Block Implicit One-Step Methods *

By Daniel S. Watanabe

Abstract. A new class of block implicit one-step methods for ordinary differential equations is presented. The methods are based on quadrature and generate function values at nonmesh points through Hermite interpolation. A general convergence theorem for block implicit methods is given, and the stability of the new class of methods is analyzed. The class contains A-stable, stiffly stable, strongly A-stable, and strongly stiffly stable methods. Numerical results demonstrating the efficiency and effectiveness of a particular block method are presented.

1. **Introduction.** Many physical systems are described by ordinary differential equations whose solutions contain time constants differing greatly in magnitude. Such equations are called stiff. When a classical numerical integration procedure is applied to a stiff system of equations, the stepsize is generally determined by the component of the solution with the largest decay rate, while the region of integration is determined by the component with the smallest rate. After the initial transient, the rapidly decaying components are insignificant, but the stepsize must remain small to prevent numerical instability. As a result, the time required to integrate a highly stiff system can become excessive.

A-stable methods are often used to overcome this problem because the stepsize of an A-stable method is governed only by the allowable discretization error. The stepsize for A-stable linear multistep methods must remain small, however, because the order of such methods cannot exceed two. Implicit one-step methods are free from this restriction on order, and several classes of A-stable implicit one-step methods of arbitrary order exist. Unfortunately, these implicit methods are relatively inefficient. Their efficiency can be improved, however, by obtaining a block of new values simultaneously. These block implicit one-step methods have been studied by Rosser [10], Shampine and Watts [11], [13], Andria, Byrne, and Hill [1], and Williams and de Hoog [14]. We present in this paper a new class of block implicit methods which appear to be competitive with linear multistep methods for stiff problems.

We shall restrict our discussion to a single equation for simplicity. The generalization to systems of equations will be obvious. We first describe general block implicit methods and our new class of methods. We then present a convergence theorem for general block implicit methods and discuss the stability of our methods. Finally, we present two numerical examples comparing one of our methods with the well-known

code DIFSUB of Gear [7], one of the best codes currently available. In both cases, the block method requires significantly fewer function evaluations than DIFSUB.

2. Block Implicit Methods. We wish to approximate the solution of

$$\text{(1)} \qquad\qquad y' = f(x, y(x)), \qquad y(a) = \alpha,$$

on the interval $[a, b]$. Rather than make specific differentiability assumptions, we shall assume y has continuous derivatives on $[a, b]$ of any order required.

Let $x_n = a + nh$ for $n = 0, 1, \ldots$ and $h > 0$. We wish to generate a sequence $\{y_n\}$ which approximates the sequence of exact values $\{y(x_n)\}$. Let $y_0 = \alpha$. An s-block method generates a block of s additional terms simultaneously and ultimately produces values for all $n \in I_h$, where $I_h = \{n: 0 \leqslant n \leqslant ms\}$ and $m = [(b - a)/sh]$. Each block of values $\{y_{n+1}, \ldots, y_{n+s}\}$, where n is a multiple of s, satisfies equations of the form

$$\text{(2)} \qquad y_{n+i} = y_n + h\phi_i(x_n, y_n, \ldots, y_{n+s}, h), \qquad i = 1, \ldots, s.$$

The increment functions ϕ_i are determined by f and are functions of $x_n, y_n, \ldots, y_{n+s}$, and h only. We shall assume they are defined in the region R of (x, z, h) space defined by $x \in [a, b - sh]$, $z_k \in (-\infty, \infty)$, $k = 0, \ldots, s$, and $h \in [0, h_0]$, where $h_0 < (b - a)/s$. The local discretization errors are defined by

$$\text{(3)} \qquad \begin{aligned} d_i(x_n, h) = h\phi_i(x_n, y(x_n), \ldots, y(x_{n+s}), h) - (y(x_{n+i}) - y(x_n)), \\ i = 1, \ldots, s, \end{aligned}$$

and the order of the ith formula (2) is defined to be the largest integer r such that

$$\text{(4)} \qquad\qquad d_i(x, h) = O(h^{r+1})$$

in the region S of (x, h) space defined by $x \in [a, b - sh]$ and $h \in [0, h_0]$.

We propose s-block methods where each increment function is an interpolatory quadrature formula employing function values and possibly derivatives at nodes in the interval $[x_n, x_{n+s}]$, and values of y at nonmesh points are obtained from the Hermite interpolation polynomial $z(x)$ interpolating the first $p_i - 1$ derivatives of y at x_{n+i} for $i = 0, \ldots, s$. Each such method can be written in the form

$$\text{(5)} \quad y_{n+i} = y_n + h \sum_{j,k} w_{ijk} h^k f^{(k)}(x_n + \theta_{ij}h, z(x_n + \theta_{ij}h)), \qquad i = 1, \ldots, s,$$

and

$$\text{(6)} \qquad\qquad z(x_n + \theta h) = \sum_{l=0}^{s} \sum_{m=0}^{p_l - 1} h^m \psi_{lm}(\theta) y_{n+l}^{(m)}.$$

Here $f^{(k)}(x, z(x))$ is the kth total derivative of $f(x, z(x))$ with respect to x,

$$f^{(k)}(x, z(x)) = (\partial/\partial x + z^{(1)}(x)\partial/\partial z)^k f(x, z(x)),$$

$\psi_{lm}(\theta)$ are the Hermite basis functions for the interval $[0, s]$, and $y_{n+l}^{(m)}$ is the mth derivative of the local solution through (x_{n+l}, y_{n+l}).

If we replace the derivatives $y_{n+l}^{(m)}$ in the Hermite interpolation polynomial (6) with the exact derivatives $y^{(m)}(x_{n+l})$, then $z^{(k)}(x)$ differs from the exact derivative

$y^{(k)}(x)$ by a quantity of $O(h^{p-k})$, where $p = \Sigma_i p_i$. Since $f^{(k)}(x, z(x))$ contains a term involving $z^{(k)}(x)$, it differs from the exact derivative $f^{(k)}(x, y(x))$ by a quantity of $O(h^{p-k})$, and thus the error in the difference equation (5) due to interpolation is of $O(h^{p+1})$. If an increment function has a degree of precision at least $q-1$, then it is easy to see that the corresponding formula has order $r = \min[p, q]$ at least. However, it is possible to achieve a higher order by careful choice of the abscissas and weights (see Method 6).

The appearance of higher derivatives in Eqs. (5) and (6) may cause concern to some. However, quadrature formulas employing derivatives need not be used. Furthermore, work by Barton, Willers, and Zahar [2] demonstrates that the automatic generation of higher derivatives is not only possible but practical for a relatively wide class of problems. Finally, our class of methods contains efficient and effective methods which do not employ higher derivatives, and it is these methods that are of primary interest.

Each particular method can be specified by the sets of parameters $p = \{p_i\}$, $\theta_i = \{\theta_{ij}\}$, and $w_i = \{w_{ij0}; w_{ij1}; \dots\}$. Some methods are:

Method 1. A 1-block method with formula of order 4 given by $p = \{2, 2\}$, $\theta_1 = \{1/2\}$, and $w_1 = \{1; 0; 1/24\}$.

Method 2. A 2-block method with formulas of order 4 given by $p = \{1, 1, 2\}$, $\theta_1 = \{0, 1/2, 1\}$, $\theta_2 = \{0, 1, 2\}$, $w_1 = \{1/6, 2/3, 1/6\}$, and $w_2 = \{1/3, 4/3, 1/3\}$.

Method 3. A 2-block method with formulas of order 5 given by $p = \{1, 2, 2\}$, $\theta_1 = \theta_2 = \{0, 1/2, 1, 3/2, 2\}$, $w_1 = \{29/180, 31/45, 2/15, 1/45, -1/180\}$, and $w_2 = \{7/45, 32/45, 4/15, 32/45, 7/45\}$.

Method 4. A 2-block method with formulas of order 6 given by $p = \{2, 2, 2\}$, $\theta_1 = \theta_2 = \{0, 1 - 1/\sqrt{3}, 1, 1 + 1/\sqrt{3}, 2\}$, $w_1 = \{31/240, (24 + 15\sqrt{3})/80, 4/15, (24 - 15\sqrt{3})/80, 1/240\}$, and $w_2 = \{2/15, 3/5, 8/15, 3/5, 2/15\}$.

Method 5. A 2-block method with formulas of order 6 given by $p = \{1, 2, 3\}$, and the same abscissas and weights as in Method 4.

Method 6. A 2-block method with formulas of order 5 and 8 given by $p = \{2, 2, 2\}$, $\theta_1 = \theta_2 = \{0, 1 - \sqrt{3/7}, 1, 1 + \sqrt{3/7}, 2\}$, $w_1 = \{13/160, (392 + 735\sqrt{3/7})/1440, 16/45, (392 - 735\sqrt{3/7})/1440, 3/160\}$, and $w_2 = \{1/10, 49/90, 32/45, 49/90, 1/10\}$. Note that θ_2 and w_2 are the 5-point Lobatto abscissas and weights.

Method 7. A 3-block method with formulas of order 6 given by $p = \{1, 1, 2, 2\}$, $\theta_1 = \theta_2 = \theta_3 = \{0, 1/2, 1, 2, 5/2, 3\}$, $w_1 = \{287/1800, 52/75, 49/360, 1/40, -4/225, 7/1800\}$, $w_2 = \{43/225, 112/225, 4/5, 31/45, -16/75, 8/225\}$, and $w_3 = \{39/200, 12/25, 33/40, 33/40, 12/25, 39/200\}$.

3. **Convergence.** The following theorem gives sufficient conditions for the convergence of an s-block method and indicates the order of the accumulated discretization error. Our methods satisfy these conditions and hence are convergent.

THEOREM. *Let there exist positive constants L, M, p, and q such that*

$$(7) \qquad |\phi_i(x, z, h) - \phi_i(x, z^*, h)| \leqslant iL \sum_{j=0}^{s} |z_j - z_j^*|, \qquad i = 1, \dots, s,$$

for (x, z, h) and $(x, z^*, h) \in R$, and

(8) $|d_i(x, h)| \leqslant iMh^{p+1}$, $i = 1, \ldots, s - 1$, $|d_s(x, h)| \leqslant sMh^{q+1}$,

for $(x, h) \in S$. Then for any $h < \min[h_0, 2/s(s + 1)L]$, the difference equations (2) have a unique solution $\{y_n\}$, defined on I_h, there is a constant N such that

$$|y_n - y(x_n)| \leqslant Nh^r, \qquad n \in I_h,$$

where $r = \min[p + 1, q]$, and the method is said to be of order r.

The form of the Lipschitz condition (7) and the discretization error bound (8) is motivated by the fact that the increment function ϕ_i is often a sum of i quadrature formulas over intervals of length h. Note the increase in the order which results from using a formula at the end of the block of higher order than those used in the interior. The proof is a straightforward generalization of the classical one, the only novel feature being the grouping of the errors into blocks, and hence is omitted.

4. Stability. We shall examine the stability of our methods by applying them to the differential equation $y' = \lambda y$, where λ is a complex constant with $\text{Re}(\lambda) < 0$. The method (5) can be interpreted as an implicit one-step method with stepsize sh. Substituting $y' = \lambda y$ in Eqs. (5) and (6), we obtain

(9) $y_{n+s} = R(h\lambda) y_n,$

where $R(\mu)$ is a rational approximation to $e^{s\mu}$. If the degrees of precision of the increment functions are greater than or equal to the degree of the Hermite polynomial (6), then R depends only on the orders of collocation p_i of the Hermite polynomial. Hence the stability of entire classes of methods can be analyzed simultaneously. Let $[p_0, \ldots, p_s]$ denote the class of methods of the form (5) whose increment functions have degrees of precision at least $\Sigma_i p_i - 1$. We shall characterize the stability of such classes in terms of the following concepts.

Definition. The class $[p_0, \ldots, p_s]$ is *A*-stable if $|R(\mu)| < 1$ for $\text{Re}(\mu) < 0$.

Definition. The class $[p_0, \ldots, p_s]$ is stiffly stable if $|R(\mu)| < 1$ for $\text{Re}(\mu) \leqslant D < 0$ and $R(\mu)$ is accurate in the neighborhood of the origin.

Definition. The class $[p_0, \ldots, p_s]$ is strongly *A*- or strongly stiffly stable if it is *A*- or stiffly stable and $|R(\mu)| \to 0$ as $\text{Re}(\mu) \to -\infty$.

We first consider the class $[p, q]$ of 1-block methods employing higher derivatives. A simple calculation shows that

$$R(\mu) = P_p(\mu)/Q_q(\mu),$$

where $P_p(\mu)$ and $Q_q(\mu)$ are polynomials of degree p and q in μ. Since $R(\mu)$ must be an approximation to e^μ of order $p + q$, it follows that $R(\mu)$ must be the (p, q) entry $E_{pq}(\mu)$ in the Padé table for e^μ. As an immediate consequence of the well-known properties of the Padé approximations to e^μ [6], we have the following result.

THEOREM. *The class $[p, p]$ is A-stable and the classes $[p, p + 1]$ and $[p, p + 2]$ are strongly A-stable for $p \geqslant 1$.*

There are implicit one-step methods possessing similar stability properties. These include the method of Hermite [8]

$$(10) \qquad y_{n+1} = y_n + \sum_{i=1}^{p} h^i \alpha_{pi} y_n^{(i)} - \sum_{i=1}^{q} h^i \beta_{qi} y_{n+1}^{(i)},$$

where α_{pi} and β_{qi} are the ith coefficients in the numerator and denominator of E_{pq}, and the implicit Runge-Kutta processes developed by Butcher [4], Ehle [6], Chipman [5], and Hulme [9]. However, the methods in the class $[p, q]$ enjoy certain advantages. They attain the same order of accuracy as the scheme (10) while employing derivatives of lower order, and, unlike the Runge-Kutta processes, they require the solution of only one, albeit complicated, nonlinear equation at each time step. For problems where the higher derivatives are easy to compute, it may be easier to solve a single complicated nonlinear equation rather than a system of simpler nonlinear equations. This potential advantage would be more pronounced for a system of differential equations.

We turn now to s-block methods. Consider the class $[p_0, \ldots, p_s]$, where $p_i = p_{s-i}$ for $i = 0, \ldots, [s/2]$. A straightforward but tedious calculation shows that

$$R(\mu) = P_p(\mu)/P_p(-\mu),$$

where $P_p(\mu)$ is a polynomial of degree $p = \Sigma_{i>0} p_i$ in μ. It follows that $|R(i\omega)| = 1$ for $\omega \in (-\infty, \infty)$ and $|R(\mu)| \longrightarrow 1$ as $\mu \longrightarrow \infty$. Hence if all the zeros of $P_p(\mu)$ have negative real parts, then $R(\mu)$ is regular for $\mathrm{Re}(\mu) < 0$ and it follows that the class is A-stable. We computed the coefficients of $P_p(\mu)$ and applied the Routh-Hurwitz conditions to $P_p(\mu)$ using a FORMAC program with exact rational arithmetic to establish the following result which we conjecture is true for all $p \geqslant 1$.

THEOREM. *The class $[p, p, p]$ is A-stable for $p \leqslant 10$.*

There are other A-stable classes with symmetric p_i. For example, the classes $[1, 2, 1]$ and $[2, 1, 2]$ are A-stable.

There are strongly A-stable and strongly stiffly stable block methods. For example, the class $[1, 1, 2]$ is strongly A-stable, and the classes $[1, 1, 3]$, $[1, 2, 2]$, $[1, 2, 3]$, $[2, 2, 3]$, $[1, 1, 1, 2]$, and $[1, 1, 2, 2]$ are strongly stiffly stable. Figure 1 shows the loci in the $h\lambda$ plane where $|R| = 1$ for these classes of methods. Since the loci are symmetric with respect to the real axis, only half of each locus is plotted. The regions of absolute stability lie to the left of the loci. The regions of instability in the left-half plane are remarkably small. The corresponding regions of instability for Gear's stiffly stable backward difference multistep methods are larger by several orders of magnitude. Hence the block methods are better suited to problems where the Jacobian has complex eigenvalues near the imaginary axis.

This discussion shows that Methods 1 and 4 are A-stable, Method 2 is strongly A-stable, and Methods 3, 5, and 7 are strongly stiffly stable. The discussion does not apply to Method 6 because the degree of precision of its first formula is too low, but it is simple to show that $R(\mu)$ for this method is the $(4, 4)$ entry in the Padé table for $e^{2\mu}$. Thus Method 6 is A-stable.
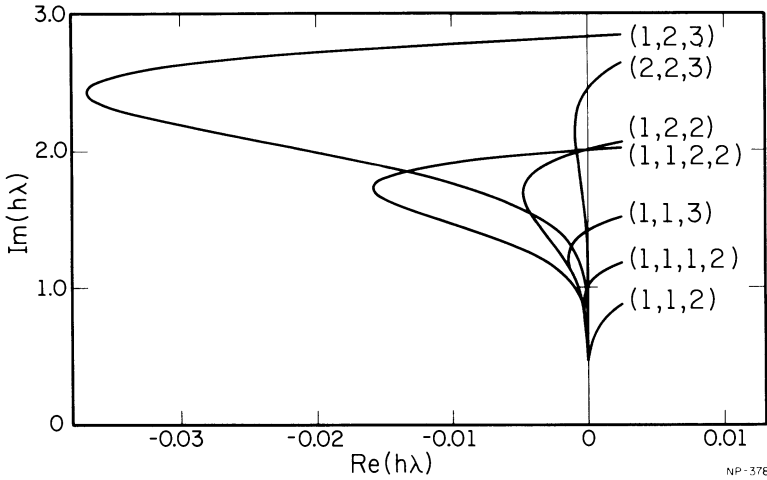
FIGURE 1

Stability regions for 2- and 3-block methods

5. Numerical Examples. Krogh [7] has proposed the following nonlinear stiff test problem. The nonlinear differential equations

$$z_i' = -\beta_i z_i + z_i^2, \quad i = 1, \dots, 4,$$

have the solutions

$$z_i = \beta_i/(1 + c_i \exp(\beta_i x)).$$

If the initial value $z_i(0) = -1$, then $c_i = -1 - \beta_i$. If we set $\mathbf{y} = U\mathbf{z}$, where $\mathbf{z} = (z_i)^T$ and U is a unitary matrix, then the differential equation for $\mathbf{y}$ is

(11) $$\mathbf{y}' = -B\mathbf{y} + U\mathbf{w}$$

where $B = U \operatorname{diag}(\beta_i)U^*$, and $\mathbf{w} = (z_i^2)^T$. The eigenvalues λ_i of the Jacobian are $2z_i - \beta_i$. If $\mathbf{y}(0) = \mathbf{z}(0)$, then $\lambda_i \longrightarrow -|\beta_i|$ as $x \longrightarrow \infty$.

We solved the nonlinear problem (11) on an IBM 360/75 using the 2-block Method 4 and the newest version of DIFSUB. We chose Method 4 because it is of order 6, is A-stable, and does not use higher derivatives. The difference equations to be solved are

(12)
$$\mathbf{y}_{n+i} = \mathbf{y}_n + h[w_{i0}\mathbf{f}(\mathbf{y}_n) + w_{i1}\mathbf{f}(\mathbf{z}(x_n + \theta_1 h)) + w_{i2}\mathbf{f}(\mathbf{y}_{n+1})$$
$$+ w_{i3}\mathbf{f}(\mathbf{z}(x_n + \theta_3 h)) + w_{i4}\mathbf{f}(\mathbf{y}_{n+2})], \quad i = 1, 2,$$

where $\mathbf{f}(\mathbf{y}) = -B\mathbf{y} + U\mathbf{w}$,

$$\mathbf{z}(x_n + \theta h) = \psi_{00}(\theta)\mathbf{y}_n + \psi_{10}(\theta)\mathbf{y}_{n+1} + \psi_{20}(\theta)\mathbf{y}_{n+2}$$

$$+ h[\psi_{01}(\theta)\mathbf{f}(\mathbf{y}_n) + \psi_{11}(\theta)\mathbf{f}(\mathbf{y}_{n+1}) + \psi_{21}(\theta)\mathbf{f}(\mathbf{y}_{n+2})],$$

$\theta_1 = 1 - 1/\sqrt{3}$, $\theta_3 = 1 + 1/\sqrt{3}$, $w_1 = \{31/240, (24 + 15\sqrt{3})/80, 4/15,$
$(24 - 15\sqrt{3})/80, 1/240\}$, $w_2 = \{2/15, 3/5, 8/15, 3/5, 2/15\}$, and the $\psi_{jk}(\theta)$ are the Hermite basis functions for the interval $[0, 2]$.

The program implementing Method 4 employs Broyden's quasi-Newton method [3] to solve the nonlinear difference equations (12). Each iteration requires four derivative evaluations, $f(y_{n+1})$, $f(y_{n+2})$, $f(z(x_n + \theta_1 h))$, and $f(z(x_n + \theta_3 h))$. Thus only two derivative evaluations are required per iteration per mesh point. Note that any implicit Runge-Kutta method of order 6 would require at least three derivative evaluations per iteration per mesh point. The special structure of the test problem is not used to simplify the iterative process as is sometimes done in such tests. The Jacobian of the difference equations is computed by numerical differentiation and is recomputed whenever the stepsize is changed. The initial approximation to the solution is obtained through rational extrapolation.

The program varies the stepsize to keep the maximum norm of the scaled local error per step less than ϵ, where ϵ is a prescribed tolerance. As in DIFSUB each component of the local error is scaled by the maximum modulus of the corresponding component of the solution observed up to that point. The local error is estimated using an idea proposed by Zadunaisky and analyzed by Stetter [12]. Starting at the point x_n, the solution is first computed for two blocks. The method is then applied to the perturbed problem

$$u'(x) = f(u(x)) + v'(x) - f(v(x)), \qquad u(x_{n+1}) = y_{n+1},$$

where $v(x)$ is the Hermite interpolant matching y_{n+i} and $y_{n+i}^{(1)}$ for $i = 0, \ldots, 4$. Obviously, the exact solution of the perturbed problem is $v(x)$. The solution of the perturbed problem is computed for one block, and the difference between the exact and computed values at x_{n+3} is used to estimate the average local error over the two blocks.

Example 1. Here as in [7] we set $\beta_1 = 1000$, $\beta_2 = 800$, $\beta_3 = -10$, $\beta_4 = 0.001$, and

$$U = (\tfrac{1}{2}) \begin{bmatrix} -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \end{bmatrix}.$$

The problem was solved with $\epsilon = 10^{-i}$ for $i = 2, 4, 6$, and 8. Table 1 presents for $\epsilon = 10^{-6}$ the total number of derivative evaluations required to reach the first mesh point after $x = 10^i$ for $i = -3, -2, \ldots, 3$, and the current stepsize h, the maximum absolute error in the components of y, and the order formula used by DIFSUB at that point. Table 2 presents for each ϵ the total number of derivative evaluations and the time in seconds required to reach the first mesh point after $x = 1000$, and the maximum absolute error in the components of y observed up to that point. Table 3 presents for each ϵ the minimum, average, and maximum values of the average of the moduli of the ratios of the components of the exact and estimated local errors.

The block method generally required fewer derivative evaluations than DIFSUB while yielding comparable accuracy. DIFSUB was faster, but it should be noted that the present program was written for ease of modification and debugging and not speed and efficiency. Furthermore, the time estimates were obtained with the IBM OS/MVT

TABLE 1

Comparison for $\epsilon = 10^{-6}$

block method				DIFSUB				
x	evaluations	h	error	x	order	evaluations	h	error
.1000 − 2	31	.5000 − 3	.4175 − 6	.1011 − 2	4	65	.7423 − 4	.1140 − 5
.1087 − 1	113	.1365 − 2	.7637 − 7	.1043 − 1	6	165	.4305 − 3	.1403 − 6
.1107 + 0	191	.1556 − 1	.9005 − 9	.1025 + 0	6	275	.8105 − 2	.2018 − 5
.1101 + 1	323	.1346 + 0	.1838 − 5	.1001 + 1	6	411	.2893 − 1	.3342 − 6
.1157 + 2	427	.2535 + 1	.9886 − 7	.1044 + 2	6	530	.6287 + 0	.2382 − 5
.1228 + 3	505	.3072 + 2	.4206 − 7	.1035 + 3	6	639	.6186 + 1	.1075 − 5
.1098 + 4	557	.1920 + 3	.2565 − 5	.1016 + 4	6	719	.8030 + 2	.4226 − 6

TABLE 2

Comparison of costs

block method				DIFSUB		
ϵ	evaluations	time	error	evaluations	time	error
.1 − 1	299	2.49	.2086 − 2	223	1.94	.2344 − 1
.1 − 3	381	2.67	.8359 − 4	447	2.32	.5349 − 3
.1 − 5	557	3.19	.2565 − 5	719	2.94	.5135 − 5
.1 − 7	865	4.05	.6335 − 7	1068	3.67	.6136 − 7

TABLE 3

Average of the moduli of the ratios of the components
of the exact and estimated local errors

ϵ	minimum	average	maximum
.1 − 1	.5428	.8533	1.2352
.1 − 3	.6581	1.0182	2.2869
.1 − 5	.7237	.9260	1.3181
.1 − 7	.5447	.9188	1.3289

TABLE 4

Comparison for $\epsilon = 10^{-6}$

block method				DIFSUB				
x	evaluations	h	error	x	order	evaluations	h	error
.1000 − 2	35	.5000 − 3	.1414 − 5	.1033 − 2	5	73	.8593 − 4	.1351 − 5
.1000 − 1	155	.5000 − 3	.5618 − 5	.1009 − 1	5	225	.1354 − 3	.1266 − 4
.1010 + 0	913	.2215 − 2	.8979 − 6	.1002 + 0	5	969	.5977 − 3	.3901 − 5
.1024 + 1	1163	.9162 − 1	.3344 − 6	.1000 + 1	5	2831	.6051 − 3	.5129 − 4
.1027 + 2	1279	.1731 + 1	.6840 − 6	.1000 + 2	5	14979	.7622 − 3	.2490 − 4
.1094 + 3	1411	.1124 + 2	.8975 − 5	−	−	−	−	−
.1034 + 4	1621	.5311 + 2	.1742 − 5	−	−	−	−	−

timing routine and may be inaccurate. The error estimation scheme appears to work remarkably well and has the desirable property of tending to overestimate the error. However, the stepsizes obtained by assuming that the local error has the form Mh^7 were sometimes too small because the coefficient M decreased rapidly over the next pair of blocks.

The overall performance of the block method is comparable to that of DIFSUB on this problem. This is surprising because the eigenvalues of the Jacobian of the differential equations lie on the negative real axis so that the A-stability of Method 4 is of no advantage. However, we would expect Method 4 to outperform DIFSUB for problems where the Jacobian has eigenvalues λ_i lying relatively close to the imaginary axis because DIFSUB, if it used a high order formula, would be restricted to stepsizes sufficiently small to keep the $h\lambda_i$ within the region of absolute stability of the formula. This expectation is confirmed by the next example.

Example 2. Here we set $\beta_1 = 100 + 1000i$, $\beta_2 = \bar{\beta}_1$, $\beta_3 = -10$, $\beta_4 = 0.01$,

$$U = (\tfrac{1}{2}) \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ -i & i & 1 & -1 \\ -i & i & -1 & 1 \end{bmatrix},$$

and $\epsilon = 10^{-6}$. The results are summarized in Table 4 which has the same format as Table 1. As expected, DIFSUB performed poorly although the maximal order was restricted to 5 as recommended by Gear for such problems. By remaining at order 5 for large x, it was forced to choose $h < 0.0008$ to keep $h\lambda_1$ and $h\lambda_2$ within the region of absolute stability. It would have required about 1.3×10^6 derivative evaluations to reach $x = 1000$.

Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, Illinois 61801

1. G. D. ANDRIA, G. D. BYRNE & D. R. HILL, "Natural spline block implicit methods," *BIT*, v. 13, 1973, pp. 131–144.

2. D. BARTON, I. M. WILLERS & R. V. M. ZAHAR, "Taylor series methods for ordinary differential equations—an evaluation," in *Mathematical Software* (J. R. Rice, Editor), Academic Press, New York, 1971, pp. 369–390.

3. C. G. BROYDEN, "A new method of solving nonlinear simultaneous equations," *Comput. J.*, v. 12, 1969, pp. 94–99.

4. J. C. BUTCHER, "Implicit Runge-Kutta processes," *Math. Comp.*, v. 18, 1964, pp. 50–64.

5. F. H. CHIPMAN, *Numerical Solution of Initial Value Problems Using A-Stable Runge-Kutta Processes*, Ph. D. Thesis, Univ. of Waterloo, Waterloo, Ontario, 1971.

6. B. L. EHLE, *On Padé Approximations to the Exponential Function and A-Stable Methods for the Numerical Solution of Initial Value Problems*, Ph. D. Thesis, Univ. of Waterloo, Waterloo, Ontario, 1969.

7. C. W. GEAR, *Numerical Initial Value Problems in Ordinary Differential Equations*, Prentice-Hall, Englewood Cliffs, N. J., 1971.

8. C. HERMITE, "Sur la formule d'interpolation de Lagrange," *J. Reine Angew. Math.*, v. 84, 1878, pp. 70–79.

9. B. L. HULME, "Discrete Galerkin and related one-step methods for ordinary differential equations," *Math. Comp.*, v. 26, 1972, pp. 881–891.

10. J. B. ROSSER, "A Runge-Kutta for all seasons," *SIAM Rev.,* v. 9, 1967, pp. 417–452.

11. L. F. SHAMPINE & H. A. WATTS, "Block implicit one-step methods," *Math. Comp.,* v. 23, 1969, pp. 731–740.

12. H. J. STETTER, "Economical global error estimation," in *Stiff Differential Systems* (R. A. Willoughby, Editor), Plenum Press, New York, 1974, pp. 245–258.

13. H. A. WATTS & L. F. SHAMPINE, "*A*-stable block implicit one-step methods," *BIT,* v. 12, 1972, pp. 252–266.

14. J. WILLIAMS & F. de HOOG, "A class of *A*-stable advanced multistep methods," *Math. Comp.,* v. 28, 1974, pp. 163–177.