

Computational Series and Multistep Methods to Integrate Forced and Damped Stiff Oscillators

J.A. Reyes* and F. García-Alonso

Department of Applied Mathematics, University of Alicante, Alicante, Spain

Abstract: In a first part, this article presents the adaptation of Scheifele functions to forced and damped oscillators, designing a series method based on these, which integrates the non perturbed problem with no truncation error. This method is highly accurate, however it is difficult to adapt to each specific problem. In order to overcome this difficulty, in a second part, we describe the transformation of the series method to a multistep scheme.

Explicit and implicit methods are formulated and combine to create a predictor-corrector method, which precisely integrates the homogenous problem.

The computational algorithm is developed and the results obtained are contrasted by the series method and by the multi-step algorithm with other known integrators.

Keywords: Numerical solutions of oscillators, highly accurate solutions, highly oscillatory problems, stiff problems, multistep algorithms, computational algorithms.

INTRODUCTION

The stretching or compression of most solid materials is modeled by means of an oscillator provided that the displacement is not too high, that is, any system which makes small oscillations around a stable point of equilibrium can be treated in first approximation as if it were an oscillator.

The theory of damped and forced harmonic movements is fundamental to many areas of physics and engineering.

Harmonic oscillators are also found in Celestial Mechanics models, as in the classic two-body problem and in the satellite problem. In orbit calculation methods, newtonian equations of movement are reduced to harmonic oscillators, by means of transformation using the Kunstaanheimo-Stiefel [1] and Burdet-Ferrándiz [2] methods. It is therefore relevant to have efficient numeric algorithms available that are capable of providing very precise approximations.

To this effect Stiefel and Bettis [3] and Bettis [4, 5], published the first fixed step and non-linear multi-step numerical methods, for the resolution of this type of oscillator. Stiefel and Scheifele [1] and Scheifele [6], developed methods, based on a generalisation of the Taylor series. Subsequently other methods have been developed for the integration of oscillators applying different ideas and with different motivations, such as Variational Iteration Method, VIM [7, 8], Parameter-Expansion Method, PEM proposed by He [9], is a very effective way for handling nonlinear problems. In [9], the following categories of

asymptotic methods are emphasized: variation approaches and homotopy perturbation method.

Deuflhard [10], Neta and Ford [11], Denk [12], Martín and Ferrándiz [13], introduced a multi-step method based on the Scheifele functions. Vigo and Ferrándiz [14-16], explained a general procedure for adapting multi-step algorithms to the integration of oscillatory problems. The underlying ideas in these methods were applied to obtain Backward Differentiation Formulas, BDF's [17]. Among the methods which use exponential interpolation techniques, those obtained by De Meyer [18], Ixaru [19] and Vanden Berghe [20] are particularly worthy of mention. In [21], a numerical Variable- Step Variable-Order, VSVO method was devised based on the Scheifele functions. Similarly, methods were designed which use Ferrándiz functions [22], instead of the Scheifele functions. These methods also integrate the homogenous problem without truncation errors. These were also used to construct multi-step algorithms [23].

This article presents the adaptation of the Scheifele functions to forced and damped oscillators, studying the properties of the adapted functions and using them to design a series method, which integrates the non perturbed problem without truncation error.

This series method is extremely accurate, however, it is difficult to adapt to each particular problem.

In order to overcome this difficulty, the conversion of the series method to a multi-step scheme is described, similar to the SMF, Scheifele-Martin-Ferrándiz methods [13, 24] for forced oscillators but with the advantage that it is a variable step and order scheme that can be applied to both forced and damped oscillators.

The multi-step scheme is obtained by approximating the derivatives of the perturbation function which appear in the series method by means of divided differences.

*Address correspondence to this author at the Department of Applied Mathematics, University of Alicante, Alicante, Spain; Tel: 965 90 34 00, Ext. 2050; Fax: 965 90 97 07; E-mail: jose.reyes@ua.es

In order to compute the coefficients of the multi-step algorithm, a recurrent calculation procedure is established, based on the relation between the divided differences and elemental and complete symmetrical functions.

Both explicit and implicit methods are formulated and based on these, predictor- corrector methods are constructed which are easy to implement in a computer, and which precisely integrate the homogenous problem exactly with just the first two functions.

The good performance of both the series and multi-step methods is demonstrated by applying them to stiff and highly oscillatory problems, contrasting the numerical results obtained with other known integrators.

FORMULATION AND BASIC IDEAS

Let $x(t)$ be the solution of the perturbed and damped oscillator, in equation

$$x''(t) + \gamma x'(t) + \alpha x(t) = \varepsilon \cdot f(x(t), x'(t), t), \quad x(0) = x_0, \quad (1)$$

$$x'(0) = x'_0,$$

it assumes that the function of perturbation is analytic, i.e., the function

$g(t) = f(x(t; x_0, x_0, t_0), x'(t; x_0, x_0, t_0), t)$ admits in $[0, T]$, a development in power series form

$$g(t) = f(x(t; x_0, x_0, t_0), x'(t; x_0, x_0, t_0), t) = \sum_{n=0}^{\infty} c_n \frac{t^n}{n!} \text{ with } c_n = g^{(n)}(0), \quad (2)$$

bringing the previous Initial Value Problem, IVP can be written

$$x''(t) + \gamma x'(t) + \alpha x(t) = \varepsilon \cdot \sum_{n=0}^{\infty} c_n \frac{t^n}{n!}, \quad (3)$$

$$x(0) = x_0, \quad x'(0) = x'_0$$

As is usual, the solution of the IVP (1), can divided into two parts, one corresponding to the solution associated with the homogeneous IVP with the given initial conditions and the other part is a solution the non homogeneous IVP in which the solution and its derivative are cancelled in $t = 0$.

The latter can be calculated by applying the principle of superposition of particular solutions to IVP

$$x_n''(t) + \gamma x_n'(t) + \alpha x_n(t) = \frac{t^n}{n!}, \quad (4)$$

$$x_n(0) = x_n'(0) = 0, \quad n \geq 0$$

combining the solutions in linear form with the coefficients of $\varepsilon \cdot c_n$.

It defines the Scheifele G -functions [6], adapted to this kind of forced and damped oscillators, as those functions G_n that verify

$$G_n(t) = x_{n-2}(t), \quad n \geq 2, \quad (5)$$

where $x_n(t)$ are the solutions of problems

$$x_n''(t) + \gamma x_n'(t) + \alpha x_n(t) = \frac{t^n}{n!}, \quad x_n(0) = x_n'(0) = 0, \quad n \geq 0, \quad (6)$$

so that the functions G_n thus defined satisfy the following recurrence relation:

$$G_n''(t) + \gamma G_n'(t) + \alpha G_n(t) = \frac{t^{n-2}}{(n-2)!}, \quad (7)$$

$$G_n(0) = G_n'(0) = 0, \quad n \geq 2$$

The Scheifele G -functions, adapted to forced and damped oscillators, have the following property

$$G_n'(t) = G_{n-1}(t) \text{ with } n \geq 3. \quad (8)$$

The expression (8) allows for all G -functions from $G_2(t)$. The first two G -functions, $G_0(t)$ and $G_1(t)$, are defined as solutions of the homogeneous problems

$$x''(t) + \gamma x'(t) + \alpha x(t) = 0, \quad x(0) = 1, \quad x'(0) = -\gamma, \quad (9)$$

$$x''(t) + \gamma x'(t) + \alpha x(t) = 0, \quad x(0) = 0, \quad x'(0) = 1, \quad (10)$$

respectively.

That is to say, the functions $G_0(t)$ and $G_1(t)$ verify that

$$G_0''(t) + \gamma G_0'(t) + \alpha G_0(t) = 0, \quad G_0(0) = 1, \quad G_0'(0) = -\gamma, \quad (11)$$

$$G_1''(t) + \gamma G_1'(t) + \alpha G_1(t) = 0, \quad G_1(0) = 0, \quad G_1'(0) = 1. \quad (12)$$

The functions $G_0(t)$ and $G_1(t)$ verify the following relationship:

$$G_0'(t) = -\alpha G_1(t) - \gamma G_0(t). \quad (13)$$

Functions $G_0(t)$ and $G_1(t)$ also verify the property expressed in (8), i.e.

$$G_2'(t) = G_1(t), \quad (14)$$

$$G_1'(t) = G_0(t). \quad (15)$$

Therefore, combining (8), (14) and (15), it is possible to ensure that

$$G_n'(t) = G_{n-1}(t), \quad n \geq 1. \quad (16)$$

By (7) and (16) we can be affirm that

$$G_n''(t) + \gamma G_n'(t) + \alpha G_n(t) = \frac{t^{n-2}}{(n-2)!}, \text{ for } n \geq 2 \quad (17)$$

$$G_{n-2}(t) + \gamma G_{n-1}(t) + \alpha G_n(t) = \frac{t^{n-2}}{(n-2)!}, \text{ for } n \geq 2 \quad (18)$$

from which we obtain the following recurrence relation

$$G_n(t) + \gamma G_{n+1}(t) + \alpha G_{n+2}(t) = \frac{t^n}{n!}, \quad n \geq 0. \quad (19)$$

Another relevant property is the fact that the G -functions can be expressed by absolutely convergent series expansion for all values of t .

Since $G_0(t)$ is solution of a linear differential equation with constant coefficients, in which the function of perturbation is analytical, then

$$G_0(t) = \sum_{k=0}^{\infty} \xi_k \frac{t^k}{k!}, \tag{20}$$

where

$$\xi_0 = G_0(0) = 1, \tag{21}$$

$$\xi_1 = G_0'(0) = -\gamma. \tag{22}$$

From (16), it follows that

$$G_n(t) = \sum_{k=0}^{\infty} \xi_k \frac{t^{n+k}}{(n+k)!}, \quad n \geq 0. \tag{23}$$

The calculation of the coefficients ξ_k to $k \geq 2$, is done by substituting (20) in (11), i.e.

$$\sum_{k=0}^{\infty} (\xi_{k+2} + \gamma \xi_{k+1} + \alpha \xi_k) \frac{t^k}{k!} = 0, \tag{24}$$

from which we deduce the following recurrence relation [6], which allows the calculation

$$\xi_0 = 1 \tag{25}$$

$$\xi_1 = -\gamma \tag{26}$$

$$\xi_k = -\gamma \xi_{k-1} - \alpha \xi_{k-2}, \quad k \geq 2. \tag{27}$$

The solution of the IVP (3), and therefore the IVP (1), is

$$x(t) = x_0 G_0(t) + (x_0' + \gamma x_0) G_1(t) + \varepsilon \sum_{n=0}^{\infty} c_n G_{n+2}(t). \tag{28}$$

Indeed:

$$\begin{aligned} x'' + \gamma x' + \alpha x &= x_0 G_0''(t) + (x_0' + \gamma x_0) G_1''(t) + \varepsilon \sum_{n=0}^{\infty} c_n G_{n+2}''(t) \\ &+ \gamma \left(x_0 G_0'(t) + (x_0' + \gamma x_0) G_1'(t) + \varepsilon \sum_{n=0}^{\infty} c_n G_{n+2}'(t) \right) \\ &+ \alpha \left(x_0 G_0(t) + (x_0' + \gamma x_0) G_1(t) + \varepsilon \sum_{n=0}^{\infty} c_n G_{n+2}(t) \right) \\ &= x_0 (G_0''(t) + \gamma G_0'(t) + \alpha G_0(t)) + (x_0' + \gamma x_0) (G_1''(t) + \gamma G_1'(t) + \alpha G_1(t)) \\ &+ \varepsilon \sum_{n=0}^{\infty} c_n (G_{n+2}''(t) + \gamma G_{n+2}'(t) + \alpha G_{n+2}(t)) \\ &= x_0 \cdot 0 + (x_0' + \gamma x_0) \cdot 0 + \varepsilon \sum_{n=0}^{\infty} c_n (G_{n+2}''(t) + \gamma G_{n+2}'(t) + \alpha G_{n+2}(t)) \\ &= \varepsilon \sum_{n=0}^{\infty} c_n \frac{t^n}{n!}. \end{aligned}$$

As for the initial conditions

$$\begin{aligned} x(0) &= x_0 G_0(0) + (x_0' + \gamma x_0) G_1(0) + \varepsilon \sum_{n=0}^{\infty} c_n G_{n+2}(0) \\ &= x_0 + (x_0' + \gamma x_0) \cdot 0 = x_0. \end{aligned}$$

$$\begin{aligned} x'(0) &= x_0 G_0'(0) + (x_0' + \gamma x_0) G_0(0) + \varepsilon \sum_{n=0}^{\infty} c_n G_{n+1}(0) \\ &= x_0 (-\gamma) + (x_0' + \gamma x_0) = x_0'. \end{aligned}$$

From (16) and (28) is obtained:

$$x'(t) = x_0 G_0'(t) + (x_0' + \gamma x_0) G_0(t) + \varepsilon \sum_{n=0}^{\infty} c_n G_{n+1}(t) \tag{29}$$

This result allows to define a numerical and computational series method [25], to integrate forced and damped oscillators.

Assume that the solution of IVP (3), is

$$x(t) = \sum_{k=0}^{\infty} a_k \frac{t^k}{k!}, \tag{30}$$

then

$$x(0) = x_0 = a_0, \tag{31}$$

$$x'(0) = x_0' = a_1. \tag{32}$$

Substituting (30) in (3), we obtain the recurrence relation

$$a_0 = x_0, \tag{33}$$

$$a_1 = x_0', \tag{34}$$

$$a_{k+2} + \gamma a_{k+1} + \alpha a_k = \varepsilon \cdot c_k, \quad k \geq 0. \tag{35}$$

Defining the new coefficients b_k

$$b_0 = a_0 = x_0, \tag{36}$$

$$b_1 = a_1 + \gamma a_0 = x_0' + \gamma x_0, \tag{37}$$

$$b_{k+2} = a_{k+2} + \gamma a_{k+1} + \alpha a_k = \varepsilon \cdot c_k, \quad k \geq 0, \tag{38}$$

is possible to express the solution of (3), through a series of G -functions

$$x(t) = \sum_{k=0}^{\infty} b_k G_k(t), \tag{39}$$

we can be considered as an extension of the Taylor series for the solutions of (1) and provides greater accuracy.

For the construction of a method of series G -functions, of order m , it performs a truncation of $m + 1$ summands in (39).

Fixed a step h , the approaches to the solution and its derivative at point $t = h$, i.e., the first step, are given respectively by the expressions

$$x_1 = b_0 G_0(h) + b_1 G_1(h) + \varepsilon \sum_{n=0}^{m-2} b_{n+2} G_{n+2}(h), \tag{40}$$

$$x_1' = b_0 G_0'(h) + b_1 G_0(h) + \varepsilon \sum_{n=0}^{m-2} b_{n+2} G_{n+1}(h). \tag{41}$$

Let us suppose that we have already calculated an approximation of the solution and its derivative in point $t = nh$, x_n and x_n' , verifying:

$$x''(t) + \gamma x'(t) + \alpha x(t) = \varepsilon \cdot f(x(t), x'(t), t), \tag{42}$$

$$x(nh) = x_n, \quad x'(nh) = x'_n,$$

to calculate an approximation of the solution and its derivative in point $(n+1)h$, it is convenient to make a change of time origin $t = \tau + nh$, yielding:

$$x''(\tau) + \gamma x'(\tau) + \alpha x(\tau) = \varepsilon \cdot f(x(\tau), x'(\tau), \tau + nh), \tag{43}$$

$$x(0) = x_n, \quad x'(0) = x'_n,$$

bringing us to the initial condition.

We calculate the coefficients by recurrence.

$$f(x(\tau), x'(\tau), \tau + nh) = \sum_{k=0}^{\infty} \frac{\tau^k}{k!} c_k$$

with $\tau = t - nh$, $\tag{44}$

$$c_k = \frac{d^k g(0)}{d\tau^k} = \frac{d^k g(nh)}{dt^k}$$

and the approximation of the solution and its derivative in point $(n+1)h$ resulting in the formulas:

$$x_{n+1} = G_0(h)x_n + G_1(h)(x'_n + \gamma x_n) + \sum_{k=0}^{m-2} b_k G_{k+2}(h), \tag{45}$$

$$x'_{n+1} = G'_0(h)x_n + G_0(h)(x'_n + \gamma x_n) + \sum_{k=0}^{m-2} b_k G_{k+1}(h), \tag{46}$$

respectively, which constitute the numerical integration method for perturbed and damped harmonic oscillators based on adapted Scheifele's G -functions series.

ZERO-STABILITY AND CONVERGENCE

The solution $x(t)$ to the given IVP (1) admits a Taylor expansion about any point $t_m \in [0, T]$ of the form:

$$x(t) = \sum_{n=0}^{\infty} a_n \frac{(t - t_m)^n}{n!} \tag{47}$$

Let us define a function

$$g_m(t) = f \left(\begin{matrix} t, x(t; x(t_m), x'(t_m), t_m), \\ x'(t; x(t_m), x'(t_m), t_m) \end{matrix} \right) = \sum_{n=0}^{\infty} c_n \frac{(t - t_m)^n}{n!}, \tag{48}$$

where a notation similar to that in (2) has been used, upon substitution of (48) into (1), we find the recursion relation

$$\begin{aligned} a_0 &= x(t_m), \\ a_1 &= x'(t_m), \\ a_n &= -\gamma a_{n-1} - \alpha a_{n-2} + \varepsilon c_{n-2}, \text{ with } n \geq 2. \end{aligned} \tag{49}$$

It is important to remark that coefficients c_n depend only on coefficients a_m with indices $m \leq n$ at most, so that the relations (49) can be solved. Nevertheless, the relations among a's and c's can be more or less involved depending on the form of function $g_m(t)$.

By defining auxiliary constants b_n through

$$\begin{aligned} b_0 &= a_0, \\ b_1 &= a_1 + \gamma a_0, \\ b_n &= a_n + \gamma a_{n-1} + \alpha a_{n-2} = \varepsilon c_{n-2}, \text{ with } n \geq 2, \end{aligned} \tag{50}$$

The exact solution can be expanded as

$$x_n(t) = \sum_{n=0}^{\infty} b_n G_n(t - t_m). \tag{51}$$

Notice that the third of equations (53) shows that b_{n+2} ($n \geq 2$) is the n order derivative of the function $g_m(t)$, at $t = t_m$. Therefore, the definition of the G -functions allows us to obtain the identity

$$x(t) = \sum_{n=0}^p b_n G_n(t - t_m) + g_m^{(p+1)}(\eta_m) G_{p+1}(t - t_m), \tag{52}$$

where $t_m < \eta_m < t$ and $p \geq 2$.

As for the derivative, it is given by

$$x'(t) = \sum_{n=0}^p b_n G'_n(t - t_m) + g_m^{(p+1)}(\eta_m) G'_{p+1}(t - t_m), \tag{53}$$

where $t_m < \eta_m < t$ and $p \geq 2$.

Given a sequence of points $\{t_m\}_{m=0}^N$ with $t_0 = 0$, $t_N = T$, evenly spaced or not, we can compute approximations (x_m, x'_m) to the exact values $(x(t_m), x'(t_m))$ by truncating (52) and (53) as

$$x_{m+1} = \sum_{n=0}^p b_n G_n(t_{m+1} - t_m), \tag{54}$$

$$x'_{m+1} = \sum_{n=0}^p b_n G'_n(t_{m+1} - t_m), \tag{55}$$

where b_n are computed from the recurrences (49) and (50) by setting $a_0 = x_m$, $a_1 = x'_m$, instead of $a_0 = x_0$, $a_1 = x'_1$.

When $t_{m+1} - t_m = h$, $m = 0, \dots, N-1$ the local truncation error is easily derived from (52) and (53) as

$$x(t_{m+1}) - x_{m+1} = g_m^{(p+1)}(\xi_m) G_{p+1}(t - t_m) = O(\varepsilon h^{p+1}), \tag{56}$$

$$x'(t_{m+1}) - x'_{m+1} = g_m^{(p+1)}(\xi_m) G'_{p+1}(t - t_m) = O(\varepsilon h^p). \tag{57}$$

The numerical scheme given in (54) and (55) is thus consistent, of order p for the solution $x(t)$.

The stability of the scheme given by (54) and (55) is easy to prove directly. Taking into account equations (49) and (50), the algorithm can be written as

$$x_{n+1} = (G_0 + \gamma G_1)x_n + G_1 x'_n + h\Phi_0(t, x_n, x'_n, f), \tag{58}$$

$$x'_{n+1} = (G'_0 + \gamma G'_1)x_n + G'_1 x'_n + h\Phi_1(t, x_n, x'_n, f), \tag{59}$$

when all G -functions are evaluated at h .

Using the identities:

$$G_0 + \gamma G_1 \equiv e^{-\frac{1}{2}\gamma t} \left(\frac{\gamma}{M} \sinh\left(\frac{Mt}{2}\right) + \cosh\left(\frac{Mt}{2}\right) \right), \quad (60)$$

$$G_1 \equiv \frac{2e^{-\frac{1}{2}\gamma t}}{M} \sinh\left(\frac{Mt}{2}\right), \quad (61)$$

$$G'_0 + \gamma G'_1 \equiv \frac{e^{-\frac{1}{2}\gamma t}}{2} \left(M \sinh\left(\frac{Mt}{2}\right) - \gamma \cosh\left(\frac{Mt}{2}\right) \right), \quad (62)$$

$$G'_1 \equiv e^{-\frac{1}{2}\gamma t} \left(\cosh\left(\frac{Mt}{2}\right) - \frac{\gamma}{M} \sinh\left(\frac{Mt}{2}\right) \right), \quad (63)$$

with $M = (\gamma^2 - 4\alpha)^{\frac{1}{2}}$, equations (58) and (59) can be transformed into

$$\begin{pmatrix} x_{n+1} \\ x'_{n+1} \end{pmatrix} = e^{-\frac{1}{2}\gamma t} \begin{pmatrix} \frac{\gamma}{M} \sinh\left(\frac{Mt}{2}\right) + \cosh\left(\frac{Mt}{2}\right) & \frac{2}{M} \sinh\left(\frac{Mt}{2}\right) \\ \frac{M}{2} \sinh\left(\frac{Mt}{2}\right) - \frac{\gamma}{2} \cosh\left(\frac{Mt}{2}\right) & \cosh\left(\frac{Mt}{2}\right) - \frac{\gamma}{M} \sinh\left(\frac{Mt}{2}\right) \end{pmatrix} \begin{pmatrix} x_n \\ x'_n \end{pmatrix} + h \begin{pmatrix} \Phi_0 \\ \Phi_1 \end{pmatrix} = A \begin{pmatrix} x_n \\ x'_n \end{pmatrix} + h\Phi. \quad (64)$$

This equation owns the form of equation (2-4) of Lambert's book [26], p. 24. We have to notice that the assumptions there are fulfilled, because the function Φ vanishes whenever $f(t) \equiv 0$, since then $c_n = 0, \forall n \geq 0$ and it verifies a Lipschitz condition, since f was assumed to be analytic.

On the other hand, the eigenvalues of matrix A are $\exp\left(\pm \frac{(M \mp \gamma)t}{2}\right)$, then the root condition holds.

Therefore, the application of Lambert's Theorem 2.1 [26] shows that.

Proposition 1 *The scheme given by (52) and (53) is zero-stable.*

Notice that the Lipschitz condition coming from the differentiability of f implies that the single-step method (54) and (55) is regular, and by virtue of its consistency we could have established directly the following

Proposition 2 *The method (58) and (59) is convergent.*

RESIDUE CALCULATION

The truncation error of G -functions series method, is

$$r_m(t) = \varepsilon \sum_{k=m-1}^{\infty} c_k \frac{t^k}{k!}, \quad (65)$$

as

$$\begin{aligned} r_m(t) &= \varepsilon g(t) - (x''_m(t) + \gamma x'_m + \alpha x_m(t)) \\ &= \varepsilon g(t) - \sum_{k=0}^m b_k (G''_k(t) + \gamma G'_k(t) + \alpha G_k(t)) \\ &= \varepsilon \sum_{k=0}^{\infty} c_k \frac{t^k}{k!} - \varepsilon \sum_{k=2}^m c_{k-2} \frac{t^{k-2}}{(k-2)!} \\ &= \varepsilon \sum_{k=0}^{\infty} c_k \frac{t^k}{k!} - \varepsilon \sum_{k=0}^{m-2} c_k \frac{t^k}{k!} = \varepsilon \sum_{k=m-1}^{\infty} c_k \frac{t^k}{k!}, \end{aligned} \quad (66)$$

from all the above, it can be concluded that the parameter of perturbation ε is a factor for $r_m(t)$.

As a result, $r_m(t)$ will be small with ε . If $\varepsilon = 0$, the generalized Scheifele method of G -functions, with only the first and second term, integrates the equation exactly (1).

Although the method of series of G -functions has a high accuracy and integrates exactly the homogeneous problem, the recurrent calculation of the coefficients c_k , development (44), it is difficult for complicated expressions of the function of perturbation, which can impede the implementation of the method of series of G -functions on a computer.

To solve this difficulty, in the next section describes the transformation of the numerical method based on series of G -functions in a multistep scheme, similar to the methods SMF [13, 24], but variable step and order, i.e., obtaining VSVO multistep schemes [14, 15].

To obtain a multistep scheme, approximates the derivative of function of perturbation, using divided differences.

Explicit and implicit methods are constructed, which allow the implementation of computational methods predictor-corrector type.

MULTISTEP EXPLICIT METHOD OF p STEP, FOR PERTURBED AND DAMPED OSCILLATORS

In order to obtain the explicit method we will substitute the derivatives by expressions in term of divided differences and next to some coefficients d_{ij} , elements of a matrix A_p^{-t} , of those we do not know a recurrence relation. Once the matrix A_p^{-t} is known, we will set up a recurrent calculus, through matrix $S_{p,n}$, to be defined later for the explicit method. The study of symmetric polynomials [27] and its relation with the divided differences, will allow us the computation of the matrix $S_{p,n}$.

To make a variable step explicit multistep method of p -steps, we use up to $(p-1)$ -th order divided difference of a function g in the grid values t_n, \dots, t_{n-p+1} we denote by $g[t_n, \dots, t_{n-k}]$ [28], with $k = 0, 1, \dots, n$, the divided difference of g with order k , in the arguments t_n, \dots, t_{n-k} of the variable $t \in [a, b]$.

The divided differences of perturbed function $g(t)$ satisfy the identity

$$g[t_n, \dots, t_{n-i}] = \sum_{j=0}^{\infty} P_j[0, -H_1, \dots, -H_i] \cdot g^{(j)}(t_n) \tag{67}$$

being $P_k(t) = \frac{t^k}{k!}$ and $H_i = t_n - t_{n-i}$ [16].

Denoting by $D_{p,n}$ the following matrix, with $1 \times p$ order

$$D_{p,n} = \left(\begin{array}{cccc} g[t_n] & !g[t_n, t_{n-1}] & \dots & (p-1)!g[t_n, \dots, t_{n-(p-1)}] \end{array} \right) \tag{68}$$

And choosing $H = \max\{H_1, \dots, H_{p-1}\}$, verifies the identity

$$D_{p,n}^t = A_p \left(\begin{array}{c} g(t_n) \\ g'(t_n) \\ \vdots \\ g^{(p-1)}(t_n) \end{array} \right) + \left(\begin{array}{c} O(H^p) \\ O(H^{p-1}) \\ \vdots \\ O(H) \end{array} \right) \tag{69}$$

where

$$A_p = \left(\begin{array}{cccccc} 1 & P_1[0] & P_2[0] & \dots & P_{p-1}[0] & \\ 0 & 1 & !P_2[0, -H_1] & \dots & !P_{p-1}[0, -H_1] & \\ 0 & 0 & 1 & \dots & 2!P_{p-1}[0, -H_1, -H_2] & \\ \vdots & \vdots & \vdots & \ddots & \vdots & \\ 0 & 0 & 0 & \dots & 1 & \end{array} \right)_{p \times p} \tag{70}$$

using a more compact notation is possible to write (69) as:

$$D_{p,n}^t = A_p \times Z_{p \times 1} + O_{p \times 1}, \text{ being}$$

$$Z_{1 \times p}^t = \left(\begin{array}{cccc} g(t_n) & g'(t_n) & \dots & g^{(p-1)}(t_n) \end{array} \right). \tag{71}$$

After truncating the former expansion and solving for $Z_{p \times 1}$, it results:

$$Z_{p \times 1} = A_p^{-1} \times (D_{p,n}^t)_{p \times 1} \text{ and } Z_{1 \times p}^t = D_{p,n} \times A_p^{-t}. \tag{72}$$

Designating by $(d_{ij})_{p \times p} = A_p^{-t} = (A_p^{-1})^t$, it can write:

$$Z_{1 \times p}^t = \left(\begin{array}{c} \sum_{i=1}^p g[t_n, \dots, t_{n-(i-1)}] d_{i1} (i-1)! \dots \\ \sum_{i=1}^p g[t_n, \dots, t_{n-(i-1)}] d_{ip} (i-1)! \end{array} \right)_{1 \times p} \tag{73}$$

Substituting in (28) and (29), we obtain by truncation

$$\begin{aligned} x_{n+1} &= x_n G_0(h) + (x'_n + x_n \gamma) G_1(h) \\ &+ \varepsilon \sum_{j=0}^{p-1} \left(\sum_{i=1}^p g[t_n, \dots, t_{n-(i-1)}] d_{i(j+1)} (i-1)! \right) G_{j+2}(h) \\ &= x_n G_0(h) + (x'_n + x_n \gamma) G_1(h) \\ &+ \varepsilon \sum_{i=1}^p g[t_n, \dots, t_{n-(i-1)}] \left(\sum_{j=0}^{p-1} d_{i(j+1)} (i-1)! G_{j+2}(h) \right), \\ x'_{n+1} &= (x'_n + x_n \gamma) G_0(h) + x_n G'_0(h) \\ &+ \varepsilon \sum_{j=0}^{p-1} \left(\sum_{i=1}^p g[t_n, \dots, t_{n-(i-1)}] d_{i(j+1)} (i-1)! \right) G_{j+1}(h) \end{aligned} \tag{75}$$

$$\begin{aligned} &= (x'_n + x_n \gamma) G_0(h) + x_n G'_0(h) \\ &+ \varepsilon \sum_{i=1}^p g[t_n, \dots, t_{n-(i-1)}] \left(\sum_{j=0}^{p-1} d_{i(j+1)} (i-1)! G_{j+1}(h) \right). \end{aligned}$$

Denoting

$$\Lambda_i = \sum_{j=0}^{p-1} d_{i(j+1)} (i-1)! G_{j+2}(h), \tag{76}$$

$$\Lambda'_i = \sum_{j=0}^{p-1} d_{i(j+1)} (i-1)! G_{j+1}(h), \text{ with } i = 1, \dots, p, \tag{77}$$

we obtain the following formulas, for a multistep explicit method

$$\begin{aligned} x_{n+1} &= x_n G_0(h) + (x'_n + x_n \gamma) G_1(h) \\ &+ \varepsilon \sum_{i=1}^p \Lambda_i g[t_n, \dots, t_{n-(i-1)}] \end{aligned}, \tag{78}$$

$$\begin{aligned} x'_{n+1} &= (x'_n + x_n \gamma) G_0(h) + x_n G'_0(h) \\ &+ \varepsilon \sum_{i=1}^p \Lambda'_i g[t_n, \dots, t_{n-(i-1)}] \end{aligned}. \tag{79}$$

In order to design a method of variable step and variable order, VSVO, be obtained the elements of matrix $A_p^{-t} = (d_{ij})_{p \times p}$ recursively.

The recurrences of elements d_{ij} , are based on the complete symmetric functions, $h_{n,r}$ and the elementary symmetrical functions $e_{n,r}$ [27, 29], defined as:

$$e_{n,0} = 1, \quad e_{n,r} = \sum_{i_1 < i_2 < \dots < i_r} t_{i_1} \dots t_{i_r}, \quad e_{n,r} = 0 \text{ with } r < 0, \tag{80}$$

$$h_{n,r} = \sum_{|\lambda|=r} \sum_{S_\lambda} t^\alpha \tag{81}$$

where $\lambda = (\lambda_1 \dots \lambda_n) \in \mathbb{N}^n$ being $|\lambda| = \lambda_1 + \dots + \lambda_n$, and

$S_\lambda = \{ \text{all the differents permutations } \alpha = (\alpha_1 \dots \alpha_n) \text{ of } \lambda \}$
with $t^\alpha = t_1^{\alpha_1} \dots t_n^{\alpha_n}$, in the case $r < 0$, it is defined as $h_{n,r} = 0$,

$$\text{particularly } h_{n,0} = 1 \text{ and } h_{n,1} = e_{n,1}. \tag{82}$$

Between the divided differences of $g(t) = t^m$, that we will denote by $t^m[t_1, \dots, t_n]$ and the complete symmetrical polynomial the next relations holds

$$t^m[t_1, \dots, t_n] = h_{n,m-n+1}.$$

If consider the point t_n , it define the complete symmetric functions,

$$q_{ij}(n) = t^{j-1} [H_n, \dots, H_{n-(i-1)}] \text{ and } \sigma_{ij}(n) = (-1)^{j-i} e_{j-1, j-i}, \tag{83}$$

in values $H_{n-k} = t_{n-k} - t^*$ with $k = 0, \dots, i-1$, and

$$t^* \in [a, b]. \tag{84}$$

The square matrices of order k , $P_{k,n} = (q_{ij}(n))$ and $S_{k,n} = (\sigma_{ij}(n))$, are inverse each other.

As $H_{n-j} = t_{n-j} - t^*$ and $H_j = t_n - t_{n-j}$ it can write $(t_n - t^*) - H_j = H_{n-j}$ with $j = 0, \dots, i-1$.

In the particular case $t^* = t_n$ we will get $H_{n-j} = -H_j$ with $j = 0, \dots, i-1$.

The divided differences of one function g satisfy the property:

$$g[t_n, t_{n-1}, \dots, t_{n-(i-1)}] = \sum_{j=0}^{i-1} q_{i(j+1)}(n) \frac{1}{j!} g^{(j)}(t^*). \tag{85}$$

If $H = \max\{|H_n|, \dots, |H_{n-(i-1)}|\}$, as $q_{ij}(n)$ have order $j-i$ in H , it can write:

$$g[t_n, t_{n-1}, \dots, t_{n-(i-1)}] = \sum_{j=0}^{p-1} q_{i(j+1)}(n) \frac{1}{j!} g^{(j)}(t^*) + O(H^{p-(i-1)})$$

with $i = 1, \dots, p$. (86)

Considering $t^* = t_n$ and expressing those equalities (86) in a matricial way, we have

$$\begin{pmatrix} g[t_n] \\ g[t_n, t_{n-1}] \\ \vdots \\ g[t_n, \dots, t_{n-(p-1)}] \end{pmatrix} = \begin{pmatrix} q_{11}(n) & \dots & q_{1p}(n) \\ q_{21}(n) & \dots & q_{2p}(n) \\ \vdots & \ddots & \vdots \\ q_{p1}(n) & \dots & q_{pp}(n) \end{pmatrix} \begin{pmatrix} g(t_n) \\ g'(t_n) \\ \vdots \\ \frac{g^{(p-1)}(t_n)}{(p-1)!} \end{pmatrix} + \begin{pmatrix} O(H^p) \\ O(H^{p-1}) \\ \vdots \\ O(H) \end{pmatrix}, \tag{87}$$

and as $q_{i(j+1)}(n) = h_{i,j}$ in the arguments $H_n, \dots, H_{n-(i-1)}$, it can write

$$\begin{pmatrix} g[t_n] \\ g[t_n, t_{n-1}] \\ \vdots \\ g[t_n, \dots, t_{n-(p-1)}] \end{pmatrix} = \begin{pmatrix} 1 & h_{1,1} & \dots & h_{1,p-1} \\ 0 & 1 & \dots & h_{2,p-2} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{pmatrix} \begin{pmatrix} g(t_n) \\ g'(t_n) \\ \vdots \\ \frac{g^{(p-1)}(t_n)}{(p-1)!} \end{pmatrix} + \begin{pmatrix} O(H^p) \\ O(H^{p-1}) \\ \vdots \\ O(H) \end{pmatrix}. \tag{88}$$

Since $\sigma_{ij}(n) = \sigma_{(i-1)(j-1)}(n) - H_{n-j+2} \sigma_{i(j-1)}(n)$ for $i, j \geq 2$, if we consider $t^* = t_n$, then:

$$S_{p,n} = (\sigma_{ij}(n))_{p \times p} \tag{89}$$

$$\text{with } \begin{cases} \sigma_{11}(n) = 1 \\ \sigma_{1j}(n) = 0, \quad 1 < j \leq p \\ \sigma_{i1}(n) = 0, \quad 1 < i \leq p \\ \sigma_{ij}(n) = \sigma_{(i-1)(j-1)}(n) - H_{n-j+2} \sigma_{i(j-1)}(n), \quad 2 \leq i, j \leq p \end{cases}$$

Getting recurrent form of the matrix A_p^{-t} through:

$$A_p^{-t} = M_p \times P_{p,n}^{-t} \times N_p = M_p \times S_{p,n}^t \times N_p, \text{ i.e.} \tag{90}$$

$$d_{ij} = \frac{(j-1)! \sigma_{ji}(n)}{(i-1)!} \text{ with } i, j = 1, \dots, p, \tag{91}$$

where $M_p = (m_{ij})_p$ is a diagonal matrix, such that $m_{ii} = \frac{1}{i!}$, with $i = 0, \dots, p-1$ and $N_p = M_p^{-1}$.

The previous expressions (89) and (90) allow us to compute the A_p^{-t} matrix by recurrence, from $S_{p,n}^t$ matrix.

Substituting (91) in (76), (77), (78) and (79), is possible define the multistep explicit method, using the next notation:

Definition 1

Let:

x_n the approximation to the value of the solution in t_n .

x'_n the approximation to the value of the derivative in t_n .

The starting values for the solution $x(t)$ in the points t_0, \dots, t_{p-1} are: x_0, \dots, x_{p-1} .

The starting values for the derivative $x'(t)$ in the points t_0, \dots, t_{p-1} are: x'_0, \dots, x'_{p-1} .

The formal expression of explicit multistep method, based on G-functions generalized, is:

$$x_{n+1} = x_n G_0(h) + (x'_n + x_n \gamma) G_1(h) + \varepsilon \sum_{i=1}^p \Lambda_i g[t_n, \dots, t_{n-(i-1)}] \tag{92}$$

with $n \geq p-1$,

$$x'_{n+1} = (x'_n + x_n \gamma) G_0(h) + x_n G'_0(h) + \varepsilon \sum_{i=1}^p \Lambda'_i g[t_n, \dots, t_{n-(i-1)}] \tag{93}$$

with $n \geq p-1$,

where

$$\Lambda_i = \sum_{j=0}^{p-1} j! \sigma_{(j+1)i}(n) G_{j+2}(h) \text{ with } i = 1, \dots, p, \tag{94}$$

$$\Lambda'_i = \sum_{j=0}^{p-1} j! \sigma_{(j+1)i}(n) G_{j+1}(h) \text{ with } i = 1, \dots, p. \tag{95}$$

Proposition 3 In the explicit method of p steps, the small perturbation parameter is always a common factor in the

expression of truncation errors in each step. If $\varepsilon = 0$, the algorithm integrates exactly the unperturbed problem.

Proof: Suppose that the value calculated for x and x' in $t = nh$ is exact, i.e., $x_n = x(nh)$ and $x'_n = x'(nh)$.

In addition also the value of $f(x_{n-j}, x'_{n-j}, (n-j)h)$ is exact, with $j = 0, 1, \dots, p-1$, i.e.

$$f(x_{n-j}, x'_{n-j}, (n-j)h) = f(x((n-j)h), x'((n-j)h), (n-j)h)$$

with $j = 0, 1, \dots, p-1$. Let $r = x((n+1)h) - x_{n+1}$, then:

$$\begin{aligned} r &= \sum_{j=0}^{\infty} \frac{x^{(j)}(nh)}{j!} h^j - \left(x_n G_0(h) + (x'_n + \gamma x_n) G_1(h) \right. \\ &\quad \left. + \varepsilon \sum_{j=1}^p \Lambda_j g[t_n, \dots, t_{n-(j-1)}] \right) \\ &= \sum_{j=0}^{\infty} x^{(j)}(nh) (G_j(h) + \gamma G_{j+1}(h) + \alpha G_{j+2}(h)) \\ &\quad - \left(x_n G_0(h) + (x'_n + \gamma x_n) G_1(h) + \varepsilon \sum_{j=1}^p \Lambda_j g[t_n, \dots, t_{n-(j-1)}] \right) \\ &= x(nh) G_0(h) + (x'(nh) + \gamma x(nh)) G_1(h) \\ &\quad + \sum_{j=2}^{\infty} (x^{(j)}(nh) + \gamma x^{(j-1)}(nh) + \alpha x^{(j-2)}(nh)) G_j(h) \\ &\quad - \left(x_n G_0(h) + (x'_n + \gamma x_n) G_1(h) + \varepsilon \sum_{j=1}^p \Lambda_j g[t_n, \dots, t_{n-(j-1)}] \right) \\ &= \sum_{j=2}^{\infty} (x^{(j)}(nh) + \gamma x^{(j-1)}(nh) + \alpha x^{(j-2)}(nh)) G_j(h) \\ &\quad - \varepsilon \sum_{j=1}^p \Lambda_j g[t_n, \dots, t_{n-(j-1)}] \\ &= \sum_{j=2}^{\infty} \varepsilon f^{(j-2)}(x_n, x'_n, nh) G_j(h) - \varepsilon \sum_{j=1}^p \Lambda_j g[t_n, \dots, t_{n-(j-1)}] \\ &= \varepsilon \left(\sum_{j=2}^{\infty} f^{(j-2)}(x_n, x'_n, nh) G_j(h) - \sum_{j=1}^p \Lambda_j g[t_n, \dots, t_{n-(j-1)}] \right). \end{aligned}$$

In the same way, let $r' = x'((n+1)h) - x'_{n+1}$, then:

$$\begin{aligned} r' &= \sum_{j=0}^{\infty} \frac{x^{(j+1)}(nh)}{j!} h^j - \left((x'_n + \gamma x_n) G_0(h) + x_n G'_0(h) \right. \\ &\quad \left. + \varepsilon \sum_{j=1}^p \Lambda'_j g[t_n, \dots, t_{n-(j-1)}] \right) \\ &= \sum_{j=0}^{\infty} x^{(j+1)}(nh) (G_j(h) + \gamma G_{j+1}(h) + \alpha G_{j+2}(h)) \\ &\quad - \left((x'_n + \gamma x_n) G_0(h) + x_n (-\alpha G_1(h) - \gamma G_0(h)) \right. \\ &\quad \left. + \varepsilon \sum_{j=1}^p \Lambda'_j g[t_n, \dots, t_{n-(j-1)}] \right) \end{aligned}$$

$$\begin{aligned} &= x'(nh) G_0(h) + (x''(nh) + \gamma x'(nh)) G_1(h) \\ &\quad + \sum_{j=2}^{\infty} (x^{(j+1)}(nh) + \gamma x^{(j)}(nh) + \alpha x^{(j-1)}(nh)) G_j(h) \\ &\quad - \left(x'_n G_0(h) - \alpha x_n G_1(h) + \varepsilon \sum_{j=1}^p \Lambda'_j g[t_n, \dots, t_{n-(j-1)}] \right) \\ &= \sum_{j=2}^{\infty} (x^{(j+1)}(nh) + \gamma x^{(j)}(nh) + \alpha x^{(j-1)}(nh)) G_j(h) \\ &\quad - \varepsilon \sum_{j=1}^p \Lambda'_j g[t_n, \dots, t_{n-(j-1)}] \\ &= \sum_{j=1}^{\infty} \varepsilon f^{(j-1)}(x_n, x'_n, nh) G_j(h) - \varepsilon \sum_{j=1}^p \Lambda_j g[t_n, \dots, t_{n-(j-1)}] \\ &= \varepsilon \left(\sum_{j=1}^{\infty} f^{(j-1)}(x_n, x'_n, nh) G_j(h) - \sum_{j=1}^p \Lambda_j g[t_n, \dots, t_{n-(j-1)}] \right). \end{aligned}$$

MULTISTEP IMPLICIT METHOD OF p STEP, FOR PERTURBED AND DAMPED OSCILLATORS

Similarly for the explicit case [16], the matrix of that we extract the coefficients d_{ij} we will denote as B_p^{-t} , being the matrix B_p :

$$B_p = \begin{pmatrix} 1 & P_1[h] & P_2[h] & \dots & P_p[h] \\ 0 & 1 & !P_2[h, 0] & \dots & !P_p[h, 0] \\ 0 & 0 & 1 & \dots & 2!P_p[h, 0, -H_1] \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{pmatrix}_{(p+1) \times (p+1)} \quad (96)$$

Designating by $(d_{ij})_{(p+1) \times (p+1)} = B_p^{-t} = (B_p^{-1})^t$, it can write

$$Z'_{1 \times (p+1)} = \begin{pmatrix} \sum_{i=1}^{p+1} g[t_{n+1}, \dots, t_{n+1-(i-1)}] d_{i1} (i-1)! \dots \\ \sum_{i=1}^{p+1} g[t_{n+1}, \dots, t_{n+1-(i-1)}] d_{i(p+1)} (i-1)! \end{pmatrix}_{1 \times p} \quad (97)$$

Proceeding analogously to the explicit case, we obtain:

$$\begin{aligned} x_{n+1} &= x_n G_0(h) + (x'_n + x_n \gamma) G_1(h) \\ &\quad + \varepsilon \sum_{j=0}^p \left(\sum_{i=1}^{p+1} g[t_{n+1}, \dots, t_{n+1-(i-1)}] d_{i(j+1)} (i-1)! \right) G_{j+2}(h) \end{aligned} \quad (98)$$

$$\begin{aligned} &= x_n G_0(h) + (x'_n + x_n \gamma) G_1(h) \\ &\quad + \varepsilon \sum_{i=1}^{p+1} g[t_{n+1}, \dots, t_{n+1-(i-1)}] \left(\sum_{j=0}^p d_{i(j+1)} (i-1)! G_{j+2}(h) \right), \\ x'_{n+1} &= (x'_n + x_n \gamma) G_0(h) + x_n G'_0(h) \\ &\quad + \varepsilon \sum_{j=0}^p \left(\sum_{i=1}^{p+1} g[t_{n+1}, \dots, t_{n+1-(i-1)}] d_{i(j+1)} (i-1)! \right) G_{j+1}(h), \end{aligned} \quad (99)$$

$$= (x'_n + x_n \gamma) G_0(h) + x_n G'_0(h) + \varepsilon \sum_{i=1}^{p+1} g[t_{n+1}, \dots, t_{n+1-(i-1)}] \left(\sum_{j=0}^p d_{i(j+1)} (i-1)! G_{j+1}(h) \right).$$

Denoting

$$\Gamma_i = \sum_{j=0}^p d_{i(j+1)} (i-1)! G_{j+2}(h), \tag{100}$$

$$\Gamma'_i = \sum_{j=0}^p d_{i(j+1)} (i-1)! G_{j+1}(h), \tag{101}$$

with $i = 1, \dots, p + 1$, we obtain the following formulas, for a multistep implicit method,

$$x_{n+1} = x_n G_0(h) + (x'_n + x_n \gamma) G_1(h) + \varepsilon \sum_{i=1}^{p+1} \Gamma_i g[t_{n+1}, \dots, t_{n+1-(i-1)}], \tag{102}$$

$$x'_{n+1} = (x'_n + x_n \gamma) G_0(h) + x_n G'_0(h) + \varepsilon \sum_{i=1}^{p+1} \Gamma'_i g[t_{n+1}, \dots, t_{n+1-(i-1)}]. \tag{103}$$

Once the matrix B_p^{-t} is known, we will set up a recurrent calculus, through matrix $S_{p,n+1}$ for the implicit method, being the matrix $S_{p,n+1}$:

$$S_{p,n+1} = (\sigma_{ij}(n))_{(p+1) \times (p+1)} \tag{104}$$

with

$$\begin{cases} \sigma_{11}(n) = 1, \quad \sigma_{1,2}(n) = -h_{n+1} \\ \sigma_{1j}(n) = 0, \quad 2 < j \leq p + 1 \\ \sigma_{i1}(n) = 0, \quad 1 < i \leq p + 1 \\ \sigma_{ij}(n) = \sigma_{(i-1)(j-1)}(n) - H_{n-j+3} \sigma_{i(j-1)}(n), \quad 2 \leq i, j \leq p + 1 \end{cases}$$

The recurrent form of the matrix B_p^{-t} , is then got through:

$$B_p^{-t} = M_{p+1} \times P_{p,n+1}^{-t} \times N_{p+1} = M_{p+1} \times S_{p,n+1}^t \times N_{p+1}, \text{ i.e.} \tag{105}$$

$$d_{ij} = \frac{(j-1)! \sigma_{ji}(n)}{(i-1)!} \text{ with } i, j = 1, \dots, p + 1, \tag{106}$$

where

$M_{p+1} = (m_{ij})_{p+1}$ is a diagonal matrix, such that $m_{ii} = \frac{1}{i!}$, with $i = 0, \dots, p$ and $N_{p+1} = M_{p+1}^{-1}$.

The expressions (104) and (105) allow us to compute the B_p^{-t} matrix by recurrence, from $S_{p,n+1}^t$.

Substituting (106) in (100), (101), (102) and (103), is possible define the multistep implicit method, using the next notation:

Definition 2

Let:

x_n the approximation to the value of the solution in t_n .

x'_n the approximation to the value of the derivative in t_n .

The starting values for the solution $x(t)$ in the points t_0, \dots, t_{p-1} are: x_0, \dots, x_{p-1} .

The starting values for the derivative $x'(t)$ in the points t_0, \dots, t_{p-1} are: x'_0, \dots, x'_{p-1} .

The formal expression of implicit multistep method, based on G -functions generalized, is:

$$x_{n+1} = x_n G_0(h) + (x'_n + x_n \gamma) G_1(h) + \varepsilon \sum_{i=1}^{p+1} \Gamma_i g[t_{n+1}, \dots, t_{n+1-(i-1)}] \tag{107}$$

with $n \geq p - 1$,

$$x'_{n+1} = (x'_n + x_n \gamma) G_0(h) + x_n G'_0(h) + \varepsilon \sum_{i=1}^{p+1} \Gamma'_i g[t_{n+1}, \dots, t_{n+1-(i-1)}] \tag{108}$$

with $n \geq p - 1$,

where

$$\Gamma_i = \sum_{j=0}^p j! \sigma_{(j+1)i} G_{j+2}(h) \text{ with } i = 1, \dots, p + 1, \tag{109}$$

$$\Gamma'_i = \sum_{j=0}^p j! \sigma_{(j+1)i} G_{j+1}(h) \text{ with } i = 1, \dots, p + 1. \tag{110}$$

Proposition 4: In the implicit method of p steps, the small perturbation parameter ε is always a common factor in the expression of truncation errors in each step. If $\varepsilon = 0$ the algorithm integrates exactly the unperturbed problem.

MULTISTEP PREDICTOR-CORRECTOR METHOD OF p STEP, FOR PERTURBED AND DAMPED OSCILLATORS

The predictor-corrector method, with variable step size, of p steps for perturbed and damped oscillators is defined like the one which have as predictor explicit method and as corrector he implicit method, with the previous definitions (92), (93) and (107), (108).

That is, the predictor-corrector method used is the type $P(EC)^\mu E^{1-t}$, with $t = \mu = 1$.

NUMERICAL EXPERIMENTS

In this section, it shows the good behaviour of the new numerical and computational methods based on G -functions, comparing the accuracy of both methods, in front of to known codes:

LSODE which obtains numerical solutions using the Livermore Stiff ODE integrator. GEAR, which obtains numerical solutions through Burlirsch-Stöer, rational extrapolation method.

MGEAR [msteppart] which is a multistep method, capable of being applied to stiff problems.

Used in solving the test problems, which are outlined below, its implementation in Maple.

Problem 1

Let's consider the following stiff problem, which appears in [19, 30, 31].

$$\begin{cases} x_1'(t) = 2x_1(t) + x_2(t) + 2 \sin(t) \\ x_2'(t) = -(\eta + 2)x_1(t) + (\eta + 1)(x_2(t) - \cos(t) + \sin(t)) \end{cases} \quad (111)$$

with initial conditions $x_1(0) = 2$, $x_2(0) = 3$ and solution independent of η :

$$x_1(t) = 2e^{-t} + \sin(t), \quad x_2(t) = 2e^{-t} + \cos(t). \quad (112)$$

The eigenvalues of the system are -1 and η , which enables its degree of stiffness to be regulated. For the case $\eta = -1000$, the following stiff problem is obtained, proposed in [26].

$$\begin{pmatrix} x_1'(t) \\ x_2'(t) \end{pmatrix} = \begin{pmatrix} -2 & 1 \\ 998 & -999 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} + \begin{pmatrix} 2 \sin(t) \\ 999(\cos(t) - \sin(t)) \end{pmatrix}, \quad (113)$$

$$x_1(0) = 2, \quad x_2(0) = 3.$$

By expressing (111) as a forced and damped oscillator, we obtain the IVP

$$x''(t) + 1001x'(t) + 1000x(t) = 1001 \cos(t) + 999 \sin(t), \quad (114)$$

$$x(0) = 2, \quad x'(0) = -1,$$

with exact solution and derivative

$$x(t) = 2e^{-t} + \sin t, \quad x'(t) = -2e^{-t} + \cos t. \quad (115)$$

To resolve this problem, has been using the method of series of G -functions primarily with $\alpha = 1000$ and $\gamma = 1001$, the integration being carried out using the algorithm described in (45) and (46). Then also has integrated IVP, by multistep predictor-corrector algorithm of p steps, for perturbed and damped oscillators.

To integrate the IVP using the method of series of G -functions, proceed as follows.

In the first integration step, the recurrence relation is obtained:

$$a_0 = x_0, \quad (116)$$

$$a_1 = x_0',$$

$$a_{k+2} = -\gamma a_{k+1} - \alpha a_k + 1001 \cos\left(k \frac{\pi}{2}\right) + 999 \sin\left(k \frac{\pi}{2}\right) \quad \text{with} \\ k \geq 0$$

and coefficients b_k are calculated through the relations:

$$b_0 = a_0 = x_0, \quad (117)$$

$$b_1 = a_1 + \gamma a_0 = x_0' + \gamma x_0,$$

$$b_{k+2} = a_{k+2} + \gamma a_{k+1} + \alpha a_k \quad \text{with, } k \geq 0.$$

Noting that the approximations of $x(h)$ and $x'(h)$ are x_1 and x_1' , respectively, the approximation of the solution and its derivative result in:

$$x_1 = b_0 G_0(h) + b_1 G_1(h) + \sum_{n=0}^{m-2} b_{n+2} G_{n+2}(h), \quad (118)$$

$$x_1' = b_0 G_0'(h) + b_1 G_0'(h) + \sum_{n=0}^{m-2} b_{n+2} G_{n+1}(h). \quad (119)$$

In order to carry out the second step of integration, it is necessary to solve the following IVP:

$$x''(t) + 1001x'(t) + 1000x(t) = 1001 \cos(t) + 999 \sin(t), \quad (120)$$

$$x(h) = x_1,$$

$$x'(h) = x_1',$$

which has the disadvantage that, when setting a step size h and calculating the value of the approximation $x(h)$ and $x'(h)$, it is not possible to move to a second step, since the perturbation function depends explicitly on time. To avoid this difficulty, we proceed in the following manner.

The independent variable $\tau = t - h$ is changed, and we obtain the IVP

$$x''(\tau) + 1001x'(\tau) + 1000x(\tau) = 1001 \cos(\tau + h) + 999 \sin(\tau + h) \quad (121)$$

$$x(0) = x_1,$$

$$x'(0) = x_1',$$

This strategy makes it possible to initialize the method.

Let's suppose that we have already calculated an approximation of the solution and its derivative in point $t = nh$, x_n and x_n' , the independent variable $\tau = t - nh$ is changed and we obtain

$$a_0 = x_n, \quad (122)$$

$$a_1 = x_n',$$

$$a_{k+2} = -\gamma a_{k+1} - \alpha a_k + 1001 \cos\left(k \frac{\pi}{2} + nh\right) + 999 \sin\left(k \frac{\pi}{2} + nh\right)$$

with $k \geq 0$

and coefficients b_k are calculated through the relations

$$b_0 = a_0, \quad (123)$$

$$b_1 = a_1 + \gamma a_0,$$

$$b_{k+2} = a_{k+2} + \gamma a_{k+1} + \alpha a_k \text{ with, } k \geq 0 .$$

The approximation of the solution $x(t)$ and $x'(t)$ in $t = (n + 1)h$, result in the formulas:

$$x_{n+1} = b_0 G_0(h) + b_1 G_1(h) + \sum_{n=0}^{m-2} b_{n+2} G_{n+2}(h), \tag{124}$$

$$x'_{n+1} = b_0 G'_0(h) + b_1 G'_1(h) + \sum_{n=0}^{m-2} b_{n+2} G'_{n+1}(h). \tag{125}$$

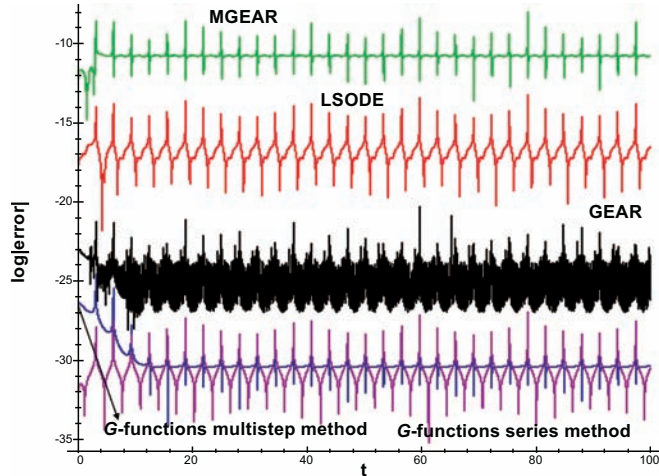


Fig. (1). Problem 1. $x(t)$ position.

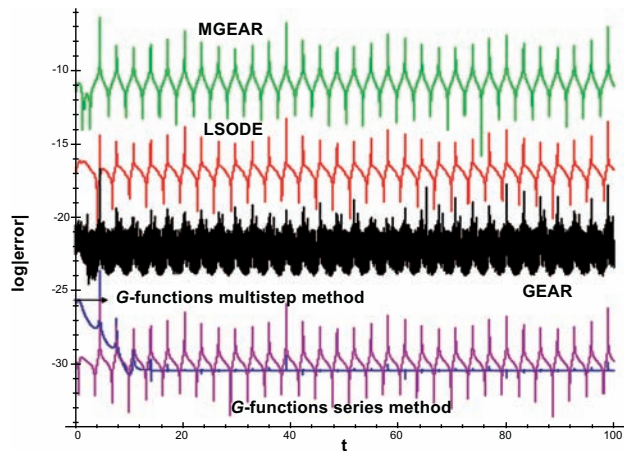


Fig. (2). Problem 1. $x'(t)$ velocity.

Fig. (1) shows the logarithm graph for the absolute value of the relative error of solution $x(t)$, with 40 digits and stepsize $h = 10^{-2}$, calculated by means of series and multistep methods, compared with the graphs corresponding for the logarithm of the absolute value of the relative error of the methods of LSODE[BACKFUNC] with $tol = 10^{-19}$, MGEAR[MSTPPART] with $errorper = Float(1, -13)$ and GEAR with $errorper = Float(1, -17)$.

Fig. (2) contrasts its derivative $x'(t)$ under the same conditions.

In Figs. (3, 4) we show an efficiency plot where G-functions multistep method its compared with integrations using well known general purpose codes. The computation

time is represented in the horizontal axis, in logarithmic scale, and the decimal logarithm of the integration error at the last point, $t = 100$, is shown in the vertical axis. The tolerances used in the standard codes are displayed in the figure into parentheses, marking each time-error point.

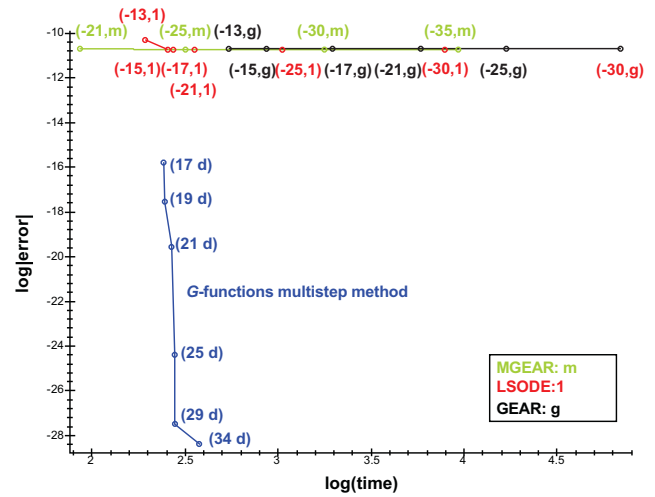


Fig. (3). Problem 1. Efficiency plot for the integration of the coordinate x at last point $t = 100$ versus computation time for different methods.

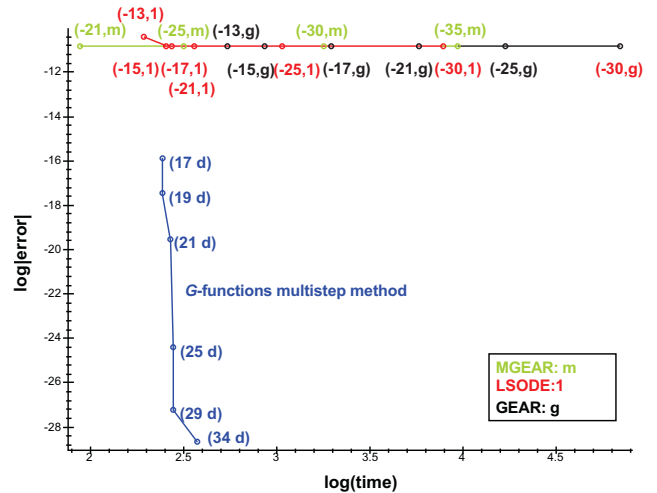


Fig. (4). Problem 1. Efficiency plot for the integration of the coordinate x' at last point $t = 100$ versus computation time for different methods.

We show the results of a few runs of it where the number of G-functions has been kept fixed at 4 and the number of digits used in the computations, that of course limit the attainable accuracy, has been varied to illustrate the behaviour of the method. The accuracy increases as the number of digits do, with a no noticeable major computational overhead. That number of digits is marked by the relevant point in the curve, with the figure followed by "d". To make the comparisons as honest as we can, the length of the mantissa used by MAPLE is adjusted according to the tolerances required to the integrator, so that for tolerances, 10^{-13} , 10^{-15} , 10^{-17} , 10^{-21} , 10^{-25} and 10^{-30} . We use 13 + 4 digits, 15 + 4 digits, 17 + 4, digits 21+4 digits,

digits 25+4 digits, and 30+4 digits, to avoid spurious increase of the computation times.

In spite of the good performance of the multistep codes chosen, Figs. (3, 4) show clearly that our algorithm is more accurate and efficient for this problem.

Problem 2

This highly oscillatory problem arises from mechanics, which appears in [12].

The mechanical oscillator is modelled by

$$x''(t) + x'(t) + 10000.25x(t) = \cos(10t), \tag{126}$$

$$x(0) = 1, \quad x'(0) = 0$$

which the exact solution and its derivative are expressed in the equations:

$$x(t) = A \cos(10t) + B \sin(10t) + e^{-\frac{t}{2}} (C \cos(100t) + D \sin(100t)) \tag{127}$$

$$x'(t) = -10A \sin(10t) + 10B \cos(10t) + e^{-\frac{t}{2}} \left(\left(100D - \frac{C}{2}\right) \cos(100t) - \left(100C + \frac{D}{2}\right) \sin(100t) \right) \tag{128}$$

with $A = \frac{158404}{1568240801}$, $B = \frac{160A}{158404}$, $C = 1 - A$,

$$D = \frac{C - 20B}{200}. \tag{129}$$

The model (126) corresponds to a vibrating system consisting of a mass of 1 kg attached to a spring of constant $k = 10000.25 \text{ kg/s}^2$. The mass starts from rest 1 m below its equilibrium position. The movement is driven by a periodic force that occurs when $t = 0$. The damping coefficient is 1 kg/s.

To resolve this problem we used the method of series of G-functions, the integration being carried out using the algorithm described in (45) and (46). Analogously to problem I, IVP has been integrated through the multistep predictor-corrector algorithm, p steps.

To integrate the IVP using the method of series of G-functions, proceed as follows.

Let's suppose that we have already calculated an approximation of the solution and its derivative in point $t = nh$, x_n and x'_n , the independent variable $\tau = t - nh$ is changed and we obtain

$$a_0 = x_n, \tag{130}$$

$$a_1 = x'_n,$$

$$a_{k+2} = -\gamma a_{k+1} - \alpha a_k + 10^k \cos\left(k \frac{\pi}{2} + 10nh\right) \text{ with } k \geq 0$$

and coefficients b_k are calculated through the relations

$$b_0 = a_0, \tag{131}$$

$$b_1 = a_1 + \gamma a_0,$$

$$b_{k+2} = a_{k+2} + \gamma a_{k+1} + \alpha a_k \text{ with, } k \geq 0.$$

The approximation of the solution $x(t)$ and $x'(t)$ in $t = (n + 1)h$, result in the formulas:

$$x_{n+1} = b_0 G_0(h) + b_1 G_1(h) + \sum_{n=0}^{m-2} b_{n+2} G_{n+2}(h), \tag{132}$$

$$x'_{n+1} = b_0 G'_0(h) + b_1 G'_1(h) + \sum_{n=0}^{m-2} b_{n+2} G'_{n+2}(h). \tag{133}$$

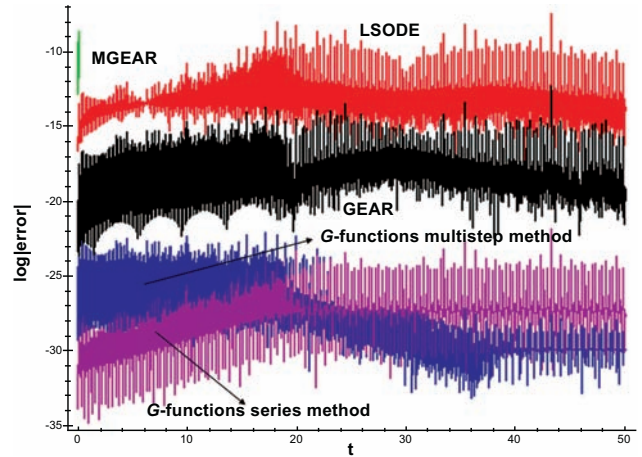


Fig. (5). Problem 2. $x(t)$ position.

Fig. (5) shows the logarithm graph for the absolute value of the relative error of solution $x(t)$, with 40 digits and stepsize $h = 5 \cdot 10^{-4}$, calculated by means of series and multistep methods, compared with the graphs corresponding for the logarithm of the absolute value of the relative error of the methods of LSODE[BACKFUNC] with $tol = 10^{-19}$, MGEAR[MSTPPART] with $errorper = Float(1, -12)$ and GEAR with $errorper = Float(1, -17)$.

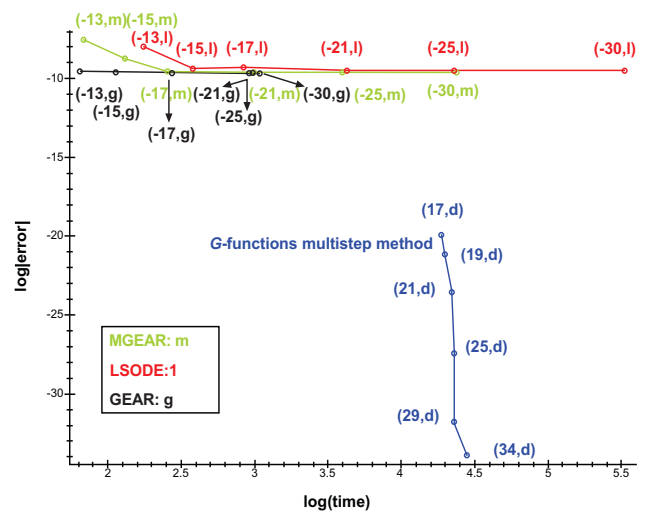


Fig. (6). Problem 2. Efficiency plot for the integration of the coordinate x at last point $t = 50$ versus computation time for different methods.

The result for the integration of the function $x(t)$ is shown in Fig. (6) in which the information is arranged as in Fig. (3).

Problem 3

Let's consider the highly oscillatory problem proposed by Petzold [32, 33], which contains a harmonic oscillator:

$$x''(t) + \lambda^2 x(t) = a \sin(\lambda t), \text{ with} \tag{134}$$

$$x(0) = 1, \quad x'(0) = -\frac{a}{2\lambda},$$

the exact solution and its derivative are expressed in the equations:

$$x(t) = \left(1 - \frac{a}{2\lambda} t\right) \cos(\lambda t), \tag{135}$$

$$x'(t) = -\lambda \left(1 - \frac{a}{2\lambda} t\right) \sin(\lambda t) - \frac{a}{2\lambda} \cos(\lambda t).$$

Although its solution can be calculated exactly by means of analytical procedures, this example has been chosen to illustrate how the G -functions series method works for highly oscillatory harmonic perturbation functions.

Has been applied, as in the two previous problems, the method of series of G -functions and multi-step predictor-corrector algorithm, p steps.

To integrate the IVP using the method of series of G -functions, proceed as follows:

$$a_0 = x_n, \tag{136}$$

$$a_1 = x'_n,$$

$$a_{k+2} = -\gamma a_{k+1} - \alpha a_k + a \lambda^k \sin\left(k \frac{\pi}{2} + \lambda n h\right) \text{ with } k \geq 0$$

and coefficients b_k are calculated through the relations

$$b_0 = a_0, \tag{137}$$

$$b_1 = a_1 + \gamma a_0,$$

$$b_{k+2} = a_{k+2} + \gamma a_{k+1} + \alpha a_k \text{ with, } k \geq 0.$$

The approximation of the solution $x(t)$ and $x'(t)$ in $t = (n + 1)h$, result in the formulas:

$$x_{n+1} = b_0 G_0(h) + b_1 G_1(h) + \sum_{n=0}^{m-2} b_{n+2} G_{n+2}(h), \tag{138}$$

$$x'_{n+1} = b_0 G'_0(h) + b_1 G'_1(h) + \sum_{n=0}^{m-2} b_{n+2} G'_{n+2}(h). \tag{139}$$

The problem is solved for the case $a = 1, \lambda = 10, \alpha = \lambda^2$ and $\gamma = 0$ [33].

As $\gamma = 0$, this problem shows that the algorithm described in (45) and (46) generalizes the method based on the Scheifele functions described in [24] for the undamped case.

Fig. (7) shows the logarithm graph for the absolute value of the relative error of solution $x(t)$, with 40 digits and stepsize $h = 10^{-3}$, calculated by means of series and multistep methods, compared with the graphs corresponding for the logarithm of the absolute value of the relative error of the methods of LSODE[BACKFUNC] with $tol = 10^{-19}$, MGEAR[MSTPPART] with $errorper = Float(1, -12)$ and GEAR with $errorper = Float(1, -17)$.

The result for the integration of the function $x(t)$ is shown in Fig. (8) in which the information is arranged as in Fig. (3).

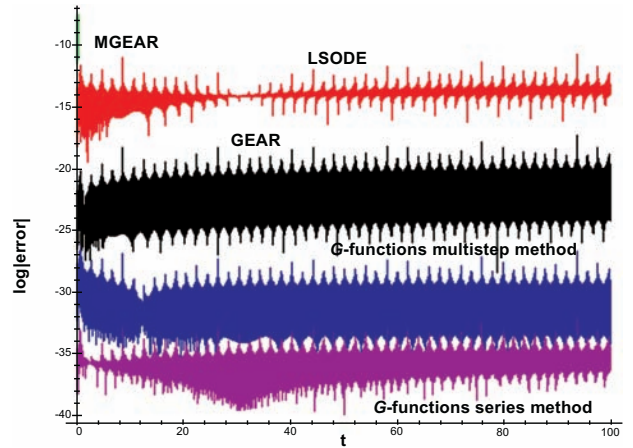


Fig. (7). Problem 3. $x(t)$ position.

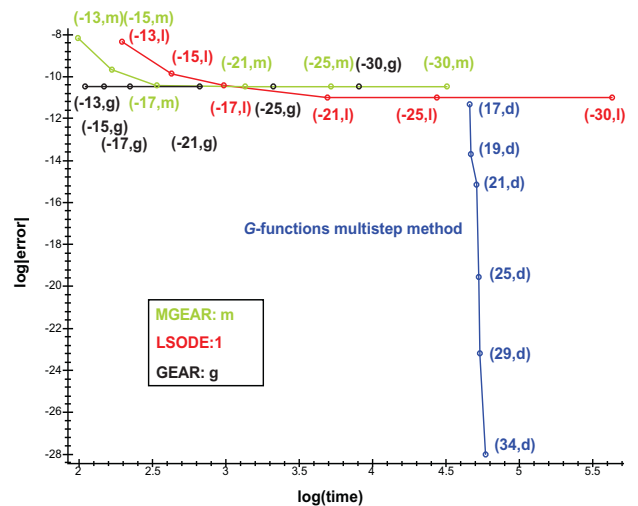


Fig. (8). Problem 3. Efficiency plot for the integration of the coordinate x at last point $t = 100$ versus computation time for different methods.

CONCLUSIONS

The first part of this article, is devoted to study of Scheifele functions adapted to forced and damped oscillators, as well as their most relevant properties, in particular, the law of recurrence useful for calculating said functions. A numerical and computational series method of integration is constructed, zero stable and convergent. This series method is based on adapted Scheifele functions. This scheme is extremely precise, however it is difficult to adapt

to each specific problem. In order to overcome this problem, in a second part, this article we describe how the series method is transformed to a multi-step scheme, the coefficients of which are calculated in a recurrent manner which permits formulation of a multi-step VSVO type scheme. Said scheme generalises the SMF method, the coefficients of which are not calculated using recurrences.

A computational algorithm is constructed on the multi-step scheme which enables it to be implemented in a computer. Both the series method and the multi-step method are contrasted against other known integrators, comparing the results obtained with the exact solution to the test problems proposed.

The benefit produced by the use of the new algorithm is demonstrated when applied to the problems for which it has been designed.

ACKNOWLEDGEMENTS

This work has been supported by GRE09-13 project of the University of Alicante, and the project of the Generalitat Valenciana GV/2011/032 for the year 2012.

CONFLICT OF INTEREST

Declared none.

REFERENCES

- [1] Stiefel EL, Scheifele G. *Linear and Regular Celestial Mechanics*. Berlin-Heldelberg-New York: Springer 1971
- [2] Ferrándiz JM. A general canonical transformation increasing the number of variables with application to the two-body problem. *Celestial Mech* 1998; 41: 343-57.
- [3] Stiefel EL, Bettis DG. Stabilization of Cowell's method. *Numer Math* 1969; 13: 154-75.
- [4] Bettis DG. Numerical integration of products of Fourier and ordinary polynomials. *Numer Math* 1970a; 14: 421-34.
- [5] Bettis DG. Stabilization of finite difference methods of numerical integration. *Celestial Mech* 1970b; 2: 282-95.
- [6] Scheifele G. On numerical integration of perturbed linear oscillating systems. *Z Angew Math Phys* 1971; 22: 186-210.
- [7] Darvishi MT, Khani F, Soliman AA. The numerical simulation for stiff systems of ordinary differential equations. *Comput Math Appl* 2007; 54(7-8): 1055-63.
- [8] He JH. Variational iteration method – a kind of non – linear analytical technique: Some examples. *Int J Non Linear Mech* 1998; 34(4): 699-708.
- [9] He JH. Some asymptotic methods for strongly nonlinear equations. *Int J Mod Phys B* 2006; 20(10): 1141-99.
- [10] Deuffhard P. A study of extrapolation methods based on multistep schemes without parasitic solutions. *Z Angew Math Phys* 1979; 30: 177-89.
- [11] Neta B, Ford CH. Families of methods for the ordinary differential equations based on trigonometric polynomials. *J Comput Appl Math* 1984; 10: 33-8.
- [12] Denk G. A new numerical method for the integration of highly oscillatory second-order ordinary differential equations. *Appl Numer Math* 1993; 13: 57-67.
- [13] Martín P, Ferrándiz JM. Multistep numerical methods based on Scheifele *G*-functions with application to satellite dynamics. *SIAM J Numer Anal* 1997; 34: 359-75.
- [14] Vigo-Aguiar J, Ferrándiz JM. A general procedure for the adaptation of multistep algorithms to the integration of oscillatory problems. *SIAM J Numer Anal* 1998a; 35(4): 1684-708.
- [15] Vigo-Aguiar J, Ferrándiz JM. VSVO adapted multistep methods for the numerical integration of second order differential equations. *Appl Math Lett* 1998b; 11: 83-9.
- [16] Vigo-Aguiar J, Ferrándiz JM. Higher-order variable-step algorithms adapted to the accurate numerical integration of perturbed oscillators. *Comput Phys* 1998c; 12: 467-70.
- [17] Vigo-Aguiar J, Andrés-Pérez F. Backwards differentiation formulae adapted to scalar linear equations. *Appl Math Lett* 2001; 14: 639-43.
- [18] De Meyer H, Vanden BG, Vanthourout J. Modified backward differentiation methods of the Adams-type based on exponential interpolation. *Comput Math Appl* 1991; 2-3: 171-9.
- [19] Ixaru LG, Vanden BG, De Meyer H. Exponentially fitted variable two-step BDF algorithm for first order ODE's. *Comput Phys Commun* 2003; 100: 56-70.
- [20] Vanden Berghe G, Ixaru LG, Van Daele M. Optimal implicit exponentially-fitted Runge-Kutta methods. *Comput Phys Commun* 2001; 140: 346-57.
- [21] Reyes JA, García-Alonso F, Ferrándiz JM, Vigo-Aguiar J. Numeric multistep variable methods for perturbed linear system integration. *Appl Math Comput* 2007; 190: 63-79.
- [22] García AF, Reyes JA, Ferrándiz JM, Vigo-Aguiar J. Accurate numerical integration of perturbed oscillatory Systems in two frequencies. *ACM Trans Math Softw* 2009a; 36 (4): article 21.
- [23] García-Alonso F, Reyes JA, Ferrándiz JM, Vigo-Aguiar J. Multistep numerical methods for the integration of oscillatory problems in several frequencies. *Adv Eng Softw* 2009b; 40(8): 543-53.
- [24] Martín P, Ferrándiz JM. Behaviour of the SMF method for the numerical integration of satellite orbits. *Celest Mech Dyn Astron* 1995; 63: 29-40.
- [25] García-Alonso F, Reyes JA, Villacampa Y. Integration of highly oscillatory problems through *G*-functions. *Open Appl Math J* 2007; 1: 9-14.
- [26] Lambert JD. *Numerical methods for ordinary differential systems*. New York: John Willey and Sons Ltd. 1991.
- [27] MacDonald IG. *Symmetric Functions and Hall Polynomials*. New York: Oxford University Press Inc. 1998.
- [28] Milne-Thomson LM. *The calculus of finite differences*. New York: The Macmillan Press 1981.
- [29] Vigo-Aguiar J. An approach to variable coefficients methods for special differential equations. *Int J Appl Math* 1999; 1(8): 911-21.
- [30] Ixaru LG, Vanden BG, De Meyer H. Frequency evaluation in exponential fitting multi-step algorithm for ODE's. *J Comput Appl Math* 2002; 140: 423-34.
- [31] Van de Vyver H. Two-step hybrid methods adapted to the numerical integration of perturbed oscillators, arXiv:math/0612637v1 [math.NA]. 21 Dec 2006.
- [32] Petzold LR. An efficient numerical method for highly oscillatory ordinary differential equations. *SIAM J Numer Anal* 1981; 18(3): 455-79.
- [33] Palacios M. Métodos multirevolución simétricos para propagación de órbitas en intervalos grandes de tiempo. *Monografías de la Real Academia de Ciencias de Zaragoza* 2003; 22: 55-66.

Received: May 8, 2012

Revised: June 8, 2012

Accepted: June 11, 2012

© Reyes and García-Alonso; Licensee *Bentham Open*.

This is an open access article licensed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted, non-commercial use, distribution and reproduction in any medium, provided the work is properly cited.