International Journal of Pure and Applied Mathematics
Volume 101 No. 4 2015, 463-476
ISSN: 1311-8080 (printed version); ISSN: 1314-3395 (on-line version)
url: http://www.ijpam.eu
doi: http://dx.doi.org/10.12732/ijpam.v101i4.2

A NON-LINEAR SCHEME BASED ON PROJECTION METHOD FOR TWO STAGE GAUSS METHOD

R. Vigneswaran

Department of Mathematics and Statistics
Faculty of Science
University of Jaffna
SRI LANKA

Abstract: A variety of linear iteration schemes with reduced linear algebra costs have been proposed to solve the non-linear equations arising in the implementation of implicit Runge-Kutta methods as an alternative to the modified Newton iteration scheme. In this paper, a non-linear scheme based on projection method is proposed to accelerate the convergence rates of linear iteration schemes. In particular, for an s-stage Runge-Kutta method, an s-step non-linear scheme is proposed, which is computationally more efficient. For two stage Gauss method, some theoretical results are established in order to improve the rate of convergence of linear iteration schemes. Finally, some numerical experiments are carried out to confirm the results established in this paper.

AMS Subject Classification: 65L04, 65L05
Key Words: implementation, projection method, non-linear scheme, rate of convergence, stiff systems

1. Introduction

In the numerical solution of a system of n differential equations

Received: September 25, 2014

$$x^{\prime} = f(x(t)), \quad x(t_0) = x_0, \quad f : \mathbb{R}^n \to \mathbb{R}^n, \tag{1}$$

using an s-stage implicit Runge-Kutta method, the solution $x(t_{r+1})$ at grid point $t_{r+1} = t_r + h$ is computed as x_{r+1} which is given by

$$x_{r+1} = x_r + h \sum_{i=1}^{s} b_i f(y_i)$$

where the internal approximations $y_1, y_2, \ldots, y_s$ satisfy the sn equations

$$y_i = x_r + h \sum_{j=1}^{s} a_{ij} f(y_j), \quad i = 1, 2, \ldots, s \tag{2}$$

and $A = [a_{ij}]$ is the real coefficient matrix of the Runge-Kutta method. Let $Y = y_1 \oplus y_2 \oplus \ldots \oplus y_s$ be sn elements column vector and let $F(Y) = f(y_1) \oplus f(y_2) \oplus \ldots \oplus f(y_s)$. Then equation (2) may be represented by $D(Y) = 0$, where D is the approximation defect defined by $D(Z) = e \otimes x_r - Z + h(A \otimes I_n)F(Z)$, where $e = (1, 1, \ldots, 1)^T$ and $A \otimes I_n$ is the tensor product of the matrix A with $n \times n$ identity matrix I_n and, in general $A \otimes B = [a_{ij}B]$. This article deals with methods suitable for stiff systems so that the matrix A is not strictly lower triangular and, in particular, is concerned with two-stage Gauss method.

Equation $D(Y) = 0$ may be solved by the Newton iteration scheme

$$D^{\prime}(Y^{m-1})(Y^m - Y^{m-1}) = D(Y^{m-1}), \quad m = 1, 2, 3, \ldots,$$

where $D^{\prime}(Z) = h(A \otimes I_n)F^{\prime}(Z) - I_{sn}$. The derivative of F at $Z = z_1 \oplus z_2 \oplus \cdots \oplus z_s$ is the block diagonal matrix $F^{\prime}(Z) = f^{\prime}(z_1) \oplus f^{\prime}(z_2) \oplus \cdots \oplus f^{\prime}(z_s)$ where each block is the derivative of f, the Jacobian, evaluated at one of the points $z_1, z_2, \ldots, z_s$. The Newton iteration is expensive because, in each iteration step, the Jacobian of f is evaluated s times and a set of sn linear equations, whose matrix depends on the Jacobian, has to be solved. To reduce this computation it is used to evaluate the Jacobian only occasionally. Let J be the Jacobian of f evaluated at some recent point x_p, updated infrequently. The modified Newton scheme evaluates $\Delta^1, \Delta^2, \Delta^3, \ldots$ and hence $Y^1, Y^2, Y^3, \ldots$, to satisfy

$$\begin{aligned} (I_{sn} - hA \otimes J)\Delta^m &= D(Y^{m-1}), \\ Y^m &= Y^{m-1} + \Delta^m, \quad m = 1, 2, 3, \ldots \ . \end{aligned} \tag{3}$$

In each step of this iteration, a set of sn linear equations has to be solved so that this scheme is still expensive. In this approach, described by Chipman [8], schemes are developed to solve these linear equations efficiently. Schemes of this

type are discussed by Bichart [1] and Collings & Tee [9]. Butcher [4] proposed a scheme of this type, using a similarity transformation of the coefficient matrix of the method, which is particularly effective when the coefficient matrix has a single point spectrum. To deal with methods where the coefficient matrix has a more general spectrum, Enright [14] proposed the use of an additional similarity transformation to transform the Jacobian matrix of the differential system to Hessenberg form. This scheme is comparatively inefficient when $n > s$ since each of the corresponding vector transformations requires $\mathcal{O}(s^2n + sn^2)$ operations. Varah [22] proposed the use of complex arithmatics to deal with the case where the coefficient matrix of the method has a complex spectrum. Burrage [2] investigated singly implicit methods where the matrix of coefficient is similar to one of diagonally implicit form. That is, the coefficient matrix has a single point spectrum so that these methods can be implemented efficiently. A variable order, variable step code based on a family of singly implicit methods has been developed by Butcher,Burrage and Chipman [6]. This code, called STRIDE, performs fairly well for stiff problems. Cash [7] proposed a family of implicit Runge-Kutta methods of a special form which again makes the system of equations effectively of lower dimension.

The other approach is to use schemes based directly on iterative procedures. Frank and Ueberhuber [15] described the use of iterated defect correction and a variety of schemes have been discussed by Butcher [5]. He suggested how higher order methods could be used in combination with diagonally implicit methods through an iterated defect correction process. In [3], this approach is analyzed for singly implicit methods but, the rate of convergence of the scheme seems unsatisfactory.

For a singly implicit method, there is a non-singular matrix S so that $S^{-1}AS = \lambda(I_s - L)^{-1}$, where L is zero except for some ones on the sub-diagonal. On applying this transformation, the scheme (3) becomes

$$\begin{aligned} [I_s \otimes (I_n - h\lambda J)]E^m &= [(I_s - L)S^{-1} \otimes I_n]D(Y^{m-1}) + (L \otimes I_n)E^n, \\ Y^m &= Y^{m-1} + (S \otimes I_n)E^m, \quad m = 1, 2, 3, \dots . \end{aligned} \tag{4}$$

Cooper and Butcher [10] proposed an iterative scheme, sacrificing superlinear convergence for reduced linear algebra cost, which may be regarded as a generalization of the scheme (4) for singly implicit methods.

They considered the scheme

$$\begin{aligned} [I_s \otimes (I_n - h\lambda J)]E^m &= (B_1S^{-1} \otimes I_n)D(Y^{m-1}) + (L_1 \otimes I_n)E^m, \\ Y^m &= Y^{m-1} + (S \otimes I_n)E^m, \quad m = 1, 2, 3, \dots, \end{aligned} \tag{5}$$

where B_1 and S are real $s \times s$ non-singular matrices and L_1 is strictly lower triangular matrix of order s, and λ is a real constant. Cooper and Butcher [10] also showed that successive over-relaxation may be applied to improve the rate of convergence for scalar test problem. Peat and Thomas [21], after extensive numerical experiments, concluded that the schemes proposed by Cooper and Butcher are, in general, the most efficient schemes for integration of stiff problems. Gladwell and Thomas [17] recommended this scheme for two-stage Gauss method. Each step of the scheme (5) requires s function evaluations and the solution of s sets of n linear equations. These s sub-steps are performed in sequence and it is not possible to compute elements of $Y^m = y_1^m \oplus y_2^m \oplus \cdots \oplus y_s^m$ until all sub-steps are completed. Cooper and Vignesvaran [11] considered a scheme where these elements are obtained in sequence and the approximation defect is updated after each sub-step is completed. Only one vector transformation is needed for each full step so that this scheme is more efficient. The rate convergence of this scheme has been improved in [24], [25], [26]. Cooper and Vignesvaran [12] proposed another scheme, which is a generalization of the basic scheme (5), to obtain improved rate of convergence, by adding extra substeps. Further improvement in the rate of convergence of this scheme has been obtained in [23]. A scheme for Gauss methods was proposed in [18] using an iterative procedure of semi-implicit type in which the Jacobian does not appear explicitly. A scheme of this type was proposed in [19] in which convergence and stability properties of the scheme are discussed in detail. Cooper and Vignesvaran [13] proposed another scheme suitable for parallel processing and they obtained the convergence results for this scheme under assumptions suitable for stiff problems.

In section 2 of this article, a non-linear scheme is proposed to accelerate the rates of convegence of linear iteration schemes. In particular an s - step non-linear scheme is presented here, which is computationally more efficient. In section 3, some numerical results are given.

2. A Non-Linear Scheme based on Projection Method

2.1. Projection Method for Linear system

Among the iterative methods for solving large linear algebraic systems of equations, those that have received the most attention have been the Jacobi and the Gauss-Seidel schemes,and their accelerated forms. But Householder [20] has dealt with another class of method which has been called projection method.

Special cases include the method of steepest decent and the other gradient based schemes used to solve linear systems. The techniques have been used to accelerate convergence of iterative process for non-linear problems.

Consider solving the linear system $Ax = b$, where A is assumed to be non-singular $n \times n$ matrix. Let x_k represent any iterate and let $\delta_k = x - x_k$, $r_k = b - Ax$, represent the error and residual respectively, where x is the true solution. A method of projection is one in which at each step, the error δ_k is resolved into two components, one of which is required to lie in a subspace selected at that step, and the other is δ_{k+1}, which is required to be less than δ_k in some norm. The subspace is selected by choosing a matrix Y_k , whose columns are linearly independent and form a basis for the subspace. In practice Y_k is generally a single vector y_k. That is, $\delta_{k-1} = \delta_k - Y_k u_k$, where u_k is a vector (or scalar if Y_k is a vector) to be selected at the k^{th} step so that $\|\delta_{k+1}\| \leq \|\delta_k\|$, where $\|\cdot\|$ is some norm. Householder shows that $\|\delta_{k+1}\|$ is minimized by choosing u_k so that $Y_k u_k$ is the projection of δ_k onto the subspace spanned by the columns of Y_k with respect to G, where G is a positive definite matrix. This implies that $\|\delta_{k+1}\|$ is minimized when $Y_k^H G(\delta_k - Y_k u_k) = 0$, where $Y_k^H = \overline{Y_k}^T$ is the Hermitian of Y_k. Here $\|\cdot\|$ is defined by $\|\delta_k\|^2 = \delta^H G \delta$.

2.2. A Non-Linear Scheme

The above idea is used to solve the non-linear system of equations $D(Y) = 0$.

Consider the iteration scheme

$$Y^{m+1} = Y^m + \mu^m E^m, \qquad m = 1, 2, 3, \ldots, \tag{6}$$

where μ^m and E^m are a scalar and a vector respectively and let $\Delta^m = Y - Y^m$. The scalar μ^m is chosen so that $\mu^m E^m$ is the projection of Δ^m onto E^m with respect to a positive definite matrix $G^H G$, where G is a $sn \times sn$ non-singular matrix.

Hence

$$\begin{aligned} \Delta^{m+1} &= \Delta^m - \mu^m E^m, \\ \mu^m &= \frac{(GE^m)^H G \Delta^m}{(GE^m)^H G E^m}. \qquad m = 1, 2, 3, \ldots \ . \end{aligned} \tag{7}$$

Suppose that the sequence $\{Y^m\}$ has limit Y. If E^m is chosen so that $E^m \to 0$ gives $D(Y^m) \to 0$, it follows that $D(Y) = 0$. Here G and E^m have to be chosen so that the scheme can be efficiently implemented and performs well. In each step of the iteration (6) the scalar μ^m has to be calculated by using (7) but

the numerator of μ^m contains Δ^m which is not known. To make the process feasible the matrix G may be chosen as $(Q \otimes I_n)D^{\prime}(Y^m)$, where Q is an $s \times s$ non-singular matrix. Since $D(Y^m) = -D^{\prime}(Y^m)\Delta^m + O(\|\Delta^m\|)^2$, $G\Delta^m$ may be approximated by $(Q \otimes I)D(Y^m)$. Since $F^{\prime}(Y^m)$ is the block diagonal matrix and each diagonal block is the Jacobian of f at one of $y_1^m, y_2^m, \ldots, y_s^m$. Thus the evaluation of $D^{\prime}(Y^m)$ requires more computation. To reduce this, the Jacobian is computed infrequently. Let J be the Jacobian evaluated at recent point x_p. Then $F^{\prime}(x_p) = I_s \otimes J$ and $D^{\prime}(Y^m) = -(I_{sn} - hA \otimes J)$.

Hence from(7), we obtain

$$\mu^m = \frac{[(Q \otimes I)(I_{sn} - hA \otimes J)E^m]^H (Q \times I_n)D(Y^m)}{[(Q \otimes I_n)(I_{sn} - hA \otimes J)E^m]^H [(Q \otimes I_n)(I_{sn} - hA \otimes J)E^m]}. \tag{8}$$

The non-singular matrix Q and E^m have still to be chosen so that the scheme can be implemented efficiently.

2.3. The *s*-Step Scheme

Here, a particular scheme is considered where the elements of $Y^m = y_1^m \oplus y_2^m \oplus \ldots \oplus y_s^m$ are obtained in sequence and are updated after each sub-step is completed. Thus this scheme is more efficient than the general scheme given by (6) with (8).

Consider the scheme

$$\begin{aligned} Y^m &= Y^{(1)}, \\ \mu_i^m &= \frac{[(Q \otimes I_n)(I_{sn} - hA \otimes J)E_i^m]^H (Q \otimes I_n)D(Y^{(i)})}{[(Q \otimes I_n)(I_{sn} - hA \otimes J)E_i^m]^H [(Q \otimes I_n)(I_{sn} - hA \otimes J)E_i^m]}, \\ Y^{(i+1)} &= Y^{(i)} + \mu_i^m E_i^m, \qquad i = 1, 2, 3, \ldots, \\ Y^{m+1} &= Y^{(s+1)}, \qquad m = 1, 2, 3, \ldots, \end{aligned} \tag{9}$$

where $E_i^m = O \oplus O \oplus \ldots \oplus O \oplus \epsilon_i^m \oplus O \oplus \cdots \oplus O$, O the zero vector and $E^m = E_1^m + E_2^m + \cdots + E_s^m$. In this scheme, $Y^{(i)} = y_1^{m+1} \oplus y_2^{m+1} \oplus \ldots \oplus y_{i-1}^{m+1} \oplus y_i^m \oplus y_{i+1}^m \oplus \ldots \oplus y_s^m$ for $i = 1, 2, 3, \ldots, s$. The non-singular matrix Q and E^m have to be chosen so that the scheme performs well. The efficiency of this scheme examined when it is applied to the linear scalar problem $x^{\prime} = qx$, $q \in \mathbb{C}$ with rapid convergence required for all complex numbers $z = hq$ with negative real parts. When the scheme (9) is applied to this linear scalar problem, the iteration matrix is obtained in the following lemma.

Lemma 2.1. *Let the scheme (9) be applied to the scalar problem* $x^{\square} = qx$. *Then there exist a stricly lower triangular matrix* $L(z)$ *and a diagonal matrix* $\Lambda(z)$ *such that* $\Delta^{m+1} = -[L(z) + \Lambda(z)]^{-1} L(z)^H \Delta^m, \quad m = 1, 2, 3, \ldots$.

Proof. Since $D(Y) = 0$, for the scalar test problem, $D(Y^{(i)}) = D(Y^{(i)}) - D(Y) = (I_s - zA)(Y - Y^{(i)}) = (I_s - zA)\Delta^{(i)}$. Further, the scalar μ_i^m is given by

$$\mu_i^m = \frac{(E_i^m)^H (I_s - zA)^H Q^H Q D(Y^{(i)})}{(E_i^m)^H (I_s - zA)^H Q^H Q (I_s - zA) E_i^m},$$

where $E_i^m = \epsilon_i^m e_{i,}$ where ϵ_i^m is a scalar and $e_1, e_2, \ldots, e_s$ are the natural base vectors for $\mathbb{R}^s$.

$\Longrightarrow$

$$\mu_i^m = \frac{e_i^H (I_s - zA)^H Q^H Q (I_s - zA) \Delta(Y^{(i)})}{e_i^H (I_s - zA)^H Q^H Q (I_s - zA) e_i}.$$

Let $l_{ij}(z) = e_i^H (I_s - zA)^H Q^H Q (I_s - zA) e_j$. It follows from the scheme (9) that

$$\Delta^{(i+1)} = \Delta^{(i)} - \mu_i^m E_i^m, \quad i = 1, 2, 3, \ldots, s.$$

By equating the ith component of both sides of the above equation, we obtain

$$e_i^H \Delta^{(i+1)} = e_i^H \Delta^{(i)} - \mu_i^m e_i^H E_i^m, \quad i = 1, 2, 3, \ldots, s.$$

$\Longrightarrow$

$$\Delta_i^{(i+1)} = \Delta_i^{(i)} - \frac{1}{l_{ii}(z)} (e_i^H (I_s - zA)^H Q^H Q (I_s - zA))(\Delta_1^{m+1} e_1 + \Delta_2^{m+1} e_2 + \ldots \Delta_{i-1}^{m+1} e_{i-1} + \Delta_i^m e_i + \ldots + \Delta_s^m e_s) \quad \text{for } i = 1, 2, 3, \ldots, s,$$

where $\Delta^m = (\Delta_1^m, \Delta_2^m, \ldots, \Delta_s^m)^T$ and $\Delta_i^{(i+1)} = \Delta_i^{m+1}$. Since $\overline{l_{ij}} = l_{ji}$, by rearranging the above equations, we can write the above equations in matrix form as

$$[L(z) + \Lambda(z)]\Delta^{m+1} = -L^H(z)\Delta^m,$$

where $L(z) = (l_{ij}(z))$ is a strictly lower triangular matrix and $\Lambda(z) = (l_{ii}(z))$ is a diagonal matrix.

$\Longrightarrow$

$$\Delta^{m+1} = -[L(z) + \Lambda(z)]^{-1} L^H(z)\Delta^m, \quad m = 1, 2, 3, \ldots . \tag{10}$$

□

The iteration matrix $M(z)$ of the iteration (10) is given by

$$M(z) = -[L(z) + \Lambda(z)]^{-1} L^H(z).$$

Since the elements of $M(z)$ are independent of the choice of E^m, the matrix Q should be chosen to minimize the maximum of the the spectral radius $\rho[M(z)]$ of $M(z)$ over $\mathbb{C}^-$, where $\mathbb{C}^- = \{z \in \mathbb{C} : \Re(z) \leq 0\}$. This seems to be very difficult and a different approach is adopted here with the aim of forcing the spectral radius to be zero for real z. Results are obtained for two stage Gauss method.

The coefficient matrix of the two stage Gauss method of order four is given by

$$A = \begin{bmatrix} a & a-b \\ a+b & a \end{bmatrix} \quad ; \quad a = \frac{1}{4}, \quad b = \frac{\sqrt{3}}{6}. \tag{11}$$

The following theorem gives an upper bound for $\rho[M(z)]$ on $\mathbb{C}^-$ for the two stage Gauss method.

Theorem 2.2. *Consider the two stage Gauss method with coefficient matrix given by (11). Suppose that $\rho[M(z)] = 0$ on the real axis $z = x$. Then there exists a non-singular matrix Q such that $Q^H Q = \begin{bmatrix} 1 & 0 \\ 0 & \frac{b-a}{a+b} \end{bmatrix}$ and $\rho[M(z)] \leq 1 - \left(\frac{a}{b}\right)^2$ for all $z \in \mathbb{C}^-$.*

Proof. For the two stage Gauss method, it follows from the above lemma that

$$M(z) = -[L(z) + \Lambda(z)]^{-1} L^H(z),$$

where $L(z) = \begin{bmatrix} 0 & 0 \\ l_{21}(z) & 0 \end{bmatrix}$ and $\Lambda(z) = \begin{bmatrix} l_{11}(z) & 0 \\ 0 & l_{22}(z) \end{bmatrix}$.

$\Longrightarrow$

$$M(z) = \begin{bmatrix} 0 & \frac{l_{21}(z)}{l_{11}(z)} \\ 0 & -\frac{l_{21}^2(z)}{l_{11}(z) l_{22}(z)} \end{bmatrix}.$$

$\Longrightarrow$

$$\rho[M(z)] = \left| \frac{l_{21}^2(z)}{l_{11}(z) l_{22}(z)} \right|. \tag{12}$$

The elements l_{ij} are given in terms of the elements of $Q^HQ = [\alpha_{ij}]$ by,

$$\begin{aligned}
l_{11} =&(1-\bar{z}a)[\alpha_{11}(1-za) - z\alpha_{12}(a+b)] - \bar{z}(a+b)[\alpha_{21}(1-za) \\
&- z\alpha_{22}(a+b)], \\
l_{12} =&-z(a-b)[\alpha_{11}(1-\bar{z}a) - \bar{z}\alpha_{12}(a+b)] + (1-za)[\alpha_{21}(1-\bar{z}a) \\
&- \bar{z}\alpha_{22}(a+b)], \\
l_{21} =&\bar{l_{12}}, \\
l_{22} =&-\bar{z}(a-b)[-z\alpha_{11}(a-b) + \alpha_{12}(1-za)] + (1-\bar{z}a)[-z\alpha_{21}(a-b) \\
&+ \alpha_{22}(1-za)].
\end{aligned} \tag{13}$$

Since $\rho[M(z)] = 0$ on the real axis $z = x$, $l_{21} = 0$ on the real axis.

In this case, it happens that $\alpha_{12} = \alpha_{21} = 0$ and $\alpha_{11}(b-a) - \alpha_{22}(a+b) = 0$. It follows that $\alpha_{22} = \dfrac{(b-a)\alpha_{11}}{b+a}$.

Let $\alpha_{11} = 1$. Then from the equations (13), we obtain

$$\begin{aligned}
l_{11} &= z\bar{z}b^2 - (z+\bar{z})a + 1, \\
l_{12} &= (a-b)(\bar{z}-z), \\
l_{21} &= \bar{l_{12}}, \\
l_{22} &= \frac{(b-a)[b^2z\bar{z} - (z+\bar{z})a + 1]}{b+a}.
\end{aligned}$$

Since $\dfrac{l_{21}^2(z)}{l_{11}(z)l_{22}(z)}$ is analytic on $\bar{\mathbb{C}}$, its modulus, $\rho[M(z)]$, attains its maximum on the imaginary axis $z = iy$ by the maximum modulus principle. It follows from (12) that

$$\rho[M(z)] = \frac{4(b^2-a^2)y^2}{[b^2(x^2+y^2) - 2xa + 1]^2}, \text{ where } z = x + iy.$$

This gives

$$\rho[M(z)] = \frac{4(b^2-a^2)y^2}{[(b^2y^2)+1]^2} \leq \frac{(b^2-a^2)}{b^2}$$

on the imaginary axis $z = iy$.

That is

$$\rho[M(z)] \leq 1 - \left(\frac{a}{b}\right)^2$$

for all $z \in \mathbb{C}^-$.

In this case we obtain

$$Q^H Q = \begin{bmatrix} 1 & 0 \\ 0 & \dfrac{b-a}{a+b} \end{bmatrix}. \tag{14}$$

□

3. Numerical Experiments

In this section, a number of numerical experiments were carried out in order to evaluate the efficiency of the non-linear scheme proposed here. To facilitate a comparison with the scheme proposed by Cooper and Butcher [10] , results are reported for two initial value problems from Gear [16] considered by Cooper and Butcher. The comparative performance is typical of the other tests carried out.

By Problem 1, we mean the system

$$\begin{aligned} x_1' &= -0.013x_1 + 1000x_1x_3, & x_1(0) &= 1, \\ x_2' &= 2500x_2x_3, & x_2(0) &= 1, \\ x_3' &= 0.013x_1 - 1000x_1x_3 - 2500x_2x_3, & x_3(0) &= 0, \end{aligned}$$

for which eigenvalues of the Jacobian, evaluated at the initial point, are 0, -0.0093 and -3500.

By Problem 2, we mean the system

$$\begin{aligned} x_1' &= -55x_1 + 65x_2 - x_1x_3, & x_1(0) &= 1, \\ x_2' &= 0.0785(x_1 - x_2),, & x_2(0) &= 1, \\ x_3' &= 0.1x_1, & x_3(0) &= 0, \end{aligned}$$

where the eigenvalues of the Jacobian at the initial point are the complex conjugate pair $-0.0062 \pm 0.01i$ and -55.

For both problems, a single step was carried out using the s-step non-linear scheme proposed here and the scheme proposed by Cooper and Butcher [10]. In each scheme, the Jacobian evaluated at $t = 0$ was used. For each scheme tested, the initial iterate Y^0 was chosen as $Y^0 = e \otimes x$, where x is the true solution at the initial point.

By Method 1 we mean the two stage Gauss method implemented according to the scheme described by Cooper and Butcher [10, p.138], with relaxation parameter $\omega = 1$.
By Method 2 we mean the same Runge- Kutta method but implemented using the non-linear scheme (9) proposed here with the matrix Q given by (14) and E^m chosen from the scheme (5). For each proplem the quantities

$$e_m = \|Y^m - Y^{m-1}\|_\infty, \quad m = 1, 2, 3, \ldots,$$

are calculated. The values e_m for which $e_m \leq$ TOL $= 10^{-9}$ are tabulated for each problem and method. Similar results are obtained for different values of TOL.

e_m	Method 1	Method 2
e_1	0.000 785 721	0.000 706 964
e_2	0.000 056 476	0.000 035 900
e_3	0.000 004 034	0.000 001 147
e_4	0.000 000 287	0.000 000 005
e_5	0.000 000 020	0.000 000 000
e_6	0.000 000 001	

Table 1: Results for Problem 1 with $h = 0.1$

e_m	Method 1	Method 2
e_1	0.213 834 612	0.202 714 555
e_2	0.012 551 776	0.009 182 527
e_3	0.000 768 439	0.000 014 639
e_4	0.000 047 528	0.000 003 623
e_5	0.000 002 979	0.000 000 057
e_6	0.000 000 210	0.000 000 000
e_7	0.000 000 015	
e_8	0.000 000 001	

Table 2: Results for Problem 2 with $h = 1.0$

Numerical results show that the Method 2 performs better than the Method 1. That is, the proposed non-linear scheme accelerates the rate of convergence of the linear scheme proposed in [10]. Similar results may be obtained for higher

order Gauss methods and the acceleration of the rates of convergence of the other linear iteration schemes using the proposed non-linear scheme may well be possible.

References

[1] T.A.Bickart, *An efficient solution process for implicit Runge-Kutta methods*, SIAM J. Numer.Anal. **14** (1977), 1022–1027. doi: http://dx.doi.org/10.1137/0714069

[2] K. Burrage, *A special familiy of Runge-Kutta methods for solving stiff differential equations*, BIT **18**(1978), 22–41.

[3] K. Burrage, *Iteration scheme for singly implicit Runge-Kutta methods*, Computer Science Report No. **24**(1981), Uni.of Auckland.

[4] J.C.Butcher, *On the implementation of implicit Runge-Kutta methods*, BIT **16** (1976), 237–240. doi: 10.1007/BF01932265

[5] J.C.Butcher, Some implementation schemes for implicit Runge-Kutta methods, *Proceeding of the Dundee Conference on Numerical Analysis ,Lecture Notes in Mathematics, Springer–Verlag*, **773**,(1980), 12-24. doi: 10.1007/BFb0094160

[6] J.C.Butcher, K.Burrage and F.H.Chipman, *STRIDE: Stable Runge-Kutta integrator for differential equations*, Report Series No. **150**, Dept. of Mathematics, Uni. of Auckland, (1979).

[7] J.R.Cash, *On a class of implicit Runge-Kutta procedures*, J. Inst. Maths. Applics. **19** (1977), 455–470. doi: 10.1093/imamat/19.4.455

[8] F.H.Chipman, *The implementation of Runge-Kutta implicit processes*, BIT **18** (1973), 391–393. doi: 10.1007/BF01933400

[9] A.G.Collings and G.J.Tee, *An analysis of Euler and implicit Runge-Kutta numerical integration schemes for structural dynamic problems*, Proceeding of the Sixth Australasian Conference on the Mechanics of Structures and Materials, 1977, **1**, 147–154.

[10] G.J.Cooper and J.C.Butcher, *An iteration scheme for implicit Runge-Kutta methods*, IMA J. Numer. Anal. **3** (1983), 127–140. doi: 10.1093/imanum/3.2.127

[11] G.J.Cooper and R. Vignesvaran, *A scheme for the implementation of implicit Runge-Kutta methods*, Computing **45** (1990), 321–332. doi: 10.1007/BF02238800

[12] G.J.Cooper and R. Vignesvaran, *Some schemes for the implementation of implicit Runge-Kutta methods*, J. Comp. App. Math. **45** (1993), 213–225. doi: http://dx.doi.org/10.1016/0377-0427(93)90276-H

[13] G.J.Cooper and R. Vignesvaran, *On the Use of Parallel Processors for Implicit Runge-Kutta Methods*, Computing **51** (1993), 135–150.

[14] W.H. Enright, *Improving the efficiency of matrix operations in the numerical solution of ODEs*, Technical Report **98**, Computer Science Dept., Univ. of Toronto, (1976).

[15] R. Frank and C.W. Ueberhuber, *Iterated defect correction for the efficient solution of stiff systems of ordinary differential equations*, BIT **18** (1977), 146–159. doi: 10.1007/BF01932286

[16] C.W. Gear, *The automatic integration of stiff ordinary differential equations*, Proc. IFIP Congress (1968), 187–193.

[17] I. Gladwell and R.M. Thomas, *Efficiency of methods for second order problems*, IMA J. Numer. Anal. **10** (1990), 181–207. doi: 10.1093/imanum/10.2.181

[18] S. Gonzalez, C. Gonzalez and J.I. Montijano, *Iterative schemes for Gauss methods*, Comp. Math. Applic. **27** (1994), 67–81. doi: http://dx.doi.org/10.1016/0898-1221(94)90150-3

[19] S. Gonzalez, J.I. Montijano and L. Randez, *Iterative schemes for three stage implicit Runge-Kutta methods*, App. Numer. Math. **17** (1995), 363–382. doi: http://dx.doi.org/10.1016/0168-9274(95)00070-B

[20] A. S. Householder, *Theory of Matrices in Numerical Analysis*, Blaisdel Publishing Company. Adivision of Ginn and Company, New York, (1964).

[21] K.D. Peat and R.M. Thomas, *Implementation of Iteration schemes for Implicit Runge-Kutta Methods*, Numerical Analysis Report No. **169**, Dept. of Mathematics, University of Manchester, (1989).

[22] J.M. Varah, *On the efficient implementation of implicit Runge-Kutta methods*, Math. Comp. **33** (1979), 557–561.

[23] R.Vigneswaran, *Improving Rates of Convergence of Iterative Schemes for Implicit Runge-Kutta Methods*, Applied Numerical Analysis and Computational Mathematics. **1(Issue 1)** (2004), 327–338. doi: 10.1002/anac.200310029

[24] R.Vigneswaran, *Some Linear Schemes for Two-stage Gauss Type Runge-Kutta Method*, Proceeding of the Third Annual Research Session at the Eastern University, Sri Lanka (2004) 160–183.

[25] R. Vigneswaran, *Some Efficient Schemes with Improving Rate of Convergence for Two stage Gauss Method*, Journal of Mathematics, Statistics and Operational Research(JMSOR) **Vol. 2 No.1**(2013), 23–29.

[26] R. Vigneswaran and S. Kajanthan, *Some Efficient Implementation Schemes for Implicit Runge-Kutta Methods*, International Journal of Pure and Applied Mathematics (ijpam) **Vol. 93 No.4**(2014), 525–540. doi: http://dx.doi.org/10.12732/ijpam.v93i4.4