

Boundary Value Methods for the Numerical Approximation of Ordinary Differential Equations

Luigi Brugnano

Dipartimento di Energetica, Università di Firenze
Via C. Lombroso 6/17, 50134 Firenze, Italy

Abstract. Many numerical methods for the approximation of ordinary differential equations (ODEs) are obtained by using Linear Multistep Formulae (LMF). Such methods, however, in their usual implementation suffer of heavy theoretical limitations, summarized by the two well known Dahlquist barriers. For this reason, Runge-Kutta schemes have become more popular than LMF, in the last twenty years. This situation has recently changed, with the introduction of Boundary Value Methods (BVMs), which are methods still based on LMF. Their main feature consists in approximating a given continuous initial value problem (IVP) by means of a discrete boundary value problem (BVP). Such use allows to avoid order barriers for stable methods. Moreover, BVMs provide several families of methods, which make them very flexible and computationally efficient. In particular, we shall see that they allow a natural implementation of efficient mesh selection strategies.

1 Introduction

Usually, the solution of an initial value ODE problem,

$$y' = f(t, y), \quad t \in [t_0, T], \quad y(t_0) = \eta, \quad (1)$$

is obtained by using a k -step LMF,

$$\sum_{i=0}^k \alpha_i y_{n+i} = h \sum_{i=0}^k \beta_i f_{n+i}. \quad (2)$$

In the previous equation, y_n denotes, as usual, the discrete approximation of the solution $y(t)$ at $t = t_n \equiv t_0 + nh$, $h = (T - t_0)/N$, and $f_n \equiv f(t_n, y_n)$. Since (2) is a k th order difference equation, then k conditions need to be imposed to obtain the discrete solution. Usually, such conditions are obtained by fixing the first k values, y_0, \dots, y_{k-1} , of the discrete solution. That is, the continuous IVP (1) is approximated by means of a discrete IVP. This approach is very straightforward. However, it suffers of heavy theoretical limitations, summarized by the two Dahlquist barriers.

It is possible to get rid of such limitations by suitably modifying the use of LMF. This is, in fact, the idea on which Boundary Value Methods rely. Early

references on such methods can be found in [4, 12]. However, only in the last three years such methods have been systematically studied, starting from [14]. In particular, a linear stability theory has been recently devised [6], which has made possible the derivation of several families of methods, each containing stable methods of arbitrarily high order. In this paper, a brief review on BVMs is presented, along with a mesh selection strategy which is very efficient for such methods.

In Sect. 2 the main facts about BVMs will be recalled, and in Sect. 3 the principal families of methods are sketched. In Sect. 4 the block version of the methods is presented, along with the mesh selection strategy. Finally, in Sect. 5 some numerical examples on difficult stiff problems are reported, showing the effectiveness of BVMs.

2 Boundary Value Methods

Suppose, when approximating (1) by means of (2), to fix the first $k_1 \leq k$ values of the discrete solution, y_0, \dots, y_{k_1-1} , and the last $k_2 \equiv k - k_1$ ones, y_{N-k_2+1}, \dots, y_N . In this way, the discrete problem becomes

$$\sum_{i=-k_1}^{k_2} \alpha_{i+k_1} y_{n+i} = h \sum_{i=-k_1}^{k_2} \beta_{i+k_1} f_{n+i}, \quad n = k_1, \dots, N - k_2, \quad (3)$$

$$y_0, \dots, y_{k_1-1}, \quad y_{N-k_2+1}, \dots, y_N, \quad \text{fixed.}$$

That is, the continuous IVP (1) is approximated by means of a discrete BVP. This approach defines a BVM with (k_1, k_2) -boundary conditions. Observe that, for $k_1 = k$ and, therefore, $k_2 = 0$, problem (3) becomes an IVP, so that BVMs contain as a proper subclass the usual initial value methods for ODEs based on LMF.

In order to completely exploit all the advantages of this new approach, that is, to derive effective BVMs, we need to generalize the known notions of stability. This is done by introducing the following polynomials [6].

Definition 1. Let $p(z)$ be a polynomial of degree k , and let $|z_1| \leq \dots \leq |z_k|$ be its roots. We say that $p(z)$ is a

- $S_{k_1 k_2}$ -polynomial if $|z_{k_1}| < 1 < |z_{k_1+1}|$;
- $N_{k_1 k_2}$ -polynomial if $|z_{k_1}| \leq 1 < |z_{k_1+1}|$, with simple zeros of unit modulus.

Observe that for $k_1 = k$ and $k_2 = 0$, one obtains the usual Schur polynomials and von Neumann polynomials, respectively.

Now, let us denote by $\rho(z)$ and $\sigma(z)$ the two polynomials associated with the LMF (2), and, as usual, let $\pi(z, q) = \rho(z) - q\sigma(z)$ denote the stability polynomial. The following definitions are then stated [6].

Definition 2. A BVM with (k_1, k_2) -boundary conditions is

- $0_{k_1 k_2}$ -stable if the corresponding polynomial $\rho(z)$ is a $N_{k_1 k_2}$ -polynomial;
- (k_1, k_2) -absolutely stable, for a given $q \in \mathbb{C}$, if the polynomial $\pi(z, q)$ is a $S_{k_1 k_2}$ -polynomial. The region

$$D_{k_1 k_2} = \{q \in \mathbb{C} : \pi(z, q) \text{ is a } S_{k_1 k_2}\text{-polynomial}\}$$

is said *region of (k_1, k_2) -absolute stability*;

- $A_{k_1 k_2}$ -stable if $\mathbb{C}^- \subseteq D_{k_1 k_2}$.

Observe that the previous definitions reduce to the usual stability notions, when $k_1 = k$ and $k_2 = 0$.

The problem of finding the $k - 1$ additional values

$$y_1, \dots, y_{k_1-1}, \quad y_{N-k_2+1}, \dots, y_N,$$

in (3) is easily solved by treating them as unknowns. This is done by introducing a set of $k - 1$ *additional equations*, independent of those provided by the *main formula* (3). Such equations are conveniently derived by a set of $k_1 - 1$ *initial additional methods*,

$$\sum_{i=0}^r \alpha_i^{(j)} y_i = h \sum_{i=0}^r \beta_i^{(j)} f_i, \quad j = 1, \dots, k_1 - 1, \quad (4)$$

and k_2 *final* ones,

$$\sum_{i=0}^r \alpha_{r-i}^{(j)} y_{N-i} = h \sum_{i=0}^r \beta_{r-i}^{(j)} f_{N-i}, \quad j = N - k_2 + 1, \dots, N. \quad (5)$$

Such additional methods have to be chosen with the same order of the main formula (3), in order to have the same order for the whole composite method (3)–(5). As we shall see in the next section, for almost all BVMs the number r of steps of the additional methods is the same of the main formula, that is $r = k$. Moreover, each BVM will be coupled with the most appropriate set of additional methods.

3 Families Of BVMs

In this section, we present the most important families of BVMs. Of such families, one is here introduced for the first time. A common feature for all these families is that all of them contain $0_{k_1 k_2}$ -stable, $A_{k_1 k_2}$ -stable methods of arbitrary high order. This, in turn, confirms that there are no more order barriers for stable BVMs.

3.1 Generalized BDF

The *Generalized Backward Differentiation Formulae (GBDF)* have the following form [6], for all $k \geq 1$,

$$\sum_{i=-\nu}^{k-\nu} \alpha_{i+\nu} y_{n+i} = h f_n, \quad n = \nu, \dots, N - k + \nu, \quad (6)$$

where

$$\nu = \begin{cases} \frac{k+2}{2}, & \text{for even } k, \\ \frac{k+1}{2}, & \text{for odd } k. \end{cases}$$

The coefficients $\{\alpha_i\}$ are uniquely determined by imposing a $O(h^{k+1})$ truncation error. Such formulae are to be used with $(\nu, k-\nu)$ -boundary condition or, equivalently, they are conveniently coupled with the following set of initial additional methods,

$$\sum_{i=0}^k \alpha_i^{(j)} y_i = h f_j, \quad j = 1, \dots, \nu - 1,$$

and final additional ones,

$$\sum_{i=0}^k \alpha_{k-i}^{(j)} y_{N-i} = h f_j, \quad j = N - k + \nu + 1, \dots, N.$$

The coefficients of the additional formulae are uniquely determined in order to have the same truncation error of the main formula (6).

Observe that the formulae obtained for $\nu = k$ are the usual BDF, which are 0-unstable, for $k \geq 7$, while GBDF are $0_{\nu, k-\nu}$ -stable, $A_{\nu, k-\nu}$ -stable, and with order of convergence k , for all $k \geq 1$.

3.2 Generalized Adams Methods

The *Generalized Adams Methods (GAMs)* are BVMs in the following form [10],

$$y_n - y_{n-1} = h \sum_{i=-\nu}^{k-\nu} \beta_{i+\nu} f_{n+i}, \quad n = \nu, \dots, N - k + \nu. \quad (7)$$

where ν is defined according to

$$\nu = \begin{cases} \frac{k+1}{2}, & \text{for odd } k, \\ \frac{k}{2}, & \text{for even } k, \end{cases} \quad (8)$$

and the coefficients $\{\beta_i\}$ are uniquely determined in order to have a method of order $k + 1$. For each $k \geq 1$, they must be used with $(\nu, k - \nu)$ -boundary conditions, and are $0_{\nu, k-\nu}$ -stable, $A_{\nu, k-\nu}$ -stable methods. They are conveniently used with the following set of additional initial methods,

$$y_j - y_{j-1} = h \sum_{i=0}^k \beta_i^{(j)} f_i, \quad j = 1, \dots, \nu - 1,$$

and final ones,

$$y_j - y_{j-1} = h \sum_{i=0}^k \beta_{k-i}^{(j)} f_{N-i}, \quad j = N - k + \nu + 1, \dots, N.$$

The coefficients of the additional methods are uniquely determined by imposing each formula to have the same order, $k + 1$, of the main method (7).

The formulae obtained in correspondence of odd values of k are also called *Extended Trapezoidal Rules (ETRs)* [3], since the formula obtained for $k = 1$ is the trapezoidal rule. Such formulae belong to the class of *symmetric schemes*, that we shall consider later.

3.3 Extended Trapezoidal Rules Of Second Kind (ETR_{2s})

Let us consider the methods having the following general form,

$$\sum_{i=-\nu}^{k-\nu} \alpha_{i+\nu} y_{n+i} = h(\beta f_n + (1 - \beta) f_{n-1}), \quad n = \nu, \dots, N - k + \nu, \quad (9)$$

where ν is chosen according to (8), and the coefficients $\{\alpha_i\}$ and β are uniquely determined by imposing a $O(h^{k+2})$ truncation error. The formulae obtained for k even will be called *unsymmetric ETR_{2s}s*, while those obtained for k odd have been called *ETR_{2s}s* [5, 8].

In particular, in the latter case one obtains $\beta = 1 - \beta = \frac{1}{2}$, and the corresponding formulae belong to the class of symmetric schemes.

All (unsymmetric) ETR_{2s}s are $0_{\nu, k-\nu}$ -stable, $A_{\nu, k-\nu}$ -stable formulae, and must be used with $(\nu, k - \nu)$ -boundary conditions. The following set of additional initial methods,

$$\sum_{i=0}^k \alpha_i^{(j)} y_i = h(\beta^{(j)} f_j + (1 - \beta^{(j)}) f_{j-1}), \quad j = 1, \dots, \nu - 1,$$

and final ones,

$$\sum_{i=0}^k \alpha_{k-i}^{(j)} y_{N-i} = h(\beta^{(j)} f_j + (1 - \beta^{(j)}) f_{j-1}), \quad j = N - k + \nu + 1, \dots, N,$$

is conveniently associated with the main formula (9). The coefficients $\{\alpha_i^{(j)}\}$ and $\beta_i^{(j)}$ of each additional method are uniquely determined by imposing the same order, $k + 1$, of the main formula.

3.4 Symmetric Schemes

We have collected as *symmetric schemes* [5, 8], BVMs having the following general properties,

- they have an odd number of steps, $k = 2\nu - 1$, and must be used with $(\nu, \nu - 1)$ -boundary conditions (i.e., they require $\nu - 1$ initial and $\nu - 1$ final additional methods);
- the corresponding polynomials $\rho(z)$ have skew-symmetric coefficients. That is, $z^k \rho(z^{-1}) = -\rho(z)$;
- the corresponding polynomials $\sigma(z)$ have symmetric coefficients. That is, $z^k \sigma(z^{-1}) = \sigma(z)$.
- $D_{\nu, \nu-1} \equiv \mathbb{C}^-$.

Such schemes are conveniently used for either approximating continuous BVPs [7], or Hamiltonian problems [5, 8].

Both ETRs and ETR_{2s} are symmetric schemes. Another important family of BVMs fits this class, namely *Top Order Methods (TOMs)* [1]. Such method have the following general form,

$$\sum_{i=0}^{\nu-1} \alpha_i (y_{n-\nu+i} - y_{n+\nu-1-i}) = h \sum_{i=0}^{\nu-1} \beta_i (f_{n-\nu+i} + f_{n+\nu-1-i}), \quad (10)$$

$$n = \nu, \dots, N - \nu + 1,$$

where the coefficients $\{\alpha_i\}$ and $\{\beta_i\}$ are determined in order to have the maximum possible order for a k -step formula, that is $p = 2k$. They can be conveniently used with the following initial

$$y_j - y_{j-1} = h \sum_{i=0}^{2k-1} \beta_i^{(j)} f_i, \quad j = 1, \dots, \nu - 1,$$

and final additional methods,

$$y_j - y_{j-1} = h \sum_{i=0}^{2k-1} \beta_{2k-1-i}^{(j)} f_{N-i}, \quad j = N - \nu + 2, \dots, N.$$

The unknown coefficients $\{\beta_i^{(j)}\}$ of the additional methods are uniquely determined by imposing that they have the same order $p = 2k$ of the main formula (10).

4 Block BVMs And Mesh Selection

The arguments presented in the last section allow to consider a BVM, together with its corresponding additional methods, as a composite method. Having fixed a suitable N , such method allows to pass from the approximation at $t = t_0$ to the one at $t = t_N$. It is then possible to discretize the interval $[t_0, T]$ by using two different meshes: a coarser one containing the $p + 1$ points

$$\tau_i = \tau_{i-1} + \hat{h}_i, \quad i = 1, \dots, p, \quad \tau_0 \equiv t_0, \quad \tau_p \equiv T,$$

and a finer one, which discretizes each subinterval $[\tau_{i-1}, \tau_i]$, $i = 1, \dots, p$, with a constant finer stepsize $h_i = \hat{h}_i/N$.

In more detail, by using the initial condition $y_0 = \eta$ provided by the continuous problem (1), we can apply the (composite) BVM over the first subinterval $[\tau_0, \tau_1]$, with constant finer stepsize $h_1 = \hat{h}_1/N$. The discrete approximation of the solution at the points

$$t_j = \tau_0 + jh_1, \quad j = 1, \dots, N,$$

is then obtained. One then uses the approximated value at $t_N \equiv \tau_1$ for computing the discrete approximation over the second subinterval $[\tau_1, \tau_2]$, by using the same BVM with finer stepsize $h_2 = \hat{h}_2/N$.

It is evident that the process can be iterated $p - 2$ more times, thus providing a discrete approximation over the entire interval $[t_0, T]$.

The resulting procedure defines the block version of BVMs [8], which has two important practical implications:

1. the stepsize variation becomes very simple, since inside each block the used stepsize is constant, while one may vary the stepsize in the coarser mesh;
2. it allows a very efficient parallel implementation of such methods [2].

In [9] a novel mesh selection strategy, which is very effective for the approximation of continuous BVPs, has been introduced. We here present a different mesh selection strategy, which is very efficient for the approximation of continuous IVP. Essentially, it is based on deferred correction [15, 19, 20], which assumes, for BVMs, a very natural implementation. For simplicity, we assume that a single block implementation of BVMs is used for approximating the continuous problem (1), which will be assumed to be scalar. Then, let us denote by

$$F_p(\mathbf{y}) = \mathbf{0} \tag{11}$$

the discrete problem obtained by applying a BVM of order p , where $\mathbf{y} = (y_0, \dots, y_N)^T$ is the discrete solution. It is not difficult to verify that

$$F_p(\mathbf{y}) \equiv A_p \mathbf{y} - hB_p f(\mathbf{y}) - \begin{pmatrix} \eta \\ \mathbf{0} \end{pmatrix},$$

where $f(\mathbf{y}) = (f(t_0, y_0), \dots, f(t_N, y_N))^T$, h is the used stepsize, and A_p, B_b are $(N+1) \times (N+1)$ matrices, whose rows contain the coefficients of the considered method,

$$A_p = \begin{pmatrix} 1 & 0 & \dots & \dots & \dots & 0 \\ \alpha_0^{(1)} & \dots & \alpha_k^{(1)} & & & \\ \vdots & & \vdots & & & \\ \alpha_0^{(k_1-1)} & \dots & \alpha_k^{(k_1-1)} & & & \\ \alpha_0 & \dots & \alpha_k & & & \\ & \ddots & & \ddots & & \\ & & \ddots & & \ddots & \\ & & & \alpha_0 & \dots & \alpha_k \\ & & & \alpha_0^{(N-k_2+1)} & \dots & \alpha_k^{(N-k_2+1)} \\ & & & \vdots & & \vdots \\ & & & \alpha_0^{(N)} & \dots & \alpha_k^{(N)} \end{pmatrix}_{(N+1) \times (N+1)},$$

$$B_p = \begin{pmatrix} 0 & 0 & \dots & \dots & \dots & 0 \\ \beta_0^{(1)} & \dots & \beta_k^{(1)} & & & \\ \vdots & & \vdots & & & \\ \beta_0^{(k_1-1)} & \dots & \beta_k^{(k_1-1)} & & & \\ \beta_0 & \dots & \beta_k & & & \\ & \ddots & & \ddots & & \\ & & \ddots & & \ddots & \\ & & & \beta_0 & \dots & \beta_k \\ & & & \beta_0^{(N-k_2+1)} & \dots & \beta_k^{(N-k_2+1)} \\ & & & \vdots & & \vdots \\ & & & \beta_0^{(N)} & \dots & \beta_k^{(N)} \end{pmatrix}_{(N+1) \times (N+1)}.$$

In the above expression, we have assumed the main formula, and the corresponding additional methods, to have the same number, k , of steps. Moreover, we shall also assume that the matrix B_p has unit row sums. In the following, this assumption will hold for all the considered methods.

Let now $\hat{\mathbf{y}} = (y(t_0), \dots, y(t_N))^T$ be the restriction of the continuous solution to the mesh. Then, since all the formulae defining the BVM have the same order p , it will be

$$F_p(\hat{\mathbf{y}}) = \tau_p \equiv \begin{pmatrix} 0 \\ \tau_1 \\ \vdots \\ \tau_N \end{pmatrix}, \quad \tau_i = O(h^{p+1}), \quad i = 1, \dots, N. \quad (12)$$

From the previous equation, one readily obtains the following first order approximation,

$$\hat{\mathbf{y}} - \mathbf{y} \approx M_p^{-1} \boldsymbol{\tau}_p, \quad (13)$$

where M_p is the Jacobian matrix of F_p evaluated at $\hat{\mathbf{y}}$,

$$M_p = A_p - hB_p J_f(\hat{\mathbf{y}}),$$

and, by denoting with f_y the Jacobian of $f(t, y)$,

$$J_f(\hat{\mathbf{y}}) = \begin{pmatrix} f_y(t_0, y(t_0)) & & \\ & \ddots & \\ & & f_y(t_N, y(t_N)) \end{pmatrix}.$$

Let now put the discrete solution \mathbf{y} inside the discrete problem obtained by using a different method, of order $q > p$, over the same mesh. It will be

$$F_q(\mathbf{y}) = -\mathbf{u}, \quad (14)$$

where, in general, the vector \mathbf{u} will not be the zero vector. The following result then holds true.

Theorem 3. *Provided that the continuous solution is suitably smooth, one has (see (12) and (14)) $\mathbf{u} = \boldsymbol{\tau}_p + O(h^{p+2})$.*

Proof. Since the method defining F_q has order q , it will be

$$F_q(\hat{\mathbf{y}}) = \boldsymbol{\tau}_q \equiv O(h^{q+1}).$$

From (13) one then obtains

$$F_q(\mathbf{y}) \approx F_q(\hat{\mathbf{y}} - M_p^{-1} \boldsymbol{\tau}_p) \approx F_q(\hat{\mathbf{y}}) - M_q M_p^{-1} \boldsymbol{\tau}_p = \boldsymbol{\tau}_q - M_q M_p^{-1} \boldsymbol{\tau}_p,$$

where $M_q = A_q - hB_q J_f(\hat{\mathbf{y}})$ denotes the Jacobian matrix of F_q evaluated at $\hat{\mathbf{y}}$. Since in the above approximations the neglected terms are $O(h^{2p})$, the thesis then follows by proving that

$$M_q \mathbf{v} = \boldsymbol{\tau}_p + O(h^{p+2}), \quad \mathbf{v} = M_p^{-1} \boldsymbol{\tau}_p.$$

The entries of $\boldsymbol{\tau}_p$ are $O(h^{p+1})$ and, consequently, those of the vector \mathbf{v} will be $O(h^p)$. That is,

$$\mathbf{v} = h^p \mathbf{c} \equiv h^p \begin{pmatrix} c(t_0) \\ \vdots \\ c(t_N) \end{pmatrix},$$

where $c(t)$ is a suitably smooth function, under the made hypothesis on $y(t)$. Since the first entry of $\boldsymbol{\tau}_p$ is zero (see (12)), then $c(t_0) = 0$. Moreover, the

function $c(t)$ is not uniquely defined, so that we may assume $c'(t_0) = 0$ as well. From the previous equations, one then obtains

$$\tau_p = M_p \mathbf{v} = h^p(A_p \mathbf{c} - hB_p J_f(\hat{\mathbf{y}})\mathbf{c}) = h^p(A_p \mathbf{c} - hB_p \mathbf{c}') + h^{p+1}B_p(\mathbf{c}' - J_f(\hat{\mathbf{y}})\mathbf{c}),$$

where the vector $\mathbf{c}' = (c'(t_0), \dots, c'(t_N))^T$. Because of the following trivial equalities,

$$A_p \mathbf{c} - hB_p \mathbf{c}' = O(h^{p+1}), \quad B_p(\mathbf{c}' - J_f(\hat{\mathbf{y}})\mathbf{c}) = (\mathbf{c}' - J_f(\hat{\mathbf{y}})\mathbf{c}) + O(h),$$

one concludes that $\tau_p = h^{p+1}(\mathbf{c}' - J_f(\hat{\mathbf{y}})\mathbf{c}) + O(h^{p+2})$. Consequently, by means of similar arguments, one finally obtains

$$\begin{aligned} M_q \mathbf{v} &= h^p(A_q \mathbf{c} - hB_q J_f(\hat{\mathbf{y}})\mathbf{c}) = h^p(A_q \mathbf{c} - hB_q \mathbf{c}') + h^{p+1}B_q(\mathbf{c}' - J_f(\hat{\mathbf{y}})\mathbf{c}) \\ &= O(h^{p+q+1}) + h^{p+1}(\mathbf{c}' - J_f(\hat{\mathbf{y}})\mathbf{c}) + O(h^{p+2}) = \tau_p + O(h^{p+2}). \end{aligned}$$

□

From the previous result and equations (11), (13), (14), the following estimate then easily follows,

$$\hat{\mathbf{y}} - \mathbf{y} \approx M_p^{-1}(F_p(\mathbf{y}) - F_q(\mathbf{y})). \quad (15)$$

We observe that, when computing (15), the matrix M_p has already been factored for solving (11).

When the methods defining F_p and F_q are both GBDF, then $B_p = B_q$ and, therefore, $\tau_p \approx F_p(\mathbf{y}) - F_q(\mathbf{y}) \equiv (A_p - A_q)\mathbf{y}$. As consequence, the estimate of the local error does not explicitly depend on f . As observed in [13, page 134], this feature makes the estimate suitable for approximating stiff problems.

Finally, we observe that, having fixed a tolerance for the error, the estimate (15), together with the usual extrapolation procedure, can be easily used for determining the appropriate stepsize.

5 Numerical Examples

We here report a few numerical examples on severe stiff test problems taken from the literature. For comparisons, we also report the performance of LSODE [16] (one of the most popular ODE solvers) and of the Matlab stiff ODE solver ODE23S [18] on the same problems.

We first consider the Robertson's problem,

$$\begin{aligned} y_1' &= -.04y_1 + 10^4 y_2 y_3, & y_1(0) &= 1, \\ y_2' &= .04y_1 - 10^4 y_2 y_3 - 3 \cdot 10^7 y_2^2, & y_2(0) &= 0, \\ y_3' &= 3 \cdot 10^7 y_2^2, & y_3(0) &= 0. \end{aligned}$$

We use the fifth-order GBDF with tolerance 10^{-7} for the error. The estimate of the error is obtained from (15) by considering the GBDF of order seven. The interval $[0, 2 \cdot 10^{20}]$ is covered with 540 steps, and stepsizes which monotonically increase from 10^{-4} up to $5.2 \cdot 10^{19}$. LSODE, with parameters $mf = 21$ and $atol = rtol = 10^{-7}$, requires 637 steps to cover the same interval. On the other hand, ODE23S, with parameters $atol = rtol = 10^{-7}$ and analytic Jacobian, fails to cover the whole integration interval.

Let us now consider the following problem due to Curtis [17, page 409],

$$y' = A_\nu(t)y + f(t), \quad t \in [0, 10\pi], \quad y(0) = \begin{pmatrix} 1 \\ 0 \end{pmatrix},$$

where $\nu = 10^3$, and

$$f(t) = \begin{pmatrix} -\sin(t) \\ \cos(t) \end{pmatrix} - A_\nu(t) \begin{pmatrix} \cos(t) \\ \sin(t) \end{pmatrix},$$

$$A_\nu(t) = M_\nu(t) \begin{pmatrix} -1001 & 0 \\ 0 & -1 \end{pmatrix} M_\nu^T(t), \quad M_\nu(t) = \begin{pmatrix} \cos(\nu t) & \sin(\nu t) \\ -\sin(\nu t) & \cos(\nu t) \end{pmatrix}.$$

The solution of the problem is given by $y(t) = (\cos(t) \sin(t))^T$, independently of the value of the parameter ν , but the problem is very stiff, despite the fact that the solution is smooth. However, the latter feature suggests that stable high order methods should perform well. For this reason, we consider the GBDF of order twenty on this problem. The stepsize is changed by using the estimate of the global errors obtained by means of the GBDF of order twenty two. Having fixed a tolerance 10^{-5} for the error, the integration interval is covered with 56 mesh points, stepsizes ranging from 0.5 to 0.64, and a maximum absolute error $2.7 \cdot 10^{-6}$. To get an idea of the performance, consider that LSODE, with parameters $mf = 21$ and $atol = rtol = 10^{-5}$, needs 2356 steps to cover the integration interval, with a maximum error $1.7 \cdot 10^{-4}$. By using the default parameters $rtol = 10^{-3}$ and $atol = 10^{-6}$, ODE23S requires more than 12000 steps to cover the integration interval, with a maximum error $2.4 \cdot 10^{-3}$.

Finally, we consider the Van der Pol's equations,

$$\begin{aligned} y_1' &= y_2, & y_1(0) &= 2, \\ y_2' &= -y_1 + \mu y_2(1 - y_1^2), & y_2(0) &= 0, \end{aligned}$$

with the parameter $\mu = 10^3$. Such equations have an attractive limit cycle, which is readily reached from the chosen starting point. We integrate up to $T = 2 \cdot 10^3$, so that a whole limit cycle is covered. By using the fifth-order GBDF, coupled with the GBDF of order seven for the error estimate, and a tolerance 10^{-7} for the error, the integration interval is covered with 1930 steps. The selected stepsizes range from $7.5 \cdot 10^{-6}$, where the solution has the most rapid variations, to $3.5 \cdot 10^1$, where it is very smooth. By using the parameters $mf = 21$ and $atol = rtol = 10^{-7}$, LSODE requires 1328 steps to cover the integration interval, while 5081 steps are required by ODE23S with parameters $atol = rtol = 10^{-7}$.

Acknowledgements. The author is very indebted with professor D. Trigiante for the helpful discussions.

References

1. Amodio, P.: *A-stable k -step linear multistep formulae of order $2k$ for the solution of stiff ODEs.* (submitted)
2. Amodio, P., Brugnano, L.: *Parallel implementation of block Boundary Value Methods for ODEs.* (submitted)
3. Amodio, P., Mazzia, F.: *A boundary value approach to the numerical solution of ODEs by multistep methods.* J. of Difference Eq. and Appl. **1** (1995) 353–367
4. Axelsson, A. O. H., Verwer, J. G.: *Boundary value techniques for initial value problems in ordinary differential equations.* Math. Comp. **45** (1985) 153–171
5. Brugnano, L.: *Essentially symplectic Boundary Value Methods for linear Hamiltonian systems.* J. Comput. Math. (to appear)
6. Brugnano, L., Trigiante, D.: *Convergence and stability of Boundary Value Methods for ordinary differential equations.* J. Comput. Appl. Math. (to appear)
7. Brugnano, L., Trigiante, D.: *High order multistep methods for boundary value problems.* Appl. Num. Math. **18** (1995) 79–94
8. Brugnano, L., Trigiante, D.: *Block Boundary Value Methods for linear Hamiltonian systems.* J. of Appl. Math. (to appear)
9. Brugnano, L., Trigiante, D.: *A new mesh selection strategy for ODEs.* (submitted)
10. Brugnano, L., Trigiante, D.: *Boundary Value Methods: the third way between Linear Multistep and Runge-Kutta methods.* Computer and Math. with Appl. (to appear)
11. Brugnano, L., Trigiante, D.: *Solving Differential Problems by Multistep Initial and Boundary Value Methods.* (in preparation)
12. Cash, J. R.: *Stable Recursions.* Academic Press, London, 1979
13. Hairer, E., Wanner, G.: *Solving ordinary differential equations II.* Springer-Verlag, Berlin, 1991
14. Lopez, L., Trigiante, D.: *Boundary Value Methods and BV-stability in the solution of initial value problems.* Appl. Numer. Math. **11** (1993) 225–239
15. Pereyra, V. L.: *On improving an approximate solution of a functional equation by deferred corrections.* Numer. Math. **8** (1966) 376–391
16. Radhakrishnan, K., Hindmarsh, A. C.: *Description and use of LSODE, the Livermore Solver for Ordinary Differential Equations.* NASA Reference Publication 1327, LLNL Report UCRL-ID-113855, 1993
17. Shampine, L. F.: *Numerical Solution of Ordinary Differential Equations.* Chapman & Hall, New York, 1994
18. Shampine, L. F., Reichelt, M. W.: *The Matlab ODE Suite.* Report, 1995
19. Stetter, H. J.: *Economical global error estimation.* In “Stiff Differential Systems”, R. A. Willoughby ed., Plenum Press, New York, 1974
20. Zadunaisky, P.: *On the estimation of errors propagated in the numerical solution of ordinary differential equations.* Numer. Math. **27** (1976) 21–39