Journal of Computational and Applied Mathematics 16 (1986) 95–103
North-Holland

Relaxed Newton-like methods for stiff differential systems

T.J. YPMA
Department of Applied Mathematics, University of the Witwatersrand, Johannesburg, 2001 South Africa

Received 4 Juny 1985
Revised 26 August 1985

Abstract: Newton-like methods are commonly used to solve the nonlinear equations arising in the numerical solution of stiff differential equations. We show that easily calculable relaxation factors may be used to improve the convergence properties of such methods. The technique is also applicable when partitioning methods are used.

Keywords: Stiff differential equations, nonlinear equations, Newton's method, partitioning, damping.

1. Introduction

The use of certain implicit numerical methods for the solution of the autonomous differential equation

$$y'(x) = f(y(x)), \qquad y\colon\ X \subseteq \mathbb{R}^1 \to U \subseteq \mathbb{R}^N, \quad f\colon\ U \subseteq \mathbb{R}^N \to V \subseteq \mathbb{R}^N \tag{1.1}$$

leads to the need to calculate, for $n = 1, 2, \ldots$, a solution $y_n^* \in U$ of a nonlinear equation of the form

$$F_n(y) = 0, \quad F_n(y) := y - h_n\beta_n f(y) - g_n \tag{1.2}$$

where h_n, $\beta_n \in \mathbb{R}^1$ and $g_n \in \mathbb{R}^N$ are given [12,13]. Equation (1.1) is autonomous only for convenience in the exposition; nonlinear equations of the same form also arise in the general case. Newton's method for solving (1.2) generates successive approximations y_n^i to y_n^* as

$$y_n^{i+1} = y_n^i + d_n^i, \quad i = 0, 1, \ldots \tag{1.3}$$

where y_n^0 is some first estimate of y_n^* and d_n^i solves the linear equation

$$\left[I - h_n\beta_n J(y_n^i)\right] d_n^i = -F_n(y_n^i), \tag{1.4}$$

where $J(y_n^i)$ is the Jacobian matrix of f, evaluated at y_n^i. The use of (1.4) is computationally expensive, so it is common practice [1–15] to replace (1.4) by an approximating linear equation

$$\left[I - h_m\beta_m \bar{J}\right] d_n^i = -F_n(y_n^i) \tag{1.5}$$

where $m \leqslant n$, $h_m\beta_m \simeq h_n\beta_n$, and $\bar{J} \in \mathscr{L}(\mathbb{R}^N)$ (where $\mathscr{L}(\mathbb{R}^N)$ is the set of real $N \times N$ matrices) is some approximation to $J(y_n^i)$ which may remain unaltered for a variety of values of i and n.

Apart from the fact that one now requires only one approximate Jacobian evaluation and one matrix factorization for all such values of i and n, (1.5) provides the freedom to impose various structures and properties on $\bar{J}$ which help make equation (1.5) relatively cheap to solve. Ideas of this sort have been exploited in the techniques of tearing [4,11,13] and partitioning [1,2,5,7,13,15]; a review of work in this area may be found in [13].

Recent numerical experience [3,9,10,14] has shown that if (1.5) is used it is worth replacing (1.3) by a relaxed iteration

$$y_n^{i+1} = y_n^i + R_n d_n^i, \quad i = 0, 1, \ldots \tag{1.6}$$

where $R_n \in \mathbb{R}^1$ is selected to improve the properties of the sequence $\{y_n^i\}$ generated by (1.5)–(1.6). In this paper we examine this idea and, following a proposal in [14], extend it to selecting $R_n \in \mathscr{L}(\mathbb{R}^N)$. The technique is easy to incorporate into existing software for solving stiff differential systems, including cases where partitioning methods are used.

Our standpoint is that R_n compensates for $h_m\beta_m \neq h_n\beta_n$ in (1.5), so that the matrix $R_n^{-1}[I - h_m\beta_m\bar{J}]$ implicitely used in (1.5)–(1.6) remains an acceptable approximation to the matrix in (1.4) even when the steplength h_n and coefficient β_n currently in use differ significantly from h_m and β_m. Thus the aim is to reduce the number of matrix factorizations used to solve (1.1). Note, however, that we cannot expect the use of R_n to reduce the (low) number of iterations usually required to solve (1.2) by (1.3)/(1.5) while the matrix in (1.5) remains a good approximation to that in (1.4).

In line with established tactics [1–4,6–11,13–15], and supported by the same arguments, our analysis is restricted to the constant coefficient linear differential equation $y'(x) = Jy(x)$, with the understanding that this analysis serves to guide us in the solution of general nonlinear differential equations.

For brevity in what follows we write a and b for $h_n\beta_n$ and $h_m\beta_m$ respectively, and drop the subscript n throughout. We assume $a, b \in \mathbb{R}^1$; $a > 0$, $b > 0$. The eigenvalues of J have the generic form $\lambda = \gamma e^{i\omega}$ with $\gamma \geq 0$ and $\omega \in [\frac{1}{2}\pi, \frac{3}{2}\pi]$, so $\mathrm{Re}(\lambda) \leq 0$. The individual eigenvalues of J are denoted λ_j and ordered as $\gamma_1 \geq \gamma_2 \geq \cdots \geq \gamma_N$. We suppose that $a\gamma_1 \gg 1$ and $a\gamma_N \ll 1$, i.e. we assume that the problem is *stiff*.

2. The scalar case

We set $R := r \in \mathbb{R}^1$ throughout this section. Then the modified Newton method (1.5)–(1.6) is linearly convergent with rate equal to the spectral radius $\rho(E(r))$ of the matrix $E(r) := I - r[I - b\bar{J}]^{-1}[I - aJ]$. Viewing the introduction of r as an attempt to compensate for $b \neq a$, we isolate this effect by assuming, throughout Section 2, that $\bar{J} = J$. Then the eigenvalues of $E(r)$ are $\theta(r, \lambda_j)$, $j = 1, \ldots, N$, where

$$\theta(r, \lambda) := 1 - r[1 - b\lambda]^{-1}[1 - a\lambda].$$

Thus one may improve the rate of convergence by selecting r to minimize $\max_j |\theta(r, \lambda_j)|$. Defining $s \in \mathbb{R}^1$ through the equation

$$b(ra)^{-1} = 1 + (a^{-1} - b^{-1})s^{-1} \tag{2.1}$$

it is easy to confirm that, given λ_j, $|\theta(r, \lambda_j)|$ is minimized if [9]

$$s = b^{-1} - \mathrm{Re}(\lambda_j) + [\mathrm{Im}(\lambda_j)]^2[a^{-1} - \mathrm{Re}(\lambda_j)]^{-1}. \tag{2.2}$$

Since this differs for different λ_j, some compromise is required. In the absence of knowledge of the values λ_j, a realistic goal is to minimize the supremum, over all λ in the negative halfplane, of $|\theta(r, \lambda)|$. The appropriate value of r is generally the Butcher factor [3,9] $r_2 := 2b(a+b)^{-1}$. In [9] it is shown that when either of the two conditions $(a\gamma_1)(b\gamma_1) < 1$ or $(a\gamma_n)(b\gamma_N) > 1$ are satisfied then the choices

$$r = \frac{1 + (a\gamma_1)(b\gamma_1)}{1 + (a\gamma_1)^2} \quad \text{or} \quad r = \frac{1 + (a\gamma_N)(b\gamma_N)}{1 + (a\gamma_N)^2},$$

respectively, are appropriate in the present context. These conditions are both of interest, the first in the initial transient region when the integration is in the process of making the transition from being nonstiff to being stiff, and the second when the system has decayed except for the effect of a forcing function. However, during most of the stiff phase both very small and very large eigenvalues typically occur (see, for example, [5]), and then these conditions are not satisfied. Accordingly the above two choices are not considered further in Section 2, though we return to them in section 3.

The choice r_2 is optimal only in the sense that it minimizes the supremum of $|\theta(r, \lambda)|$over all λ in the negative halfplane. However, various other objectives could be used to select a value of r; we consider the following ideas suggested by [12,14]. Since (1.2) produces $y^* = af(y^*) + g$, it follows from (1.5)–(1.6) that

$$\begin{aligned} y^{i+1} - y^* &= y^i - y^* - r[I - bJ]^{-1}\big(y^i - y^* - a[f(y^i) - f(y^*)]\big) \\ &= \big(I - r[I - bJ]^{-1}[I - aJ]\big)(y^i - y^*), \quad i = 0, 1, \ldots. \end{aligned}$$

Assuming that the eigenvectors $u_j \in \mathbb{C}^N$ of J form a basis for $\mathbb{R}^N$, and writing

$$y^m - y^* = \sum_{j=1}^{N} \alpha_m^j u_j, \quad m = 0, 1, \ldots,$$

for the appropriate constants α_m^j, this leads to

$$\alpha_{i+1}^j = \big(1 - r[1 - b\lambda_j]^{-1}[1 - a\lambda_j]\big)\alpha_i^j = \theta(r, \lambda_j)\alpha_i^j, \quad j = 1, \ldots, N. \tag{2.3}$$

Expression (2.3) describes the damping effect of iteration (1.5)–(1.6) on the components of the error corresponding to various eigenvalues of J [12,14]. Clearly errors corresponding to very large eigenvalues will be heavily damped if we select r as the Tischer factor [14] $r_1 := a^{-1}b$, while for small eigenvalues choosing r to be $r_3 := 1$ [14] is appropriate. The value r_2 is a compromise between these extremes. In this context, selecting r entails deciding which error components to damp out most rapidly.

Since "the aim of the corrector step is to attempt to gain stability by eliminating the components of the error vector, corresponding to the large eigenvalues of the Jacobian, that have been amplified by the predictor" [8], either r_1 or r_2 should be used.

To examine the effect of choosing r_l we investigate $\theta_l := |\theta(r_l, \lambda)|$, $l = 1, 2, 3$; where $\lambda \equiv \gamma e^{i\omega}$ with $\gamma \geq 0$, $\omega \in [\frac{1}{2}\pi, \frac{3}{2}\pi]$. Regions of nonconvergence correspond to $\theta_l \geq 1$. Firstly, $\theta_3 \geq 1$ iff

$$b\gamma \leq \big(1 - (a\gamma)^2\big)/2(\cos\omega - a\gamma); \tag{2.4}$$

the limits on $b\gamma$ for the cases $\omega = \frac{1}{2}\pi$, π are shown in Fig. 1. Clearly using $r_3 \equiv 1$ leads to convergence difficulties if $b < \frac{1}{2}a$ and $a\gamma$ is at all large, particularly if λ is near the imaginary

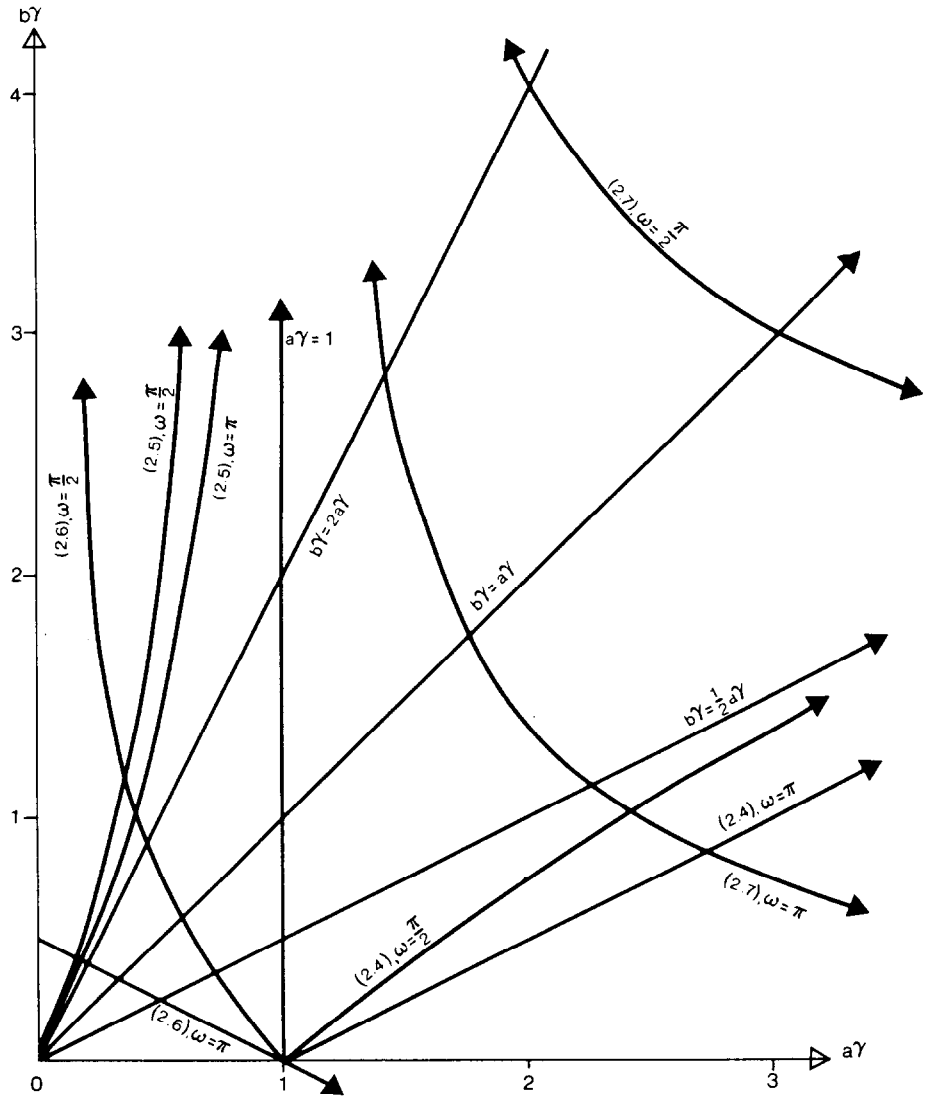

Fig. 1. Bounds on by defined by inequalities (2.4)–(2.7).

axis. This shows that the undamped iteration (1.3)/(1.5) may fail if doubling of the steplength, which leads to $a > 2b$, occurs. Similarly, $\theta_1 \geqslant 1$ iff $a\gamma \leqslant 1$ and

$$b\gamma \geqslant 2a\gamma(1 - a\gamma \cos \omega)/\big(1 - (a\gamma)^2\big); \tag{2.5}$$

the limits on $b\gamma$ for the cases $\omega = \frac{1}{2}\pi$, π are also shown on Fig. 1. The use of $r_1 \equiv a^{-1}b$ thus leads to convergence difficulties if $2a < b$ and $a\gamma$ is small, especially if λ is near the real axis. In this case halving the steplength may lead to failure of the iteration. By contrast, $\theta_2 < 1$ for all λ. Since many practical methods for solving (1.1) involve either doubling or halving of steplengths (see e.g. [10]) the use of r_1 or r_3 cannot *generally* be recommended.

We briefly examine the relative damping effects of using r_l. We find $\theta_2 \leqslant \theta_3$ iff

$$b\gamma \geqslant \left(1 - (a\gamma)^2\right)/2(a\gamma - \cos\omega), \tag{2.6}$$

the ratio $\theta_2\theta_2^{-1}$ tending to $b(a+b)^{-1}$ for large γ; while $\theta_1 \leqslant \theta_2$ iff $a\gamma > 1$ and

$$b\gamma \geqslant 2a\gamma(1 - a\gamma\cos\omega)/\left((a\gamma)^2 - 1\right), \tag{2.7}$$

the ratio $\theta_1\theta_2^{-1}$ tending to 0 for large γ. The limiting cases $\omega = \frac{1}{2}\pi, \pi$ of (2.6) and (2.7) are also shown on Fig. 1. So for all stiff eigenvalues ($a\gamma > 1$) r_1 and r_2 are preferable to r_3, while for the larger eigenvalues r_1 is preferable to r_2 *from the point of view of damping,* and hence of stability.

Since the rate of convergence depends on the spectral radius ρ_l of the matrix $E(r_l)$, we compare ρ_1 to ρ_2, examining the case *least* favourable to r_2. The maximum value $\left|(a-b)(a+b)^{-1}\right|$ of ρ_2 is attained if J has a purely imaginary eigenvalue, so we assume this is the case. The value of ρ_1 is determined by the eigenvalue λ which minimizes $|1 - b\lambda|$, and it is plausible to assume that this λ satisfies $|a\lambda| < 1$ (in fact, $|\lambda| \simeq |\lambda_N|$ is to be expected). With this assumption we see that $\rho_2\rho_1^{-1} < 1$, and expect $\rho_2\rho_1^{-1} \simeq a(a+b)^{-1}$.

Finally, we quantify the advantage of using r_2 rather than r_3, by comparing the size of the relative error $\left|(b-a)a^{-1}\right|$ for which $\rho_2 = \alpha$ to the value for which $\rho_3 = \alpha$ (given $\alpha \in (0, 1)$). Again we take the case *least* favourable to r_2, hence assume J has a purely imaginary eigenvalue, $\hat{\lambda}$. The value of ρ_3 is determined by the eigenvalue λ which maximizes $|b\lambda||1 - b\lambda|^{-1}$, and this maximum is minimized if this λ is real and satisfies $|b\lambda||1 - b\lambda|^{-1} = |b\hat{\lambda}||1 - b\hat{\lambda}|^{-1}$ (we expect $|\lambda| \simeq |\lambda_1|$). If using r_2 with $ba^{-1} = t_2$ leads to the same spectral radius as the use of r_3 with $ba^{-1} = t_3$, then with the above assumptions together with $(t_2 - 1)(t_3 - 1) > 0$, we obtain by substituting for b in terms of $t_2 a$ and $t_3 a$ in the relevant expressions in $\rho_2^2 = \rho_3^2$ the equation

$$(t_2 - 1)/(t_2 + 1) = (t_3 - 1)a\gamma/(1 + t_3 a\gamma),$$

and hence after some manipulation

$$|t_2 - 1| = (2a\gamma)(1 + a\gamma)^{-1}|t_3 - 1|.$$

Thus even in this worst case the relative error $\left|(b-a)a^{-1}\right|$ we can tolerate and still get a particular rate of convergence is essentially twice as large when using r_2 as for r_3, under the plausible assumption $a\gamma \gg 1$.

We briefly summarize the advantages of using r_2. Firstly, we have convergence for *every* eigenvalue, no matter how much b differs from a. The stiff eigenvalues are better damped using r_2 rather than r_3, thus improving the stability characteristics of the method when $b \neq a$. The rate of convergence when $b \neq a$ is also improved when r_2 is used, with the range of relative errors $\left|(b-a)a^{-1}\right|$ permitted to get a specific rate of convergence being essentially doubled when r_2 is used instead of r_3. This explains the success noted in practice when r_2 is used, as in [3,9,10,14]. However, from the point of view of damping the largest eigenvalues the most, the choice of r_1 is better than r_2 although this may result in a decrease in the overall rate of convergence.

3. Partitioning

We adopt as the model partitioning method that outlined in Section 2 of [1]. Suppose that for some k the eigenvalues of J are partitioned as

$$|a\lambda_{k+1}| < 1, \qquad |a\lambda_k| > 1. \tag{3.1}$$

Let $K \in \mathscr{L}(\mathbb{R}^N)$ be such that the similarity transformation $K^{-1}JK = G$ reduces J to *block upper triangular* form, with K, K^{-1} and G partitioned conformally with (3.1) as

$$K = (K_1\ K_2), \qquad K^{-1} = \begin{pmatrix} L_1^T \\ L_2^T \end{pmatrix}, \qquad G = \begin{pmatrix} G_{11} & G_{12} \\ 0 & G_{22} \end{pmatrix},$$

and suppose that $\lambda_1, \ldots, \lambda_k$ are the eigenvalues of G_{11}. Such a transformation of J to G will in practice only be calculated approximately; for example, the block QR iteration of [2] might be used, with the undesired elements set to 0 when they have become sufficiently small. Set $\bar{J} := K\bar{G}K^{-1}$ and $R := KDK^{-1}$, where $\bar{G}$ and the *diagonal* matrix $D \in \mathscr{L}(\mathbb{R}^N)$ are given by

$$G = \begin{pmatrix} G_{11} & G_{12} \\ 0 & E \end{pmatrix}, \qquad D = \begin{pmatrix} D_1 & 0 \\ 0 & D_2 \end{pmatrix},$$

with D also partitioned conformally with (3.1). The matrix E is usually taken as 0 [1,2,7,13], but more generally we shall allow E to be any upper triangular matrix such that $I - bE$ is nonsingular. One might, for example, let E be the diagonal or upper triangular portion of G_{22}; we shall show that this is useful provided that E is sufficiently close to G_{22} that the eigenvalues of $[I - bE]^{-1}[I - aG_{22}]$ are in some sense close to those of $[I - bG_{22}]^{-1}[I - aG_{22}]$. Then (1.5)–(1.6) becomes

$$y^{i+1} = y^i - R[I - b\bar{J}]^{-1}F(y^i) \equiv y^i - KD[I - b\bar{G}]^{-1}K^{-1}F(y^i), \tag{3.2}$$

and we obtain as in Section 2

$$y^{i+1} - y^* = \left[K\left(I - D[I - b\bar{G}]^{-1}[I - aG]\right)K^{-1}\right](y^i - y^*). \tag{3.3}$$

Substituting for D, $\bar{G}$ and G we find

$$I - D[I - b\bar{G}]^{-1}[I - aG] = \begin{pmatrix} I - D_1[I - bG_{11}]^{-1}[I - aG_{11}] & -D_1[I - bG_{11}]^{1}Y \\ 0 & I - D_2[I - bE]^{-1}[I - aG_{22}] \end{pmatrix} \tag{3.4}$$

where $Y \equiv -aG_{12} + bG_{12}[I - bE]^{-1}[I - aG_{22}]$. Thus the eigenvalues of the iteration matrix in (3.3) are the eigenvalues of $I - D_1[I - bG_{11}]^{-1}[I - aG_{11}]$ and $I - D_2[I - bE]^{-1}[I - aG_{22}]$, and D_1 and D_2 may be chosen to modify these eigenvalues as desired.

For any form of G_{11}, G_{22} and E, and any choice of D_1 and D_2, we have from (3.2) the following technique for calculating y_{i+1} [2]:

(i) compute $d \equiv K^{-1}F(y^i)$ and partition $d = \begin{pmatrix} d_1 \\ d_2 \end{pmatrix}$,

(ii) solve $[I - bE]c_2 = d_2$,

(iii) solve $[I - bG_{11}]c_1 = d_1 + bG_{12}c_2$, (3.5)

(iv) set $y^{i+1} = y^i - K\begin{pmatrix} D_1c_1 \\ D_2c_2 \end{pmatrix}$.

To make (ii) cheap to solve the choices $E = 0$ or E upper triangular are adopted; thus producing y^{i+1} requires only the full solution of one relatively small $(k \times k)$ system of linear equations, and the effort of calculating the similarity transformation is exploited both in reducing the size of the system of linear equations to be solved and selecting the relaxation factors D_1 and D_2, as we shall show. Our method is identical to those proposed in [1,2,7,11,13,15] except for our use of the diagonal relaxation matrices D_1 and D_2 and the possibility that $E \neq 0$; thus the technique requires only a trivial modification of existing software for solving stiff differential equations by partitioning, while the extra computational effort is small. Beware that the partitioning desired will change as the integration proceeds, since for most stiff problems the step size increases as various transients die out. Thus initially G_{11} might be empty, but as the integration proceeds G_{11} will grow in size, so that eventually the cost of partitioning escalates to the extent that partitioning is no longer worthwhile. One must then revert to the full matrix method and the ideas of Section 2.

We now examine the choice of the relaxation matrix D. Let $D = \operatorname{diag}(d_1, \ldots, d_N)$. If G_{11} is upper triangular then $\lambda_1, \ldots, \lambda_k$ are known and real, so the optimal choice $d_j = (1 - a\lambda_j)^{-1}(1 - b\lambda_j)$, $j = 1, \ldots, k$ is obvious and calculable. If G_{22} is upper triangular then an obvious choice is $E = G_{22}$, in which case the choice $d_j = (1 - a\lambda_j)^{-1}(1 - b\lambda_j)$, $j = k+1, \ldots, N$ is again at hand. In both of these cases the eigenvalues of the relevant matrices on the diagonal of (3.4) become 0. In the second case the identical effect on (3.4) could have been obtained by setting $E = 0$ and $d_j = (1 - a\lambda_j)^{-1}$, $j = k+1, \ldots, N$; this also reduces the expense of (3.5) by eliminating (3.5) (ii) but produces different values of y^{i+1}.

Generally it is impractical to transform G to upper triangular form, and G_{11} and G_{22} have some other (e.g. upper Hessenberg) form. In this case we let D_1 and D_2 be of the form rI and, in analogy with Section 2, we select suitable values of r depending on the objectives we wish to attain.

We first consider the upper left-hand corner of (3.4); the eigenvalues of this matrix are $\theta(r, \lambda_j)$, $j = 1, \ldots, k$ so the situation is precisely that of Section 2, with N replaced by k, except that we here have the additional assumption $a\gamma_j > 1$, $j = 1, \ldots, k$. In particular we may expect $(a\gamma_k)(b\gamma_k) \geq 1$, and to improve the rate of convergence we therefore apply the appropriate factor of [9] mentioned in Section 2, namely

$$r_4 = \left[1 + (a\gamma_k)(b\gamma_k)\right] / \left[1 + (a\gamma_k)^2\right].$$

The value γ_k can be cheaply estimated by applying an inverse power method to G_{11}, a partial factorization of which may be available from the work done in solving (3.5) (iii) [6,7]. Notice r_4 varies continuously with γ_k, $r_4 = r_2$ if $(a\gamma_k)(b\gamma_k) = 1$, and r_4 tends to r_1 as γ_k increases. Note, however, that although one aims to partition the eigenvalues so that $a\gamma_k > 1$, in practice $a\gamma_k < 1$ often occurs [1,2,7]; if $(a\gamma_k)(b\gamma_k) < 1$ then the arguments of [9] show that r_2 is preferable to r_4 for improving the rate of convergence. To achieve the latter objective our recommendation is therefore exactly that of [9] adapted to partitioning: estimate γ_k, use r_4 if $(a\gamma_k)(b\gamma_k) > 1$, else use r_2. If one wishes to avoid the expense of estimating γ_k then use r_2 which, although possibly not optimal, is safe and better than using $r = 1$, i.e. not relaxing at all.

We have noted that r_4 tends to r_1 as γ_k increases. Now $a\gamma_1, \ldots, a\gamma_k$ are all at least moderately large, Section 2 shows that r_1 is a desirable choice from the point of view of damping the larger eigenvalues rapidly, while Fig. 1 shows that the arguments of Section 2 against the use of r_1, namely for nonconvergence if $2a < b$ (see (2.5)), fall away provided $a\gamma_k$ is not much less than 1.

Thus r_1 is now also a safe choice, its use does not require that any eigenvalues be estimated, and it is better than r_2 or r_4 from the point of view of damping the largest eigenvalues although it may decrease the overall rate of convergence.

We now consider the choice of r in $D_2 = rI$. If $E = 0$ then the eigenvalues of $I - D_2[I - bE]^{-1}[I - aG_{22}]$ are $1 - r(1 - a\lambda_j)$, $j = k+1, \ldots, N$. The technique of [9] may be used to show that the value of r which minimizes the maximum of $\{ |1 - r(1 - a\lambda_j)| : j = k+1, \ldots, N \}$ is $r_5 = (1 + a^2\gamma_{k+1}^2)^{-1}$. The value γ_{k+1} is easily estimated by direct power iteration on G_{22}, hence r_5 is cheaply calculable and is the optimal choice for rapid convergence. If the expense of estimating γ_{k+1} is to be avoided, the pessimistic estimate $a\gamma_{k+1} = 1$ may be used, resulting in the choice $r = \frac{1}{2}$.

If $E \neq 0$, suppose E is so close to G_{22} that the eigenvalues of $[I - bE]^{-1}[I - aG_{22}]$ are essentially the same as those of $[I - bG_{22}]^{-1}[I - aG_{22}]$. Then r may be chosen to minimize $\max |\theta(r, \lambda_j)|$, $j = k+1, \ldots, N$, as in Section 2, noting that now $(a\gamma_{k+1})(b\gamma_{k+1}) < 1$ is expected. To improve the rate of convergence we therefore use the appropriate factor of [9] mentioned in Section 2, namely

$$r_6 = [1 + (a\gamma_{k+1})(b\gamma_{k+1})] / (1 + (a\gamma_{k+1})^2)$$

where γ_{k+1} is estimated as above. If the expense of estimating γ_{k+1} is to be avoided then r_2 or r_3 may be used; r_1 should be avoided since Fig. 1 (2.5) shows it is dangerous in the region $a\gamma < 1$, while there are no large eigenvalues to be damped here. We prefer r_2 to r_3 since it is obtained from r_6 under the pessimistic estimate $(a\gamma_{k+1})(b\gamma_{k+1}) = 1$ and damps those eigenvalues with $a\gamma$ near 1 better than does r_3, as shown by Fig. 1 (2.6).

We note that in [1,2,6,7,11,13] it is usually assumed that $b = a$, implying that the matrix in (3.5) (iii) is refactored whenever a changes. This assumption is partly justified by the fact that the reduced matrix $[I - aG_{11}]$ may be relatively cheaply refactored if G_{11} has some desirable (e.g. upper Hessenberg) form [1,2,6,7]. Although such refactorization is fairly cheap, it is obviously still cheaper not to refactor at all, thus many implementations use $b \neq a$, only refactoring when convergence has become unacceptably slow or, pre-empting this, when the relative change $|b - a|a^{-1}$ becomes large. The introduction of relaxation matrices D_1 and D_2 improves the convergence when $b \neq a$ and thus permits the use of a larger range of values $b \neq a$ before refactoring becomes necessary, consequently reducing the overall computational expense. Moreover, as pointed out in Section 2, relaxation improves the stability characteristics of the method when $b \neq a$, since the larger eigenvalue terms are more rapidly damped out. Since the introduction of relaxation is so cheap and simple it is worth doing even though the benefits may be relatively minor in the present context of partitioning.

Acknowledgements

Part of this work was done while visiting the Department of Computer Science, University of Auckland, New Zealand, and supported by a research grant from the Council for Scientific and Industrial Research, Pretoria, South Africa. I am grateful to John Butcher for helpful discussions concerning parts of this work, and to a referee for suggesting several improvements.

References

[1] Å. Björck, Some methods for separating stiff components in initial values problems, *Proceedings of the 1983 Dundee Conference on Numerical Analysis* (Springer, Berlin, 1984).

[2] Å. Björck, A block QR algorithm for partitioning stiff differential systems, *BIT* **23** (1983) 329–345.

[3] K. Burrage, J.C. Butcher and F.H. Chipman, An implementation of singly implicit Runge–Kutta methods, *BIT* **20** (1980) 326–340.

[4] M.B. Carver and S.R. MacEwen, On the use of sparse matrix approximation to the Jacobian in integrating large sets of ordinary differential equations, *SIAM J. Sci. Stat. Comp.* **2** (1981) 51–64.

[5] A.R. Curtis, Jacobian matrix properties and their impact on choice of software for stiff ODE systems, *IMA J. Numer. Anal.* **3** (1983) 397–415.

[6] W.H. Enright, Improving the efficiency of matrix operations in the numerical solution of stiff ordinary differential equations, *ACM Trans. Math. Soft.* **4** (1978) 127–137.

[7] W.H. Enright and M.S. Kamel, Partitioning of stiff systems and exploiting the resulting structure, *ACM Trans. Math. Soft.* **5** (1979) 374–385.

[8] C.W. Gear and Y. Saad, Iterative solution of linear equations in ODE codes, *SIAM J. Sci. Stat. Comp.* **4** (1981) 583–601.

[9] F.T. Krogh and K. Stewart, Asymptotic ($h \to \infty$) absolute stability for BDFs applied to stiff differential equations, *ACM Trans. Math. Soft.* **10** (1984) 45–57.

[10] F.T. Krogh and K. Stewart, Preliminary comparison test results for STRUT, LSODE and LSODA, Memorandum 499, Jet Prop. Lab., Pasadena, CA, 1984.

[11] H.H. Robertson, Numerical integration of systems of stiff ordinary differential equations with special structure, *J. Inst. Maths. Applics.* **18** (1976) 249–263.

[12] L.F. Shampine, Implementation of implicit formulas for the solution of ode's, *SIAM J. Sci. Stat. Comp.* **1** (1980) 103–118.

[13] G. Söderlind, On the efficient solution of nonlinear equations in numerical methods for stiff differential systems, Report TRITA-NA-8114, Royal Instit. Tech., Stockholm (1981.

[14] P.E. Tischer, The cyclic use of linear multistep formulas for the solution of stiff differential equations, Ph.D. Thesis, Monash University, Melbourne 1983.

[15] D.S. Watkins and R.W. Hanson Smith, The numerical solution of separably stiff systems by precise partitioning, *ACM Trans. Math. Soft.* **9** (1983) 293–301.