CWI

Centrum voor Wiskunde en Informatica
Centre for Mathematics and Computer Science

B.P. Sommeijer, W. Couzy, P.J. van der Houwen

A-stable parallel block methods

Department of Numerical Mathematics Report NM-R8918 August

The Centre for Mathematics and Computer Science is a research institute of the Stichting Mathematisch Centrum, which was founded on February 11, 1946, as a nonprofit institution aiming at the promotion of mathematics, computer science, and their applications. It is sponsored by the Dutch Government through the Netherlands Organization for the Advancement of Research (N.W.O.).

A-Stable Parallel Block Methods

B.P. Sommeijer, W. Couzy & P.J. van der Houwen

Centre for Mathematics and Computer Science
Post box 4079, 1009 AB Amsterdam, The Netherlands

In this paper we study the stability of a class of block methods which are suitable for use on parallel computers. A-stable methods of orders 3 and 4 and $A(\alpha)$-stable methods with $\alpha > 89.9°$ of order 5 are constructed. On multiprocessor computers these methods are of the same computational complexity as implicit linear multistep methods on one-processor computers.

1980 Mathematics Subject Classification: 65M10, 65M20
1982 CR Categories: 5.17
Key Words and Phrases: stability, block methods, parallelism.

1. Introduction

We will construct highly stable block methods for solving the initial-value problem for

(1.1) $$\frac{dy(t)}{dt} = f(y(t)).$$

Most block methods in the literature are methods where the block vector of solution values corresponds to equally spaced t-values (abscissas). In this paper, nonequally spaced abscissas are allowed. We shall construct A-stable methods of orders three and four, and $A(\alpha)$-stable methods of order five with $\alpha \approx \pi/2$. These methods are designed in such a way that they can take full advantage of a parallel machine, thus reducing the work to an amount comparable with that of conventional linear multistep methods on uniprocessor machines. For example, the methods are of equal computational complexity as the celebrated backward differentiation formulas (BDFs), but they possess a significantly larger stability region than the BDFs.

2. Parallel Block Methods

The block methods studied in this paper are a direct generalization of the implicit one-step method

(2.1) $$y_{n+1} = ay_n + hbf(y_n) + hdf(y_{n+1}), \quad n = 0, 1, \dots ,$$

where h is the stepsize and y_n an approximation to $y(t_n)$. By introducing block vectors

(2.2) $$\mathbf{Y}_{n+1} := (y_{n,1}, \dots , y_{n,k})^T, \quad \mathbf{c} := (c_1, \dots , c_k)^T, \quad c_k = 1,$$

where $y_{n,i}$ denotes a numerical approximation to the exact solution value $y(t_n+c_ih)$, and assuming that (1.1) is a scalar equation, we can define the block method

(2.3) $$\mathbf{Y}_{n+1} = A\mathbf{Y}_n + hBf(\mathbf{Y}_n) + hDf(\mathbf{Y}_{n+1}),$$

where A, B and D are k-by-k matrices. Here we use the convention that for any given vector $v=(v_j)$, f(v) denotes the vector with entries $f(v_j)$. This method can be considered as the block analogue of (2.1). A characteristic of these methods is that, unlike conventional block methods based on linear multistep methods, the block point vector **c** is allowed to have k-1 noninteger components. In order to start the method, one needs the initial vector $\mathbf{Y}_0$, which requires, in general, as many starting values as there are distinct values c_j (j=1,...,k). Notice that the last component of $\mathbf{Y}_{n+1}$ contains the step point value y_{n+1}. Furthermore, we remark that, in general, $y_{n,i} \neq y_{m,j}$, even if $n+c_i=m+c_j$.

The method (2.3) is straightforwardly extended to systems of ODEs and therefore also to nonautonomous equations. It is suitable for use on parallel computers if the matrix D is *diagonal*, since such a form decouples the various components as far as implicitness is concerned; the corresponding methods will be called *parallel block methods*. Using k processors, each processor has to evaluate a component of $f(\mathbf{Y}_n)$ and to solve a system of equations whose dimension is that of the system of ODEs (1.1). If Newton's method is used for solving the system of equations, then each processor needs the Jacobian matrix $I-hd_{jj}\partial f/\partial y$ and its LU-decomposition. Either the various processors have to compute themselves the data they need, or one may consider the use of additional processors for computing the Jacobian matrices and their LU-decompositions. Let us consider the second strategy. As soon as the

Report NM-R8918
Centre for Mathematics and Computer Science
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

additional processors have completed an update of the matrix $\partial f/\partial y$ and computed the LU-decompositions of the k matrices $I-hd_{jj}\partial f/\partial y$, then the first k processors can replace their data by the new data. However, usually the computational job of computing Jacobian matrices and LU-decompositions is so substantial that the speed of updating may not be great enough. In such cases, the use of matrices D with equal diagonal elements is recommendable, because then the Jacobian matrices $I-hd_{jj}\partial f/\partial y$ are all identical, so that only one instead of k decompositions are required. Therefore, methods where D is of the form dI, I being the identity matrix, have some advantage.

The order conditions for methods of the form (2.3) are extremely simple. By defining the vectors (cf. [2])

(2.4a)
$$C_0 := Ae - \mathbf{e}; \quad C_1 := A(\mathbf{c} - \mathbf{e}) + B\mathbf{e} + D\mathbf{e} - \mathbf{c};$$
$$C_j := A(\mathbf{c} - \mathbf{e})^j + j[B(\mathbf{c} - \mathbf{e})^{j-1} + D\mathbf{c}^{j-1}] - \mathbf{c}^j, \quad j = 2, 3, \dots,$$

where $\mathbf{e}$ denotes the vector with unit entries, the conditions for pth-order consistency take the form

(2.4b) $$C_j = \mathbf{0}, \quad j = 0, 1, \dots, p.$$

Here, powers of vectors are meant to be componentwise powers.

In order to compare the components of these vectors with the error constants corresponding to conventional linear multistep methods, we introduce the *normalized* error vectors

(2.5) $$\mathbf{E}_j := \frac{C_j}{j!(B + D)\mathbf{e}}.$$

When a linear k-step method is written in the form (2.3) with $\mathbf{c}=(-k+2,\dots,-2,-1,0,1)^T$, then the last component of $\mathbf{E}_j$ equals the normalized error constant of the linear k-step method.

3. Stability

The (linear) stability of block methods can be investigated by applying the method to the test equation $y'=\lambda y$. This will lead to a recursion of the form

(3.1) $$Y_{n+1} = M(z)\, Y_n, \quad M(z) := [I - zD]^{-1}[A + zB], \quad z := \lambda h.$$

M will be called the *amplification matrix* and its eigenvalues the *amplification factors*.

In our stability analysis we shall use the following result on the power bound of a matrix N: let $\|\cdot\|$ denote the spectral norm, then (cf., e.g., Varga [4, p. 65]).

(3.2) $$\|N^n\| = O(n^{q-1}[\rho(N)]^n) \text{ as } n \to \infty,$$

where $\rho(N)$ is the spectral radius of N and where all diagonal submatrices of the Jordan normal form of N which have spectral radius $\rho(N)$ are at most q-by-q. If $\rho(N) < 1$ or $\rho(N) = q = 1$, then we call the matrix N *power bounded.*

Following the familiar stability definitions used for RK and LM methods, we shall call the region where the amplification matrix M(z) is power bounded, the *stability region* of the block method. If the stability region contains the origin, then the method is called *zero-stable.* The region where $\|M^n\|$ tends to zero will be called the *strong stability region.* If the (strong) stability region of a block method contains the left half plane, then the block method is called *(strongly) A-stable,* if the amplification matrix of an A-stable method has vanishing eigenvalues at infinity, then the method is called *L-stable.* In order to compare the stability regions of the various methods, we introduce the notion of $A(\alpha,\beta,\gamma)$-stability which gives more detailed information on the stability region.

Definition 3.1. A method is said to be *$A(\alpha,\beta,\gamma)$-stable* if (i) its region of stability contains the infinite wedge $\{z: -\alpha < \pi - \arg(z) < \alpha\}$, $0 < \alpha \le \pi/2$, and all points in the nonpositive halfplane with $|z| > \beta$, and (ii) γ is the smallest nonnegative number such that the magnitude of the eigenvalues of the amplification matrix do not exceed $1 + \gamma$ when z runs through the region of instability lying in the nonpositive halfplane. □

The parameter γ may be considered as the degree of instability of the method. According to this definition, we have that A-stable methods are $A(\pi/2,0,0)$-stable.

If we set A=D=I and B=O in the amplification matrix M defined in (3.1), then the method is a 'system of backward Euler methods'. Such methods have excellent stability properties (e.g., the property of L-stability), but are only of first order. However, the matrices A, B and D can be chosen such that the order p is raised to 2k. This section investigates to what extent the favourable stability properties of the backward Euler method are preserved when the order is increased.

3.1. Two-Dimensional Block Methods

First we consider the case k=2 and choose the coefficient matrices of the form

(3.3) $$A = \begin{pmatrix} a_1 & 1-a_1 \\ a_2 & 1-a_2 \end{pmatrix}, \quad B = \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}, \quad D = \begin{pmatrix} d_1 & 0 \\ 0 & d_2 \end{pmatrix}, \quad \mathbf{c} = (c,1)^T.$$

Imposing the conditions for second-order consistency we can express the entries of the matrix B in terms of the five free parameters c, a_j and d_j:

(3.4a) $$b_{j1} = \frac{1}{2}(1 - c)\, a_j + \frac{c_j(2d_j - c_j)}{2(1 - c)}, \quad b_{j2} = c_j + (1 - c)a_j - b_{j1} - d_j, \quad c_1 = c, \; c_2 = 1.$$

The components C_{ij} of the vectors C_i are given by

$$C_{ij} = (1 - \frac{i}{2})(c - 1)^i\, a_j + i c_j^{i-1} d_j + \frac{i}{2} c_j(c_j - 2d_j)\,(c - 1)^{i-2} - c_j^i, \quad i \geq 3, \; j = 1, 2.$$

An elementary calculation shows that C_{3j} vanishes if

(3.4b) $$a_j = \frac{c_j}{(c - 1)^3}\,[3(c-1)(c_j-2d_j) + 2c_j(3d_j-c_j)],$$

and that C_{4j} also vanishes if, in addition,

(3.4c) $$d_1 = \frac{c}{2(c + 1)}, \quad d_2 = \frac{c - 2}{2(c - 3)}.$$

The characteristic equation of the amplification matrix in (3.1) can be written in the form

(3.5) $$\det[A + zB - \zeta(I - zD)] = \det\begin{pmatrix} a_1 + b_{11}z - \zeta(1 - d_1 z) & 1 - a_1 + b_{12}z \\ a_2 + b_{21}z & 1 - a_2 + b_{22}z - \zeta(1 - d_2 z) \end{pmatrix} = 0.$$

This yields the quadratic equation

(3.5') $$P(\zeta,z) = p(z)\zeta^2 + q(z)\zeta + r(z) = 0,$$

where p, q and r are quadratic functions of z. We shall determine the z-region where this polynomial has its roots ζ within the unit circle, that is, the region of strong stability. In addition, we should impose the condition of zero-stability, i.e., the condition that the two eigenvalues $\alpha=1$ and $\alpha=a_1-a_2$ of A are on the unit disk those on the unit circle being simple, i.e.,

(3.6) $$-1 \leq a_1 - a_2 < 1.$$

Furthermore, if we want the property of A-stability, then it is convenient to impose the 'stability at infinity' condition, that is, the polynomial $P(\zeta,\infty)$ is required to be a Schur polynomial (this condition limits the range of the free parameters). Since

$$P(\zeta,\infty) = z^2[d_1 d_2 \zeta^2 + (b_{11}d_2 + b_{22}d_1)\zeta + \det(B)],$$

we obtain by virtue of the Hurwitz criterion the conditions

(3.7) $$|\, b_{11}d_2 + b_{22}d_1 \,| < \mathrm{sign}(d_1 d_2)[d_1 d_2 + \det(B)], \quad \det(B) < d_1 d_2.$$

We shall discuss the second-, third- and fourth-order cases separately.

3.1.1. Second-order methods. If we are satisfied with second-order accuracy, then we may choose the free parameters a_j and d_j in (3.4a) such that the matrix B vanishes while preserving the property of A-stability. For example, if c=0 then the method reduces to

(3.8) $$A = \begin{pmatrix} 0 & 1 \\ -1/3 & 4/3 \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}, \quad D = \begin{pmatrix} 0 & 0 \\ 0 & 2/3 \end{pmatrix}, \quad \mathbf{c} = (0,1)^T,$$

which is equivalent with the familiar two-step backward differentiation formula.

3.1.2. Third-order methods. Third-order accuracy is achieved by choosing $C_{31}=C_{32}=0$, leaving us with three free parameters for monitoring the stability of the method. We find

$$(3.9)\qquad a_1 = \frac{c(c^2 - 3c + 6d_1)}{(c - 1)^3}, \quad b_{11} = \frac{c^2 - 2cd_1 - c^2 d_1}{(c - 1)^2}, \quad b_{12} = \frac{c - 2cd_1 - d_1}{(c - 1)^2},$$

$$a_2 = \frac{3c + 12d_2 - 6cd_2 - 5}{(c - 1)^3}, \quad b_{21} = \frac{2 - 5d_2 - c + 2cd_2}{(c - 1)^2}, \quad b_{22} = \frac{(c - 2)^2 - d_2(c^2 - 6c + 8)}{(c - 1)^2},$$

leaving c, d_1 and d_2 as the free parameters. Taking into account the conditions of zero-stability and 'stability at infinity' (conditions (3.6) and (3.7)), we performed a numerical search in the (c,d_1,d_2)-space. It turned out that the regions of A-stable (c,d_1,d_2)-values are so small that A-stable points and strongly unstable points are close together, that is, a small perturbation of these values causes the method to violate the A-stability conditions. For example, the values

$$(3.10)\qquad c = 0.917387, \quad d_1 = 0.319523, \quad d_2 = 0.347067,$$

generate such a 'marginally' A-stable method. There is, however, an alternative approach. It is easily verified that putting $a_2=C_{32}=0$ yields methods providing third-order approximations at the step points t_n and second-order approximations at the points t_n+ch. It turns out that in the space of free parameters the regions of A-stable methods are larger so that it is easier to find A-stable methods by a numerical search. For example, we found the A-stable, third-order method

$$(3.11)\qquad A = \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} \frac{147}{220} & \frac{161}{220} \\ -\frac{50}{33} & \frac{23}{66} \end{pmatrix}, \quad D = \begin{pmatrix} \frac{7}{10} & 0 \\ 0 & \frac{13}{6} \end{pmatrix}, \quad \mathbf{c} = \frac{1}{10}(21, 10)^T$$

with the normalized error vectors $E_3 \approx (.19, 0)^T$ and $E_4 \approx (.20, -.017)^T$. The amplification factors at the origin equal 0 and 1, and the maximal amplification factor at infinity is ≈ 0.94.

3.1.3. Fourth-order methods. Fourth-order accuracy for both components is obtained by choosing $C_{31}=C_{32}=C_{41}=C_{42}=0$. Alternatively, replacing $C_{41}=0$ by $a_2=0$, reduces the order of the first component to 3, without affecting the order of the second component. In both approaches we are left with one free parameter for monitoring the stability of the method. Unfortunately, the stability regions of these fourth-order methods are rather limited and do not even allow for A(α)-stability. Thus, in the class (3.3) the fourth-order methods seem to be of no interest.

3.2. Three-Dimensional Block Methods

For k=3 we expect to find A-stable methods of order four and we may hope for A(α)-stable methods of order five. These two cases will be investigated in the following subsections. We recall that the block point vector **c** is of the form $(c_1,c_2,c_3=1)^T$.

3.2.1. Fourth-order methods. Let us choose the matrix A such that

$$(3.12a)\qquad a_{i3} = 1 - a_{i1} - a_{i2}, \quad i = 1, 2, 3,$$

then C_0 vanishes. The vectors C_j vanish for j=1, 2, 3, 4 if the entries b_{ij} and d_j in the matrices B and D satisfy the linear systems

$$(3.12b)\qquad \begin{pmatrix} 1 & 1 & 1 & 1 \\ c_1-1 & c_2-1 & 0 & c_i \\ (c_1-1)^2 & (c_2-1)^2 & 0 & c_i^2 \\ (c_1-1)^3 & (c_2-1)^3 & 0 & c_i^3 \end{pmatrix} \begin{pmatrix} b_{i1} \\ b_{i2} \\ b_{i3} \\ d_i \end{pmatrix} = \begin{pmatrix} c_i - a_{i1}(c_1-1) - a_{i2}(c_2 - 1) \\ \frac{1}{2}[c_i^2 - a_{i1}(c_1-1)^2 - a_{i2}(c_2 - 1)^2] \\ \frac{1}{3}[c_i^3 - a_{i1}(c_1-1)^3 - a_{i2}(c_2 - 1)^3] \\ \frac{1}{4}[c_i^4 - a_{i1}(c_1-1)^4 - a_{i2}(c_2 - 1)^4] \end{pmatrix}, \quad i = 1, 2, 3.$$

These conditions show that there is a family of fourth-order block methods with eight free parameters: a_{i1}, a_{i2} (i=1,2,3), c_1 and c_2.

In order to ensure zero-stability, we require that A has its two parasitic eigenvalues within the unit circle. Writing the characteristic equation of A in the form

$$(\zeta - 1)(\zeta^2 + q_0\zeta + r_0) = 0,$$

we find that

(3.13a) $q_0 = a_{31} + a_{32} - a_{11} - a_{22}, \quad r_0 = a_{11}a_{12} + a_{31}a_{12} + a_{32}a_{21} - a_{11}a_{32} - a_{21}a_{12} - a_{22}a_{31},$

so that we have zero-stability if

(3.13b) $|q_0| < r_0 + 1 \quad \text{and} \quad r_0 < 1.$

Taking the constraint (3.13b) into account, we performed a numerical search over the free parameters and looked for A-stable methods. This has led to the following method

$$(3.14)\quad A = \begin{pmatrix} -1 & \frac{1}{2} & \frac{3}{2} \\ \frac{1}{2} & 1 & -\frac{1}{2} \\ -1 & \frac{1}{2} & \frac{3}{2} \end{pmatrix}, \quad B = \begin{pmatrix} \frac{5\cdot13\cdot43}{2^{11}} & \frac{15161}{2^5\cdot3^2\cdot11} & \frac{29\cdot43\cdot83}{2^{11}\cdot3^2\cdot5} \\ \frac{-73}{2\cdot3^2\cdot7} & \frac{-467}{2\cdot3^3\cdot7} & \frac{-7\cdot37}{2\cdot3^3\cdot13} \\ \frac{5\cdot16069}{2^{11}\cdot3^2\cdot7} & \frac{54419}{2^5\cdot3^3\cdot5\cdot7} & \frac{41927}{2^{11}\cdot3^2} \end{pmatrix}, \quad D = \begin{pmatrix} \frac{13\cdot1303}{2^9\cdot5\cdot11} & 0 & 0 \\ 0 & \frac{277}{2\cdot3^2\cdot13} & 0 \\ 0 & 0 & \frac{16001}{2^9\cdot3^2\cdot5} \end{pmatrix},$$

with $\mathbf{c} = \frac{1}{4}(20, 13, 4)^T$.

This method has normalized error vector $\mathbf{E}_5 \approx (.13, .27, .075)^T$. Its amplification factors at the origin are 0, 1/2 and 1, and at infinity the maximal amplification factor is $\approx$ 0.92.

The above direct search method is rather expensive, and therefore we also applied an alternative approach where we tried to minimize the quantity

$$(3.15)\quad \sum_{i=0}^{m} \sum_{j=1}^{k} |\mu_{ij}|^{q_{ij}}$$

over the free parameters b_{i2} and d_i (i=1,2,3), c_1 and c_2. Here, k=3, the q_{ij} are control parameters and μ_{ij}, j=1, ... , k denote the eigenvalues of the amplification matrix $M(z_i)$ defined in (3.1) with z_i running through a set of m points lying on the imaginary axis. In this way we found the method

$$(3.16)\quad A = \frac{1}{1600}\begin{pmatrix} 2820 & -183 & -1037 \\ -7100 & -3423 & 12123 \\ -1020 & -1607 & 4227 \end{pmatrix}, \quad B = \frac{1}{400}\begin{pmatrix} -398 & -92 & -177 \\ 6282 & -92 & 2143 \\ 1098 & 272 & 507 \end{pmatrix}, \quad D = \frac{1}{5}\begin{pmatrix} 8 & 0 & 0 \\ 0 & 8 & 0 \\ 0 & 0 & 8 \end{pmatrix}, \quad \mathbf{c} = (3,5,1)^T.$$

Method (3.16) has normalized error vector $\mathbf{E}_5 \approx (3.67, .19, .064)^T$. At the origin the amplification factors are 0.81, 0.81 and 1, and at infinity the maximal amplification factor is $\approx$ 0.37.

3.2.2. Fifth-order methods. Along the same lines as we constructed the fourth-order method (3.16), we proceeded with the fifth-order case. Now only five free parameters are available, say d_i (i=1,2,3), c_1 and c_2. Imposing the constraint (3.13b), we searched for A-stable methods in the parameter-space. Unfortunately, we did not succeed in finding A-stable methods. However, we found a few $A(\alpha,\beta,\gamma)$-stable methods (see Definition 3.1 for a specification of $A(\alpha,\beta,\gamma)$-stability) which may be considered as A-stable for most practical applications.

We mention the A(α,β,γ)-stable method with $\alpha \approx 89.9988°$, $\beta \approx 0.16$ and $\gamma \approx 2.6_{10}\text{-}6$ generated by

$$\mathbf{c} = \begin{pmatrix} -2.747 \\ -2.122 \\ 1 \end{pmatrix}, \quad A = \begin{pmatrix} -.37354856915573 & 1.3772028209449 & -.0036542517891531 \\ 0.45636214490330 & 0.58957191150098 & -.045934056404276 \\ -71.558907928027 & 69.945110840701 & 2.6137970873262 \end{pmatrix},$$

(3.17)

$$B = \begin{pmatrix} -.089579683013023 & -.020791477924637 & 0.0023118793010643 \\ 0.037434812789650 & 0.78549538208108 & 0.024702269787981 \\ -18.279469309687 & -29.674965823418 & -1.6401568285440 \end{pmatrix}, \quad D = \begin{pmatrix} 0.261 & 0 & 0 \\ 0 & 0.581 & 0 \\ 0 & 0 & 0.832 \end{pmatrix}$$

and with normalized error vector $E_6 \approx (.007, .0038, -.015)^T$. At the origin the amplification factors are 0.92, 0.92, and 1, and at infinity the maximal amplification factor is ≈ 0.993. The stability region is given in Figure 3.1.

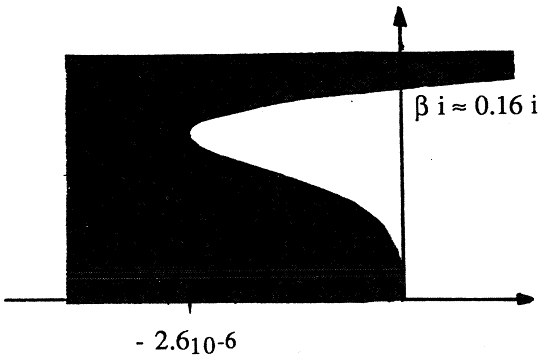

Figure 3.1. Stability region of (3.17).

Finally, we present the A(α,β,γ)-stable method with $\alpha \approx 89.98°$, $\beta \approx 0.30$ and $\gamma \approx 6.9_{10}\text{-}5$ generated by

$$\mathbf{c} = \begin{pmatrix} 1.6153 \\ 4.7871 \\ 1 \end{pmatrix}, \quad A = \begin{pmatrix} 0.58694824150708 & -.042737729478577 & 0.45578948797150 \\ 73.394943213338 & 2.5499812910344 & -74.944924504372 \\ 1.3881897627759 & -.0035265226034516 & -0.38466324017241 \end{pmatrix},$$

(3.18)

$$B = \begin{pmatrix} 0.784348212088875 & 0.023439431423946 & 0.033345158796322 \\ -30.332265183768 & -1.5938561820999 & -18.934741340575 \\ -.012761141648945 & 0.0022604702667178 & -.092097195902230 \end{pmatrix}, \quad D = \begin{pmatrix} 0.57487 & 0 & 0 \\ 0 & 0.83102 & 0 \\ 0 & 0 & 0.2618 \end{pmatrix}$$

and with normalized error vector $E_6 \approx (.004, -.016, .007)^T$. At the origin the amplification factors are 0.88, 0.88 and 1, and at infinity the maximal amplification factor is ≈ 0.89. The stability region is given in Figure 3.2.

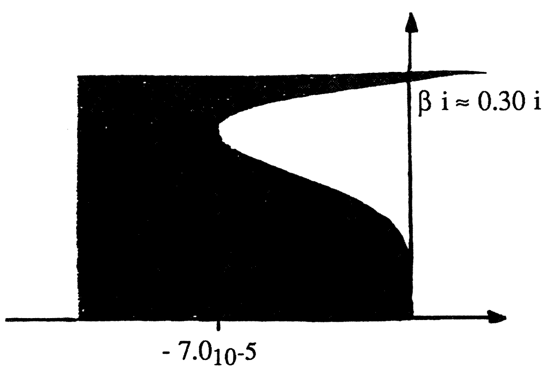

Figure 3.2. Stability region of (3.18).

3.3. Survey of Method Characteristics

We conclude with a survey of the parameters α, β and γ characterizing the stability regions of the block methods derived in this paper (see Definition 3.1) and compare them with those of the BDFs (details about the BDF methods can be found in [1]). In Table 3.1 these values are listed.

In addition, we give the normalized error vectors defined in (2.5) of all methods. For a uniform presentation, we first formulated the BDFs as block methods. We recall that a k-step BDF method can be cast in the form (2.3) with block point vector $\mathbf{c} = (2-k, ..., -1, 0, 1)^T$.

Finally, we remark that a k-step, kth-order BDF requires k starting values, independent of its formulation, whereas the block methods of this paper need only 2 (for p=3) or 3 (for p=4,5) starting values, which reduces the implementational complexity.

Table 3.1. Normalized error vectors and values of α, β and γ.

Method	Order p	E_{p+1}^T	α	β	γ
BDF_3	3	(0, 0, 1/4)	88.4°	1.94	0.046
(3.11)	3	(.20, -.017)	90°	0	0
BDF_4	4	(0, 0, 0, 1/5)	73.2°	4.72	0.191
(3.14)	4	(.13, .27, .075)	90°	0	0
(3.16)	4	(3.67, .19, .064)	90°	0	0
BDF_5	5	(0, 0, 0, 0, 1/6)	51.8°	9.94	0.379
(3.17)	5	(.007, .0038, -.015)	>89.9°	0.16	0.0000026
(3.18)	5	(.004, -.016, .007)	>89.9°	0.30	0.000069

4. Numerical Experiments

We illustrate the performance of the methods by integrating a few test problems.

4.1. Accuracy Test

To verify the order of the various methods we integrated the test problem proposed by Kaps [3]:

$$\frac{dy_1}{dt} = -(2+\varepsilon^{-1})\, y_1 + \varepsilon^{-1} y_2, \qquad \frac{dy_2}{dt} = y_1 - y_2(1+y_2), \qquad y_1(0) = y_2(0) = 1, \qquad 0 \le t \le T, \tag{4.1}$$

with exact solution $y_1=\exp(-2t)$ and $y_2=\exp(-t)$ for all values of the parameter ε. In Table 4.1, we have listed the values Δ, where Δ denotes the number of correct decimal digits at the endpoint (i.e., we write the maximum norm of the error at t=T in the form $10^{-\Delta}$). In all experiments the theoretical order of the method is shown for sufficiently small values of h (if p is the order of the method, then, on halving the step size, the value of Δ should increase by $\approx 0.3p$).

Table 4.1. Values of Δ for problem (4.1) at T=1 with $\varepsilon=10^{-8}$.

Method	Order p	h=1/4	h=1/8	h=1/16	h=1/32	h=1/64	h=1/128	h=1/256
BDF_3	3	2.8	3.7	4.6	5.5	6.5	7.4	8.3
(3.11)	3	2.8	3.6	4.4	5.2	6.1	7.0	7.9
BDF_4	4	3.4	4.7	5.9	7.1	8.4	9.6	10.7
(3.14)	4	3.8	5.2	9.5	7.9	8.9	10.0	11.2
(3.16)	4	3.1	3.9	4.8	5.9	7.1	8.2	9.4
BDF_5	5	4.0	5.6	7.2	8.7	10.2	12.0	
(3.17)	5	2.6	4.0	5.5	7.3	9.2	10.3	
(3.18)	5	4.7	5.4	6.4	7.7	9.2	10.1	

4.2. Stability Test

We tested the stability of the methods by integrating a problem in which the Jacobian matrix has purely imaginary eigenvalues:

$$(4.2) \qquad \frac{dy_1}{dt} = -\alpha y_2 + (1+\alpha)\cos(t), \quad \frac{dy_2}{dt} = \alpha y_1 - (1+\alpha)\sin(t), \quad y_1(0)=0, \; y_2(0)=1, \quad 0 \le t \le T,$$

with exact solution $y_1=\sin(t)$ and $y_2=\cos(t)$ for all values of the parameter α.

In Table 4.2, the results are listed for T=100. Values of Δ corresponding to stepsizes that are theoretically unstable are underlined and overflow is indicated by ∗. The unstable results of the BDFs is in agreement with their regions of instability indicated in Table 3.1 (the phenomenon that BDF_5 becomes stable again for sufficiently small h is due to the fact that its imaginary interval of instability is given by i[0.71, 9.94]).

Table 4.2. Values of Δ for problem (4.2) at T=100 with α=10.

Method	Order p	h=4/5	h=2/5	h=1/5	h=1/10	h=1/20	h=1/40	h=1/80
BDF_3	3	2.0	2.9	3.9	*	*	4.9	7.5
(3.11)	3	2.1	2.8	3.4	4.0	4.6	5.3	6.3
BDF_4	4	2.2	*	*	*	2.9	8.2	9.9
(3.14)	4	2.8	4.0	4.9	5.8	6.8	8.0	9.6
(3.16)	4	1.6	2.7	3.8	4.9	5.8	6.8	8.2
BDF_5	5	-0.1	*	*	*	8.5	10.3	12.7
(3.17)	5	1.2	2.0	3.4	4.7	6.2	7.6	9.0
(3.18)	5	2.9	3.9	5.1	6.4	7.6	8.6	10.0

Next, we show that the 'almost' A-stable fifth-order methods (3.17) and (3.18) behave as A-stable methods in practice. We performed experiments for α=1 and α=4 with h=1/8: for α=1 both integration processes are *theoretically* unstable, and for α=4 the processes are stable. In Table 4.3 the results are listed for increasing length of the integration interval: these results clearly show that both methods perform perfectly stably for α=1.

Table 4.3. Values of Δ for problem (4.2) for h=1/8.

Method	α=1: theoretically unstable			α=4: theoretically stable		
	T=10	T=100	T=1000	T=10	T=100	T=1000
(3.17)	3.6	3.8	3.6	4.0	3.9	3.9
(3.18)	4.5	4.3	4.8	5.4	5.4	5.4

4.3. Performance Test on the ALLIANT FX/4

Finally, we tested the methods (3.11) and (3.18) on the ALLIANT FX/4 by integrating the problem (4.1) of Kaps. In Table 4.4, we have listed timings and the rate of efficiency defined by

$$\frac{1}{k} \, \frac{\text{execution time on one processor}}{\text{execution time on k processors}}.$$

These results show that the gain factor is close to its optimal value.

Table 4.4. Timings (in seconds) for problem (4.1) at T=1 with $\varepsilon=10^{-8}$ and h=1/256.

method	Number of processors 1	2	3	4	Efficiency rate
(3.11)	0.43	0.23	0.23		0.93
(3.18)	0.66	0.45	0.25	0.25	0.88

References

[1] **Gear, C.W. (1971):** *Numerical initial value problems in ordinary differential equations*, Prentice Hall, Englewood Cliffs, N.J..

[2] **Houwen, P.J. van der & Sommeijer, B.P. (1989):** *Block Runge-Kutta methods on parallel computers*, Report NM-R8906, Centre for Mathematics and Computer Science, Amsterdam (submitted for publication).

[3] **Kaps, P. (1981):** *Rosenbrock-type methods*, in: *Numerical methods for stiff initial value problems* (eds.: G. Dahlquist & R. Jeltsch), Bericht Nr.9, Inst. für Geometrie und Praktische Mathematik der RWTH Aachen.

[4] **Varga, R.S. (1962):** *Matrix iterative analysis*, Prentice Hall, Englewood Cliffs, N.J..