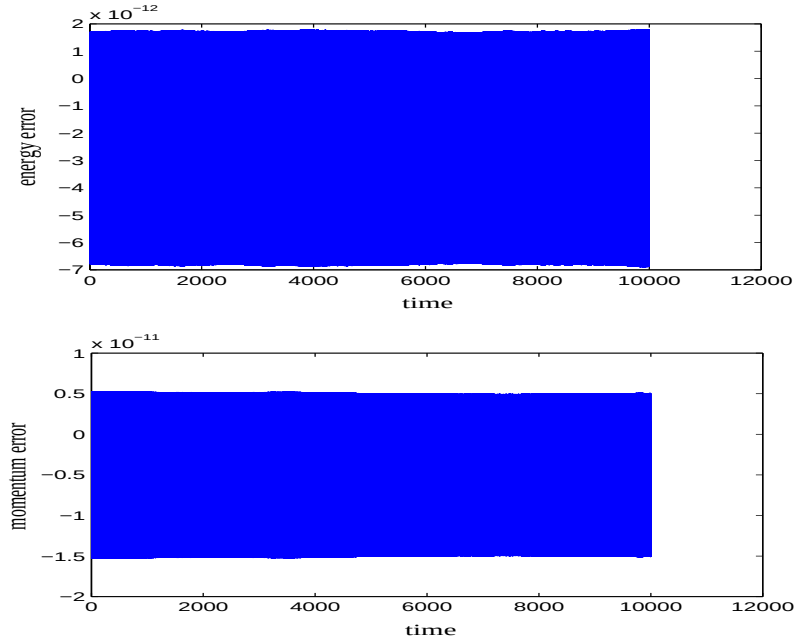


Figure 3.18: Conservation of invariants of rigid body motion with symmetric projection for methods N and P.

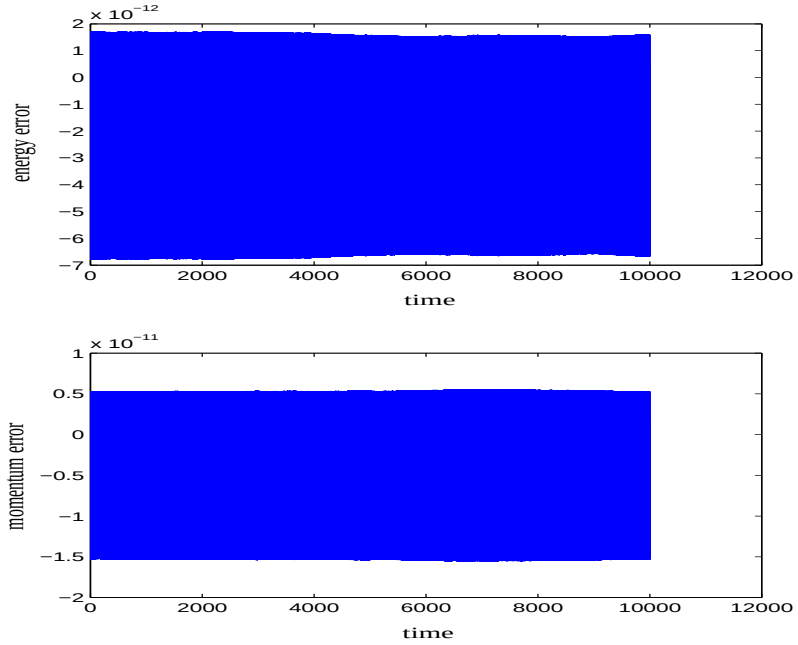


Figure 3.19: Conservation of invariants of rigid body motion with symmetric projection for method N.

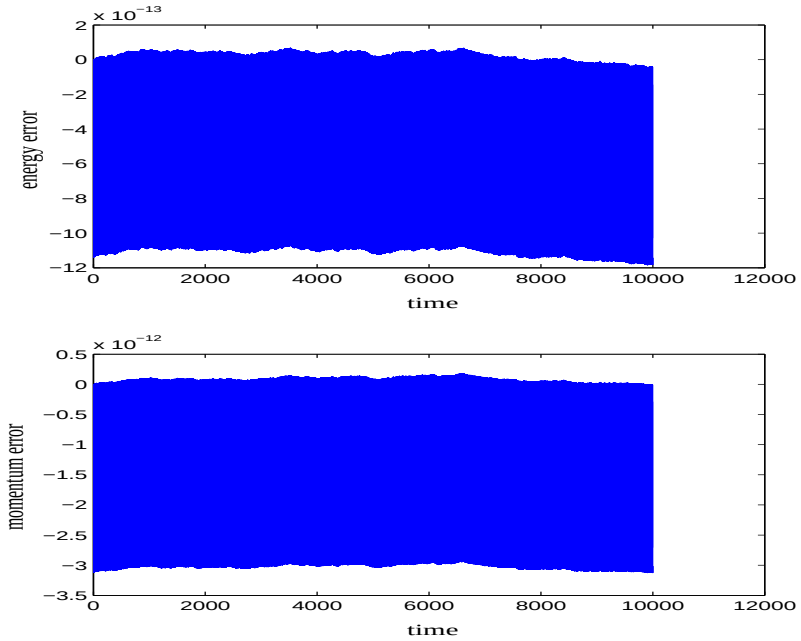


Figure 3.20: Conservation of invariants of rigid body motion for method GLM43.

# Chapter 4

# Conclusions

In this thesis, we have presented the preservation of quadratic invariants of Hamiltonian systems using G-symplectic general linear methods. The multi-value nature of these methods leads to the corruption of numerical solution, subject to the parasitic solution. Symmetric projection approach has been employed for eliminating this parasitism effect. We used various problems such as simple pendulum, perturbed Kepler, harmonic oscillator and rigid body motion problem. In all experiments we have used fixed step-size.

In Chapter 1, an introduction of Hamiltonian systems and quadratic invariants were given. We have also reviewed the general linear methods for numerical solution of ordinary differential equation systems. Chapter 2 deals with a detailed study of G-symplectic general linear methods and projection methods. In particular, control of parasitism via symmetric projection was investigated. In Chapter 3, G-symplectic general linear methods with parasitism are implemented together with symmetric projection onto the invariant manifold. Standard projection fails to control parasitism in methods over exponentially long time intervals. Moreover, use of single G-symplectic general linear methods with parasitism is also not a favourable candidate if we consider symmetric projection. Good invariant preservation is achieved if we use symmetric projection together with G-symplectic general linear methods having parasitic growth parameters with opposite signs.

# Bibliography


[1] K. Burrage, and J. C. Butcher. Non-linear stability of a general class of differential equation methods. *BIT Numerical Mathematics*, $20 : 185 - 203, 1980$.

[2] J. M. Sanz-Serna. Symplectic integrators for Hamiltonian problems. *Acta Numer.*, $1 : 243 - 286, 1992$.

[3] T. Eirola, and J. M. Sanz-Serna. Conservation of integrals and symplectic structure in the integration of differential equations by multistep methods. *Numer. Math*, $61 : 281 - 290, 1992$.

[4] Y. F. Tang. The symplecticity of multi-step methods. *Computers Math. Applic.*, $25 : 83 - 90, 1993$.

[5] H. Yoshida. Recent progress in the theory and application of symplectic integrators. *Celestial Mech. and Dyn. Astr.*, $56 : 27 - 43, 1993$.

[6] A. Iserles. A first course in the numerical analysis of differential equations. *Cambridge University Press*, 1996.

[7] E. Hairer and G. Wanner. Solving ordinary differential equations II: Stiff and differential-algebraic problems. *Springer, second edition*, 1996.

[8] K. Feng. The step-transition operators for multi-step methods of ODEs, *J. Comput. Math.*, $16 : 193 - 202, 1998$.

[9] E. Hairer. Symmetric projection methods for differential equations on manifolds. *BIT Numerical Mathematics*, $4 : 726 - 734, 2000$.

[10] S. Maeda. Symplectic Runge-Kutta methods from the viewpoint of symmetry. *J. Math. Tokushima Univ*, $34 : 15 - 22, 2000$.

[11] E. Hairer, C. Lubich, and G. Wanner. Geometric numerical integration: Structure-preserving algorithms for ordinary differential equations. *Springer, second edition*, 2005.

[12] J. C. Butcher. General linear methods. *Acta Numerica*, $15 : 157 - 256, 2006$.

[13] J. C. Butcher. Numerical methods for ordinary differential equations. *Wiley, second edition*, 2008.

[14] E. Hairer. Conjugate-symplecticity of linear multistep methods. *J. Comput. Math.*, $26 : 657 - 659, 2008$.

[15] J. C. Butcher, and L. L. Hewitt. The existence of symplectic general linear methods. *Numer. Algorithms*, $51 : 77 - 84, 2009$.

[16] Y. Habib. Long-Term Behaviour of G-symplectic Methods, PhD Thesis, The University of Auckland, 2010.

[17] D. Ambrosio Raffaele, and E. Hairer. Long-term stability of multi-value methods for ordinary differential equations. *Journal of Scientific Computing*, $1 - 14, 2013$.

[18] D. Ambrosio Raffaele, E. Hairer, and C. J. Zbinden. G-symplecticity implies conjugate-symplecticity of the underlying one-step method. *BIT Numer. Math*, $53 : 867 - 872, 2013$.

[19] J. C. Butcher. Dealing with parasitic behaviour in G-symplectic integrators. *Recent Developments in the Numerics of Nonlinear Hyperbolic Conservation Laws, Notes on Numerical Fluid Mechanics and Multidisciplinary Design. Springer-Verlag, Berlin*, $120 : 105 - 123, 2013$.

[20] J. C. Butcher, Y. Habib, A. T. Hill, and T. J. T. Norton. The control of parasitism in G-symplectic methods. *SIAM J. Numer. Anal*, $52(5) : 2440 - 2465, 2014$.