



Trigonometric collocation methods based on Lagrange basis polynomials for multi-frequency oscillatory second-order differential equations



Bin Wang^{a,*}, Xinyuan Wu^b, Fanwei Meng^a

^a School of Mathematical Sciences, Qufu Normal University, Qufu, Shandong 273165, PR China

^b Department of Mathematics, Nanjing University, Nanjing 210093, PR China

ARTICLE INFO

Article history:

Received 8 April 2016

Received in revised form 6 September 2016

MSC:

65L05

65L06

34C15

34E05

Keywords:

Trigonometric collocation methods

Lagrange polynomials

Multi-frequency oscillatory second-order systems

Variation-of-constants formula

ABSTRACT

In the present work, a kind of trigonometric collocation methods based on Lagrange basis polynomials is developed for effectively solving multi-frequency oscillatory second-order differential equations $q''(t) + Mq(t) = f(q(t))$. The properties of the obtained methods are investigated. It is shown that the convergent condition of these methods is independent of $\|M\|$, which is very crucial for solving oscillatory systems. A fourth-order scheme of the methods is presented. Numerical experiments are implemented to show the remarkable efficiency of the methods proposed in this paper.

© 2016 Elsevier B.V. All rights reserved.

1. Introduction

The numerical treatment of multi-frequency oscillatory systems is a computational problem of an overarching importance in a wide range of applications, such as quantum physics, circuit simulations, flexible body dynamics and mechanics (see, e.g. [1–7] and the references therein). The main theme of the present paper is to construct and analyse a kind of efficient collocation methods for solving multi-frequency oscillatory second-order differential equations of the form

$$q''(t) + Mq(t) = f(q(t)), \quad q(0) = q_0, \quad q'(0) = q'_0, \quad t \in [0, t_{\text{end}}], \quad (1)$$

where M is a $d \times d$ positive semi-definite matrix implicitly containing the frequencies of the oscillatory problem and $f: \mathbb{R}^d \rightarrow \mathbb{R}^d$ is an analytic function. The solution of this system is a multi-frequency nonlinear oscillator because of the presence of the linear term Mq . System (1) is a highly oscillatory problem when $\|M\| \gg 1$. In recent years, various numerical methods for approximating solutions of oscillatory systems have been developed by many researchers. Readers are referred to [8–16] and the references therein. Once it is further assumed that M is symmetric and f is the negative gradient of a

* Corresponding author.

E-mail addresses: wangbinmaths@gmail.com (B. Wang), xywu@nju.edu.cn (X. Wu), fwmeng@mail.qfnu.edu.cn (F. Meng).

real-valued function $U(q)$, the system (1) is identical to the following initial value Hamiltonian system

$$\begin{aligned}\dot{q} &= \nabla_p H(q, p), & q(0) &= q_0, \\ \dot{p} &= -\nabla_q H(q, p), & p(0) &= p_0 \equiv q'_0\end{aligned}\quad (2)$$

with the Hamiltonian function

$$H(q, p) = \frac{1}{2} p^T p + \frac{1}{2} q^T M q + U(q). \quad (3)$$

This is an important system which has received much attention by many authors (see, e.g. [1,2,4,5,17]).

In [18], the authors took advantage of shifted Legendre polynomials to obtain a local Fourier expansion of the system (1) and derived a kind of collocation methods (trigonometric collocation methods). The analysis and the results of numerical experiments in [18] showed that the trigonometric collocation methods are more efficient in comparison with some alternative approaches that have previously appeared in the literature. Motivated by the work in [18], this paper is devoted to the formulation and analysis of another trigonometric collocation methods for solving multi-frequency oscillatory second-order systems (1). We will consider a more classical approach and use Lagrange polynomials to obtain the methods. Because of this different approach, compared with the methods in [18], the obtained methods have a simpler scheme and can be implemented in practical computations at a lower cost. These trigonometric collocation methods are designed by interpolating the function f of (1) by Lagrange basis polynomials, and incorporating the variation-of-constants formula with the idea of collocation methods. It is noted that these integrators are a kind of collocation methods and they share all the interesting features of collocation methods. We analyse the properties of the trigonometric collocation methods. We also consider the convergence of the fixed-point iteration for the methods. It is important to emphasize that for the trigonometric collocation methods, the convergent condition is independent of $\|M\|$, which is a very important property for solving oscillatory systems.

This paper is organized as follows. In Section 2, we formulate the scheme of trigonometric collocation methods based on Lagrange basis polynomials. The properties of the obtained methods are analysed in Section 3. In Section 4, a fourth-order scheme of the methods is presented and numerical tests confirm that the method proposed in this paper yields a dramatic improvement. Conclusions are included in Section 5.

2. Formulation of the methods

To begin with we restrict the multi-frequency oscillatory system (1) to the interval $[0, h]$ with any $h > 0$:

$$q''(t) + Mq(t) = f(q(t)), \quad q(0) = q_0, \quad q'(0) = q'_0, \quad t \in [0, h]. \quad (4)$$

With regard to the variation-of-constants formula for (1) given in [19], we have the following result on the exact solution $q(t)$ of the system (1) and its derivative $q'(t) = p(t)$:

$$\begin{aligned}q(t) &= \phi_0(t^2 M) q_0 + t \phi_1(t^2 M) p_0 + t^2 \int_0^1 (1-z) \phi_1((1-z)^2 t^2 M) f(q(tz)) dz, \\ p(t) &= -t M \phi_1(t^2 M) q_0 + \phi_0(t^2 M) p_0 + t \int_0^1 \phi_0((1-z)^2 t^2 M) f(q(tz)) dz,\end{aligned}\quad (5)$$

where $t \in [0, h]$ and

$$\phi_i(M) := \sum_{l=0}^{\infty} \frac{(-1)^l M^l}{(2l+i)!}, \quad i = 0, 1. \quad (6)$$

From this result, it follows that

$$\begin{aligned}q(h) &= \phi_0(V) q_0 + h \phi_1(V) p_0 + h^2 \int_0^1 (1-z) \phi_1((1-z)^2 V) f(q(hz)) dz, \\ p(h) &= -h M \phi_1(V) q_0 + \phi_0(V) p_0 + h \int_0^1 \phi_0((1-z)^2 V) f(q(hz)) dz,\end{aligned}\quad (7)$$

where $V = h^2 M$.

The main point in designing practical schemes to solve (1) is based on replacing $f(q)$ in (7) by some expansion. In this paper, we interpolate $f(q)$ as

$$f(q(\xi h)) \sim \sum_{j=1}^s l_j(\xi) f(q(c_j h)), \quad \xi \in [0, 1], \quad (8)$$

where

$$l_j(x) = \prod_{k=1, k \neq j}^s \frac{x - c_k}{c_j - c_k} \quad (9)$$

for $j = 1, \dots, s$ are the Lagrange basis polynomials in interpolation and c_1, \dots, c_s are distinct real numbers (usually $s \geq 1$, $0 \leq c_i \leq 1$). Then replacing $f(q(\xi h))$ in (7) by the series (8) yields an approximation of $q(h)$, $p(h)$ as follows:

$$\begin{aligned} \tilde{q}(h) &= \phi_0(V)q_0 + h\phi_1(V)p_0 + h^2 \sum_{j=1}^s I_{1,j} f(\tilde{q}(c_j h)), \\ \tilde{p}(h) &= -hM\phi_1(V)q_0 + \phi_0(V)p_0 + h \sum_{j=1}^s I_{2,j} f(\tilde{q}(c_j h)), \end{aligned} \quad (10)$$

where

$$I_{1,j} := \int_0^1 l_j(z)(1-z)\phi_1((1-z)^2V)dz, \quad I_{2,j} := \int_0^1 l_j(z)\phi_0((1-z)^2V)dz. \quad (11)$$

According to the variation-of-constants formula (5) for (4), the approximation (10) satisfies the following system

$$\begin{aligned} \tilde{q}'(\xi h) &= \tilde{p}(\xi h), \quad \tilde{q}(0) = q_0, \\ \tilde{p}'(\xi h) &= -M\tilde{q}(\xi h) + \sum_{j=1}^s l_j(\xi) f(\tilde{q}(c_j h)), \quad \tilde{p}(0) = p_0. \end{aligned} \quad (12)$$

In what follows we first approximate $f(\tilde{q}(c_j h))$, $I_{1,j}$, $I_{2,j}$ appearing in (10) and then a kind of collocation methods can be formulated.

2.1. The computation of $f(\tilde{q}(c_j h))$

It follows from (12) that $\tilde{q}(c_i h)$, $i = 1, 2, \dots, s$, can be obtained by solving the following discrete problems:

$$\tilde{q}''(c_i h) + M\tilde{q}(c_i h) = \sum_{j=1}^s l_j(c_i) f(\tilde{q}(c_j h)), \quad \tilde{q}(0) = q_0, \quad \tilde{q}'(0) = p_0. \quad (13)$$

By setting $\tilde{q}_i = \tilde{q}(c_i h)$ with $i = 1, 2, \dots, s$, (13) can be solved by the variation-of-constants formula (5) in the form:

$$\tilde{q}_i = \phi_0(c_i^2 V)q_0 + c_i h \phi_1(c_i^2 V)p_0 + (c_i h)^2 \sum_{j=1}^s \tilde{I}_{c_i,j} f(\tilde{q}_j), \quad i = 1, 2, \dots, s,$$

where

$$\tilde{I}_{c_i,j} := \int_0^1 l_j(c_i z)(1-z)\phi_1((1-z)^2 c_i^2 V)dz, \quad i, j = 1, \dots, s. \quad (14)$$

2.2. The computation of $I_{1,j}$, $I_{2,j}$, $\tilde{I}_{c_i,j}$

With the definition (9), the integrals $I_{1,j}$, $I_{2,j}$, $\tilde{I}_{c_i,j}$ appearing above can be computed as follows:

$$\begin{aligned} I_{1,j} &= \int_0^1 l_j(z)(1-z)\phi_1((1-z)^2V)dz \\ &= \prod_{k=1, k \neq j}^s \sum_{l=0}^{\infty} \int_0^1 \frac{z - c_k}{c_j - c_k} (1-z)^{2l+1} dz \frac{(-1)^l V^l}{(2l+1)!} \\ &= \sum_{l=0}^{\infty} \left(\prod_{k=1, k \neq j}^s \frac{\frac{1}{2l+3} - c_k}{c_j - c_k} \right) \frac{(-1)^l V^l}{(2l+2)!} = \sum_{l=0}^{\infty} l_j \left(\frac{1}{2l+3} \right) \frac{(-1)^l V^l}{(2l+2)!}, \\ I_{2,j} &= \int_0^1 l_j(z)\phi_0((1-z)^2V)dz = \prod_{k=1, k \neq j}^s \sum_{l=0}^{\infty} \int_0^1 \frac{z - c_k}{c_j - c_k} (1-z)^{2l} dz \frac{(-1)^l V^l}{(2l)!} \end{aligned}$$

$$\begin{aligned}
&= \sum_{l=0}^{\infty} \left(\prod_{k=1, k \neq j}^s \frac{\frac{1}{2l+2} - c_k}{c_j - c_k} \right) \frac{(-1)^l V^l}{(2l+1)!} = \sum_{l=0}^{\infty} l_j \left(\frac{1}{2l+2} \right) \frac{(-1)^l V^l}{(2l+1)!}, \\
\tilde{I}_{c_i, j} &= \int_0^1 l_j(c_i z) (1-z) \phi_1((1-z)^2 c_i^2 V) dz \\
&= \prod_{k=1, k \neq j}^s \sum_{l=0}^{\infty} \int_0^1 \frac{c_i z - c_k}{c_j - c_k} (1-z)^{2l+1} dz \frac{(-1)^l (c_i^2 V)^l}{(2l+1)!} \\
&= \sum_{l=0}^{\infty} \left(\prod_{k=1, k \neq j}^s \frac{\frac{c_i}{2l+3} - c_k}{c_j - c_k} \right) \frac{(-1)^l (c_i^2 V)^l}{(2l+2)!} = \sum_{l=0}^{\infty} l_j \left(\frac{c_i}{2l+3} \right) \frac{(-1)^l (c_i^2 V)^l}{(2l+2)!}, \quad i, j = 1, \dots, s.
\end{aligned}$$

When the matrix M is symmetric and positive semi-definite, we have the decomposition of M as follows:

$$M = P^T W^2 P = \Omega_0^2 \quad \text{with } \Omega_0 = P^T W P,$$

where P is an orthogonal matrix and $W = \text{diag}(\lambda_k)$ with nonnegative diagonal entries which are the square roots of the eigenvalues of M . Then the above integrals become

$$\begin{aligned}
I_{1,j} &= P^T \int_0^1 l_j(z) W^{-1} \sin((1-z)W) dz P = P^T \text{diag} \left(\int_0^1 l_j(z) \lambda_k^{-1} \sin((1-z)\lambda_k) dz \right) P, \\
I_{2,j} &= P^T \int_0^1 l_j(z) \cos((1-z)W) dz P = P^T \text{diag} \left(\int_0^1 l_j(z) \cos((1-z)\lambda_k) dz \right) P, \\
\tilde{I}_{c_i, j} &= P^T \int_0^1 l_j(c_i z) (c_i W)^{-1} \sin((1-z)c_i W) dz \\
&= P^T \text{diag} \left(\int_0^1 l_j(c_i z) (c_i \lambda_k)^{-1} \sin((1-z)c_i \lambda_k) dz \right) P, \quad i, j = 1, \dots, s.
\end{aligned}$$

It is noted that $W^{-1} \sin((1-z)W)$, $(c_i W)^{-1} \sin((1-z)c_i W)$ are well-defined also for singular W . The case of $\lambda_k = 0$ gives:

$$\begin{aligned}
\int_0^1 l_j(z) \lambda_k^{-1} \sin((1-z)\lambda_k) dz &= \int_0^1 l_j(z) (1-z) dz, \\
\int_0^1 l_j(z) \cos((1-z)\lambda_k) dz &= \int_0^1 l_j(z) dz, \\
\int_0^1 l_j(c_i z) (c_i \lambda_k)^{-1} \sin((1-z)c_i \lambda_k) dz &= \int_0^1 l_j(c_i z) (1-z) dz,
\end{aligned}$$

which can be evaluated easily since $l_j(z)$ is a polynomial function. If $\lambda_k \neq 0$, they can be evaluated as follows:

$$\begin{aligned}
\int_0^1 l_j(z) \lambda_k^{-1} \sin((1-z)\lambda_k) dz &= 1/\lambda_k \int_0^1 l_j(z) \sin((1-z)\lambda_k) dz = 1/\lambda_k^2 \int_0^1 l_j(z) d \cos((1-z)\lambda_k) \\
&= 1/\lambda_k^2 l_j(1) - 1/\lambda_k^2 l_j(0) \cos(\lambda_k) - 1/\lambda_k^2 \int_0^1 l_j'(z) \cos((1-z)\lambda_k) dz \\
&= 1/\lambda_k^2 l_j(1) - 1/\lambda_k^2 l_j(0) \cos(\lambda_k) + 1/\lambda_k^3 \int_0^1 l_j'(z) d \sin((1-z)\lambda_k) \\
&= 1/\lambda_k^2 l_j(1) - 1/\lambda_k^2 l_j(0) \cos(\lambda_k) - 1/\lambda_k^3 l_j'(0) \sin(\lambda_k) \\
&\quad - 1/\lambda_k^3 \int_0^1 l_j''(z) \sin((1-z)\lambda_k) dz \\
&= 1/\lambda_k^2 l_j(1) - 1/\lambda_k^2 l_j(0) \cos(\lambda_k) - 1/\lambda_k^3 l_j'(0) \sin(\lambda_k) \\
&\quad - 1/\lambda_k^4 l_j''(1) + 1/\lambda_k^4 l_j''(0) \cos(\lambda_k) \\
&\quad + 1/\lambda_k^5 l_j^{(3)}(0) \sin(\lambda_k) + 1/\lambda_k^5 \int_0^1 l_{l,j}^{(4)}(z) \sin((1-z)\lambda_k) dz \\
&= \dots \\
&= \sum_{k=0}^{\lfloor \deg(l_j)/2 \rfloor} (-1)^k / \lambda_k^{2k+2} \left(l_j^{(2k)}(1) - l_j^{(2k)}(0) \cos(\lambda_k) - 1/\lambda_k l_j^{(2k+1)}(0) \sin(\lambda_k) \right), \\
&\quad i = 1, 2, \dots, s,
\end{aligned}$$

where $\deg(l_j)$ is the degree of l_j and $\lfloor \deg(l_j)/2 \rfloor$ denotes the integral part of $\deg(l_j)/2$. Similarly, we obtain

$$\begin{aligned} & \int_0^1 l_j(z) \cos((1-z)\lambda_k) dz \\ &= \sum_{k=0}^{\lfloor \deg(l_j)/2 \rfloor} (-1)^k / \lambda_k^{2k+1} \left(l_j^{(2k)}(0) \sin(\lambda_k) + 1/\lambda_k l_j^{(2k+1)}(0) - 1/\lambda_k^2 l_j^{(2k+1)}(0) \cos(\lambda_k) \right), \\ & \int_0^1 l_j(c_i z) (c_i \lambda_k)^{-1} \sin((1-z)c_i \lambda_k) dz \\ &= \sum_{k=0}^{\lfloor \deg(l_j)/2 \rfloor} (-1)^k / (c_i \lambda_k)^{2k+2} \left(l_j^{(2k)}(c_i) - l_j^{(2k)}(0) \cos(c_i \lambda_k) - 1/\lambda_k l_j^{(2k+1)}(0) \sin(c_i \lambda_k) \right), \quad i, j = 1, 2, \dots, s. \end{aligned} \quad (15)$$

2.3. The scheme of trigonometric collocation methods

We are now in a position to present a kind of trigonometric collocation methods for the multi-frequency oscillatory second-order ODEs (1).

Definition 2.1. A trigonometric collocation method for integrating the multi-frequency oscillatory system (1) is defined as

$$\begin{aligned} \tilde{q}_i &= \phi_0(c_i^2 V) q_0 + c_i h \phi_1(c_i^2 V) p_0 + (c_i h)^2 \sum_{j=1}^s \tilde{I}_{c_i, j} f(\tilde{q}_j), \quad i = 1, 2, \dots, s, \\ \tilde{q}(h) &= \phi_0(V) q_0 + h \phi_1(V) p_0 + h^2 \sum_{j=1}^s I_{1, j} f(\tilde{q}_j), \\ \tilde{p}(h) &= -h M \phi_1(V) q_0 + \phi_0(V) p_0 + h \sum_{j=1}^s I_{2, j} f(\tilde{q}_j), \end{aligned} \quad (16)$$

where h is the stepsize and $I_{1, j}$, $I_{2, j}$, $\tilde{I}_{c_i, j}$ can be computed as stated in Section 2.2.

Remark 1. In [18], the authors took advantage of shifted Legendre polynomials to obtain a local Fourier expansion of the system (1) and derived trigonometric Fourier collocation methods (TFCMs). TFCMs are the subclass of s -stage ERKN methods which were presented in [19] with the following Butcher tableau:

$$\begin{array}{c|ccc} c_1 & \sum_{j=0}^{r-1} \Pi_{1, j, c_1}(V) b_1 \hat{P}_j(c_1) & \cdots & \sum_{j=0}^{r-1} \Pi_{1, j, c_1}(V) b_s \hat{P}_j(c_s) \\ \vdots & \vdots & \ddots & \vdots \\ c_s & \sum_{j=0}^{r-1} \Pi_{1, j, c_s}(V) b_1 \hat{P}_j(c_1) & \cdots & \sum_{j=0}^{r-1} \Pi_{1, j, c_s}(V) b_s \hat{P}_j(c_s) \\ \hline & \sum_{j=0}^{r-1} \Pi_{1, j}(V) b_1 \hat{P}_j(c_1) & \cdots & \sum_{j=0}^{r-1} \Pi_{1, j}(V) b_s \hat{P}_j(c_s) \\ \hline & \sum_{j=0}^{r-1} \Pi_{2, j}(V) b_1 \hat{P}_j(c_1) & \cdots & \sum_{j=0}^{r-1} \Pi_{2, j}(V) b_s \hat{P}_j(c_s) \end{array} \quad (17)$$

where

$$\begin{aligned} \Pi_{1, j}(V) &:= \int_0^1 \hat{P}_j(z) (1-z) \phi_1((1-z)^2 V) dz, \\ \Pi_{2, j}(V) &:= \int_0^1 \hat{P}_j(z) \phi_0((1-z)^2 V) dz, \\ \Pi_{1, j, c_i}(V) &:= \int_0^1 \hat{P}_j(c_i z) (1-z) \phi_1((1-z)^2 c_i^2 V) dz, \end{aligned}$$

r is an integer with the requirement: $2 \leq r \leq s$, \hat{P}_j are shifted Legendre polynomials over the interval $[0, 1]$ and c_l , b_l , $l = 1, 2, \dots, s$ are the node points and the quadrature weights of a quadrature formula, respectively.

It is noted that the method (16) is also a subclass of s -stage ERKN methods with the following Butcher tableau:

$$\begin{array}{c|ccc} c_1 & \tilde{I}_{c_1,1} & \cdots & \tilde{I}_{c_1,s} \\ \vdots & \vdots & \ddots & \vdots \\ c_s & \tilde{I}_{c_s,1} & \cdots & \tilde{I}_{c_s,s} \\ \hline & I_{1,1} & \cdots & I_{1,s} \\ \hline & I_{2,1} & \cdots & I_{2,s} \end{array} \quad (18)$$

where

$$\begin{aligned} I_{1,j} &:= \int_0^1 l_j(z)(1-z)\phi_1((1-z)^2V)dz, \\ I_{2,j} &:= \int_0^1 l_j(z)\phi_0((1-z)^2V)dz, \\ \tilde{I}_{c_i,j} &:= \int_0^1 l_j(c_i z)(1-z)\phi_1((1-z)^2c_i^2V)dz. \end{aligned}$$

From (17) and (18), it follows clearly that the coefficients of (18) are simpler than (17). Therefore, the scheme of the methods derived in this paper is much simpler than that given in [18]. The obtained methods can be implemented in practical computations at a lower cost, which will be shown by the numerical experiments in Section 4. The reason for this point is that we use a more classical approach and choose Lagrange polynomials to give a local Fourier expansion of the system (1).

Remark 2. It can be observed from the two tableaux (17)–(18) that the methods here presented are different from those presented in [18]. We also note that in the recent monograph [20], it has been shown that the approach of constructing energy-preserving methods for Hamiltonian problems which are based upon the use of shifted Legendre polynomials (such as in [21]) and Lagrange polynomials constructed on Gauss–Legendre nodes (such as in [22]) leads to precisely the same methods. Therefore, by choosing special real numbers c_1, \dots, c_s for (18) and special quadrature formulae for (17), the methods given in this paper may have some connections with those in [18]. We will discuss the connections in a future research.

Remark 3. It is noted that the method (16) can be applied to the system (1) with an arbitrary matrix M since the trigonometric collocation methods do not need the symmetry of M . Moreover, the method (16) exactly integrates the linear system $q'' + Mq = 0$ and it has an additional advantage of energy preservation for linear systems. The method approximates the solution in the interval $[0, h]$. We then lend the procedure with equal ease to next interval. Namely, we can consider the obtained result as the initial condition for a new initial value problem in the interval $[h, 2h]$. In this way, the method (16) can approximate the solution in an arbitrary interval $[0, t_{\text{end}}]$ with $t_{\text{end}} = Nh$.

When $M = 0$, (1) reduces to a special and important class of systems of second-order ODEs expressed in the traditional form

$$q''(t) = f(q(t)), \quad q(0) = q_0, \quad q'(0) = q'_0, \quad t \in [0, t_{\text{end}}]. \quad (19)$$

For this case, with the definition (6) and the results of $I_{1,j}$, $I_{2,j}$, $\tilde{I}_{c_i,j}$ in Section 2.2, the trigonometric collocation method (16) becomes the following scheme.

Definition 2.2. An RKN-type collocation method for integrating the traditional second-order ODEs (19) is defined as

$$\begin{aligned} \tilde{q}_i &= q_0 + c_i h p_0 + (c_i h)^2 \sum_{j=1}^s \frac{l_j\left(\frac{c_i}{3}\right)}{2} f(\tilde{q}_j), \quad i = 1, 2, \dots, s, \\ \tilde{q}(h) &= q_0 + h p_0 + h^2 \sum_{j=1}^s \frac{l_j\left(\frac{1}{3}\right)}{2} f(\tilde{q}_j), \\ \tilde{p}(h) &= p_0 + h \sum_{j=1}^s l_j\left(\frac{1}{2}\right) f(\tilde{q}_j), \end{aligned} \quad (20)$$

where h is the stepsize.

Remark 4. It is noted that the method (20) is the subclass of s -stage RKN methods with the following Butcher tableau:

$$\begin{array}{c|c} c & \bar{A} = (\bar{a}_{ij})_{s \times s} \\ \hline & \bar{b}^T \\ & b^T \end{array} = \begin{array}{c|ccc} c_1 & l_1\left(\frac{c_1}{3}\right)/2 & \cdots & l_s\left(\frac{c_1}{3}\right)/2 \\ \vdots & \vdots & \ddots & \vdots \\ c_s & l_1\left(\frac{c_s}{3}\right)/2 & \cdots & l_s\left(\frac{c_s}{3}\right)/2 \\ \hline & l_1\left(\frac{1}{3}\right)/2 & \cdots & l_s\left(\frac{1}{3}\right)/2 \\ \hline & l_1\left(\frac{1}{2}\right) & \cdots & l_s\left(\frac{1}{2}\right) \end{array} \quad (21)$$

This point means that by letting $M = 0$, the trigonometric collocation methods yield a subclass of RKN methods for solving traditional second-order ODEs, which demonstrates the wider applications of the methods.

3. Properties of the methods

For the exact solution of (2) at $t = h$, let $\mathbf{y}(h) = (q^\top(h), p^\top(h))^\top$. Then the oscillatory Hamiltonian system (2) can be rewritten in the form

$$\mathbf{y}'(\xi h) = F(\mathbf{y}(\xi h)) := \begin{pmatrix} p(\xi h) \\ -Mq(\xi h) + f(q(\xi h)) \end{pmatrix}, \quad \mathbf{y}_0 = \begin{pmatrix} q_0 \\ p_0 \end{pmatrix} \quad (22)$$

for $0 \leq \xi \leq 1$. The Hamiltonian is

$$H(\mathbf{y}) = \frac{1}{2}p^\top p + \frac{1}{2}q^\top Mq + U(q). \quad (23)$$

On the other hand, denoting the numerical method (16) as

$$\boldsymbol{\omega}(h) = (\tilde{q}^\top(h), \tilde{p}^\top(h))^\top,$$

the numerical solution satisfies

$$\boldsymbol{\omega}'(\xi h) = \begin{pmatrix} \tilde{p}(\xi h) \\ -M\tilde{q}(\xi h) + \sum_{j=1}^s l_j(\xi)f(\tilde{q}(c_j h)) \end{pmatrix}, \quad \boldsymbol{\omega}_0 = \begin{pmatrix} q_0 \\ p_0 \end{pmatrix}. \quad (24)$$

The next lemma is useful for the following analysis.

Lemma 3.1. Let $g : [0, h] \rightarrow \mathbb{R}^d$ have j continuous derivatives in the interval $[0, h]$. Then

$$\int_0^1 P_j(\tau)g(\tau h)d\tau = \mathcal{O}(h^j),$$

where $P_j(\tau)$ is an orthogonal polynomial of degree j on the interval $[0, 1]$.

Proof. We assume that $g(\tau h)$ can be expanded in Taylor series at the origin for sake of simplicity. Then, for all $j \geq 0$, by considering that $P_j(\tau)$ is orthogonal to all polynomials of degree $n < j$:

$$\int_0^1 P_j(\tau)g(\tau h)d\tau = \sum_{n=1}^{\infty} \frac{g^{(n)}(0)}{n!} h^n \int_0^1 P_j(\tau)\tau^n d\tau = \mathcal{O}(h^j). \quad \square$$

3.1. The order of energy preservation

In this subsection we are concerned with the order of preservation of the Hamiltonian energy.

Theorem 3.2. Under the condition that c_l , $l = 1, 2, \dots, s$ are chosen as the node points of a s -point Gauss–Legendre's quadrature over the integral $[0, 1]$, we have

$$H(\boldsymbol{\omega}(h)) - H(\mathbf{y}_0) = \mathcal{O}(h^{2s+1}),$$

where the constant symbolized by \mathcal{O} is independent of h .

Proof. By virtue of [Lemma 3.1](#), (23) and (24), one has

$$\begin{aligned} H(\omega(h)) - H(\mathbf{y}_0) &= h \int_0^1 \nabla H(\omega(\xi h))^T \omega'(\xi h) d\xi \\ &= h \int_0^1 \left((M\tilde{q}(\xi h) - f(\tilde{q}(\xi h)))^T, \tilde{p}(\xi h)^T \right) \\ &\quad \cdot \begin{pmatrix} \tilde{p}(\xi h) \\ -M\tilde{q}(\xi h) + \sum_{j=1}^s l_j(\xi) f(\tilde{q}(c_j h)) \end{pmatrix} d\xi \\ &= h \int_0^1 \tilde{p}(\xi h)^T \left(\sum_{j=1}^s l_j(\xi) f(\tilde{q}(c_j h)) - f(\tilde{q}(\xi h)) \right) d\xi. \end{aligned}$$

Moreover, we have

$$f(\tilde{q}(\xi h)) - \sum_{j=1}^s l_j(\xi) f(\tilde{q}(c_j h)) = \frac{f^{(s+1)}(\tilde{q}(\xi h))|_{\xi=\zeta}}{(n+1)!} \prod_{i=1}^s (\xi h - c_i h).$$

Here $f^{(s+1)}(\tilde{q}(\xi h))$ denote the $(s+1)$ th-order derivative of $f(\tilde{q}(t))$ with respect to t . Then, we obtain

$$\begin{aligned} H(\omega(h)) - H(\mathbf{y}_0) &= -h \int_0^1 \tilde{p}(\xi h)^T \frac{f^{(s+1)}(\tilde{q}(\xi h))|_{\xi=\zeta}}{(n+1)!} \prod_{i=1}^s (\xi h - c_i h) d\xi \\ &= -h^{s+1} \int_0^1 \tilde{p}(\xi h)^T \frac{f^{(s+1)}(\tilde{q}(\xi h))|_{\xi=\zeta}}{(n+1)!} \prod_{i=1}^s (\xi - c_i) d\xi. \end{aligned}$$

Since c_l , $l = 1, 2, \dots, s$ are chosen as the node points of a s -point Gauss–Legendre's quadrature over the integral $[0, 1]$, $\prod_{i=1}^s (\xi - c_i)$ is an orthogonal polynomial of degree s on the interval $[0, 1]$. Therefore, it follows from [Lemma 3.1](#) that

$$H(\omega(h)) - H(\mathbf{y}_0) = -h^{s+1} \mathcal{O}(h^s) = \mathcal{O}(h^{2s+1}). \quad \square$$

3.2. The order of quadratic invariant

We next turn to the quadratic invariant $Q(\mathbf{y}) = q^T D p$ of (1). The quadratic form Q is a first integral of (1) if and only if $p^T D p + q^T D(f(q) - Mq) = 0$ for all $p, q \in \mathbb{R}^d$. This implies that D is a skew-symmetric matrix and that $q^T D(f(q) - Mq) = 0$ for any $q \in \mathbb{R}^d$. The following result states the degree of accuracy of the method (16).

Theorem 3.3. Under the condition in [Theorem 3.2](#), we have

$$Q(\omega(h)) - Q(\mathbf{y}_0) = \mathcal{O}(h^{2s+1}),$$

where the constant symbolized by \mathcal{O} is independent of h .

Proof. From $Q(\mathbf{y}) = q^T D p$ and $D^T = -D$, it follows that

$$\begin{aligned} Q(\omega(h)) - Q(\mathbf{y}_0) &= h \int_0^1 \nabla Q(\omega(\xi h))^T \omega'(\xi h) d\xi \\ &= h \int_0^1 \left(-\tilde{p}(\xi h)^T D, \tilde{q}(\xi h)^T D \right) \begin{pmatrix} \tilde{p}(\xi h) \\ -M\tilde{q}(\xi h) + \sum_{j=1}^s l_j(\xi) f(\tilde{q}(c_j h)) \end{pmatrix} d\xi. \end{aligned}$$

Since $q^T D(f(q) - Mq) = 0$ for any $q \in \mathbb{R}^d$, we obtain

$$\begin{aligned} Q(\omega(h)) - Q(\mathbf{y}_0) &= h \int_0^1 \tilde{q}(\xi h)^T D \left(-M\tilde{q}(\xi h) + \sum_{j=1}^s l_j(\xi) f(\tilde{q}(c_j h)) \right) d\xi \\ &= h \int_0^1 \tilde{q}(\xi h)^T D \frac{f^{(s+1)}(\tilde{q}(\xi h))|_{\xi=\zeta}}{(n+1)!} \prod_{i=1}^s (\xi h - c_i h) d\xi \\ &= h^{s+1} \int_0^1 \tilde{q}(\xi h)^T D \frac{f^{(s+1)}(\tilde{q}(\xi h))|_{\xi=\zeta}}{(n+1)!} \prod_{i=1}^s (\xi - c_i) d\xi \\ &= \mathcal{O}(h^{s+1}) \mathcal{O}(h^s) = \mathcal{O}(h^{2s+1}). \quad \square \end{aligned}$$

3.3. The order

To express the dependence of the solutions of $\mathbf{y}'(t) = F(\mathbf{y}(t))$ on the initial values, for any given $\tilde{t} \in [0, h]$, we denote by $\mathbf{y}(\cdot, \tilde{t}, \tilde{\mathbf{y}})$ the solution satisfying the initial condition $\mathbf{y}(\tilde{t}, \tilde{t}, \tilde{\mathbf{y}}) = \tilde{\mathbf{y}}$ and set

$$\Phi(s, \tilde{t}, \tilde{\mathbf{y}}) = \frac{\partial \mathbf{y}(s, \tilde{t}, \tilde{\mathbf{y}})}{\partial \tilde{\mathbf{y}}}. \quad (25)$$

Recalling the elementary theory of ODEs, we have the following standard result (see, e.g. [23])

$$\frac{\partial \mathbf{y}(s, \tilde{t}, \tilde{\mathbf{y}})}{\partial \tilde{t}} = -\Phi(s, \tilde{t}, \tilde{\mathbf{y}})F(\tilde{\mathbf{y}}). \quad (26)$$

The following theorem states the result on the order of the trigonometric collocation methods.

Theorem 3.4. Under the condition in Theorem 3.2, the trigonometric collocation method (16) satisfies

$$\mathbf{y}(h) - \omega(h) = \mathcal{O}(h^{2s+1}),$$

where the constant symbolized by \mathcal{O} is independent of h .

Proof. It follows from (25) and (26) that

$$\begin{aligned} \mathbf{y}(h) - \omega(h) &= \mathbf{y}(h, 0, \mathbf{y}_0) - \mathbf{y}(h, h, \omega(h)) = - \int_0^h \frac{d\mathbf{y}(h, \tau, \omega(\tau))}{d\tau} d\tau \\ &= - \int_0^h \left[\frac{\partial \mathbf{y}(h, \tau, \omega(\tau))}{\partial \tilde{t}} + \frac{\partial \mathbf{y}(h, \tau, \omega(\tau))}{\partial \tilde{\mathbf{y}}} \omega'(\tau) \right] d\tau \\ &= h \int_0^1 \Phi(h, \xi h, \omega(\xi h)) \left[F(\omega(\xi h)) - \omega'(\xi h) \right] d\xi \\ &= h \int_0^1 \Phi(h, \xi h, \omega(\xi h)) \left(f(\tilde{q}(\xi h)) - \sum_{j=1}^s l_j(\xi) f(\tilde{q}(c_j h)) \right) d\xi. \end{aligned}$$

We rewrite $\Phi(h, \xi h, \omega(\xi h))$ as a block matrix:

$$\Phi(h, \xi h, \omega(\xi h)) = \begin{pmatrix} \Phi_{11}(\xi h) & \Phi_{12}(\xi h) \\ \Phi_{21}(\xi h) & \Phi_{22}(\xi h) \end{pmatrix},$$

where Φ_{ij} ($i, j = 1, 2$) are all $d \times d$ matrices.

We then yield

$$\begin{aligned} \mathbf{y}(h) - \omega(h) &= h \begin{pmatrix} \int_0^1 \Phi_{12}(\xi h) \frac{f^{(s+1)}(\tilde{q}(\xi h))|_{\xi=\zeta}}{(n+1)!} \prod_{i=1}^s (\xi h - c_i h) d\xi \\ \int_0^1 \Phi_{22}(\xi h) \frac{f^{(s+1)}(\tilde{q}(\xi h))|_{\xi=\zeta}}{(n+1)!} \prod_{i=1}^s (\xi h - c_i h) d\xi \end{pmatrix} \\ &= h^{s+1} \begin{pmatrix} \int_0^1 \Phi_{12}(\xi h) \frac{f^{(s+1)}(\tilde{q}(\xi h))|_{\xi=\zeta}}{(n+1)!} \prod_{i=1}^s (\xi - c_i) d\xi \\ \int_0^1 \Phi_{22}(\xi h) \frac{f^{(s+1)}(\tilde{q}(\xi h))|_{\xi=\zeta}}{(n+1)!} \prod_{i=1}^s (\xi - c_i) d\xi \end{pmatrix} = h^{s+1} \mathcal{O}(h^s) = \mathcal{O}(h^{2s+1}). \quad \square \end{aligned}$$

3.4. Convergence analysis of the iteration

Theorem 3.5. Assume that M is symmetric and positive semi-definite and that f satisfies a Lipschitz condition in the variable q , i.e., there exists a constant L with the property that $\|f(q_1) - f(q_2)\| \leq L \|q_1 - q_2\|$. If

$$0 < h < \frac{1}{\sqrt{L \max_{i,j=1,\dots,s} \int_0^1 |l_j(c_i z)(1-z)| dz}}, \quad (27)$$

then the fixed-point iteration for the method (16) is convergent.

Proof. Following Definition 2.1, the first formula of (16) can be rewritten as

$$Q = \phi_0(c^2V)(e \otimes q_0) + hc\phi_1(c^2V)(e \otimes p_0) + h^2A(V)f(Q), \quad (28)$$

where $c = (c_1, \dots, c_s)^\top$, $e = (1, \dots, 1)^\top$, $Q = (\tilde{q}_1, \dots, \tilde{q}_s)^\top$, $f(Q) = (f(\tilde{q}_1)^\top, \dots, f(\tilde{q}_s)^\top)^\top$, $A(V) = (a_{ij}(V))_{s \times s}$, and $a_{ij}(V)$ and the block diagonal matrices are defined respectively by

$$\begin{aligned} a_{ij}(V) &:= \int_0^1 l_j(c_i z)(1-z)\phi_1((1-z)^2 c_i^2 V) dz, \\ \phi_0(c^2V) &:= \text{diag}(\phi_0(c_1^2V), \dots, \phi_0(c_s^2V))^\top, \\ c\phi_1(c^2V) &:= \text{diag}(c_1\phi_1(c_1^2V), \dots, c_s\phi_1(c_s^2V))^\top. \end{aligned}$$

By Proposition 2.1 in [24], we know that $\|\phi_1((1-z)^2 c_i^2 V)\| \leq 1$ and then we get

$$\|a_{ij}(V)\| \leq \int_0^1 |l_j(c_i z)(1-z)| dz.$$

Let

$$\varphi(x) = \phi_0(c^2V)(e \otimes q_0) + hc\phi_1(c^2V)(e \otimes p_0) + h^2A(V)f(x).$$

Then

$$\begin{aligned} \|\varphi(x) - \varphi(y)\| &= \|h^2A(V)f(x) - h^2A(V)f(y)\| \leq h^2L \|A(V)\| \|x - y\| \\ &\leq h^2L \max_{i,j=1,\dots,s} \int_0^1 |l_j(c_i z)(1-z)| dz \|x - y\|, \end{aligned}$$

which means that $\varphi(x)$ is a contraction from the assumption (27). The well-known Contraction Mapping Theorem then ensures the convergence of the fixed-point iteration. \square

Remark 5. It is noted that the convergence of the methods is independent of $\|M\|$. This point is of prime importance especially for highly oscillatory systems since we usually have $\|M\| \gg 1$, which will be shown by the numerical results of Problem 2 in Section 4.

3.5. Stability and phase properties

In this part we are concerned with the stability and phase properties. We consider the test equation:

$$q''(t) + \omega^2 q(t) = -\epsilon q(t) \quad \text{with } \omega^2 + \epsilon > 0, \quad (29)$$

where ω represents an estimation of the dominant frequency λ and $\epsilon = \lambda^2 - \omega^2$ is the error of that estimation. Applying (16) to (29) produces

$$\begin{pmatrix} \tilde{q} \\ h\tilde{p} \end{pmatrix} = S(V, z) \begin{pmatrix} q_0 \\ hp_0 \end{pmatrix},$$

where the stability matrix $S(V, z)$ is given by

$$S(V, z) = \begin{pmatrix} \phi_0(V) - z\bar{b}^\top(V)N^{-1}\phi_0(c^2V) & \phi_1(V) - z\bar{b}^\top(V)N^{-1}(c \cdot \phi_1(c^2V)) \\ -V\phi_1(V) - zb^\top(V)N^{-1}\phi_0(c^2V) & \phi_0(V) - zb^\top(V)N^{-1}(c \cdot \phi_1(c^2V)) \end{pmatrix}$$

with $N = I + zA(V)$, $\bar{b}(V) = (I_{1,1}, \dots, I_{1,s})^\top$, $b(V) = (I_{2,1}, \dots, I_{2,s})^\top$.

Accordingly, we have the following definitions of stability and dispersion order and dissipation order for our method (16).

Definition 3.6 ([25]). Let $\rho(S)$ be the spectral radius of S .

$$R_s = \{(V, z) \mid V > 0 \text{ and } \rho(S) < 1\}$$

is called the *stability region of the method* (16).

$$R_p = \{(V, z) \mid V > 0, \rho(S) = 1 \text{ and } \text{tr}(S)^2 < 4 \det(S)\}$$

is called the *periodicity region of the method* (16). The quantities

$$\phi(\zeta) = \zeta - \arccos\left(\frac{\text{tr}(S)}{2\sqrt{\det(S)}}\right), \quad d(\zeta) = 1 - \sqrt{\det(S)}$$

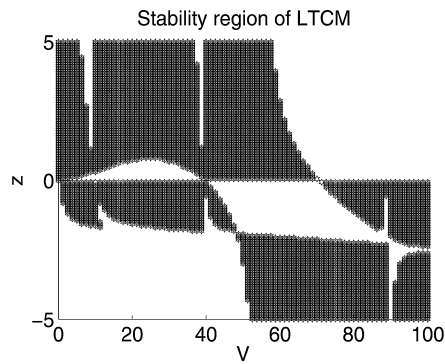


Fig. 1. Stability region (shaded area) of the method LTCM.

are called the dispersion error and the dissipation error of the method (16), respectively, where $\zeta = \sqrt{V + z}$. Then, a method is said to be dispersive of order r and dissipative of order s , if $\phi(\zeta) = O(\zeta^{r+1})$ and $d(\zeta) = O(\zeta^{s+1})$, respectively. If $\phi(\zeta) = 0$ and $d(\zeta) = 0$, then the method is said to be zero dispersive and zero dissipative, respectively.

4. Numerical experiments

As an example of the trigonometric collocation methods (16), we choose the node points of a two-point Gauss–Legendre’s quadrature over the integral $[0, 1]$ as follows

$$c_1 = \frac{3 - \sqrt{3}}{6}, \quad c_2 = \frac{3 + \sqrt{3}}{6}. \quad (30)$$

Then we choose $s = 2$ in (16) and denote the corresponding fourth-order method as LTCM.

The stability region of this method is shown in Fig. 1. We note that in order to obtain any information for the stability regions, we need to consider various values of V and z . Here we choose the subsets $V \in [0, 100]$, $z \in [-5, 5]$ and these regions shown in Fig. 1 only give an indication of the stability of this method.

The dissipative error and dispersion error are given respectively by

$$d(\zeta) = \frac{\epsilon^2}{24(\epsilon + \omega^2)^2} \zeta^4 + O(\zeta^5), \quad \phi(\zeta) = \frac{\epsilon^2}{6(\epsilon + \omega^2)^2} \zeta^3 + O(\zeta^4).$$

It is noted that when $M = 0$, the method LTCM reduces to a fourth-order RKN method with the Butcher tableau (21) and (30).

In order to show the efficiency and robustness of the fourth-order method, the other integrators we select for comparison are:

- TFCM: a fourth-order trigonometric Fourier collocation method in [18] with $c_1 = \frac{3-\sqrt{3}}{6}$, $c_2 = \frac{3+\sqrt{3}}{6}$, $b_1 = b_2 = 1/2$, $r = 2$;
- SRKM1: the symplectic Runge–Kutta method of order five in [26] based on Radau quadrature;
- EPCM1: the “extended Lobatto IIIA method of order four” in [27], which is an energy-preserving collocation method (the case $s = 2$ in [22]);
- EPRKM1: the energy-preserving Runge–Kutta method of order four (formula (19) in [21]).

Since all these methods are implicit, we use the classical waveform Picard algorithm. For each experiment, first we show the convergence rate of iterations for different error tolerances. Then for different methods, we set the error tolerance as 10^{-16} and set the maximum number of iteration as 5. We display the global errors and the energy errors if the problem is a Hamiltonian system. The numerical experiments have been carried out on a personal computer and the algorithm has been implemented by using the MATLAB-R2013a.

Problem 1. Consider the Hamiltonian equation which governs the motion of an artificial satellite (this problem has been considered in [28]) with the Hamiltonian

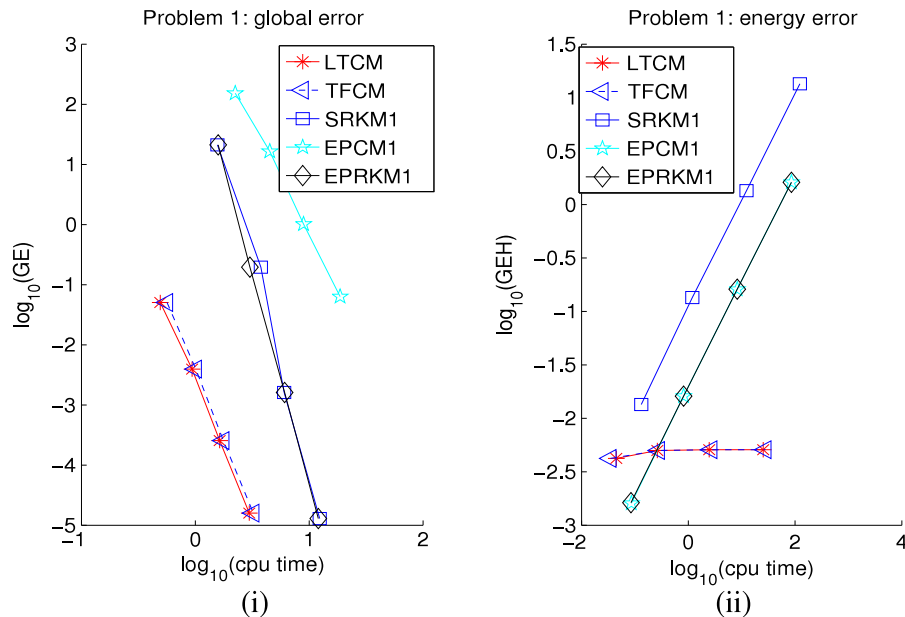
$$H(q, p) = \frac{1}{2} p^\top p + \frac{1}{2} \kappa q^\top q + \lambda \left(\frac{(q_1 q_3 + q_2 q_4)^2}{r^4} - \frac{1}{12 r^2} \right),$$

where $q = (q_1, q_2, q_3, q_4)^\top$ and $r = q^\top q$. The initial conditions are given on an elliptic equatorial orbit by

$$q_0 = \sqrt{\frac{r_0}{2}} \left(-1, -\frac{\sqrt{3}}{2}, -\frac{1}{2}, 0 \right)^\top, \quad p_0 = \frac{1}{2} \sqrt{K^2 \frac{1+e}{2}} \left(1, \frac{\sqrt{3}}{2}, \frac{1}{2}, 0 \right)^\top.$$

Table 1Results for **Problem 1**: The total CPU time (s) of iterations for different error tolerances (tol).

Methods	tol = 1.0e − 006	tol = 1.0e − 008	tol = 1.0e − 010	tol = 1.0e − 012
LTCM	6.8215	8.8964	8.8500	10.5551
TFCM	9.7892	9.7553	9.9806	13.0105
SRKM1	67.0230	64.1777	75.9390	86.8317
EPCM1	104.4341	112.9710	126.4438	145.6188
EPRKM1	56.2409	64.3123	75.2503	84.9962

**Fig. 2.** Results for **Problem 1**. (i): The logarithm of the global error (GE) over the integration interval against the logarithm of CPU time. (ii): The logarithm of the maximum global error of Hamiltonian energy (GEH) against the logarithm of CPU time.

Here $M = \frac{\kappa}{2}$ and κ is the total energy of the elliptic motion which is defined by $\kappa = \frac{K^2 - 2|p_0|^2}{r_0} - V_0$ with $V_0 = -\frac{\lambda}{12r_0^3}$. The parameters of this problem are chosen as $K^2 = 3.98601 \times 10^5$, $r_0 = 6.8 \times 10^3$, $e = 0.1$, $\lambda = \frac{3}{2}K^2J_2R^2$, $J_2 = 1.08625 \times 10^{-3}$, $R = 6.37122 \times 10^3$. First the problem is solved in the interval $[0, 10^4]$ with the stepsize $h = \frac{1}{10}$ to show the convergence rate of iterations. See **Table 1** for the CPU time of iterations for different error tolerances. Then this equation is integrated in $[0, 1000]$ with the stepsizes $1/2^i$, $i = 2, 3, 4, 5$. The global errors against CPU time are shown in **Fig. 2(i)**. We finally integrate this problem with a fixed stepsize $h = 1/20$ in the interval $[0, t_{\text{end}}]$ with $t_{\text{end}} = 10, 100, 10^3, 10^4$. The maximum global errors of Hamiltonian energy against CPU time are presented in **Fig. 2(ii)**.

Problem 2. Consider the Fermi–Pasta–Ulam Problem [5].

Fermi–Pasta–Ulam Problem is a Hamiltonian system with the Hamiltonian

$$H(y, x) = \frac{1}{2} \sum_{i=1}^{2m} y_i^2 + \frac{\omega^2}{2} \sum_{i=1}^m x_{m+i}^2 + \frac{1}{4} \left[(x_1 - x_{m+1})^4 + \sum_{i=1}^{m-1} (x_{i+1} - x_{m+i-1} - x_i - x_{m+i})^4 + (x_m + x_{2m})^4 \right],$$

where x_i is a scaled displacement of the i th stiff spring, x_{m+i} represents a scaled expansion (or compression) of the i th stiff spring, and y_i , y_{m+i} are their velocities (or momenta). This system can be rewritten as

$$x''(t) + Mx(t) = -\nabla U(x), \quad t \in [t_0, t_{\text{end}}],$$

where

$$M = \begin{pmatrix} \mathbf{0}_{m \times m} & \mathbf{0}_{m \times m} \\ \mathbf{0}_{m \times m} & \omega^2 I_{m \times m} \end{pmatrix},$$

$$U(x) = \frac{1}{4} \left[(x_1 - x_{m+1})^4 + \sum_{i=1}^{m-1} (x_{i+1} - x_{m+i-1} - x_i - x_{m+i})^4 + (x_m + x_{2m})^4 \right].$$

Table 2Results for [Problem 2](#): The total CPU time (s) of iterations for different error tolerances (tol).

Methods	tol = 1.0e – 006	tol = 1.0e – 008	tol = 1.0e – 010	tol = 1.0e – 012
LTCM ($\omega = 100$)	7.1570	9.7010	9.6435	12.2449
LTCM ($\omega = 200$)	7.5169	10.0160	9.2135	11.1672
TFCM ($\omega = 100$)	7.6434	10.3224	10.3341	12.7998
TFCM ($\omega = 200$)	7.8861	11.1322	10.0578	12.3621
SRKM1 ($\omega = 100$)	32.0491	39.4922	48.5822	57.0720
SRKM1 ($\omega = 200$)	58.2410	70.5585	86.1757	99.6403
EPCM1 ($\omega = 100$)	50.8899	70.5920	87.9782	102.9839
EPCM1 ($\omega = 200$)	121.2714	149.7104	189.4323	220.1096
EPRKM1 ($\omega = 100$)	31.0881	39.0050	47.6389	56.4456
EPRKM1 ($\omega = 200$)	55.2205	68.8459	82.5919	98.5277

Following [5], we choose

$$m = 3, \quad x_1(0) = 1, \quad y_1(0) = 1, \quad x_4(0) = \frac{1}{\omega}, \quad y_4(0) = 1$$

with zero for the remaining initial values.

First the problem is solved in the interval $[0, 1000]$ with the stepsize $h = \frac{1}{100}$ and $\omega = 100, 200$ to show the convergence rate of iterations. See [Table 2](#) for the total CPU time of iterations for different error tolerances. It can be observed that when ω increases, the convergence rate of LTCM and TFCM is almost unaffected. However, the convergence rate of the other methods varies greatly when ω becomes large.

Then we integrate the system in the interval $[0, 50]$ with $\omega = 50, 100, 150, 200$ and the stepsizes $h = 1/(20 \times 2^j)$, $j = 1, 2, 3, 4$. The global errors are shown in [Fig. 4](#). Finally we integrate this problem with a fixed stepsize $h = 1/100$ in the interval $[0, t_{\text{end}}]$ with $t_{\text{end}} = 1, 10, 100, 1000$. The maximum global errors of Hamiltonian energy are presented in [Fig. 4](#). Here it is noted that some results are too large, thus we do not plot the corresponding points in [Figs. 3–4](#). Similar situation occurs in the next two problems.

Problem 3. Consider the nonlinear Klein–Gordon equation [29]

$$\begin{aligned} \frac{\partial^2 u}{\partial t^2} - \frac{\partial^2 u}{\partial x^2} &= -u^3 - u, \quad 0 < x < L, \quad t > 0, \\ u(x, 0) &= A \left(1 + \cos \left(\frac{2\pi}{L} x \right) \right), \quad u_t(x, 0) = 0, \quad u(0, t) = u(L, t), \end{aligned}$$

where $L = 1.28$, $A = 0.9$. Carrying out a semi-discretization on the spatial variable by using second-order symmetric differences yields

$$\frac{d^2 U}{dt^2} + MU = F(U), \quad 0 < t \leq t_{\text{end}},$$

where $U(t) = (u_1(t), \dots, u_N(t))^T$ with $u_i(t) \approx u(x_i, t)$, $i = 1, \dots, N$,

$$M = \frac{1}{\Delta x^2} \begin{pmatrix} 2 & -1 & & & -1 \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ -1 & & & -1 & 2 \end{pmatrix}_{N \times N}$$

with $\Delta x = L/N$, $x_i = i\Delta x$, $F(U) = (-u_1^3 - u_1, \dots, -u_N^3 - u_N)^T$ and $N = 32$. The corresponding Hamiltonian of this system is

$$H(U', U) = \frac{1}{2} U'^T U' + \frac{1}{2} U^T M U + \frac{1}{2} u_1^2 + \frac{1}{4} u_1^4 + \dots + \frac{1}{2} u_N^2 + \frac{1}{4} u_N^4.$$

Here we choose $N = 32$. The problem is solved in the interval $[0, 500]$ with the stepsize $h = \frac{1}{100}$ to show the convergence rate of iterations. See [Table 3](#) for the total CPU time of iterations for different error tolerances. Then we solve this problem in $[0, 20]$ with stepsizes $h = 1/(3 \times 2^j)$, $j = 1, 2, 3, 4$. [Fig. 5\(i\)](#) shows the global errors. Finally this problem is integrated with a fixed stepsize $h = 0.002$ in the interval $[0, t_{\text{end}}]$ with $t_{\text{end}} = 1, 10, 100, 1000$. The maximum global errors of Hamiltonian energy are presented in [Fig. 5\(ii\)](#).

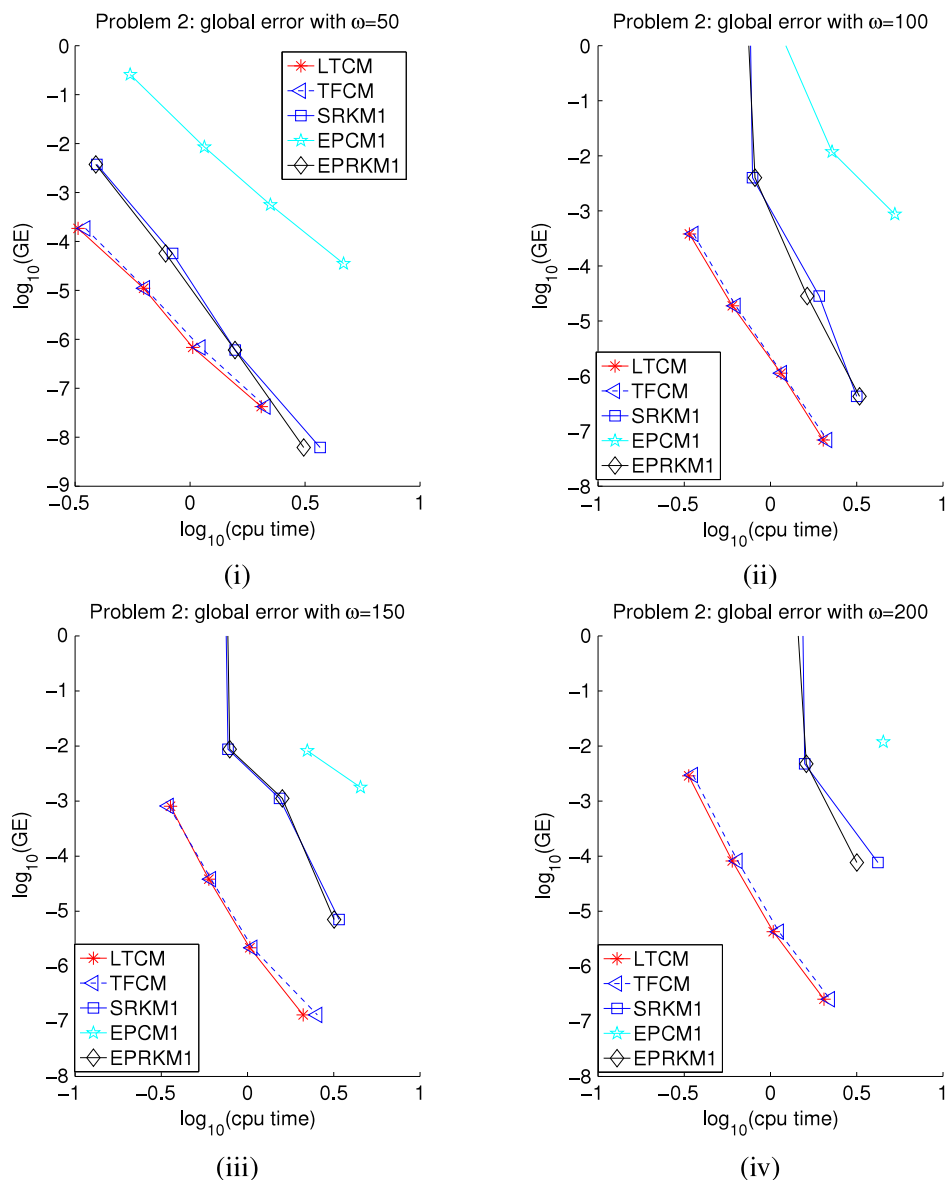


Fig. 3. Results for Problem 2. The logarithm of the global error (GE) over the integration interval against the logarithm of CPU time.

Table 3

Results for Problem 3: The total CPU time (s) of iterations for different error tolerances (tol).

Methods	$tol = 1.0e - 006$	$tol = 1.0e - 008$	$tol = 1.0e - 010$	$tol = 1.0e - 012$
LTCM	5.9325	7.9263	8.1816	10.0602
TFCM	6.5318	8.7008	8.8934	10.7489
SRKM1	24.1600	29.4173	34.5310	39.5161
EPCM1	37.2757	46.4011	53.1403	66.2339
EPRKM1	22.6571	27.8341	33.5435	39.4533

Problem 4. Consider the wave equation

$$\frac{\partial^2 u}{\partial t^2} - a(x) \frac{\partial^2 u}{\partial x^2} + 92u = f(t, x, u), \quad 0 < x < 1, \quad t > 0,$$

$$u(0, t) = 0, \quad u(1, t) = 0, \quad u(x, 0) = a(x), \quad u_t(x, 0) = 0$$

with $a(x) = 4x(1-x)$, $f(t, x, u) = u^5 - a^2(x)u^3 + \frac{a^5(x)}{4} \sin^2(20t) \cos(10t)$. The exact solution is $u(x, t) = a(x) \cos(10t)$.

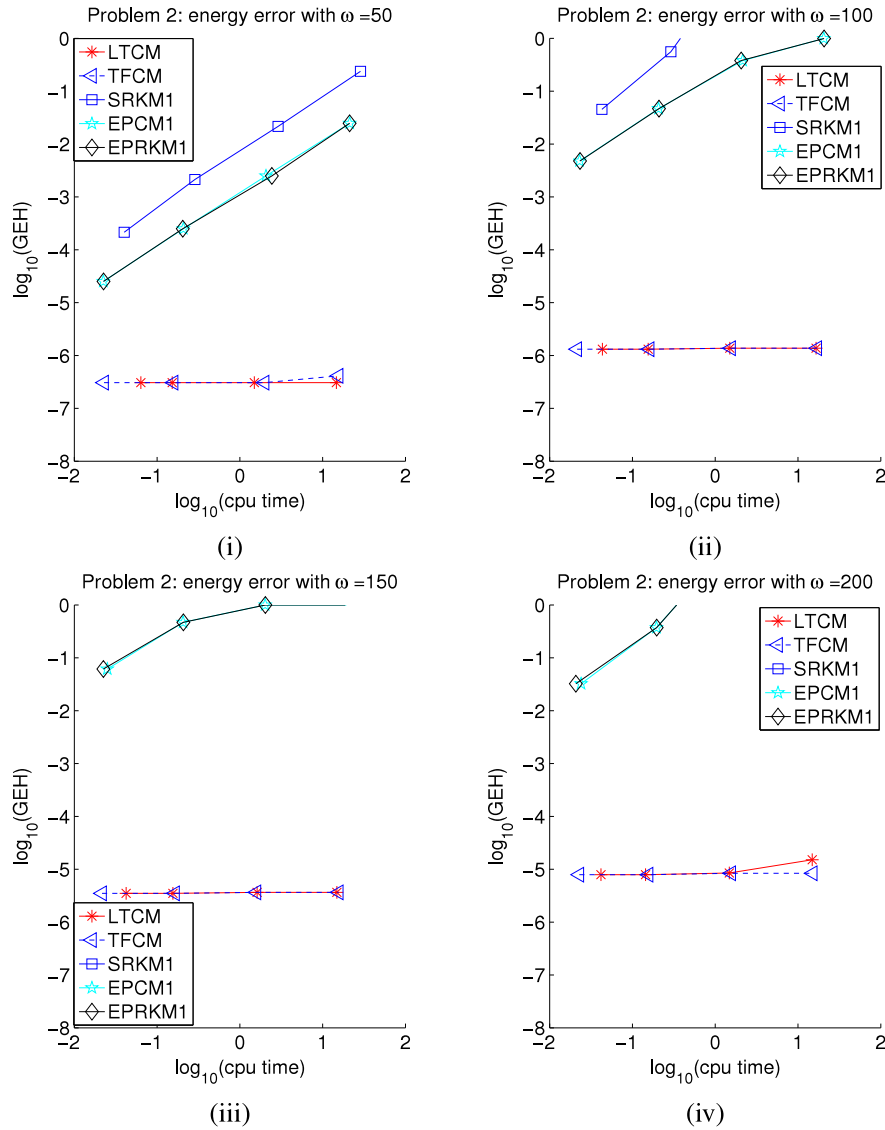


Fig. 4. Results for Problem 2. The logarithm of the maximum global error of Hamiltonian energy (GEH) against the logarithm of CPU time.

Using semi-discretization on the spatial variable with second-order symmetric differences, we obtain

$$\frac{d^2 U}{dt^2} + MU = F(t, U), \quad U(0) = (a(x_1), \dots, a(x_{N-1}))^T, \quad U'(0) = \mathbf{0}, \quad 0 < t \leq t_{\text{end}},$$

where $U(t) = (u_1(t), \dots, u_{N-1}(t))^T$ with $u_i(t) \approx u(x_i, t)$, $x_i = i\Delta x$, $\Delta x = 1/N$, $i = 1, \dots, N-1$,

$$M = 92I_{N-1} + \frac{1}{\Delta x^2} \begin{pmatrix} 2a(x_1) & -a(x_1) & & & \\ -a(x_2) & 2a(x_2) & -a(x_2) & & \\ & \ddots & \ddots & \ddots & \\ & & -a(x_{N-2}) & 2a(x_{N-2}) & -a(x_{N-2}) \\ & & & -a(x_{N-1}) & 2a(x_{N-1}) \end{pmatrix},$$

and

$$F(t, U) = (f(t, x_1, u_1), \dots, f(t, x_{N-1}, u_{N-1}))^T.$$

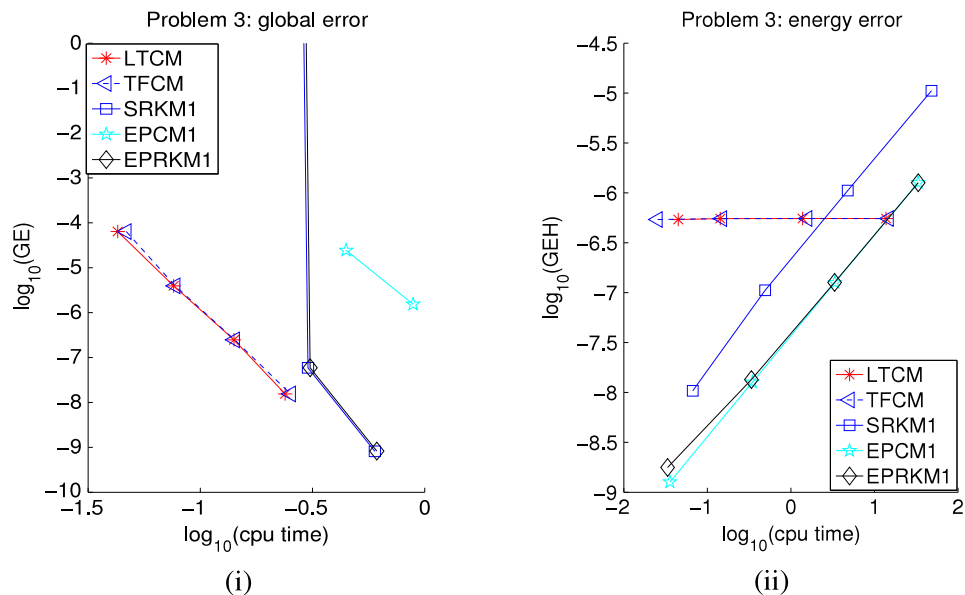


Fig. 5. Results for Problem 3. (i): The logarithm of the global error (GE) over the integration interval against the logarithm of CPU time. (ii): The logarithm of the maximum global error of Hamiltonian energy (GEH) against the logarithm of CPU time.

Table 4

Results for Problem 4: The total CPU time (s) of iterations for different error tolerances (tol).

Methods	$tol = 1.0e - 006$	$tol = 1.0e - 008$	$tol = 1.0e - 010$	$tol = 1.0e - 012$
LTCM	1.8980	1.8737	2.1212	2.3196
TFCM	1.9213	1.9345	2.2227	2.3736
SRKM1	13.8634	16.6963	19.0854	22.6142
EPCM1	23.5110	28.1288	32.2263	36.8443
EPRKM1	13.5526	17.2289	18.8744	23.0066

The problem is solved in the interval $[0, 100]$ with the stepsize $h = \frac{1}{40}$ to show the convergence rate of iterations. See Table 4 for the total CPU time of iterations for different error tolerances. Then system is integrated in the interval $[0, 100]$ with $N = 40$ and $h = 1/2^j$, $j = 5, 6, 7, 8$. The global errors are shown in Fig. 6.

Remark 6. It follows from the numerical results that our method LTCM is very promising as compared with the classical methods SRKM1, EPCM1 and EPRKM1. Although LTCM has a similar performance as TFCM in preserving the solution and the energy, it can be observed from Figs. 2(i), 3 and 5(i) that LTCM performs a little better than TFCM in presenting the solution. Moreover, it follows from Tables 1–4 that LTCM has a better convergence performance of iterations than TFCM, which means that LTCM can have a lower computational cost when the same error tolerance is required in the iteration procedure.

Remark 7. From Figs. 2(ii), 4 and 5(ii), it can be observed that the energy-preserving Runge–Kutta method EPRKM1 cannot preserve the Hamiltonian energy, and the errors seem to grow with the CPU time when the stepsize is reduced. The reason for this phenomenon may be that EPRKM1 does not take advantage of the special structure brought by the linear term Mq of the oscillatory differential equations (1) and its convergence depends on $\|M\|$. The method LTCM developed in this paper makes good use of the matrix M appearing in the oscillatory systems (1) and its convergent condition is independent of $\|M\|$, which makes LTCM perform well in preserving Hamiltonian energy although it is not an energy-preserving method.

5. Conclusions and discussions

In this paper we have investigated a kind of trigonometric collocation methods based on Lagrange basis polynomials, the variation-of-constants formula and the idea of collocation methods for solving multi-frequency oscillatory second-order differential equations (1) efficiently. It has been shown that the convergent condition of these trigonometric collocation methods is independent of $\|M\|$, which is very important and crucial for solving highly oscillatory systems. The numerical experiments with some model problems show that our method derived in this paper has remarkable efficiency in comparison with some existing methods in the literature.

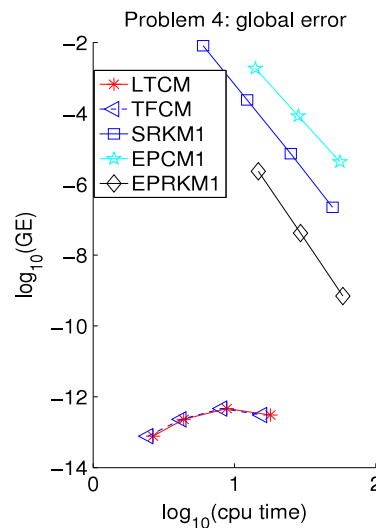


Fig. 6. Results for Problem 4: The logarithm of the global error (GE) over the integration interval against the logarithm of CPU time.

Acknowledgements

The authors are sincerely thankful to two anonymous reviewers for their valuable suggestions, which help improve the presentation of the manuscript significantly.

This paper was supported by National Natural Science Foundation of China under Grant 11401333, 11671200, 11671227, by Natural Science Foundation of Shandong Province under Grant ZR2014AQ003, and by China Postdoctoral Science Foundation under Grant 2015M580578.

References

- [1] D. Cohen, E. Hairer, C. Lubich, Numerical energy conservation for multi-frequency oscillatory differential equations, *BIT* 45 (2005) 287–305.
- [2] D. Cohen, T. Jahnke, K. Lorenz, C. Lubich, Numerical integrators for highly oscillatory Hamiltonian systems: a review, in: A. Mielke (Ed.), *Analysis, Modeling and Simulation of Multiscale Problems*, Springer, Berlin, 2006, pp. 553–576.
- [3] B. García-Archilla, J.M. Sanz-Serna, R.D. Skeel, Long-time-step methods for oscillatory differential equations, *SIAM J. Sci. Comput.* 20 (1999) 930–963.
- [4] E. Hairer, C. Lubich, Long-time energy conservation of numerical methods for oscillatory differential equations, *SIAM J. Numer. Anal.* 38 (2000) 414–441.
- [5] E. Hairer, C. Lubich, G. Wanner, *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*, second ed., Springer-Verlag, Berlin, Heidelberg, 2006.
- [6] X. Wu, B. Wang, W. Shi, Efficient energy-preserving integrators for oscillatory Hamiltonian systems, *J. Comput. Phys.* 235 (2013) 587–605.
- [7] X. Wu, X. You, B. Wang, *Structure-Preserving Algorithms for Oscillatory Differential Equations*, Springer-Verlag, Berlin, Heidelberg, 2013.
- [8] M. Hochbruck, C. Lubich, A Gautschi-type method for oscillatory second-order differential equations, *Numer. Math.* 83 (1999) 403–426.
- [9] B. Wang, G. Li, Bounds on asymptotic-numerical solvers for ordinary differential equations with extrinsic oscillation, *Appl. Math. Model.* 39 (2015) 2528–2538.
- [10] B. Wang, K. Liu, X. Wu, A Filon-type asymptotic approach to solving highly oscillatory second-order initial value problems, *J. Comput. Phys.* 243 (2013) 210–223.
- [11] B. Wang, X. Wu, A new high precision energy-preserving integrator for system of oscillatory second-order differential equations, *Phys. Lett. A* 376 (2012) 1185–1190.
- [12] B. Wang, X. Wu, H. Zhao, Novel improved multidimensional Störmer–Verlet formulas with applications to four aspects in scientific computation, *Math. Comput. Modelling* 57 (2013) 857–872.
- [13] B. Wang, H. Yang, F. Meng, Sixth-order symplectic and symmetric explicit ERKN schemes for solving multi-frequency oscillatory nonlinear Hamiltonian equations, *Calcolo* (2016). <http://dx.doi.org/10.1007/s10092-016-0179-y>.
- [14] X. Wu, B. Wang, J. Xia, Explicit symplectic multidimensional exponential fitting modified Runge–Kutta–Nyström methods, *BIT* 52 (2012) 773–795.
- [15] M. Hochbruck, A. Ostermann, Explicit exponential Runge–Kutta methods for semilinear parabolic problems, *SIAM J. Numer. Anal.* 43 (2005) 1069–1090.
- [16] M. Hochbruck, A. Ostermann, J. Schweitzer, Exponential Rosenbrock-type methods, *SIAM J. Numer. Anal.* 47 (2009) 786–803.
- [17] D. Cohen, Conservation properties of numerical integrators for highly oscillatory Hamiltonian systems, *IMA J. Numer. Anal.* 26 (2006) 34–59.
- [18] B. Wang, A. Iserles, X. Wu, Arbitrary-order trigonometric Fourier collocation methods for multi-frequency oscillatory systems, *Found. Comput. Math.* 16 (2016) 151–181.
- [19] X. Wu, X. You, W. Shi, B. Wang, ERKN integrators for systems of oscillatory second-order differential equations, *Comput. Phys. Comm.* 181 (2010) 1873–1887.
- [20] L. Brugnano, F. Iavernaro, *Line Integral Methods for Conservative Problems*, CRC Press, Boca Raton (FL), 2016.
- [21] L. Brugnano, F. Iavernaro, D. Trigiante, A simple framework for the derivation and analysis of effective one-step methods for ODEs, *Appl. Math. Comput.* 218 (2012) 8475–8485.
- [22] E. Hairer, Energy-preserving variant of collocation methods, *JNAIAM J. Numer. Anal. Ind. Appl. Math.* 5 (2010) 73–84.
- [23] J.K. Hale, *Ordinary Differential Equations*, Robert E. Krieger Publishing company, Huntington, New York, 1980.
- [24] J. Li, X. Wu, Adapted Falkner-type methods solving oscillatory second-order differential equations, *Numer. Algorithms* 62 (2013) 355–381.
- [25] X. Wu, A note on stability of multidimensional adapted Runge–Kutta–Nyström methods for oscillatory systems, *Appl. Math. Model.* 36 (2012) 6331–6337.
- [26] Sun Geng, Construction of high order symplectic Runge–Kutta methods, *J. Comput. Math.* 11 (1993) 250–260.
- [27] F. Iavernaro, D. Trigiante, High-order symmetric schemes for the energy conservation of polynomial Hamiltonian problems, *JNAIAM J. Numer. Anal. Ind. Appl. Math.* 4 (1) (2009).
- [28] E.L. Stiefel, G. Scheifele, *Linear and Regular Celestial Mechanics*, Springer-Verlag, New York, 1971.
- [29] S. Jiménez, L. Vázquez, Analysis of four numerical schemes for a nonlinear Klein–Gordon equation, *Appl. Math. Comput.* 35 (1990) 61–93.