

José A. Reyes, Mónica Cortés-Molina, and Fernando García-Alonso\*

# Multistep variable methods for exact integration of perturbed stiff linear systems

**Abstract:** A family of real and analytical functions with values within the ring of  $M(m, R)$  is introduced. The solution for linear systems of differential equations is expressed as a series of  $\Phi$ -functions. This new multistep method is defined for variable-step and variable-order, maintains the good properties of the  $\Phi$ -function series method. It incorporates to compute the coefficients of the algorithm a recurrent algebraic procedure, based in the existing relation between the divided differences and the elemental and complete symmetrical functions. In addition, under certain hypotheses, the new multistep method calculates the exact solution of the perturbed problem.

The new method is implemented in a computational algorithm which enables us to resolve in a general manner some physics and engineering IVP's modeled by means systems of differential equations. The good behaviour and precision of the method is evidenced by contrasting the results with other-reputed algorithms and even with methods based on Scheifele's  $G$ -functions.

**Keywords:** numerical solutions for perturbed linear systems of ODE's, perturbed stiff linear systems of ODE's, multistep methods, perturbed oscillators

**MSC 2010:** 65L04, 65L05, 65L06

DOI: 10.1515/jnma-2013-1001

Received March 25, 2013; accepted December 2, 2014

## 1 Introduction

Perturbed harmonic oscillators are of particular interest in many areas of physics and engineering. They are also of great interest in Astrodynamics, as newtonian equations of motion can be reduced to harmonic oscillators by means of the Kustaanheimo–Stiefel [19] and Burdet–Ferrándiz [2] transformations. In addition, all natural phenomena, which can be modelled using perturbed oscillators, permit a model using perturbed differential linear equation systems.

The numerical methods used to solve these kinds of systems should preferably have the property that, if the terms of perturbation disappear from the independent variable  $t$  at an arbitrary moment, the numerical method then integrates the non-perturbed system without any discretisation error [19]. Methods with this desirable property are found in [3, 5, 10, 11, 15, 16].

In [4] the  $\Phi$ -function series method is presented, for the precise integration of perturbed linear differential equations. Compared with the  $G$ -function series [16], this method has the advantage of integrating the perturbed problem without any discretisation error, under certain conditions and with only the first two terms of the series. It also maintains the desirable properties of the Scheifele methods.

The  $\Phi$ -function series method is extremely precise. However, it is difficult to calculate coefficients for each specific case it is applied to, which makes it difficult to implement on a computer.

In order to solve this problem, this article describes a variable-step, variable-order multistep method (VSVO), which maintains the good properties of the  $\Phi$ -function series method and incorporates an algebraic procedure to compute the coefficients of the algorithm.

---

**José A. Reyes, Mónica Cortés-Molina:** Department of Applied Mathematics, University of Alicante, Carretera de San Vicente del Raspeig s/n Ap 99-E-03080, Alicante, Spain

**\*Corresponding Author: Fernando García-Alonso:** Department of Applied Mathematics, University of Alicante, Carretera de San Vicente del Raspeig s/n Ap 99-E-03080, Alicante, Spain. Email: fernando.garcia@ua.es

This multistep method is obtained by approximating the perturbation function derivatives, which appear in the series method, using divided differences. This allows us to establish a recurrent calculation procedure to compute the coefficients of the algorithm. This recurrent procedure is based on the relationship existing between the divided differences of the perturbation function and the elementary and complete symmetric functions.

Both explicit and implicit multistep VSVO algorithms are constructed and, from these, a predictor-corrector algorithm. A computational algorithm is also designed to implement the method on a computer.

The excellent behaviour of the method can be seen when it is applied to stiff and highly oscillatory problems, comparing the numerical results obtained with those calculated by other well-known integrators and even with methods based on Scheifele's G-functions.

## 2 Basic ideas and formulations

Let us consider the following IVP:

$$\begin{aligned} \mathbf{x}' + A \mathbf{x} &= \varepsilon \cdot \mathbf{f}(\mathbf{x}(t), t) \\ \mathbf{x}(0) &= \mathbf{x}_0, \quad t \in [a, b] = I \end{aligned} \quad (2.1)$$

where  $\mathbf{x} : R \rightarrow R^m$ ,  $A \in M(m, R)$  and  $\mathbf{f} : R^m \times R \rightarrow R^m$ . The components of the vector perturbation field  $\mathbf{f}(\mathbf{x}(t), t)$  are  $f_i(\mathbf{x}(t), t)$  with  $i = 1, \dots, m$  and the field is continuous, with continuous derivatives until a certain order that satisfies the conditions for existence and uniqueness of solution. This type of system is called a perturbed linear system.

Assuming that  $\mathbf{g}(t) = \mathbf{f}(\mathbf{x}(t), t)$  is analytical in  $I$  with regard to  $t$ , where it is sufficient that  $\mathbf{f}$  is analytical in its arguments. In terms of the linear operator derivation  $D$ , with respect to the variable  $t$ , (2.1) can be written as follows:

$$\begin{aligned} (D + A)\mathbf{x} &= \varepsilon \cdot \mathbf{g}(t) \\ \mathbf{x}(0) &= \mathbf{x}_0, \quad t \in [0, T] = I \end{aligned} \quad (2.2)$$

for which it is supposed that  $\mathbf{x}(t)$  will be the only solution, in  $I$ , which can be developed in a power series.

Applying the operator  $(D + B)$  to (2.2), where  $B \in M(m, R)$ , and noting  $L_2 = D^2 + (A + B)D + BA$ , the new IVP is obtained:

$$\begin{aligned} L_2(\mathbf{x}) &= (D + B)\varepsilon\mathbf{g}(t) \\ \mathbf{x}(0) &= \mathbf{x}_0, \quad \mathbf{x}'(0) = -A\mathbf{x}_0 + \varepsilon\mathbf{g}(0) = \mathbf{x}'_0 \end{aligned} \quad (2.3)$$

whose exact solution  $\mathbf{x}(t)$  is the same as that of (2.1) and (2.2).

The idea that leads us to consider this 'enlarged' IVP, is that of cancelling the perturbation with the operator  $(D + B)$ .

Given that  $\mathbf{g}(t)$  is analytical in its arguments, we can write

$$\mathbf{g}(t) = \mathbf{f}(\mathbf{x}(t), t) = \sum_{n=0}^{\infty} \frac{\mathbf{g}^{(n)}(0)}{n!} t^n = \sum_{n=0}^{\infty} \frac{t^n}{n!} \mathbf{c}_n \quad (2.4)$$

with [4]

$$\begin{aligned} L_2(\mathbf{x}) &= \varepsilon \sum_{n=1}^{\infty} \frac{t^n}{n!} (\mathbf{c}_{n+1} + B\mathbf{c}_n) \\ \mathbf{x}(0) &= \mathbf{x}_0, \quad \mathbf{x}'(0) = -A\mathbf{x}_0 + \varepsilon\mathbf{g}(0) = \mathbf{x}'_0. \end{aligned} \quad (2.5)$$

The solution of the IVP (2.5) is obtained by adding a specific unperturbed IVP solution with null initial conditions to the general solution of the perturbed IVP with given initial conditions. The former can be obtained by resolving the following specific IVPs:

$$\begin{aligned} X_j'' + (A + B)X_j' + BAX_j &= \frac{t^j}{j!} I_m, \quad j = 0, 1, \dots \\ X_j(0) &= \underline{0}, \quad X_j'(0) = \underline{0} \end{aligned} \quad (2.6)$$

where  $X_j$  is a real function with values in the ring  $M(m, R)$  of the squared matrices of order  $m$ , with  $I_m$  and  $\underline{0}$  being, respectively, the unit and neutral elements of said ring. The solutions of (2.6) are the so-called  $\Phi$ -functions.

**Definition 2.1.**

$$\Phi_{j+2}(t) = X_j(t), \quad j \geq 0, \quad j \in \mathbb{N}. \tag{2.7}$$

**Proposition 2.1** (law of derivation). The  $\Phi$ -functions verify:

$$\Phi'_j(t) = \Phi_{j-1}(t), \quad j \geq 3, \quad j \in \mathbb{N}. \tag{2.8}$$

**Proposition 2.2** (law of recurrence). The  $\Phi$ -functions verify the following recurrence law:

$$\Phi_{j-2}(t) + (A + B)\Phi_{j-1}(t) + BA\Phi_j(t) = \frac{t^{j-2}}{(j-2)!} I_m, \quad j \geq 4, \quad j \in \mathbb{N}. \tag{2.9}$$

In order to complete the construction of the  $\Phi$ -functions, given in (2.7), are defined  $\Phi_0(t)$  and  $\Phi_1(t)$ .

**Definition 2.2.**  $\Phi_0(t)$  and  $\Phi_1(t)$ , are respectively, the solutions of the following IVP

$$X''(t) + (A + B)X'_t + BAX(t) = \underline{0}, \quad X(0) = I_m, \quad X'(0) = \underline{0} \tag{2.10}$$

$$X''(t) + (A + B)X'_t + BAX(t) = \underline{0}, \quad X(0) = \underline{0}, \quad X'(0) = I_m. \tag{2.11}$$

The law of derivation presented in Proposition 2.1, is completed by the proposition below.

**Proposition 2.3.**

$$\Phi'_2(t) = \Phi_1(t).$$

**Theorem 2.1.** *The solution of the IVP*

$$\begin{aligned} L_2(\mathbf{x}) &= \underline{0} \\ \mathbf{x}(0) &= \mathbf{x}_0, \quad \mathbf{x}'(0) = -A\mathbf{x}_0 + \varepsilon \mathbf{g}(0) = \mathbf{x}'_0 \end{aligned} \tag{2.12}$$

is [4]

$$\Phi_0(t)\mathbf{x}_0 + \Phi_1(t)\mathbf{x}'_0. \tag{2.13}$$

**Theorem 2.2.** *The solution of the IVP (2.5), in terms of  $\Phi$ -functions [4], is given by*

$$\mathbf{x}(t) = \Phi_0(t)\mathbf{x}_0 + \Phi_1(t)\mathbf{x}'_0 + \varepsilon \sum_{n=2}^{\infty} \Phi_n(t) (\mathbf{c}_{n-1} + B\mathbf{c}_{n-2}). \tag{2.14}$$

**Proposition 2.4** (truncation error). Carrying out a truncation of  $m + 1$   $\Phi$ -functions, with  $m \geq 2$ ,

$$\mathbf{x}_m(t) = \Phi_0(t)\mathbf{x}_0 + \Phi_1(t)\mathbf{x}'_0 + \varepsilon \sum_{n=0}^{m-2} \Phi_{n+2}(t) (\mathbf{c}_{n+1} + B\mathbf{c}_n)$$

the truncation error corresponding to  $\mathbf{x}_m(t)$ , shall be given by [4]

$$E_m(t) = \varepsilon \sum_{n=m-1}^{\infty} \frac{t^n}{n!} (\mathbf{c}_{n+1} + B\mathbf{c}_n). \tag{2.15}$$

As a result the truncation error is small with  $\varepsilon$ . If  $\varepsilon = 0$ , that is, if the perturbation disappear in an arbitrary instant of the independent variable  $t$ , the  $\Phi$ -functions integrates without discretisation error (2.5).

The above results and their proofs are presented in detail in [4].

### 3 Multistep methods

The series method described in [4] is very precise, however there is a difficulty adapting it to each specific problem. To solve this difficulty, we will proceed to describe the conversion of the series method to a multistep method similar to the SMF [10, 11] and EI<sub>p</sub>PC [15], VSVO [26] methods, which has the advantage of precisely integrating the perturbed problem under certain conditions.

Denoting by  $t_n = t_{n-1} + h_n$ , with  $n = 1, 2, \dots$ , an approach to the solution  $\mathbf{x}(t)$ , in the point  $t = t_1$ , ie  $\mathbf{x}_1 = \mathbf{x}(t_1)$  is given by:

$$\mathbf{x}_1 = \Phi_0(h_1)\mathbf{x}_0 + \Phi_1(h_1)\mathbf{x}'_0 + \varepsilon \sum_{n=0}^{m-2} \Phi_{n+2}(h_1) (\mathbf{c}_{n+1} + B\mathbf{c}_n). \quad (3.1)$$

Let us suppose that we have calculated an approximation to the solution  $\mathbf{x}(t)$  and its derivative  $\mathbf{x}'(t)$  in the point  $t = t_n$ , we shall call these approximations  $\mathbf{x}_n$  and  $\mathbf{x}'_n$ , respectively.

As

$$\begin{aligned} L_2(\mathbf{x}(t)) &= (D + B)\varepsilon \mathbf{g}(t) \\ \mathbf{x}(t_n) &= \mathbf{x}_n, \quad \mathbf{x}'(t_n) = \mathbf{x}'_n = -A\mathbf{x}_n + \varepsilon \mathbf{g}(t_n) \end{aligned} \quad (3.2)$$

to calculate an approximation to the solution at the point  $t_{n+1}$ , the change was made to the independent variable  $t = \tau + t_n$ , becoming (3.2) in the system:

$$\begin{aligned} L_2(\mathbf{x}(\tau)) &= (D + B)\varepsilon \mathbf{g}(\tau) \\ \mathbf{x}(0) &= \mathbf{x}_n, \quad \mathbf{x}'(0) = \mathbf{x}'_n. \end{aligned} \quad (3.3)$$

Calculating the expansion coefficients

$$\mathbf{f}(\mathbf{x}(\tau), \tau + t_n) = \mathbf{g}(\tau) = \sum_{k=0}^{\infty} \frac{\tau^k}{k!} \mathbf{q}_k \quad (3.4)$$

where

$$\mathbf{q}_k = \frac{d^k \mathbf{g}(0)}{d\tau^k} = \frac{d^k \mathbf{g}(t_n)}{dt^k} = \left( g_1^{(k)}(t_n), \dots, g_m^{(k)}(t_n) \right)^t. \quad (3.5)$$

The approximation to the solution in point  $t_{n+1} = (n + 1)h$ , is given by

$$\mathbf{x}_{n+1} = \Phi_0(h_{n+1})\mathbf{x}_n + \Phi_1(h_{n+1})\mathbf{x}'_n + \varepsilon \sum_{k=0}^{m-2} \Phi_{k+2}(h_{n+1}) (\mathbf{q}_{k+1} + B\mathbf{q}_k). \quad (3.6)$$

#### 3.1 Explicit multistep method MDFSpE, for perturbed systems

In order to obtain an explicit method the derivatives of the perturbation function are substituted by divided differences.

To make a variable step explicit multistep method of  $p$ -step, series

$$\sum_{k=0}^{\infty} \Phi_k(t) (\mathbf{q}_{k+1} + B\mathbf{q}_k)$$

is truncated, such that the higher order derivative is  $(p - 1)$ , i.e,  $\mathbf{q}_{p-1} = \mathbf{g}^{(p-1)}(t_n)$ , so that:

$$\begin{aligned} \mathbf{x}_{n+1} &= (\Phi_0(h_{n+1}) - \Phi_1(h_{n+1})A) \mathbf{x}_n + \varepsilon \Phi_1(h_{n+1})\mathbf{q}_0 \\ &+ \varepsilon \sum_{k=0}^{p-2} \Phi_{k+2}(h) (\mathbf{q}_{k+1} + B\mathbf{q}_k) + \varepsilon \Phi_{p+1}(h_{n+1})B\mathbf{q}_{p-1}. \end{aligned} \quad (3.7)$$

Properly rearranging (3.7), we obtain:

$$\mathbf{x}_{n+1} = (\Phi_0(h_{n+1}) - \Phi_1(h_{n+1})A) \mathbf{x}_n + \varepsilon \sum_{k=0}^{p-1} (\Phi_{k+1}(h_{n+1}) + \Phi_{k+2}(h_{n+1})B) \mathbf{q}_k. \quad (3.8)$$

Therefore it would be necessary to use divided differences  $(p - 1)$ -th order of each of the component fields of the function  $\mathbf{g}$ , in the values  $t_n, \dots, t_{n-p+1}$ , that is  $g_i[t_n, \dots, t_{n-p+1}]$  function with  $i = 1, \dots, m$ .

As the divided differences of an arbitrary function  $g(t)$ , satisfy the identity

$$g[t_n, \dots, t_{n-i}] = \sum_{j=0}^{\infty} P_j[0, -H_1, \dots, -H_i] \cdot g^{(j)}(t_n) \tag{3.9}$$

with  $P_k(t) = t^k/k!$  and  $H_i = t_n - t_{n-i}$  [15, 25].

Denoting by

$$D_{p,n} = ((j - 1)!g_i[t_n, \dots, t_{n-j+1}])_{j=1, \dots, p}^{i=1, \dots, m} \tag{3.10}$$

the next matrix, with  $m \times p$  order and  $H = \max \{H_1, \dots, H_{p-1}\}$ , it verify the identity

$$D_{p,n}^t = A_p \left( g_j^{(i-1)}(t_n) \right)_{j=1, \dots, m}^{i=1, \dots, p} + O_{p \times m} \tag{3.11}$$

where  $O_{p \times m}$  is a matrix of order  $p \times m$ , whose  $i$ -th row is

$$(O(H^{p-i+1}) \dots O(H^{p-i+1}))$$

and  $A_p$  is the non singular matrix of order  $p$ :

$$A_p = \begin{pmatrix} 1 & P_1[0] & P_2[0] & \dots & P_{p-1}[0] \\ 0 & 1 & 1!P_2[0, -H_1] & \dots & 1!P_{p-1}[0, -H_1] \\ 0 & 0 & 1 & \dots & 2!P_{n-1}[0, -H_1, -H_2] \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{pmatrix}_{p \times p} \tag{3.12}$$

Using a more compact notation

$$D_{p,n}^t = A_p \times Z_{p \times m} + O_{p \times m}, \quad Z_{p \times m} = \left( g_j^{(i-1)}(t_n) \right)_{j=1, \dots, m}^{i=1, \dots, p} \tag{3.13}$$

Truncating the expansion obtained and solving  $Z_{p \times m}$  it results in:

$$Z_{p \times m} = A_p^{-1} \times (D_{p,n}^t)_{p \times m} \tag{3.14}$$

replacing in (3.8) the derivatives of the components fields of the perturbed function, we can write

$$\mathbf{x}_{n+1} = (\Phi_0(h_{n+1}) - \Phi_1(h_{n+1})A) \mathbf{x}_n + \varepsilon \sum_{k=0}^{p-1} (\Phi_{k+1}(h_{n+1}) + \Phi_{k+2}(h_{n+1})B) \mathbf{p}_{k+1} \tag{3.15}$$

being  $\mathbf{p}_j$  with  $j = 1, \dots, p$  the  $j$ -th. column of the matrix  $D_{p,n} \times A_p^{-t}$ .

Let  $(d_{ij})_{j=1, \dots, p}^{i=1, \dots, p}$  be the matrix  $A_p^{-t}$ , then

$$\mathbf{p}_j = \begin{pmatrix} \sum_{i=1}^p g_1[t_n, \dots, t_{n-(i-1)}] d_{ij}(i - 1)! \\ \sum_{i=1}^p g_2[t_n, \dots, t_{n-(i-1)}] d_{ij}(i - 1)! \\ \vdots \\ \sum_{i=1}^p g_m[t_n, \dots, t_{n-(i-1)}] d_{ij}(i - 1)! \end{pmatrix} \tag{3.16}$$

which substituted in  $\sum_{k=0}^{p-1} (\Phi_{k+1}(h_{n+1}) + \Phi_{k+2}(h_{n+1})B) \mathbf{p}_{k+1}$  allow us to write

$$\sum_{k=0}^{p-1} (\Phi_{k+1}(h_{n+1}) + \Phi_{k+2}(h_{n+1})B) \mathbf{p}_{k+1} = \sum_{i=1}^p \left( \sum_{k=0}^{p-1} (i-1)! d_{i,k+1} (\Phi_{k+1}(h_{n+1}) + \Phi_{k+2}(h_{n+1})B) \begin{pmatrix} \mathbf{g}_1[t_n, \dots, t_{n-(i-1)}] \\ \mathbf{g}_2[t_n, \dots, t_{n-(i-1)}] \\ \vdots \\ \mathbf{g}_m[t_n, \dots, t_{n-(i-1)}] \end{pmatrix} \right). \quad (3.17)$$

Defining

$$\Lambda_i = \sum_{k=0}^{p-1} (i-1)! d_{i,k+1} (\Phi_{k+1}(h_{n+1}) + \Phi_{k+2}(h_{n+1})B), \quad i = 1, \dots, p \quad (3.18)$$

we obtain

$$\sum_{k=0}^{p-1} (\Phi_{k+1}(h_{n+1}) + \Phi_{k+2}(h_{n+1})B) \mathbf{p}_{k+1} = \sum_{i=1}^p \Lambda_i \begin{pmatrix} \mathbf{g}_1[t_n, \dots, t_{n-(i-1)}] \\ \mathbf{g}_2[t_n, \dots, t_{n-(i-1)}] \\ \vdots \\ \mathbf{g}_m[t_n, \dots, t_{n-(i-1)}] \end{pmatrix}. \quad (3.19)$$

Replacing (3.19) in (3.15), we obtain the next formula for a explicit multistep method

$$\mathbf{x}_{n+1} = (\Phi_0(h_{n+1}) - \Phi_1(h_{n+1})A) \mathbf{x}_n + \varepsilon \sum_{i=1}^p \Lambda_i \begin{pmatrix} \mathbf{g}_1[t_n, \dots, t_{n-(i-1)}] \\ \mathbf{g}_2[t_n, \dots, t_{n-(i-1)}] \\ \vdots \\ \mathbf{g}_m[t_n, \dots, t_{n-(i-1)}] \end{pmatrix}. \quad (3.20)$$

We introduce the next notation, we obtain the next definition.

**Definition 3.1.** Let  $\mathbf{x}_n$  be the approximation to the value of the solution at the point  $t_n$ , and let

$$\Lambda_i = \sum_{k=0}^{p-1} (i-1)! d_{i,k+1} (\Phi_{k+1}(h_{n+1}) + \Phi_{k+2}(h_{n+1})B), \quad (d_{ij})_{j=1, \dots, p}^{i=1, \dots, p} = A_p^{-t}$$

$$F_{n,i}^t = (\mathbf{g}_1[t_n, \dots, t_{n-(i-1)}], \dots, \mathbf{g}_m[t_n, \dots, t_{n-(i-1)}]).$$

The starting values for  $\mathbf{x}$  are:  $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \dots, \mathbf{x}_{p-1}$ .

The explicit method MDFSpE, variable step size of  $p$  step, for perturbed linear systems, it is formulated through the next equation:

$$\mathbf{x}_{n+1} = (\Phi_0(h_{n+1}) - \Phi_1(h_{n+1})A) \mathbf{x}_n + \varepsilon \sum_{i=1}^p \Lambda_i F_{n,i}^t, \quad n \geq p-1.$$

### 3.2 Implicit multistep method MDFSpl, for perturbed systems

In order to obtain an implicit method of  $p$  steps, we will use the same idea as in the previous section, the  $\sum_{k=0}^{\infty} \Phi_k(t)(\mathbf{q}_{k+1} + B\mathbf{q}_k)$  series is truncated, such that the higher order derivative is  $p$ , i.e,  $\mathbf{q}_p = \mathbf{g}^{(p)}(t_n)$ , to correspond with the latest  $\mathbf{g}[t_{n+1}, \dots, t_{n+1-p}]$  divided difference, so that:

$$\mathbf{x}_{n+1} = (\Phi_0(h_{n+1}) - \Phi_1(h_{n+1})A) \mathbf{x}_n + \varepsilon \Phi_1(h_{n+1})\mathbf{q}_0 + \varepsilon \sum_{k=0}^{p-1} \Phi_{k+2}(h_{n+1})(\mathbf{q}_{k+1} + B\mathbf{q}_k) + \varepsilon \Phi_{p+2}(h)B\mathbf{q}_p \quad (3.21)$$

rearrange (3.21), we obtain:

$$\mathbf{x}_{n+1} = (\Phi_0(h_{n+1}) - \Phi_1(h_{n+1})A) \mathbf{x}_n + \varepsilon \sum_{k=0}^p (\Phi_{k+1}(h_{n+1}) + \Phi_{k+2}(h_{n+1})B) \mathbf{q}_k. \quad (3.22)$$

It will be necessary to use the divided differences of order  $p$ -th of each of the fields of components of the function of  $\mathbf{g}$ , in the values  $t_{n+1}, \dots, t_{n+1-p}$ .

Given that  $h_{n+1} = t_{n+1} - t_n$ , it is defined

$$H_i = t_n - t_{n-i}. \quad (3.23)$$

Denoting by

$$D_{p,n+1} = \left( (j-1)! g_i[t_{n+1}, \dots, t_{n-j+2}] \right)_{j=1, \dots, p+1}^{i=1, \dots, m} \quad (3.24)$$

the next matrix, with  $m \times (p+1)$  order, and  $H = \max\{h_{n+1}, H_1, \dots, H_{p-1}\}$ , it verify the identity

$$D_{p,n+1}^t = B_p \left( g_j^{(i-1)}(t_n) \right)_{j=1, \dots, m}^{i=1, \dots, p+1} + O_{(p+1) \times m} \quad (3.25)$$

where  $O_{(p+1) \times m}$  is a matrix of order  $(p+1) \times m$ , whose  $i$ -th row is

$$(O(H^{p-i+2}) \dots O(H^{p-i+2}))$$

and  $B_p$  is the non singular matrix of order  $p+1$

$$B_p = \begin{pmatrix} 1 & P_1[h_{n+1}] & P_2[h_{n+1}] & \dots & P_p[h_{n+1}] \\ 0 & 1 & 1!P_2[h_{n+1}, 0] & \dots & 1!P_p[h_{n+1}, 0] \\ 0 & 0 & 1 & \dots & 2!P_p[h_{n+1}, 0, -H_1] \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{pmatrix}_{(p+1) \times (p+1)}. \quad (3.26)$$

Using a more compact notation:

$$D_{p,n+1}^t = B_p \times Z_{(p+1) \times m} + O_{(p+1) \times m}, \quad Z_{(p+1) \times m} = \left( g_j^{(i-1)}(t_n) \right)_{j=1, \dots, m}^{i=1, \dots, p+1}. \quad (3.27)$$

Truncating the expansion obtained and solving for  $Z_{(p+1) \times m}$  results in:

$$Z_{(p+1) \times m} = B_p^{-1} \times (D_{p,n+1}^t)_{(p+1) \times m} \quad (3.28)$$

replacing in (3.22) the derivatives of the components fields of the perturbed function, we can write

$$\mathbf{x}_{n+1} = (\Phi_0(h_{n+1}) - \Phi_1(h_{n+1})A) \mathbf{x}_n + \varepsilon \sum_{k=0}^p (\Phi_{k+1}(h_{n+1}) + \Phi_{k+2}(h_{n+1})B) \mathbf{p}_{k+1} \quad (3.29)$$

being  $\mathbf{p}_j$  with  $j = 1, \dots, p+1$  the  $j$ -th column of the matrix  $D_{p,n+1} \times B_p^{-t}$ .

Let  $(d_{ij})_{j=1, \dots, (p+1)}^{i=1, \dots, (p+1)}$  be the matrix  $B_p^{-t}$ , then

$$\mathbf{p}_j = \begin{pmatrix} \sum_{i=1}^{p+1} g_1[t_{n+1}, \dots, t_{n+1-(i-1)}] d_{ij}(i-1)! \\ \sum_{i=1}^{p+1} g_2[t_{n+1}, \dots, t_{n+1-(i-1)}] d_{ij}(i-1)! \\ \vdots \\ \sum_{i=1}^{p+1} g_m[t_{n+1}, \dots, t_{n+1-(i-1)}] d_{ij}(i-1)! \end{pmatrix} \quad (3.30)$$

which substituted in  $\sum_{k=0}^p (\Phi_{k+1}(h_{n+1}) + \Phi_{k+2}(h_{n+1})B) \mathbf{p}_{k+1}$  allow us to write:

$$\begin{aligned} \sum_{k=0}^p (\Phi_{k+1}(h_{n+1}) + \Phi_{k+2}(h_{n+1})B) \mathbf{p}_{k+1} &= \sum_{i=1}^{p+1} \left( \sum_{k=0}^p (i-1)! d_{i,k+1} (\Phi_{k+1}(h_{n+1}) \right. \\ &\quad \left. + \Phi_{k+2}(h_{n+1})B) \begin{pmatrix} g_1[t_{n+1}, \dots, t_{n+1-(i-1)}] \\ g_2[t_{n+1}, \dots, t_{n+1-(i-1)}] \\ \vdots \\ g_m[t_{n+1}, \dots, t_{n+1-(i-1)}] \end{pmatrix} \right). \end{aligned} \quad (3.31)$$

Defining

$$\Gamma_i = \sum_{k=0}^p (i-1)! d_{i,k+1} (\Phi_{k+1}(h_{n+1}) + \Phi_{k+2}(h_{n+1})B), \quad i = 1, \dots, p+1 \quad (3.32)$$

we obtain

$$\sum_{k=0}^p (\Phi_{k+1}(h_{n+1}) + \Phi_{k+2}(h_{n+1})B) \mathbf{p}_{k+1} = \sum_{i=1}^{p+1} \Gamma_i \begin{pmatrix} g_1[t_{n+1}, \dots, t_{n+1-(i-1)}] \\ g_2[t_{n+1}, \dots, t_{n+1-(i-1)}] \\ \vdots \\ g_m[t_{n+1}, \dots, t_{n+1-(i-1)}] \end{pmatrix}. \quad (3.33)$$

Replacing (3.33) in (3.29), we obtain the next formula for a implicit multistep method:

$$\mathbf{x}_{n+1} = (\Phi_0(h_{n+1}) - \Phi_1(h_{n+1})A) \mathbf{x}_n + \varepsilon \sum_{i=1}^{p+1} \Gamma_i \begin{pmatrix} g_1[t_{n+1}, \dots, t_{n+1-(i-1)}] \\ g_2[t_{n+1}, \dots, t_{n+1-(i-1)}] \\ \vdots \\ g_m[t_{n+1}, \dots, t_{n+1-(i-1)}] \end{pmatrix}. \quad (3.34)$$

The next notation is introduced, giving rise to the following definition.

**Definition 3.2.** Let  $\mathbf{x}_n$  be the approximation to the value of the solution at the point  $t_n$ , and let

$$\Gamma_i = \sum_{k=0}^p (i-1)! d_{i,k+1} (\Phi_{k+1}(h_{n+1}) + \Phi_{k+2}(h_{n+1})B), \quad B_p^{-t} = (d_{ij})_{j=1, \dots, (p+1)}^{i=1, \dots, (p+1)}$$

$$T_{n,i}^t = (g_1[t_{n+1}, \dots, t_{n+1-(i-1)}], \dots, g_m[t_{n+1}, \dots, t_{n+1-(i-1)}]).$$

The starting values for  $\mathbf{x}$  are:  $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \dots, \mathbf{x}_{p-1}$ .

The implicit method MDFSpI, variable step size of  $p$  step, for perturbed linear systems, it is formulated through the next equation:

$$\mathbf{x}_{n+1} = (\Phi_0(h_{n+1}) - \Phi_1(h_{n+1})A) \mathbf{x}_n + \varepsilon \sum_{i=1}^{p+1} \Gamma_i T_{n,i}, \quad n \geq p-1.$$

### 3.3 Predictor-corrector multistep method MDFSpPC, for perturbed systems

We define the predictor-corrector method, with variable step size of  $p$  step MDFSpPC for perturbed linear systems, which has as predictor to MDFpE and as corrector to MDFSpI.

The predictor-corrector method used is like  $P(EC)^\mu E^{1-t}$  with  $\mu = t = 1$ .

## 4 Recurrent calculus of the matrices $A_p^{-t}$ , $B_p^{-t}$ and new definition the multistep methods

The next methods described in Definitions 3.1 and 3.2, present the difficulty that the coefficients of the matrices  $A_p^{-t} = (d_{ij})_{j=1, \dots, p}^{i=1, \dots, p}$  and  $B_p^{-t} = (d_{ij})_{j=1, \dots, (p+1)}^{i=1, \dots, (p+1)}$ , are not expressed in a recurrent way, which leads to difficulties in its codification to automatize its calculation.

Once this problem is resulted, the methods will be able to be codified and will enable one to chose the step size and the number of steps, that each execution requires.

The problem is then reduced to find a recurrent formula, that allows us to calculate the elements of matrices  $A_p^{-t}$  and  $B_p^{-t}$ .



#### 4.1 Recurrent calculus of $A_p^{-t}$ and new definition the multistep explicit method

To obtain the recurrent calculus of the matrix elements  $A_p^{-t} = (d_{ij})_{j=1,\dots,p}^{i=1,\dots,p}$ , allowing us to construct a variable-step, variable-order method (VSVO), we use elementary symmetric functions  $e_{n,r}$  and complete symmetric functions  $h_{n,r}$  [9, 15, 23], defined as:

$$e_{n,0} = 1, \quad e_{n,r} = \sum_{i_1 < i_2 < \dots < i_r} t_{i_1} \cdots t_{i_r}, \quad e_{n,r} = 0, \quad r < 0 \quad (4.1)$$

$$h_{n,r} = \sum_{|\alpha|=r} \sum_{S_\lambda} t^\alpha, \quad \lambda = (\lambda_1 \cdots \lambda_n) \in \mathbb{N}^n, \quad |\lambda| = \lambda_1 + \dots + \lambda_n \quad (4.2)$$

and  $S_\lambda = \{\text{all the different permutations } \alpha = (\alpha_1 \cdots \alpha_n) \text{ of } \lambda\}$  with  $t^\alpha = t_1^{\alpha_1} \cdots t_n^{\alpha_n}$ .

Particularly  $h_{n,0} = 1$  and  $h_{n,1} = e_{n,1}$ . In the case  $r < 0$ , it is defined as  $h_{n,r} = 0$ .

Between the divided differences of  $g(t) = t^m$ , that we will denoted by  $t^m[t_1, \dots, t_n]$  and the complete symmetrical polynomial the next relation holds:  $t^m[t_1, \dots, t_n] = h_{n,m-n+1}$ .

Considering the point  $t_n$ , it is defined the complete symmetric functions [15] and [23]:

$$q_{i,j}(n) = t^{j-1}[H_n, \dots, H_{n-(i-1)}], \quad \sigma_{i,j}(n) = (-1)^{j-i} e_{j-1,j-i} \quad (4.3)$$

in the values  $H_{n-k} = t_{n-k} - t^*$  with  $k = 0, \dots, i-1$  and  $t^* \in [a, b]$ .

The square matrices of order  $k$ ,

$$P_{k,n} = (q_{i,j}(n))_{j=1,\dots,k}^{i=1,\dots,k}, \quad S_{k,n} = (\sigma_{i,j}(n))_{j=1,\dots,k}^{i=1,\dots,k} \quad (4.4)$$

are inverse to each other.

As  $H_{n-j} = t_{n-j} - t^*$  and  $H_j = t_n - t_{n-j}$  we 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:

$$g[t_n, \dots, t_{n-(i-1)}] = \sum_{j=0}^{\infty} q_{i,j+1}(n) \frac{1}{j!} g^{(j)}(t^*). \quad (4.5)$$

If  $H = \max\{|H_n|, \dots, |H_{n-(i-1)}|\}$ , as  $q_{i,j}(n)$  have order  $j-i$  in  $H$ , due to the last result, we can write

$$g[t_n, t_{n-1}, \dots, t_{n-(i-1)}] = \sum_{j=0}^{p-1} q_{i,j+1}(n) \frac{1}{j!} g^{(j)}(t^*) + O(H^{p-(i-1)}), \quad i = 1, \dots, p. \quad (4.6)$$

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

$$\begin{pmatrix} g[t_n] \\ g[t_n, t_{n-1}] \\ \vdots \\ g[t_n, \dots, t_{n-(p-1)}] \end{pmatrix} = \begin{pmatrix} q_{1,1}(n) & \cdots & q_{1,p}(n) \\ q_{2,1}(n) & \cdots & q_{2,p}(n) \\ \vdots & \ddots & \vdots \\ q_{p,1}(n) & \cdots & q_{p,p}(n) \end{pmatrix} \begin{pmatrix} \frac{g(t_n)}{1!} \\ \frac{g'(t_n)}{1!} \\ \vdots \\ \frac{g^{(p-1)}(t_n)}{(p-1)!} \end{pmatrix} + \begin{pmatrix} O(H^p) \\ O(H^{p-1}) \\ \vdots \\ O(H) \end{pmatrix} \quad (4.7)$$

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

$$\begin{pmatrix} g[t_n] \\ g[t_n, t_{n-1}] \\ \vdots \\ g[t_n, \dots, t_{n-(p-1)}] \end{pmatrix} = \begin{pmatrix} 1 & h_{1,1} & \cdots & h_{1,p-1} \\ 0 & 1 & \cdots & h_{2,p-2} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} \begin{pmatrix} \frac{g(t_n)}{1!} \\ \frac{g'(t_n)}{1!} \\ \vdots \\ \frac{g^{(p-1)}(t_n)}{(p-1)!} \end{pmatrix} + \begin{pmatrix} O(H^p) \\ O(H^{p-1}) \\ \vdots \\ O(H) \end{pmatrix}. \quad (4.8)$$

As  $\sigma_{i,j}(n) = \sigma_{i-1,j-1}(n) - H_{n-j+2} \sigma_{i,j-1}(n)$  for  $i, j \geq 2$  [23], if we consider  $t^* = t_n$ , then

$$S_{p,n} = (\sigma_{ij}(n))_{j=1,\dots,p}^{i=1,\dots,p}, \quad \begin{cases} \sigma_{1,1}(n) = 1 \\ \sigma_{1,j}(n) = 0, & 1 < j \leq p \\ \sigma_{i,1}(n) = 0, & 1 < i \leq p \\ \sigma_{i,j}(n) = \sigma_{i-1,j-1}(n) - H_{n-j+2} \sigma_{i,j-1}(n), & 2 \leq i, j \leq p. \end{cases} \quad (4.9)$$

The recurrent form of the matrix  $A_p^{-t}$  is got through:

$$A_p^{-t} = M_p \times P_{p,n}^{-t} \times N_p = M_p \times S_{p,n}^t \times N_p \tag{4.10}$$

that is to say

$$d_{i,j} = \frac{(j-1)! \sigma_{j,i}(n)}{(i-1)!} \quad i, j = 1, \dots, p \tag{4.11}$$

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

The expressions (4.10) and (4.11) allow us to compute the  $A_p^{-t}$  matrix by recurrence, from  $S_{p,n}^t$  matrix.

Substituting (4.11) in Definition 3.1, we obtain the explicit method modified.

**Definition 4.1.** Let  $\mathbf{x}_n$  be the approximation to the value of the solution at the point  $t_n$ , and let

$$\Lambda_i = \sum_{k=1}^p (k-1)! \sigma_{k,i} (\Phi_k(h_{n+1}) + \Phi_{k+1}(h_{n+1})B), \quad i = 1, \dots, p$$

$$F_{n,i}^t = (g_1[t_n, \dots, t_{n-(i-1)}], \dots, g_m[t_n, \dots, t_{n-(i-1)}]).$$

The starting values for  $\mathbf{x}$  are:  $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \dots, \mathbf{x}_{p-1}$ .

The explicit method MDFSpE, variable step size of  $p$  step, for perturbed linear systems, it is formulated through the next equation:

$$\mathbf{x}_{n+1} = (\Phi_0(h_{n+1}) - \Phi_1(h_{n+1})A) \mathbf{x}_n + \varepsilon \sum_{i=1}^p \Lambda_i F_{n,i}, \quad n \geq p-1.$$

## 4.2 Recurrent calculus of $B_p^{-t}$ and new definition the multistep implicit method

As in the case above, we use elementary symmetric functions  $e_{n,r}$  and complete symmetric functions  $h_{n,r}$  to obtain the recurrent calculus matrix elements  $B_p^{-t} = (d_{ij})_{j=1, \dots, (p+1)}^{i=1, \dots, (p+1)}$ , allowing us to construct a variable-step, variable-order method (VSVO).

Taking  $h_{n+1} = t_{n+1} - t^*$ , the divided differences of one function  $g$  satisfy the property:

$$g_1[t_{n+1}, t_n, \dots, t_{n-(i-1)}] = \sum_{j=0}^{\infty} q_{i+1,j+1}(n) \frac{1}{j!} g^{(j)}(t^*). \tag{4.12}$$

If  $H = \max \{|h_{n+1}|, |H_n|, \dots, |H_{n-(i-1)}|\}$ , as  $q_{i,j}(n)$  have order  $j-i$  in  $H$ , due to the last result, we can write

$$g_1[t_{n+1}, \dots, t_{n-(i-1)}] = \sum_{j=0}^{p-1} q_{i+1,j+1}(n) \frac{1}{j!} g^{(j)}(t^*) + O(H^{p-(i-1)}), \quad i = 1, \dots, p. \tag{4.13}$$

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

$$\begin{pmatrix} g[t_{n+1}] \\ g[t_{n+1}, t_n] \\ \vdots \\ g[t_{n+1}, \dots, t_{n-(p-1)}] \end{pmatrix} = \begin{pmatrix} q_{1,1}(n) & \cdots & q_{1,p+1}(n) \\ q_{2,1}(n) & \cdots & q_{2,p+1}(n) \\ \vdots & \ddots & \vdots \\ q_{p+1,1}(n) & \cdots & q_{p+1,p+1}(n) \end{pmatrix} \begin{pmatrix} g(t_n) \\ \frac{g'(t_n)}{1!} \\ \vdots \\ \frac{g^{(p)}(t_n)}{p!} \end{pmatrix} + \begin{pmatrix} O(H^{p+1}) \\ O(H^p) \\ \vdots \\ O(H) \end{pmatrix} \tag{4.14}$$

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

$$\begin{pmatrix} g[t_{n+1}] \\ g[t_{n+1}, t_n] \\ \vdots \\ g[t_{n+1}, \dots, t_{n-(p-1)}] \end{pmatrix} = \begin{pmatrix} 1 & h_{1,1} & \cdots & h_{1,p+1} \\ 0 & 1 & \cdots & h_{2,p+1} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} \begin{pmatrix} g(t_n) \\ \frac{g'(t_n)}{1!} \\ \vdots \\ \frac{g^{(p)}(t_n)}{p!} \end{pmatrix} + \begin{pmatrix} O(H^{p+1}) \\ O(H^p) \\ \vdots \\ O(H) \end{pmatrix}. \tag{4.15}$$

As  $\sigma_{i,j}(n) = \sigma_{i-1,j-1}(n) - H_{n-j+3} \sigma_{i,j-1}(n)$  for  $i, j \geq 2$  [23], if we consider  $t^* = t_n$ , then

$$S_{p,n+1} = (\sigma_{ij}(n))_{j=1 \dots (p+1)}^{i=1 \dots (p+1)}, \quad \begin{cases} \sigma_{1,1}(n) = 1 \\ \sigma_{1,2}(n) = -h_{n+1} \\ \sigma_{1,j}(n) = 0, \quad 2 < j \leq p+1 \\ \sigma_{i,1}(n) = 0, \quad 1 < i \leq p+1 \\ \sigma_{i,j}(n) = \sigma_{i-1,j-1}(n) - H_{n-j+3} \sigma_{i,j-1}(n), \quad 2 \leq i, \quad j \leq p+1. \end{cases} \quad (4.16)$$

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

$$B_p^{-t} = M_{p+1} \times S_{p,n+1}^t \times N_{p+1} \quad (4.17)$$

that is

$$d_{i,j} = \frac{(j-1)! \sigma_{j,i}(n)}{(i-1)!}, \quad i, j = 1, \dots, p+1. \quad (4.18)$$

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

The expressions (4.17) and (4.18) allow us to compute the  $B_p^{-t}$  matrix by recurrence, from  $S_{p,n+1}^{-t}$  matrix.

Substituting (4.18) in Definition 3.2, we obtain the implicit method modified.

**Definition 4.2.** Let  $\mathbf{x}_n$  be the approximation to the value of the solution at the point  $t_n$ , and let

$$\Gamma_i = \sum_{k=0}^p k! \sigma_{k+1,i} (\Phi_{k+1}(h_{n+1}) + \Phi_{k+2}(h_{n+1})B), \quad i = 1, \dots, p+1$$

$$T_{n,i}^t = (g_1[t_{n+1}, \dots, t_{n+1-(i-1)}], \dots, g_m[t_{n+1}, \dots, t_{n+1-(i-1)}]).$$

The starting values for  $\mathbf{x}$  are:  $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \dots, \mathbf{x}_{p-1}$ .

The implicit method MDFSpI, variable step size of  $p$  step, for perturbed linear systems, it is formulated through the next equation:

$$\mathbf{x}_{n+1} = (\Phi_0(h_{n+1}) - \Phi_1(h_{n+1})A) \mathbf{x}_n + \varepsilon \sum_{i=1}^{p+1} \Gamma_i T_{n,i}, \quad n \geq p-1.$$

### 4.3 New predictor-corrector multistep method

We define the predictor-corrector method, with variable step size of  $p$  step MDFSpPC for perturbed linear systems, which has as predictor to MDFSpE and as corrector to MDFSpI, with the previous definition.

## 5 Numerical experiments

In this section we use the MDFSpPC method to solve the test problems proposed in [4], showing the validity of this VSVO method, as we can obtain similar precision to that obtained with the  $\Phi$ -function series method without having to design the recurrences in each case.

The solutions obtained using the MDFSpPC method for stiff and highly oscillatory problems are compared with those calculated using the best-known codes:

- LSODE methods, causes a numerical solution to be found using the Livermore Stiff ODE solver.
- GEAR causes a numerical solution to be found by way of a Burlirsch-Stoer rational extrapolation method.
- MGEAR [msteppart] is a multistep method suitable for stiff systems.

Using in the last ones the implementations of MAPLE to ensure that the results are not distorted by a deficient programming that favours the new code.

The results obtained using the new method are also compared with those calculated using the multistep methods based on Scheifele's  $G$ -functions, specifically with the EIpc [15].

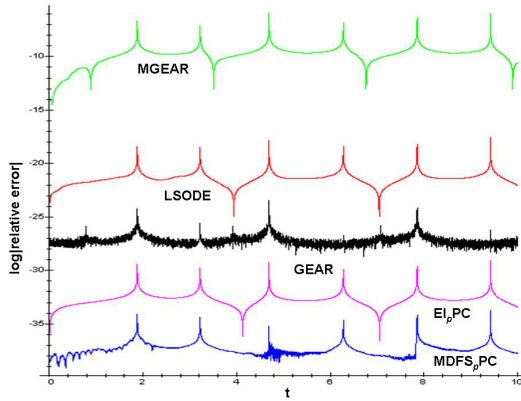


Figure 1. Problem 1, relative error.

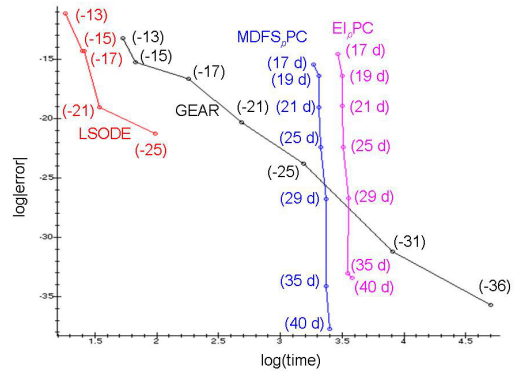


Figure 2. Problem 1, efficiency plot for the integration of  $x(t)$  at last point.

### 5.1 Problem 1

Let us consider the following stiff problem, which appears in [6, 7, 22]:

$$\begin{cases} x_1'(t) = 2x_1(t) + x_2(t) + 2 \sin(t) \\ x_2'(t) = -(\beta + 2)x_1(t) + (\beta + 1)(x_2(t) - \cos(t) + \sin(t)) \end{cases} \quad (5.1)$$

with initial conditions  $x_1(0) = 2, x_2(0) = 3$ , and exact solution, independent of  $\beta$ ,

$$\begin{aligned} x_1(t) &= 2e^{-t} + \sin(t) \\ x_2(t) &= 2e^{-t} + \cos(t). \end{aligned} \quad (5.2)$$

The eigenvalues of the system are  $-1$  and  $\beta$  which enables its degree of stiffness to be regulated. For the case  $\beta = -1000$ , the following stiff problem is obtained, proposed in [8]:

$$\begin{pmatrix} x_1'(t) \\ x_2'(t) \end{pmatrix} + \begin{pmatrix} 2 & -1 \\ -998 & 999 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} = \begin{pmatrix} 2 \sin(t) \\ 999(\cos(t) - \sin(t)) \end{pmatrix} \quad (5.3)$$

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

In this article has been resolved (5.3) by the method MDFSpPC, based on the series of  $\Phi$ -functions [4], using the matrix

$$B = \begin{pmatrix} -1 & -\frac{2}{999} \\ 999 & 1 \end{pmatrix}$$

as a matrix for annulment of the perturbation function. The IPV expanded which is obtained of applied the differential operator  $(D + B)$  to (5.3), is expressed as:

$$L_2(\mathbf{x}) = \begin{pmatrix} x_1''(t) \\ x_2''(t) \end{pmatrix} + \begin{pmatrix} 1 & -\frac{1001}{999} \\ 1 & 1000 \end{pmatrix} \begin{pmatrix} x_1'(t) \\ x_2'(t) \end{pmatrix} + \begin{pmatrix} -\frac{2}{999} & -1 \\ 1000 & 0 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad (5.4)$$

this being the problem solved by the method MDFSpPC.

The results are compared with those obtained by integrators, LSODE, MGEAR, GEAR, implemented in MAPLE and the multistep method EIppPC [15], based on the Scheifele  $G$ -functions.

In Fig. 1 contrasts the decimal logarithm of module of the relative error of the solution  $\mathbf{x}(t)$ , calculated using  $\Phi$ -functions method MDFSpPC, with step size  $h = 10^{-3}$ , 40 digits and  $p = 11$ , with the numerical integration codes MGEAR[msteppart] with errorper = Float(1, -13), LSODE with a tolerance of  $10^{-25}$ , GEAR with errorper = Float(1, -22) and EIppPC with step size  $h = 10^{-3}$ , 40 digits and  $p = 11$ .

In Fig. 2 we show an efficiency plot where  $\Phi$ -functions multistep methods are 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, 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.

We show the results of a few runs of it where the number of  $\Phi$ -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  $10^{-13}$ ,  $10^{-15}$ ,  $10^{-17}$ ,  $10^{-21}$ , and  $10^{-25}$  in LSODE and  $10^{-13}$ ,  $10^{-15}$ ,  $10^{-17}$ ,  $10^{-21}$ ,  $10^{-25}$ ,  $10^{-31}$ , and  $10^{-36}$  in GEAR. We use 13 + 4 digits, 15 + 4 digits, 17 + 4 digits, 21 + 4 digits, 25 + 4 digits, 31 + 4 digits, and 36 + 4 digits to avoid spurious increase of the computation times.

### 5.2 Problem 2

Let's consider the highly oscillatory problem proposed by Petzold [13] and [12], which contains a harmonic oscillator:

$$\begin{aligned} \begin{pmatrix} x_1'(t) \\ x_2'(t) \end{pmatrix} + \begin{pmatrix} 0 & \lambda^2 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} &= \begin{pmatrix} a \sin(\lambda t) \\ 0 \end{pmatrix} \\ \begin{pmatrix} x_1(0) \\ x_2(0) \end{pmatrix} &= \begin{pmatrix} -\frac{a}{2\lambda} \\ 1 \end{pmatrix}. \end{aligned} \tag{5.5}$$

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

For more easily a matrix  $B$ , which annihilates the function of disturbance, is applied the procedure described by Steffensen [20] and [21].

We define a new variable  $x_3(t) = a \sin(\lambda t)$ , which allows to express the system (5.5) as follows:

$$\begin{aligned} \begin{pmatrix} x_1'(t) \\ x_2'(t) \\ x_3'(t) \end{pmatrix} + \begin{pmatrix} 0 & \lambda^2 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{pmatrix} &= \begin{pmatrix} a \sin(\lambda t) \\ 0 \\ a\lambda \cos(\lambda t) \end{pmatrix} \\ \begin{pmatrix} x_1(0) \\ x_2(0) \\ x_3(0) \end{pmatrix} &= \begin{pmatrix} -\frac{a}{2\lambda} \\ 1 \\ 0 \end{pmatrix} \end{aligned} \tag{5.6}$$

and exact solution

$$\begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{pmatrix} = \begin{pmatrix} -\lambda \left(1 - \frac{a}{2\lambda} t\right) \sin(\lambda t) - \frac{a}{2\lambda} t \cos(\lambda t) \\ \left(1 - \frac{a}{2\lambda} t\right) \cos(\lambda t) \\ a \sin(\lambda t) \end{pmatrix}. \tag{5.7}$$

By applying the operator  $(D + B)$  to the system (5.6), we obtain:

$$\begin{aligned} L_2(\mathbf{x}) &= \begin{pmatrix} x_1''(t) \\ x_2''(t) \\ x_3''(t) \end{pmatrix} + \begin{pmatrix} 0 & \lambda^2 & -1 \\ -1 & 1 & 0 \\ \lambda^2 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1'(t) \\ x_2'(t) \\ x_3'(t) \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & \lambda^4 & 0 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \\ \mathbf{x}(0) &= \begin{pmatrix} x_1(0) \\ x_2(0) \\ x_3(0) \end{pmatrix} = \begin{pmatrix} -\frac{a}{2\lambda} \\ 1 \\ 0 \end{pmatrix} \\ \mathbf{x}'(0) &= \begin{pmatrix} x_1'(0) \\ x_2'(0) \\ x_3'(0) \end{pmatrix} = \begin{pmatrix} -\lambda^2 \\ -\frac{a}{2\lambda} \\ a\lambda \end{pmatrix}. \end{aligned} \tag{5.8}$$

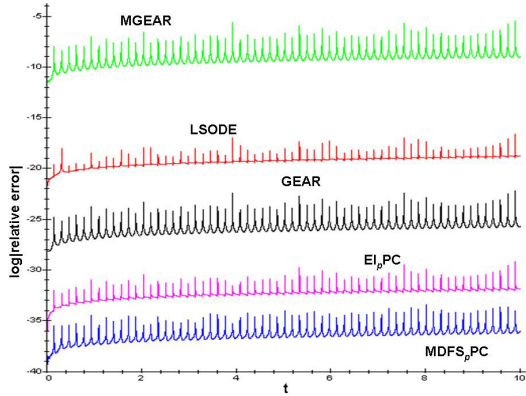


Figure 3. Problem 2, relative error.

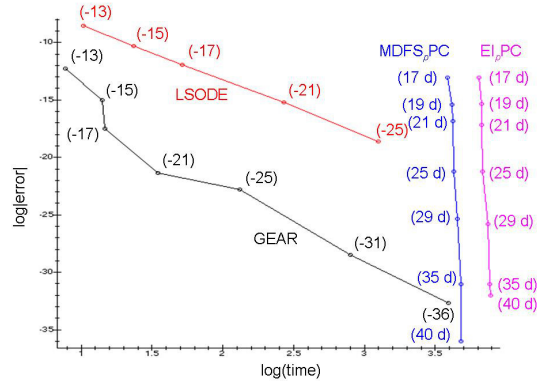


Figure 4. Problem 2, efficiency plot for the integration of  $\mathbf{x}(t)$  at last point.

This problem is resolved by  $\lambda = 10$ , by applying the MDFSpPC method.

The results are compared with those obtained by integrators, LSODE, MGEAR, GEAR, implemented in MAPLE and the multistep method EIPpPC [15], based on the Scheifele  $G$ -functions.

In Fig. 3 contrasts the decimal logarithm of module of the relative error of the solution  $\mathbf{x}(t)$ , calculated using  $\Phi$ -functions method MDFSpPC, with step size  $h = 10^{-3}$ , 40 digits and  $p = 17$ , with the numerical integration codes MGEAR[msteppart] with errorper = Float(1, -13), LSODE with a tolerance of  $10^{-25}$ , GEAR with errorper = Float(1, -31) and EI<sub>p</sub>PC with step size  $h = 10^{-3}$ , 40 digits and  $p = 17$ .

The results for the integration of the function  $\mathbf{x}$  are shown in Fig. 4 in which the information is arranged as in Fig. 2.

### 5.3 Problem 3

Denk [1] proposed the following highly oscillatory problem:

$$\begin{pmatrix} x_1'(t) \\ x_2'(t) \end{pmatrix} + \begin{pmatrix} 0 & -1 \\ \kappa^2 & 0 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} = \kappa^2 \begin{pmatrix} 0 \\ t \end{pmatrix} \tag{5.9}$$

where  $x_1(0) = 10^{-5}$ ,  $x_2(0) = 1 - 10^{-5}\kappa \cot k$  with  $\kappa = 314.16$ .

With eigenvalues  $\pm \kappa i$  and with an exact solution

$$\begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} = \begin{pmatrix} t + 10^{-5} (\cos(\kappa t) - \cot(\kappa) \sin(\kappa t)) \\ 1 - 10^{-5}\kappa (\sin(\kappa t) + \cot(\kappa) \cos(\kappa t)) \end{pmatrix}. \tag{5.10}$$

A new variable  $x_3(t) = -\kappa^2 t$  is defined [20] and [21], which permits the system (5.9) to be expressed as follows:

$$\begin{pmatrix} x_1'(t) \\ x_2'(t) \\ x_3'(t) \end{pmatrix} + \begin{pmatrix} 0 & -1 & 0 \\ \kappa^2 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{pmatrix} = \kappa^2 \begin{pmatrix} 0 \\ t \\ -1 \end{pmatrix} \tag{5.11}$$

$$\begin{pmatrix} x_1(0) \\ x_2(0) \\ x_3(0) \end{pmatrix} = \begin{pmatrix} 10^{-5} \\ 1 - 10^{-5}\kappa \cot(\kappa) \\ 0 \end{pmatrix}$$

and exact solution

$$\begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{pmatrix} = \begin{pmatrix} t + 10^{-5} (\cos(\kappa t) - \cot(\kappa) \sin(\kappa t)) \\ 1 - 10^{-5}\kappa (\sin(\kappa t) + \cot(\kappa) \cos(\kappa t)) \\ -\kappa^2 t \end{pmatrix}. \tag{5.12}$$

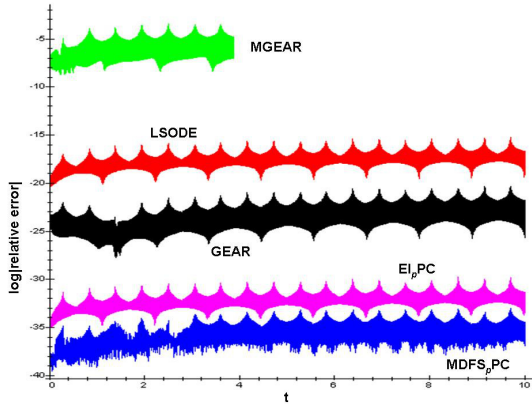


Figure 5. Problem 3, relative error.

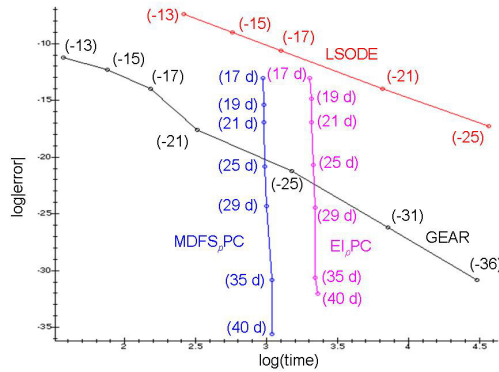


Figure 6. Problem 3, efficiency plot for the integration of  $\mathbf{x}(t)$  at last point.

By applying the operator  $(D + B)$  to the system (5.11), with the  $B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$  result:

$$\begin{aligned}
 L_2(\mathbf{x}) &= \begin{pmatrix} x_1''(t) \\ x_2''(t) \\ x_3''(t) \end{pmatrix} + \begin{pmatrix} 1 & -1 & 0 \\ \kappa^2 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1'(t) \\ x_2'(t) \\ x_3'(t) \end{pmatrix} + \begin{pmatrix} 0 & -1 & 0 \\ 0 & 0 & 0 \\ 0 & -1 & 0 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \\
 \mathbf{x}(0) &= \begin{pmatrix} x_1(0) \\ x_2(0) \\ x_3(0) \end{pmatrix} = \begin{pmatrix} 10^{-5} \\ 1 - 10^{-5}\kappa \cot \kappa \\ 0 \end{pmatrix} \\
 \mathbf{x}'(0) &= \begin{pmatrix} x_1'(0) \\ x_2'(0) \\ x_3'(0) \end{pmatrix} = \begin{pmatrix} 1 - 10^{-5}\kappa \cot \kappa \\ -10^{-5}\kappa^2 \\ -\kappa^2 \end{pmatrix}.
 \end{aligned} \tag{5.13}$$

In Fig. 5 contrasts the decimal logarithm of module of the relative error of the solution  $\mathbf{x}(t)$ , calculated using  $\Phi$ -functions method MDFSpPC, with step size  $h = 10^{-3}$ , 40 digits and  $p = 2$ , with the numerical integration codes MGEAR[msteppart] with errorper = Float(1, -10), LSODE with a tolerance of  $10^{-25}$ , GEAR with errorper = Float(1, -22) and EI<sub>p</sub>PC with step size  $h = 10^{-3}$ , 40 digits and  $p = 8$ .

The results for the integration of the function  $\mathbf{x}(t)$  are shown in Fig. 6 in which the information is arranged as in Fig. 2.

### 5.4 Problem 4

This example shows an application of the MDFSpPC to a problem of quasiperiodic orbits studied by [18], which can also be found in [14, 17, 22, 24], among others.

Let

$$x''(t) + x(t) = 10^{-3}e^{it}, \quad x(0) = 1, \quad x'(0) = 0.9995i, \tag{5.14}$$

for which the analytical solution is:

$$x(t) = (1 - 5 \cdot 10^{-4}it)e^{it} = (\cos(t) + 5 \cdot 10^{-4}t \sin(t)) + i(\sin(t) - 5 \cdot 10^{-4}t \cos(t)). \tag{5.15}$$

The solution represents motion on a perturbation of a circular orbit in the complex plane. The problem may be solved either as a single equation in complex arithmetic or a pair of uncoupled equations.



Noting  $x(t) = u(t) + iv(t)$ , and by substituting in (5.14), we get the following second order system:

$$\begin{cases} u''(t) + u(t) = 10^{-3} \cos(t) \\ v''(t) + v(t) = 10^{-3} \sin(t) \end{cases} \quad (5.16)$$

$$u(0) = 1, \quad u'(0) = 0, \quad v(0) = 0, \quad v'(0) = 0.9995.$$

By defining the variables  $x_1(t) = u(t)$ ,  $x_2(t) = u'(t)$ ,  $x_3(t) = v(t)$  and  $x_4(t) = v'(t)$  becomes the system of first order linear equations:

$$\begin{pmatrix} x_1'(t) \\ x_2'(t) \\ x_3'(t) \\ x_4'(t) \end{pmatrix} + \begin{pmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{pmatrix} = 10^{-3} \begin{pmatrix} 0 \\ \cos(t) \\ 0 \\ \sin(t) \end{pmatrix} \quad (5.17)$$

$$\begin{pmatrix} x_1(0) \\ x_2(0) \\ x_3(0) \\ x_4(0) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0.9995 \end{pmatrix}$$

with double eigenvalues  $\pm i$  and exact solution:

$$\begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{pmatrix} = \begin{pmatrix} \cos(t) + 5 \cdot 10^{-4} t \sin(t) \\ -0.9995 \sin(t) + 5 \cdot 10^{-4} t \cos(t) \\ \sin(t) - 5 \cdot 10^{-4} t \cos(t) \\ 0.9995 \cos(t) + 5 \cdot 10^{-4} t \sin(t) \end{pmatrix}. \quad (5.18)$$

The matrix which annihilates the disturbance function is

$$B = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix}.$$

Applying the operator  $(D + B)$  to the system (5.17) with the result:

$$L_2(\mathbf{x}) = \begin{pmatrix} x_1''(t) \\ x_2''(t) \\ x_3''(t) \\ x_4''(t) \end{pmatrix} + \begin{pmatrix} 1 & -1 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & -1 \\ 0 & -1 & 1 & 0 \end{pmatrix} \begin{pmatrix} x_1'(t) \\ x_2'(t) \\ x_3'(t) \\ x_4'(t) \end{pmatrix} + \begin{pmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \\ -1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (5.19)$$

$$\begin{pmatrix} x_1(0) \\ x_2(0) \\ x_3(0) \\ x_4(0) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0.9995 \end{pmatrix}, \quad \begin{pmatrix} x_1'(0) \\ x_2'(0) \\ x_3'(0) \\ x_4'(0) \end{pmatrix} = \begin{pmatrix} 1 \\ -0.999 \\ 0.9995 \\ 0 \end{pmatrix}.$$

In Fig. 7 contrasts the decimal logarithm of the relative error model of the solution  $\mathbf{x}(t)$ , calculated using  $\Phi$ -functions method MDFSpPC, with step size  $h = 10^{-3}$ , 40 digits and  $p = 10$ , with the numerical integration codes MGEAR[msteppart] with errorper = Float(1, -13), LSODE with a tolerance of  $10^{-25}$ , GEAR with errorper = Float(1, -22) and ElpPC with step size  $h = 10^{-3}$ , 40 digits and  $p = 10$ .

## 6 Conclusions

The multistep method MDFSpPC introduced in this paper it is based on the ideas developed by G. Scheifele and it is constructed taking as point of departure the  $\Phi$ -functions series method.



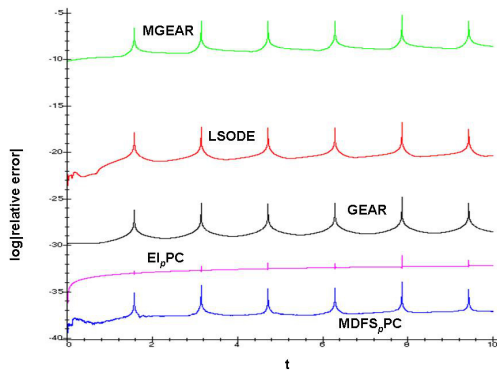


Figure 7. Problem 4, relative error.

The new method allows a more accurate integration of a wide range of problems, and under certain hypotheses, the multistep method calculates the exact solution of the perturbed problem.

Moreover the multistep method is defined for variable-step and variable-order, VSVO, maintains the good properties of the  $\Phi$ -functions series method and it incorporates an algebraic recurrent procedure to compute the coefficients of the algorithm, what facilitates its implementation on a computer.

This new method can successfully compete with well known general and special-purpose integrators as shown in the examples, where it gains in the attainable accuracy and the efficiency of several orders of magnitude have been shown for different stiff perturbed problems.

**Funding:** This work has been supported by GRE09-13 project of the University of Alicante and the project of the Generalitat Valenciana GV/2011/032.

## References

- [1] G. Denk, A new numerical method for the integration of highly oscillatory second-order ordinary differential equations, *Appl. Numer. Math.* **13** (1993), 57–67.
- [2] J. M. Ferrándiz, A general canonical transformation increasing the number of variables with application to the two-body problem, *Celestial Mech.* **41** (1988), 343–357.
- [3] F. Garcia-Alonso, J. A. Reyes, J. M. Ferrándiz, and J. Vigo-Aguiar, Accurate numerical integration of perturbed oscillatory systems in two frequencies, *ACM Trans. Math. Software TOMS*, **36** (2009), No. 4, article 21.
- [4] F. Garcia-Alonso and J. A. Reyes, A new approach for exact integration of some perturbed stiff linear systems of oscillatory type, *Appl. Math. Comput.* **215** (2009), 2649–2662.
- [5] F. Garcia-Alonso, J. A. Reyes, and Y. Villacampa, A new approach for multistep numerical methods in several frequencies for perturbed oscillators, *Adv. Engrg. Softw.* **45** (2012), 252–260.
- [6] L. G. Ixaru, G. Vanden Berghe, and H. De Meyer, Exponentially fitted variable two-step BDF algorithm for first order ODE's, *Comp. Phys. Commun.* **100** (2003), 56–70.
- [7] L. G. Ixaru, G. Vanden Berghe, and H. De Meyer, Frequency evaluation in exponential fitting multistep algorithms for ODE's, *J. Comp. Appl. Math.* **140** (2002), 423–434.
- [8] J. D. Lambert, *Numerical Methods for Ordinary Differential Systems*, John Wiley and Sons Ltd., New York, 1991.
- [9] I. G. MacDonald, *Symmetric Functions and Hall Polynomials*, Oxford University Press Inc., New York, 1998.
- [10] P. Martin and J. M. Ferrándiz, Behaviour of the SMF method for the numerical integration of satellite orbits, *Celestial Mech.* **63** (1995), 29–40.
- [11] P. Martin and J. M. Ferrándiz, Multistep numerical methods based on Scheifele  $G$ -functions with application to satellite dynamics, *SIAM J. Numer. Anal.* **34** (1997), 359–375.
- [12] M. Palacios, Métodos multirevolución simétricos para propagación de órbitas en intervalos grandes de tiempo, *Mono-grafías de la Real Academia de ciencias de Zaragoza* **22** (2003), 55–66.
- [13] L. R. Petzold, An Efficient numerical method for highly oscillatory ordinary differential equations, *SIAM J. Numer. Anal.* **18** (1981), No. 3, 455–479.

- [14] J. I. Ramos, Piecewise-linearized methods for initial-value problems with oscillating solutions, *Appl. Math. Comput.* **181** (2006), 123–146.
- [15] J. A. Reyes, F. García-Alonso, J. M. Ferrándiz, and J. Vigo-Aguiar, Numeric multistep variable methods for perturbed linear system integration, *Appl. Math. Comput.* **190** (2007), 63–79.
- [16] G. Scheifele, On numerical integration of perturbed linear oscillating systems, *Z. Angew. Math. Phys.* **22** (1971), 186–210.
- [17] T. E. Simos and J. Vigo-Aguiar, Exponentially fitted symplectic integrator, *Phys. Rev. E.* **67** (2003), 1–7.
- [18] E. L. Stiefel and D. G. Bettis, Stabilization of Cowell's method, *Numer. Math.* **13** (1969), 154–175.
- [19] E. L. Stiefel and G. Scheifele, *Linear and Regular Celestial Mechanics*, Springer, Berlin–Heldelberg–New York, 1971.
- [20] J. F. Steffensen, On the differential equations of Hill in the theory of the motion of the Moon, *Acta Math.* **93** (1955), 169–177.
- [21] J. F. Steffensen, On the differential equations of Hill in the theory of the motion of the Moon (II), *Acta Math.* **95** (1956), 25–37.
- [22] H. Van de Vyver, Two-step hybrid methods adapted to the numerical integration of perturbed oscillators, arXiv:math/0612637v1 [math.NA], 2006.
- [23] J. Vigo-Aguiar, An approach to variable coefficients methods for special differential equations, *Int. J. Appl. Math.* **1** (1999), No. 8, 911–921.
- [24] J. Vigo-Aguiar and J. M. Ferrándiz, A general procedure for the adaptation of multistep algorithms to the integration of oscillatory problems, *SIAM J. Numer. Anal.* **35** (1998), No. 4, 1684–1708.
- [25] J. Vigo-Aguiar and J. M. Ferrándiz, Higher-order variable-step algorithms adapted to the accurate numerical integration of perturbed oscillators, *Computer in Physics* **12** (1998), 467–470.
- [26] J. Vigo-Aguiar and J. M. Ferrándiz, VSVO adapted multistep methods for the numerical integration of second order differential equations, *Appl. Math. Lett.* **11** (1998), 83–89.