Journal of Computational Mathematics
Vol.31, No.2, 2013, 175–189.
http://www.global-sci.org/jcm
doi:10.4208/jcm.1212-m3987

THE INTEGRATION OF STIFF SYSTEMS OF ODEs USING NDFs AND MEBDFs*

Elisabete Alberdi Celaya
Department of Applied Mathematics, EUIT de Minas y Obras Públicas, Universidad del País Vasco UPV/EHU, Colina de Beurko s/n, 48902 Barakaldo, Spain
Email: elisabete.alberdi@ehu.es
Juan José Anza
Department of Applied Mathematics, ETS de Ingeniería de Bilbao, Universidad del País Vasco UPV/EHU, Alameda de Urquijo s/n, 48013 Bilbao, Spain
Email: juanjose.anza@ehu.es

Abstract

In this paper we modify the MEBDF method using the NDFs as predictors instead of the BDFs. We have done it in three different ways: changing both predictors of the MEBDF, changing only the first predictor and changing only the second one. We have called the new methods MENDF, MENBDF and MEBNDF respectively. The new methods are A-stable up to order 4 and the stability properties of the new methods are better than the MEBDF method.

Mathematics subject classification: 65L05.
Key words: BDF, NDF, MEBDF, Stiff ODEs, Stability.

1. Introduction

In this paper we will consider the initial value problem:

$$y'(x) = f\left(x, y(x)\right), \quad y\left(x_0\right) = y_0, \tag{1.1}$$

on the finite interval $T = [x_0, x_n]$, and being $y : [x_0, x_n] \to \mathbb{R}^m$ and $f : [x_0, x_n] \times \mathbb{R}^m \to \mathbb{R}^m$ continuous functions. When we are solving systems of stiff ordinary differential equations by numerical integration, it is important to use an accurate algorithm which has good stability properties [6]. Many researches have been focused on the generation of efficient algorithms for the numerical integration of stiff systems and some of them have been based on backward differentiation formulae (BDF) [9]. The BDFs give us the possibility to use high order formulae in highly stable schemes, but their biggest drawback is the poor stability properties of the highest orders formulae, when the eigenvalues of the Jacobian matrix lie close to the imaginary axis.

A great effort to derive methods with better accuracy and stability properties than the ones of the BDFs has been made. One of the modifications made to the BDFs in this line are the NDFs (Numerical Differentiation formulae) [17]. It is a computationally cheap modification that consists of anticipating a difference of order $(k+1)$ multiplied by a constant $\kappa\gamma_k$ in the BDF formula of order k. This term makes the NDFs more accurate than the BDFs and not

* Received December 19, 2011 / Revised version received October 25, 2012 / Accepted December 25, 2012 / Published online March 14, 2013 /

much less stable. This modification was proposed only for orders $k = 1, 2, 3, 4$, because it is inefficient for orders greater than 4.

The search of higher order and more stable methods has been followed in two main directions. The first of these two directions consists of using superfuture point schemes and the second one uses higher derivatives of the solutions. In [2, 4] Cash introduces methods using superfuture points to solve stiff IVPs. These methods are known as extended BDF (EBDF) and modified extended BDF (MEBDF). They use two BDF predictors and one implicit multistep corrector. Both methods are A-stable up to order 4 and $A(\alpha)$-stable up to order 9, and the class MEBDF has better stability properties than the class EBDF. In [5] a code based on the MEBDF is described and its performance on a set of stiff problems is discussed. In [13] Matrix free MEBDF (MF-MEBDF) methods are introduced to optimize the computations of the EBDF.

A different variation of the BDFs was introduced by Fredebeul [8], the A-BDF method. In this method the implicit and explicit BDF are used in the same formula, with a free parameter, being $A(\alpha)$-stable up to order 7. And in [11], a modification to the methods A-BDF and EBDF is introduced, the method called A-EBDF, in which larger absolute stability regions than the ones of the A-BDF and the EBDF are obtained.

Among the modifications made to the BDFs by using higher derivatives, we can find [7], where a class of second derivative formulae A-stable for order 4 is developed. In [3], Cash introduces another class of second derivative methods which uses the EBDF scheme. This class is A-stable up to order 6. In more recent researches such as [12, 15], different classes of second derivative multistep methods are derived in which very good stability properties are reached again.

The purpose of this paper is to follow the MEBDF scheme but by substituting the BDF predictors by the NDF formulae [17]. We did this in [1], when we changed the predictors of the EBDFs and we obtained new classes of formulae with smaller local truncation error and better stability properties. We have changed the predictors of the MEBDFs in three different ways: changing only the first predictor, changing only the second one or by changing both predictors. We have called the new methods MENBDF, MEBNDF and MENDF respectively and all of them have better stability properties than the MEBDFs. In Section 2, we give details about the modifications introduced in MEBDF, such as, MENDF, MENBDF and MEBNDF. In Section 3, the stability analysis is developed. Finally in Section 4, some computational aspects are included and results of several problems are reported in Section 5.

2. The Use of NDFs as the Predictors of the MEBDF Class

In order to understand the new methods we have developed, we will start analysing the properties of the NDF and MEBDF, to finally derive the MENDF, MENBDF and MEBNDF algorithms.

2.1. NDF scheme

Among the codes that have been created to solve stiff problems, the most popular and widely used are the backward differentiation formulae, BDFs [9]. These numerical methods are A-stable only up to order 2, but they have good stability properties also when working in high orders. The k-step BDF can be expressed in this way using backward differences:

$$\sum_{j=1}^{k} \frac{1}{j} \nabla^j y_{n+k} = h f_{n+k}. \tag{2.1}$$

We get the well-known expression of the BDFs after developing the backward differences of expression (2.1):

$$\sum_{j=0}^{k} \hat{\alpha}_j y_{n+j} = h f_{n+k}. \tag{2.2}$$

The leading term of the BDFs truncation error is this one:

$$C_1 h^{k+1} y^{(k+1)} (x_n), \tag{2.3}$$

where the error constant C_1 is given by

$$C_1 = \frac{-1/\gamma_k}{k+1}, \tag{2.4}$$

and

$$\gamma_k = \sum_{j=1}^{k} \frac{1}{j} = \begin{cases} 1, & k = 1, \\ 3/2, & k = 2, \\ 11/6, & k = 3, \\ 25/12, & k = 4, \\ 137/60, & k = 5. \end{cases} \tag{2.5}$$

In [17], Shampine introduces a new family of formulae suitable for the solution of stiff problems. This new family called numerical differentiation formulae, NDFs, consist of anticipating the difference of order $(k+1)$ multiplied by the term $\kappa\gamma_k$ in the BDF formula of order k. These new methods have the form:

$$\sum_{j=1}^{k} \frac{1}{j} \nabla^j y_{n+k} = h f_{n+k} + \kappa\gamma_k \nabla^{k+1} y_{n+k}. \tag{2.6}$$

And again, developing the backward differences of expression (2.6) an equivalent expression for the NDFs is achieved:

$$\sum_{j=0}^{k} \hat{\alpha}_j y_{n+j} = h f_{n+k} + \kappa\gamma_k \nabla^{k+1} y_{n+k}. \tag{2.7}$$

Klopfenstein and Shampine introduced the scalar parameter κ, so that the angle of $A(\alpha)$-stability was maximized at the same time that the error was reduced. The NDFs are more accurate than the BDFs, although a couple of the higher-order formulae are less stable. The leading term of the local truncation error of the NDFs is given by:

$$C_2 h^{k+1} y^{(k+1)} (x_n). \tag{2.8}$$

We have called C_2 to the error constant of the LTE of the method NDF (2.8):

$$C_2 = \frac{-1/\gamma_k}{k+1} - \kappa. \tag{2.9}$$

The better accuracy of the NDFs implies that they can achieve the same accuracy as BDFs with a bigger step size. More properties of BDF and NDF methods are shown in Table 2.1.

Table 2.1: The Klopfenstein-Shampine NDFs and their efficiency and $A(\alpha)$-stability relative to the BDFs [17].

k	NDF coefficient κ	Step ratio percent	Stability angle BDF	Stability angle NDF
1	-0.1850	26%	90	90
2	-1/9	26%	90	90
3	-0.0823	26%	86	80
4	-0.0415	12%	73	66

2.2. MEBDF scheme

With the aim of increasing the stability of the BDF methods, Cash extended these methods by introducing superfuture points [2]. The method was called EBDF (extended backward differentiation formula) and A-stable schemes of order up to 4 were obtained. Later, in [4], a class of modified extended backward differentiation formulae (MEBDF) is introduced where the forth stage of the EBDF scheme was changed by the next expression:

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = h\hat{\beta}_k f_{n+k} + h\beta_{k+1}\bar{f}_{n+k+1} + h\left(\beta_k - \hat{\beta}_k\right)\bar{f}_{n+k}, \tag{2.10}$$

where the coefficients β_k are the ones that correspond to the method EBDF and $\hat{\beta}_k$ are the coefficients of the method BDF. A table of coefficients β_k and $\hat{\beta}_k$ can be found in [2] and [9] respectively.

Assuming that the solutions y_n, y_{n+1}, ..., y_{n+k-1} are available, the way in which the formula (2.10) is used is by carrying out the following steps:

1. Compute the first predictor $\bar{y}_{n+k}$ as the solution of the conventional k-step backward differentiation formula:
$$y_{n+k} + \sum_{j=0}^{k-1} \hat{\alpha}_j y_{n+j} = h\hat{\beta}_k f_{n+k}\ , \quad (y_{n+k} := \bar{y}_{n+k}). \tag{2.11}$$
2. Compute the second predictor $\bar{y}_{n+k+1}$ advancing a new step with the same k-step BDF formula:
$$y_{n+k+1} + \sum_{j=0}^{k-1} \hat{\alpha}_j y_{n+j+1} = h\hat{\beta}_k f_{n+k+1}\ , \quad (y_{n+k+1} := \bar{y}_{n+k+1}). \tag{2.12}$$
3. Evaluate $\bar{f}_{n+k}$=$f(x_{n+k}, \bar{y}_{n+k})$ and $\bar{f}_{n+k+1}$=$f(x_{n+k+1}, \bar{y}_{n+k+1})$.
4. Insert $\bar{f}_{n+k}$ and $\bar{f}_{n+k+1}$ in (2.10) and solve for a new y_{n+k} which will be the numerical solution of the MEBDF method.

The local truncation error of the MEBDF method is given by the expression:

$$\begin{aligned} LTE_k = h^{k+2}\Bigg[C_1\left(\beta_{k+1}\left(1 - \frac{\hat{\alpha}_{k-1}}{\hat{\alpha}_k}\right) + (\beta_k - \hat{\beta}_k)\right)\frac{\partial f}{\partial y}y^{(k+1)} \\ +C_3 y^{(k+2)}\Bigg](x_n) + O\left(h^{k+3}\right), \end{aligned} \tag{2.13}$$

where C_1 is the error constant of the local truncation error of the BDF method given by (2.4), and C_3 is the error constant of the local truncation error for the formula (2.10).

Lemma 2.1. *Given that the formula* (2.10) *is of order* $(k+1)$ *and the BDF/NDF used in* (2.11) *and* (2.12) *are of order k, the predictor-corrector algorithm (1)-(4) has order* $(k+1)$. *The demonstration of this lemma can be found in reference* [10].

2.3. MENDF, MENBDF, MEBNDF methods' scheme

2.3.1. MENDF

The MENDF method consists of applying the NDF method as the predictor in stages 1 and 2 of the MEBDF method. Next, we follow the same steps as in the MEBDF scheme: we evaluate $\bar{f}_{n+k}=f(x_{n+k},\bar{y}_{n+k})$ and $\bar{f}_{n+k+1}=f\left(x_{n+k+1},\bar{y}_{n+k+1}\right)$ and we insert the terms $\bar{f}_{n+k},\ \bar{f}_{n+k+1}$ in the expression (2.10).

First predictor: The first time we apply the predictor NDF, the value $\bar{y}_{n+k}$ is obtained. The difference between the exact value and the calculated is given by this expression:

$$y(x_{n+k})-\bar{y}_{n+k}=C_2h^{k+1}y^{(k+1)}\left(x_n\right)+O\left(h^{k+2}\right), \tag{2.14}$$

where C_2 is the error constant of the method NDF given by (2.9).

Second predictor: The second time we use the predictor NDF we get the value $\bar{y}_{n+k+1}$, and the difference between the exact and the calculated value is this:

$$y(x_{n+k+1})-\bar{y}_{n+k+1}=C_2\left(1-\frac{\hat{\alpha}_{k-1}}{\hat{\alpha_k}}-\frac{\kappa\gamma_k(k+1)}{\hat{\alpha}_k}\right)h^{k+1}y^{(k+1)}\left(x_n\right)+O\left(h^{k+2}\right). \tag{2.15}$$

In [1] we verified for $k=2$ that the local truncation error of the method after applying the second predictor NDF, is the one proposed by the expression (2.15).

Corrector: Eventually, we apply the corrector (expression (2.10)) and the local truncation error of the overall method is obtained:

$$LTE_k=h^{k+2}\left[\left(\beta_{k+1}A_k+C_2(\beta_k-\hat{\beta}_k)\right)\frac{\partial f}{\partial y}y^{(k+1)}+C_3y^{(k+2)}\right](x_n)+O\left(h^{k+3}\right), \tag{2.16}$$

where: $A_k=C_2\left(1-\frac{\hat{\alpha}_{k-1}}{\hat{\alpha_k}}-\frac{\kappa\gamma_k(k+1)}{\hat{\alpha}_k}\right)$.

2.3.2. MENBDF, MEBNDF

In the previous Section we have used the NDF as the predictor of the MEBDF algorithm in stages 1 and 2. But the option of using NDFs in the predicting stages of the MEBDF is not the unique option. We can also apply the NDF as the first predictor and the BDF as the second one or the BDF as the first predictor and the NDF as the second one. Hence, the options available for the predictors are the following: BDF-BDF, NDF-NDF, NDF-BDF, BDF-NDF. In all the cases the local truncation error can be expressed in this way:

$$h^{k+2}\left[\left(\beta_{k+1}A_k+C_i(\beta_k-\hat{\beta}_k)\right)\frac{\partial f}{\partial y}y^{(k+1)}+C_3y^{(k+2)}\right](x_n)+O\left(h^{k+3}\right). \tag{2.17}$$

The value of the constants A_k and C_i is different depending on the predictors:

- Case MEBDF (BDF-BDF-MEBDF):

$$A_k = C_1\left(-\frac{\hat{\alpha}_{k-1}}{\hat{\alpha}_k}+1\right), \quad C_i = C_1. \tag{2.18}$$

- Case MENDF (NDF-NDF-MEBDF):

$$A_k = C_2\left(-\frac{\hat{\alpha}_{k-1}}{\hat{\alpha}_k}-\kappa(k+1)\gamma_k\frac{1}{\hat{\alpha}_k}+1\right), \quad C_i = C_2. \tag{2.19}$$

- Case MENBDF (NDF-BDF-MEBDF):

$$A_k = \left(-C_2\frac{\hat{\alpha}_{k-1}}{\hat{\alpha}_k}+C_1\right), \quad C_i = C_2. \tag{2.20}$$

- Case MEBNDF (BDF-NDF-MEBDF):

$$A_k = \left(-C_1\frac{\hat{\alpha}_{k-1}}{\hat{\alpha}_k}-C_1\kappa(k+1)\gamma_k\frac{1}{\hat{\alpha}_k}+C_2\right), \quad C_i = C_1. \tag{2.21}$$

The values of A_k depend on the predictors used in the general scheme of the MEBDFs. The same occured in the case of the EBDFs. And in both families, EBDF and MEBDF, the value of the constant A_k is the same when the same predictors are used. Values of A_k for EBDF, EBNDF, ENBDF and ENDF can be found in [1].

3. Stability Analysis

3.1. Stability function of MENDF

We now examine the stability behaviour of our new methods. The region of absolute stability of the overall method MENDF is found using Schur's theorem, see [16]. To do this, we will apply the method MENDF to the test equation $y' = \lambda y$. That is to say, $hf_j = h\lambda y_j$ is introduced in expression (2.10) and expression (2.7) is used as the first and the second predictor. We will set y_{n-1}=1, ..., y_{n+k-1}=r^k, the algorithm will be computed in order to obtain y_{n+k}=r^{k+1} and the characteristic equation will be achieved, being $\hat{h} = h\lambda$:

$$A\hat{h}^3 + B\hat{h}^2 + C\hat{h} + D = 0, \tag{3.1}$$

where

$$\begin{cases} A = -\hat{\beta}_k r^{k+1}, \quad D = \left(\hat{\alpha}_k - \kappa\gamma_k\right)^2 T, \\ B = 2\left(\hat{\alpha}_k - \kappa\gamma_k\right)\hat{\beta}_k r^{k+1} + T - \beta_{k+1}S + (\beta_k - \hat{\beta}_k)R, \\ C = -\hat{\beta}_k\left(\hat{\alpha}_k - \kappa\gamma_k\right)^2 r^{k+1} - 2\left(\hat{\alpha}_k - \kappa\gamma_k\right)T + \left(\hat{\alpha}_k - \kappa\gamma_k\right)\beta_{k+1}S \\ \qquad -\beta_{k+1}\left(-\hat{\alpha}_{k-1} - (k+1)\kappa\gamma_k\right)R - (\beta_k - \hat{\beta}_k)R(\hat{\alpha}_k - \kappa\gamma_k), \end{cases} \tag{3.2}$$

with

$$\begin{cases} R = (-1)^{k+1}\binom{k+1}{0}\kappa\gamma_k + \sum\limits_{j=1}^{k} r^j\left(-\hat{\alpha}_{j-1} + (-1)^{k+1-j}\binom{k+1}{j}\kappa\gamma_k\right), \\ S = (-1)^k r\kappa\gamma_k - \sum\limits_{j=1}^{k-1} r^{j+1}\left(-\hat{\alpha}_{j-1} + (-1)^{k+1-j}\binom{k+1}{j}\kappa\gamma_k\right), \\ T = \sum\limits_{j=0}^{k}\alpha_j r^{j+1}. \end{cases} \tag{3.3}$$

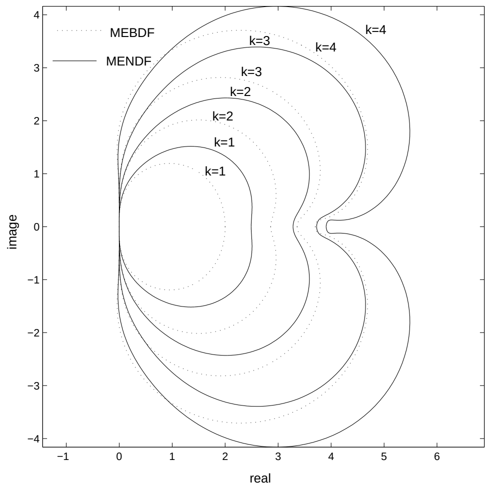

Fig. 3.1. Regions of stability of the methods MENDF and MEBDF. The stability regions are the outside of the plotted curves.

In Fig. 3.1 the stability regions of the MEBDF and MENDF methods are shown.

We will include the calculations done for the case $k = 2$:

$$\sum_{j=0}^{2} \hat{\alpha}_j y_{n+j} = h\bar{f}_{n+2} + \kappa\gamma_2 \nabla^3 \bar{y}_{n+2}. \tag{3.4}$$

Developing $\nabla^3 \bar{y}_{n+2}$ and applying the method given by (3.4) to the test equation, we can work out $\bar{y}_{n+2}$:

$$\bar{y}_{n+2} = \frac{y_{n+1}\left(-\hat{\alpha}_1 - 3\kappa\gamma_2\right) + y_n\left(-\hat{\alpha}_0 + 3\kappa\gamma_2\right) - \kappa\gamma_2 y_{n-1}}{\left(\hat{\alpha}_2 - \kappa\gamma_2 - \hat{h}\right)}. \tag{3.5}$$

NDF is used again as the second predictor to get $\bar{y}_{n+3}$:

$$\bar{y}_{n+3} = \frac{\bar{y}_{n+2}\left(-\hat{\alpha}_1 - 3\kappa\gamma_2\right) + y_{n+1}\left(-\hat{\alpha}_0 + 3\kappa\gamma_2\right) - \kappa\gamma_2 y_n}{\left(\hat{\alpha}_2 - \kappa\gamma_2 - \hat{h}\right)}. \tag{3.6}$$

Substituting (3.5) into (3.6) we have $\bar{y}_{n+3}$:

$$\bar{y}_{n+3} = \left[\frac{\left(y_{n+1}\left(-\hat{\alpha}_1 - 3\kappa\gamma_2\right) + y_n\left(-\hat{\alpha}_0 + 3\kappa\gamma_2\right) - \kappa\gamma_2 y_{n-1}\right)\left(-\hat{\alpha}_1 - 3\kappa\gamma_2\right)}{\left(\hat{\alpha}_2 - \kappa\gamma_2 - \hat{h}\right)^2} + \frac{\left(y_{n+1}\left(-\hat{\alpha}_0 + 3\kappa\gamma_2\right) - \kappa\gamma_2 y_n\right)\left(\hat{\alpha}_2 - \kappa\gamma_2 - \hat{h}\right)}{\left(\hat{\alpha}_2 - \kappa\gamma_2 - \hat{h}\right)^2} \right].$$

We calculate the derivatives of $\bar{y}_{n+2}$ and $\bar{y}_{n+3}$:

$$f\left(\bar{y}_{n+2}\right) = \lambda \frac{y_{n+1}\left(-\hat{\alpha}_1 - 3\kappa\gamma_2\right) + y_n\left(-\hat{\alpha}_0 + 3\kappa\gamma_2\right) - \kappa\gamma_2 y_{n-1}}{\left(\hat{\alpha}_2 - \kappa\gamma_2 - \hat{h}\right)}, \tag{3.7}$$

$$f(\bar{y}_{n+3}) = \lambda \left[\frac{\left(y_{n+1}\left(-\hat{\alpha}_1 - 3\kappa\gamma_2\right) + y_n\left(-\hat{\alpha}_0 + 3\kappa\gamma_2\right) - \kappa\gamma_2 y_{n-1}\right)\left(-\hat{\alpha}_1 - 3\kappa\gamma_2\right)}{\left(\hat{\alpha}_2 - \kappa\gamma_2 - \hat{h}\right)^2} \right.$$
$$\left. + \frac{\left(y_{n+1}\left(-\hat{\alpha}_0 + 3\kappa\gamma_2\right) - \kappa\gamma_2 y_n\right)\left(\hat{\alpha}_2 - \kappa\gamma_2 - \hat{h}\right)}{\left(\hat{\alpha}_2 - \kappa\gamma_2 - \hat{h}\right)^2} \right]. \tag{3.8}$$

Finally y_{n+2} is obtained using expression (2.10):

$$\alpha_0 y_n + \alpha_1 y_{n+1} + \alpha_2 y_{n+2} - \hat{h}\hat{\beta}_2 y_{n+2}$$
$$- \hat{h}\beta_3 \left[\frac{\left(y_{n+1}\left(-\hat{\alpha}_1 - 3\kappa\gamma_2\right) + y_n\left(-\hat{\alpha}_0 + 3\kappa\gamma_2\right) - \kappa\gamma_2 y_{n-1}\right)\left(-\hat{\alpha}_1 - 3\kappa\gamma_2\right)}{\left(\hat{\alpha}_2 - \kappa\gamma_2 - \hat{h}\right)^2} \right.$$
$$\left. + \frac{\left(y_{n+1}\left(-\hat{\alpha}_0 + 3\kappa\gamma_2\right) - \kappa\gamma_2 y_n\right)\left(\hat{\alpha}_2 - \kappa\gamma_2 - \hat{h}\right)}{\left(\hat{\alpha}_2 - \kappa\gamma_2 - \hat{h}\right)^2} \right]$$
$$- \hat{h}\left(\beta_2 - \hat{\beta}_2\right) \frac{y_{n+1}\left(-\hat{\alpha}_1 - 3\kappa\gamma_2\right) + y_n\left(-\hat{\alpha}_0 + 3\kappa\gamma_2\right) - \kappa\gamma_2 y_{n-1}}{\left(\hat{\alpha}_2 - \kappa\gamma_2 - \hat{h}\right)} = 0.$$

Substituting $y_{n+j} = r^{j+1}$ the following equation is obtained:

$$\left(\alpha_0 r + \alpha_1 r^2 + \alpha_2 r^3 - \hat{h}\hat{\beta}_2 r^3\right)\left(\hat{\alpha}_2 - \kappa\gamma_2 - \hat{h}\right)^2$$
$$- \hat{h}\beta_3 \left[\left(r^2\left(-\hat{\alpha}_1 - 3\kappa\gamma_2\right) + r\left(-\hat{\alpha}_0 + 3\kappa\gamma_2\right) - \kappa\gamma_2\right)\left(-\hat{\alpha}_1 - 3\kappa\gamma_2\right)\right.$$
$$\left. + \left(r^2\left(-\hat{\alpha}_0 + 3\kappa\gamma_2\right) - \kappa\gamma_2 r\right)\left(\hat{\alpha}_2 - \kappa\gamma_2 - \hat{h}\right)\right]$$
$$- \hat{h}\left(\beta_2 - \hat{\beta}_2\right)\left[r^2\left(-\hat{\alpha}_1 - 3\kappa\gamma_2\right) + r\left(-\hat{\alpha}_0 + 3\kappa\gamma_2\right) - \kappa\gamma_2\right]\left(\hat{\alpha}_2 - \kappa\gamma_2 - \hat{h}\right) = 0.$$

And the coefficients of the polynomial in $\hat{h}^n$ are given by:

$$\begin{cases} \hat{h}^3: \quad A = -\hat{\beta}_2 r^3, \\ \hat{h}^2: \quad B = 2\left(\hat{\alpha}_2 - \kappa\gamma_2\right)\hat{\beta}_2 r^3 + T - \beta_3 S + \left(\beta_2 - \hat{\beta}_2\right) R, \\ \hat{h}: \quad C = -\hat{\beta}_2\left(\hat{\alpha}_2 - \kappa\gamma_2\right)^2 r^3 - 2\left(\hat{\alpha}_2 - \kappa\gamma_2\right) T + \left(\hat{\alpha}_2 - \kappa\gamma_2\right)\beta_3 S \\ \qquad\qquad - \beta_3\left(-\hat{\alpha}_1 - 3\kappa\gamma_2\right) R - \left(\beta_2 - \hat{\beta}_2\right) R\left(\hat{\alpha}_2 - \kappa\gamma_2\right), \\ \hat{h}^0: \quad D = \left(\hat{\alpha}_2 - \kappa\gamma_2\right)^2 T, \end{cases}$$

where

$$\begin{cases} R = r^2\left(-\hat{\alpha}_1 - 3\kappa\gamma_2\right) + r\left(-\hat{\alpha}_0 + 3\kappa\gamma_2\right) - \kappa\gamma_2, \\ S = r\kappa\gamma_2 - r^2\left(-\hat{\alpha}_0 + 3\kappa\gamma_2\right), \quad T = \sum_{j=0}^{2} \alpha_j r^{j+1}. \end{cases}$$

The stability angles of the method MENDF can be found in Table 3.1.

3.2. Stability function of MENBDF

We will apply the MENBDF method to the test equation $y' = \lambda y$. The NDF will be used as the first predictor and the BDF as the second one. We will substitute $hf_j = \hat{h} y_j$ in expression

Table 3.1: $A(\alpha)$-stability of the methods MEBDF, MEBNDF, MENBDF, MENDF.

k	p (order)	$A(\alpha)$ MEBDF	$A(\alpha)$ MEBNDF	$A(\alpha)$ MENBDF	$A(\alpha)$ MENDF
1	2	90	90	90	90
2	3	90	90	90	90
3	4	90	90	90	90
4	5	88.36	88.41	88.88	88.93

(2.10), where $\hat{h} = h\lambda$. Setting y_{n-1}=1, ..., y_{n+k-1}=r^k and computing the method, we will reach the solution y_{n+k}=r^{k+1} as well as the characteristic equation:

$$A\hat{h}^3 + B\hat{h}^2 + C\hat{h} + D = 0. \tag{3.9}$$

Coefficients of the polynomial in $\hat{h}^n$:

$$\begin{cases} A = -\hat{\beta}_k r^{k+1}, \quad D = \hat{\alpha}_k\left(\hat{\alpha}_k - \kappa\gamma_k\right)T, \\ B = \left(2\hat{\alpha}_k - \kappa\gamma_k\right)\hat{\beta}_k r^{k+1} + T - \beta_{k+1}S + (\beta_k - \hat{\beta}_k)R, \\ C = -\hat{\beta}_k\hat{\alpha}_k\left(\hat{\alpha}_k - \kappa\gamma_k\right)r^{k+1} - \left(2\hat{\alpha}_k - \kappa\gamma_k\right)T \\ \qquad + \left(\hat{\alpha}_k - \kappa\gamma_k\right)\beta_{k+1}S + \beta_{k+1}\hat{\alpha}_{k-1}R - (\beta_k - \hat{\beta}_k)R\hat{\alpha}_k, \end{cases} \tag{3.10}$$

where

$$\begin{cases} R = (-1)^{k+1}\binom{k+1}{0}\kappa\gamma_k + \sum\limits_{j=1}^{k} r^j\left(-\hat{\alpha}_{j-1} + (-1)^{k+1-j}\binom{k+1}{j}\kappa\gamma_k\right), \\ S = \sum\limits_{j=0}^{k-2}\hat{\alpha}_j r^{j+2}, \quad T = \sum\limits_{j=0}^{k}\alpha_j r^{j+1}. \end{cases} \tag{3.11}$$

The stability angles of the method are in Table 3.1.

3.3. Stability function of MEBNDF

Proceeding in the same way as before, the characteristic polynomial of the MEBNDF method is obtained:

$$A\hat{h}^3 + B\hat{h}^2 + C\hat{h} + D = 0. \tag{3.12}$$

Coefficients of the polynomial in $\hat{h}^n$:

$$\begin{cases} A = -\hat{\beta}_k r^k, \quad D = \hat{\alpha}_k\left(\hat{\alpha}_k - \kappa\gamma_k\right)T, \\ B = \left(2\hat{\alpha}_k - \kappa\gamma_k\right)\hat{\beta}_k r^k + T - \beta_{k+1}S - (\beta_k - \hat{\beta}_k)R, \\ C = -\hat{\beta}_k\hat{\alpha}_k\left(\hat{\alpha}_k - \kappa\gamma_k\right)r^k - \left(2\hat{\alpha}_k - \kappa\gamma_k\right)T + \hat{\alpha}_k\beta_{k+1}S \\ \qquad + \beta_{k+1}\left(-\hat{\alpha}_{k-1} - (k+1)\kappa\gamma_k\right)R + (\beta_k - \hat{\beta}_k)R(\hat{\alpha}_k - \kappa\gamma_k), \end{cases} \tag{3.13}$$

Table 3.2: $A(\alpha)$-stability of the methods EBDF and MEBDF for different predictors.

$k = 4$	Predictors BDF-BDF	Predictors BDF-NDF	Predictors NDF-BDF	Predictors NDF-NDF
EBDF	87.61	87.68	87.49	87.54
MEBDF	88.36	88.41	88.88	88.93

where

$$\begin{cases} R = \sum\limits_{j=0}^{k-1} \hat{\alpha}_j r^j, \quad T = \sum\limits_{j=0}^{k} \alpha_j r^j, \\ S = (-1)^k \kappa\gamma_k - \sum\limits_{j=1}^{k-1} r^j \left(-\hat{\alpha}_{j-1} + (-1)^{k+1-j} \binom{k+1}{j} \kappa\gamma_k \right). \end{cases} \tag{3.14}$$

3.4. Comparison with the stability regions of the EBDFs

When in [1] we changed the predictor of the EBDFs, we obtain methods which were A-stable up to $k = 3$ and the same occurs when changing the predictor in the MEBDFs. In [1] we saw that for $k = 4$ the $A(\alpha)$-stability angle of the EBNDF was larger than the one corresponding to the EBDF. We also saw, again for $k = 4$, that the $A(\alpha)$-stability angles of the EBDF and the EBNDF were larger than the angles corresponding to the ENDF and the ENBDF, respectively.

This time, for $k = 4$, when we have changed the predictors of the MEBDF scheme into NDFs, we have obtained methods with larger stability regions than the ones corresponding to the MEBDF. And in the same way that occured with the MEBDF, which region of stability is bigger than the one of the EBDF, our new methods (MEBNDF, MENBDF and MENDF) have better stability properties than the EBNDF, the ENBDF and the ENDF.

4. Algorithmic Implementation

The methods BDF and NDF used in the predicting stages, can be written using backward differences (expressions (2.1) and (2.6)). In order to use the same scheme as the predictor during the programming of the MEBDF, we will write the corrector of the MEBDF method (expression (2.10)) using backward differences too:

$$\begin{aligned} &\sum_{j=0}^{k} \alpha_j y_{n+j} = h\hat{\beta}_k f_{n+k} + h\beta_{k+1}\bar{f}_{n+k+1} + h(\beta_k - \hat{\beta}_k)\bar{f}_{n+k} \\ \Rightarrow &\sum_{j=1}^{k} m_{k,j} \nabla^j y_{n+k} = h\hat{\beta}_k f_{n+k} + h\beta_{k+1}\bar{f}_{n+k+1} + h(\beta_k - \hat{\beta}_k)\bar{f}_{n+k}, \end{aligned} \tag{4.1}$$

where

$$M = (m_{k,j}) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ \frac{18}{23} & \frac{5}{23} & 0 & 0 \\ \frac{132}{197} & \frac{48}{197} & \frac{17}{197} & 0 \\ \frac{1500}{2501} & \frac{606}{2501} & \frac{284}{2501} & \frac{111}{2501} \end{pmatrix}. \tag{4.2}$$

The coefficients corresponding to k are in the k-th row of the matrix M.

In [17], an alternative way to write the left hand side of the expressions (2.1) and (2.6), expressions corresponding to BDFs and NDFs, is introduced:

$$\sum_{j=1}^{k} \frac{1}{j} \nabla^j y_{n+k} = \gamma_k \left(y_{n+k} - y_{n+k}^{(0)} \right) + \sum_{j=1}^{k} \gamma_j \nabla^j y_{n+k-1}, \tag{4.3}$$

where

$$\begin{cases} \gamma_j = \sum\limits_{l=1}^{j} \frac{1}{l}, \\ y_{n+k}^{(0)} = \sum\limits_{j=0}^{k} \nabla^j y_{n+k-1} = \nabla^0 y_{n+k-1} + \nabla y_{n+k-1} + ... + \nabla^k y_{n+k-1}, \\ y_{n+k} - y_{n+k}^{(0)} = \nabla^{k+1} y_{n+k}. \end{cases} \tag{4.4}$$

The identity (4.3) shows that equations (2.1) and (2.6) are equivalent to:

$$(1-\kappa)\,\gamma_k \left(y_{n+k} - y_{n+k}^{(0)} \right) + \sum_{j=1}^{k} \gamma_j \nabla^j y_{n+k-1} = h f_{n+k}. \tag{4.5}$$

In the case of the BDFs, $\kappa = 0$ and for the NDFs, the values of κ are in Table 2.1. For the predictors, we have evaluated the implicit formula (4.5) using the Newton method. The correction to the current iterate $y_{n+k}^{(i+1)} = y_{n+k}^{(i)} + \Delta^{(i)}$ has been obtained by solving:

$$\begin{aligned} &\left(I - \frac{h}{(1-\kappa)\,\gamma_k} J \right) \Delta^{(i)} \\ =&\frac{h}{(1-\kappa)\,\gamma_k} f\left(x_{n+k}, y_{n+k}^{(i)} \right) - \frac{1}{(1-\kappa)\,\gamma_k} \sum_{j=1}^{k} \gamma_j \nabla^j y_{n+k-1} - \left(y_{n+k}^{(i)} - y_{n+k}^{(0)} \right), \end{aligned} \tag{4.6}$$

where J is the Jacobian of $f(x, y)$.

Using the previous idea, we have developed and alternative formula of the left hand side of (4.1). In this way we have obtained the next expression for the corrector of the MEBDF:

$$\sum_{j=1}^{k} m_{k,j} \nabla^j y_{n+k} = \tilde{\gamma}_{k,k} \left(y_{n+k} - y_{n+k}^{(0)} \right) + \sum_{j=1}^{k} \tilde{\gamma}_{k,j} \nabla^j y_{n+k-1}, \tag{4.7}$$

where

$$(\tilde{\gamma}_{k,j}) = \left(\sum_{l=1}^{j} m_{k,l} \right) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ \frac{18}{23} & 1 & 0 & 0 \\ \frac{132}{197} & \frac{180}{197} & 1 & 0 \\ \frac{1500}{2501} & \frac{2106}{2501} & \frac{2390}{2501} & 1 \end{pmatrix}. \tag{4.8}$$

Taking into account expressions (4.7) and (4.8), expression (4.1) can be written as:

$$\begin{aligned} &\left(y_{n+k} - y_{n+k}^{(0)} \right) + \sum_{j=1}^{k} \frac{\tilde{\gamma}_{k,j}}{\tilde{\gamma}_{k,k}} \nabla^j y_{n+k-1} \\ &= h \frac{\hat{\beta}_k}{\tilde{\gamma}_{k,k}} f_{n+k} + h \frac{\beta_{k+1}}{\tilde{\gamma}_{k,k}} \bar{f}_{n+k+1} + h \frac{(\beta_k - \hat{\beta}_k)}{\tilde{\gamma}_{k,k}} \bar{f}_{n+k}, \end{aligned} \tag{4.9}$$

where y_{n+k} is computed as the solution of the implicit formula (4.9) using Newton's method. And the correction of the current iterate $y_{n+k}^{(i+1)} = y_{n+k}^{(i)} + \Delta^{(i)}$, is obtained by solving the next equation:

$$\begin{aligned} \left(I - h \frac{\hat{\beta}_k}{\tilde{\gamma}_{k,k}} J \right) \Delta^{(i)} = h \frac{\hat{\beta}_k}{\tilde{\gamma}_{k,k}} f\left(x_{n+k}, y_{n+k}^{(i)} \right) + h \frac{\beta_{k+1}}{\tilde{\gamma}_{k,k}} \bar{f}_{n+k+1} + h \frac{(\beta_k - \hat{\beta}_k)}{\tilde{\gamma}_{k,k}} \bar{f}_{n+k} \\ - \sum_{j=1}^{k} \frac{\tilde{\gamma}_{k,j}}{\tilde{\gamma}_{k,k}} \nabla^j y_{n+k-1} - \left(y_{n+k}^{(i)} - y_{n+k}^{(0)} \right), \end{aligned} \tag{4.10}$$

Table 5.1: Results for integration of Example 1.

x	y_i	Exact solution	Error in MEBDF	Error in MEBNDF	Error in MENBDF	Error in MENDF
5	y_1	$0.673794699908547 \cdot 10^{-2}$	$1.1205 \cdot 10^{-6}$	$6.8914 \cdot 10^{-7}$	$2.6859 \cdot 10^{-7}$	$1.2205 \cdot 10^{-7}$
	y_2	$0.673794699908547 \cdot 10^{-2}$	$8.8475 \cdot 10^{-8}$	$8.7257 \cdot 10^{-7}$	$4.6561 \cdot 10^{-8}$	$1.9257 \cdot 10^{-7}$
10	y_1	$0.453999297624848 \cdot 10^{-4}$	$8.1129 \cdot 10^{-10}$	$8.5045 \cdot 10^{-10}$	$1.2195 \cdot 10^{-11}$	$3.7435 \cdot 10^{-11}$
	y_2	$0.453999297624848 \cdot 10^{-4}$	$9.2483 \cdot 10^{-10}$	$9.1614 \cdot 10^{-10}$	$3.7448 \cdot 10^{-11}$	$8.9026 \cdot 10^{-11}$
20	y_1	$0.206115362243856 \cdot 10^{-8}$	$5.7370 \cdot 10^{-15}$	$3.3317 \cdot 10^{-15}$	$2.0302 \cdot 10^{-15}$	$2.3632 \cdot 10^{-15}$
	y_2	$0.206115362243856 \cdot 10^{-8}$	$1.8692 \cdot 10^{-15}$	$1.2429 \cdot 10^{-15}$	$3.4064 \cdot 10^{-15}$	$9.9083 \cdot 10^{-16}$

Table 5.2: Results for integration of Example 2.

x	y_i	Exact solution	Error in MEBDF	Error in MEBNDF	Error in MENBDF	Error in MENDF
1	y_1	$0.303265331217737 \cdot 10^{0}$	$7.0144 \cdot 10^{-5}$	$1.8023 \cdot 10^{-5}$	$1.5822 \cdot 10^{-3}$	$1.3052 \cdot 10^{-3}$
	y_2	$0.303265330376617 \cdot 10^{0}$	$6.4172 \cdot 10^{-4}$	$4.5880 \cdot 10^{-4}$	$3.6462 \cdot 10^{-4}$	$2.1813 \cdot 10^{-4}$
	y_3	$-0.303265329336016 \cdot 10^{0}$	$2.4834 \cdot 10^{-4}$	$1.0859 \cdot 10^{-4}$	$3.2916 \cdot 10^{-5}$	$1.2826 \cdot 10^{-4}$
5	y_1	$0.410424993119494 \cdot 10^{-1}$	$2.7327 \cdot 10^{-5}$	$2.4341 \cdot 10^{-5}$	$2.7694 \cdot 10^{-5}$	$2.4149 \cdot 10^{-5}$
	y_2	$0.410424993119494 \cdot 10^{-1}$	$2.7327 \cdot 10^{-5}$	$2.4341 \cdot 10^{-5}$	$2.7694 \cdot 10^{-5}$	$2.4149 \cdot 10^{-5}$
	y_3	$-0.410424993119494 \cdot 10^{-1}$	$2.7327 \cdot 10^{-5}$	$2.4341 \cdot 10^{-5}$	$2.7694 \cdot 10^{-5}$	$2.4149 \cdot 10^{-5}$
10	y_1	$0.336897349954273 \cdot 10^{-2}$	$2.3204 \cdot 10^{-6}$	$2.0679 \cdot 10^{-6}$	$2.3595 \cdot 10^{-6}$	$2.0593 \cdot 10^{-6}$
	y_2	$0.336897349954273 \cdot 10^{-2}$	$2.3204 \cdot 10^{-6}$	$2.0679 \cdot 10^{-6}$	$2.3595 \cdot 10^{-6}$	$2.0593 \cdot 10^{-6}$
	y_3	$-0.336897349954273 \cdot 10^{-2}$	$2.3204 \cdot 10^{-6}$	$2.0679 \cdot 10^{-6}$	$2.3595 \cdot 10^{-6}$	$2.0593 \cdot 10^{-6}$

where J is the Jacobian of $f(x,y)$.

5. Numerical Results

In this section we present some numerical results to compare the performance of the methods MEBNDF, MENBDF and MENDF with that of MEBDF method.

Example 1. We consider the following stiff system as considered by Cash in [2]:

$$\begin{cases} y_1' = -\alpha y_1 - \beta y_2 + (\alpha+\beta-1)e^{-x}, \\ y_2' = \beta y_1 - \alpha y_2 + (\alpha-\beta-1)e^{-x}, \end{cases} \qquad \text{with initial value} \qquad y(0) = (1,1)^T.$$

The eigenvalues of the Jacobian matrix are $-\alpha \pm \beta i$, and its exact solution is this one: $y_1(x) = y_2(x) = e^{-x}$. We have solved the problem for $\alpha = 1, \beta = 15$. In Table 5.1 we show the results obtained for the integration of this problem. We have taken 200 steps and $k = 3$ to integrate Example 1. It can be seen that the results obtained by the new methods are superior or similar to the ones obtained by the MEBDF. The MENDF method is the one with the smallest error at the end of the interval of integration, and the MEBNDF is the next. Methods MENBDF and MEBDF show similar errors at the end of the interval of integration.

Example 2. Consider the system of differential equations considered in [11]:

$$\begin{cases} y_1' = -20y_1 - 0.25y_2 - 19.75y_3, \\ y_2' = 20y_1 - 20.25y_2 + 0.25y_3, \\ y_3' = 20y_1 - 19.75y_2 - 0.25y_3, \end{cases} \qquad \text{with initial value} \qquad y(0) = (1,0,-1)^T.$$

Table 5.3: Results for integration of Example 3.

x	y_i	Exact solution	Error in MEBDF	Error in MEBNDF	Error in MENBDF	Error in MENDF
	y_1	$0.996787780748254 \cdot 10^{0}$	$2.0504 \cdot 10^{-3}$	$1.8468 \cdot 10^{-3}$	$2.2934 \cdot 10^{-3}$	$2.0479 \cdot 10^{-3}$
0.1	y_2	$0.673794699908547 \cdot 10^{-2}$	$2.0503 \cdot 10^{-3}$	$1.8468 \cdot 10^{-3}$	$2.2934 \cdot 10^{-3}$	$2.0479 \cdot 10^{-3}$
	y_3	$0.674409121143880 \cdot 10^{-2}$	$1.9529 \cdot 10^{-3}$	$1.6577 \cdot 10^{-3}$	$2.2008 \cdot 10^{-3}$	$1.8026 \cdot 10^{-3}$
	y_1	$0.951229424514602 \cdot 10^{0}$	$6.2063 \cdot 10^{-9}$	$5.4726 \cdot 10^{-9}$	$6.2242 \cdot 10^{-9}$	$5.3693 \cdot 10^{-9}$
0.5	y_2	$0.138879438649640 \cdot 10^{-10}$	$1.6818 \cdot 10^{-11}$	$6.8684 \cdot 10^{-12}$	$1.8554 \cdot 10^{-11}$	$9.3807 \cdot 10^{-12}$
	y_3	$0.138879438649640 \cdot 10^{-10}$	$3.2927 \cdot 10^{-11}$	$1.5989 \cdot 10^{-11}$	$7.4311 \cdot 10^{-11}$	$5.7377 \cdot 10^{-11}$
	y_1	$0.904837418035960 \cdot 10^{0}$	$5.8876 \cdot 10^{-9}$	$5.1992 \cdot 10^{-9}$	$5.9030 \cdot 10^{-9}$	$5.0985 \cdot 10^{-9}$
1	y_2	$0.192874984796392 \cdot 10^{-21}$	$2.5275 \cdot 10^{-20}$	$1.5331 \cdot 10^{-20}$	$1.5831 \cdot 10^{-20}$	$9.5993 \cdot 10^{-21}$
	y_3	$0.192874984796392 \cdot 10^{-21}$	$2.5179 \cdot 10^{-20}$	$1.4970 \cdot 10^{-20}$	$6.3200 \cdot 10^{-20}$	$3.7731 \cdot 10^{-20}$

Table 5.4: Results for integration of Example 4.

x	y_i	Exact solution	Error in MEBDF	Error in MEBNDF	Error in MENBDF	Error in MENDF
	y_1	$0.673794699908547 \cdot 10^{-2}$	$4.3118 \cdot 10^{-5}$	$8.0532 \cdot 10^{-5}$	$1.2225 \cdot 10^{-6}$	$2.2914 \cdot 10^{-5}$
5	y_2	$0.673794699908547 \cdot 10^{-2}$	$9.0623 \cdot 10^{-5}$	$3.5681 \cdot 10^{-5}$	$4.6216 \cdot 10^{-5}$	$2.2964 \cdot 10^{-5}$
	y_3	5	$8.8818 \cdot 10^{-16}$	$8.8818 \cdot 10^{-16}$	$8.8818 \cdot 10^{-16}$	$8.8818 \cdot 10^{-16}$
	y_1	$0.453999297624848 \cdot 10^{-4}$	$1.4443 \cdot 10^{-5}$	$3.3460 \cdot 10^{-6}$	$2.9658 \cdot 10^{-6}$	$1.3461 \cdot 10^{-6}$
10	y_2	$0.453999297624848 \cdot 10^{-4}$	$4.5723 \cdot 10^{-6}$	$1.1982 \cdot 10^{-5}$	$7.5281 \cdot 10^{-7}$	$1.5005 \cdot 10^{-6}$
	y_3	10	$1.7764 \cdot 10^{-14}$	$1.7764 \cdot 10^{-14}$	$1.7764 \cdot 10^{-14}$	$1.7764 \cdot 10^{-14}$
	y_1	$0.206115362243856 \cdot 10^{-8}$	$3.4504 \cdot 10^{-7}$	$4.2692 \cdot 10^{-9}$	$9.3518 \cdot 10^{-9}$	$4.5262 \cdot 10^{-9}$
20	y_2	$0.206115362243856 \cdot 10^{-8}$	$1.0910 \cdot 10^{-8}$	$2.4810 \cdot 10^{-7}$	$9.6023 \cdot 10^{-9}$	$6.3324 \cdot 10^{-9}$
	y_3	20	$1.7764 \cdot 10^{-14}$	$1.7764 \cdot 10^{-14}$	$1.7764 \cdot 10^{-14}$	$1.7764 \cdot 10^{-14}$

The exact solution of this problem is:

$$\begin{cases} y_1(x) = \frac{1}{2}\left(e^{-0.5x} + e^{-20x}\left(\cos 20x + \sin 20x\right)\right), \\ y_2(x) = \frac{1}{2}\left(e^{-0.5x} - e^{-20x}\left(\cos 20x - \sin 20x\right)\right), \\ y_3(x) = -\frac{1}{2}\left(e^{-0.5x} + e^{-20x}\left(\cos 20x - \sin 20x\right)\right). \end{cases}$$

The system has been integrated using MEBDF, MEBNDF, MENBDF and MENDF. The results are tabulated in Table 5.2. We have taken 50 steps and $k = 3$ to integrate Example 2. It can be seen that again, it is the MENDF which gets the smallest error at the end of the interval of integration, and the MEBNDF is the next.

Example 3. We consider the following stiff system considered by Hosseini and Hojjati in [13].

$$\begin{cases} y_1' = -0.1y_1 - 49.9y_2, \\ y_2' = -50y_2, \\ y_3' = 70y_2 - 120y_3, \end{cases} \qquad \text{with initial value} \qquad y(0) = (2, 1, 2)^T,$$

and with the stiffness ratio 1200. The exact solution is

$$\begin{cases} y_1(x) = e^{-50x} + e^{-0.1x}, \\ y_2(x) = e^{-50x}, \\ y_3(x) = e^{-50x} + e^{-120x}. \end{cases}$$

Table 5.5: Results for integration of Example 5.

x	y_i	Exact solution	Error in MEBDF	Error in MEBNDF	Error in MENBDF	Error in MENDF
3	y_1	$-0.247825652536129 \cdot 10^{-6}$	$1.9890 \cdot 10^{-10}$	$1.7682 \cdot 10^{-10}$	$2.0120 \cdot 10^{-10}$	$1.7511 \cdot 10^{-10}$
	y_2	$0.497870683678639 \cdot 10^{-1}$	$1.9975 \cdot 10^{-5}$	$1.7758 \cdot 10^{-5}$	$2.0206 \cdot 10^{-5}$	$1.7587 \cdot 10^{-5}$
5	y_1	$-0.453908515921664 \cdot 10^{-8}$	$3.6390 \cdot 10^{-12}$	$3.2348 \cdot 10^{-12}$	$3.6807 \cdot 10^{-12}$	$3.2032 \cdot 10^{-12}$
	y_2	$0.673794699908547 \cdot 10^{-2}$	$2.7004 \cdot 10^{-6}$	$2.4006 \cdot 10^{-6}$	$2.7313 \cdot 10^{-6}$	$2.3770 \cdot 10^{-6}$

In Table 5.3 we list the error of the computed solutions. We have taken 50 steps and $k = 4$ to integrate Example 3. We observe that the performance of the new methods is correct, not being the results obtained by our methods inferior to the ones obtained by the MEBDF. The MENDF is the method which gets the best results. We have to take into account that apart from obtaining the best accuracy, the MENDF is which presents the best $A(\alpha)$-stability when $k = 4$.

Example 4. We consider another stiff initial value problem considered in [12]:

$$\begin{cases} y_1' = -\alpha y_1 - \beta y_2 + (\alpha + \beta - 1)e^{-x}, \\ y_2' = \beta y_1 - \alpha y_2 + (\alpha - \beta - 1)e^{-x}, \\ y_3' = 1, \end{cases} \qquad \text{with initial value} \qquad y(0) = (1, 1, 0)^T.$$

The exact solution is $y_1(x) = y_2(x) = e^{-x}$, and $y_3(x) = x$. In Table 5.4, we give the results obtained for the solution of this problem for the case $\alpha = 1, \beta = 15$, for 200 steps and $k = 4$. Again, our new methods are superior or similar to the MEBDF method, being the MENDF the one obtaining the most accurate results.

Example 5. Consider the non-linear system of differential equations considered in [14]:

$$\begin{cases} y_1' = \lambda y_1 + y_2^2, \\ y_2' = -y_2, \end{cases} \qquad \text{with initial value} \qquad y(0) = \left(-\frac{1}{\lambda + 2}, 1\right)^T,$$

where $\lambda = 10^4$. The exact solution of this problem is this one:

$$y_1(x) = \frac{-e^{-2x}}{(\lambda + 2)}, \quad y_2(x) = e^{-x}.$$

We have integrated this problem in $T = [0, 5]$, we have taken 60 steps and $k = 4$. The results are tabulated in Table 5.5 and the same conclusions as in the previous four examples are repeated.

6. Conclusions

In this paper we have built three different new methods (MENDF, MENBDF and MEBNDF) taking as basis the MEBDF method. This new methods follow the MEBDF scheme but they use NDF predictors in one of the predicting stages or in both of them. The result is a set of three methods that maintains the accuracy and the stability characteristics of the MEBDF method in orders $p = 2, 3, 4$. In order $p = 5$, the new methods also maintain the accuracy of the MEBDF while improving the $A(\alpha)$-stability.

We have used backward differences to program the predicting stages (BDF and NDF), as well as the expression of the corrector. The algorithmic implementation done is the same for all the MEBDF class (MEBDF, MENDF, MENBDF and MEBNDF), so it represents and efficient way to program four different methods.

Acknowledgements. The author E. Alberdi would like to thank Lea Artibai ikastetxea for the opportunity she has been given to do research on this area.

References

[1] E. Alberdi and J.J. Anza, A predictor modification to the EBDF method for stiff systems, *J. Comput. Math.*, **29**:2 (2011), 199-214.

[2] J.R. Cash, On the integration of stiff systems of ODEs using extended backward differentiation formulae, *Numer. Math.*, **34**:2 (1980), 235-246.

[3] J.R. Cash, Second derivative extended backward differentation formulas for the numerical integration of stiff systems, *SIAM J. Numer. Anal.*, **18**:1 (1981), 21-36.

[4] J.R. Cash, The integration of stiff initial value problems in ODEs using modified extended backward differentiation formula, *Comput. Math. Appl.*, **9**:5 (1983), 645-657.

[5] J.R. Cash and S. Considine, An MEBDF code for stiff initial value problems, *ACM Trans. Math. Software*, **18**:2 (1992), 142-155.

[6] G. Dahlquist, A special stability problem for linear multistep methods, *BIT*, **3** (1963), 27-43.

[7] W.H. Enright, Second derivative multistep methods for stiff ordinary differential equations, *SIAM J. Numer. Anal.*, **11**:2 (1974), 321-331.

[8] C. Fredebeul, A-BDF: a generalization of the backward differentiation formulae, *SIAM J. Numer. Anal.*, **35**:5 (1998), 1917-1938.

[9] C.W. Gear, Numerical Initial Value Problems in Ordinary Differential Equations, Prentice-Hall, New Jersey, 1971.

[10] E. Hairer and G. Wanner, Solving ordinary differential equations, II, Stiff and Differential-Algebraic Problems, Springer, Berlin, 1996.

[11] G. Hojjati, M.Y. Rahimi Ardabili and S.M. Hosseini, A-EBDF: an adaptative method for numerical solution of stiff systems of ODEs, *Math. Comput. Simul.*, **66** (2004), 33-41.

[12] G. Hojjati, M.Y. Rahimi Ardabili and S.M. Hosseini, New second derivative multistep methods for stiff systems, *Appl. Math. Model.*, **30** (2006), 466-476.

[13] S.M. Hosseini and G. Hojjati, Matrix-free MEBDF method for the solution of stiff systems of ODEs, *Math. Comput. Modell.*, **29** (1999), 67-77.

[14] Gamal A.F. Ismail and Iman H. Ibrahim, A new higher effective P-C methods for stiff systems, *Math. Comput. Simul.*, **47** (1998), 541-552.

[15] Gamal A.F. Ismail and Iman H. Ibrahim, New efficient second derivative multistep methods for stiff systems, *Appl. Math. Model.*, **23** (1999), 279-288.

[16] J.D. Lambert, Computational Methods in Ordinary Differential Equations, Wiley, New York, 1972.

[17] L.F. Shampine and M.W. Reichelt, The MATLAB ODE suite, *SIAM J. Sci. Comput.*, **18**:1 (1997), 1-22.