Efficient iterations for Gauss methods on second order problems.

S. González–Pinto [a], S. Pérez–Rodríguez [a] and R. Rojas–Bello [b]

[a] *Dpto. de Análisis Matemático, Universidad de La Laguna, 38271 La Laguna-Tenerife, Spain. email: spinto@ull.es*

[b] *Área de Matemáticas, Universidad Nacional Experimental de Guayana, Puerto Ordaz, Venezuela.*

Abstract

We consider some important aspects about the implementation of high order implicit formulas (specially the Gauss methods) for solving second-order differential systems having high frequencies and small amplitudes superimposed. The choice of an appropriate iterative scheme is discussed in detail. Important topics about the predictors (initial guesses) are analyzed and a variable order strategy to select the best predictor at each integration step is supplied. A few numerical experiments on some standard test problems confirm the theory presented.

Keywords: Second Order Initial Value Problems, Runge-Kutta Nyström Methods, Gauss Methods, Iterative Schemes, Initial Guesses.

AMS classification: 65L05.

1 Introduction

We consider the numerical solution of oscillatory problems in the possible presence of stiffness for second order differential systems of special type

$$y''(t) = f(t, y(t)), \quad y(t_0) = y_0,\ y'(t_0) = y'_0, \quad t \in [t_0, t_{end}],\ y, y', f \in \mathbb{R}^m. \tag{1.1}$$

The stiffness in this case means that the solution $y(t)$ combines components with dominant short frequencies and components with large frequencies and small amplitudes. This phenomenon

[1] Work supported by projects BMF2001-2562, MTM2004-06466-C02-02 and PI 2003/142 (Gobierno de Canarias grant).

appears in many practical problems of type (1.1).

Most numerical methods proposed to integrate (1.1) can be arranged in three groups, the linear multistep methods (LM), the direct hybrid methods and the one step methods. The linear multistep approach suffers the inconvenience of having the convergence order limited to two if P-stability is required [17]. To overcome the barrier of P-stability Cash [1,2], Thomas [20], Chawla et al. [4,5] have considered hybrids variants of multistep methods, by introducing additional off-step points in the LM formula. P-stable hybrid methods of orders 4,5,...,8, have been derived by these authors. Inside the class of Runge-Kutta-Nyström (RKN) methods it is possible to get P-stable formulas of arbitrarily large order [10]. Van der Houwen and Sommeijer [15], Sharp et al. [19] and Franco et al. [6] among others, have obtained high order P-stable Diagonally Implicit Runge-Kutta-Nyström methods (DIRKN) with high orders in the phase.

The attractiveness of DIRKN methods comes from the special structure of their coefficient matrix, which allows to reduce the algebraic cost when solving their internal stages by some Newton-type iteration. However, for getting the same accuracy in terms of convergence order DIRKN methods require more internal stages than other P-stable highly implicit RKN methods, such as those derived from the Gauss family (or Lobatto IIIA family) [13, p.72-75] (the RKN version). For that reason, the search of a cheap implementation for these highly implicit RKN methods is an interesting alternative and possibly more efficient than the DIRKN approach.

In this paper we pursue several objectives addressed to get an efficient implementation of highly implicit RKN methods. We will be mainly concerned with the Runge-Kutta Gauss family, but the results can be extended to other highly implicit methods. Our first goal will be to analyze the usual Simplified Newton Iteration (SNI) in some detail. Next, an alternative iteration which reduces the computational effort with regard to SNI will be developed (sections 2 and 3). Initial guesses of several orders, with no large amplification factors for the accumulated errors at each integration step, and a criterion to select the best one at each step will be supplied in section 4. A few numerical experiments comparing several iterations will be presented in section 5.

2 Alternatives to the Simplified Newton Iteration for RKN methods

An s-stage Runge-Kutta-Nyström method RKN $(\bar{A}, c, b, \bar{b})$ advances the numerical solution from (t_n, y_n, y'_n) to $(t_{n+1} = t_n + h, y_{n+1}, y'_{n+1})$ by means of the formula

$$y_{n+1} = y_n + hy'_n + h^2(\bar{b}^T \otimes I)F(Y_n), \quad y'_{n+1} = y'_n + h(b^T \otimes I)F(Y_n), \tag{2.1}$$

where the internal stages $Y_n^T = (Y_{n,1}^T, \ldots, Y_{n,s}^T) \in \mathbb{R}^{ms}$ are computed from

$$Y_n = e \otimes y_n + h(c \otimes y'_n) + h^2(\bar{A} \otimes I)F(Y_n), \tag{2.2}$$

with $F^T(Y_n) := (f^T(t_n + c_1 h, Y_{n,1}), \ldots, f^T(t_n + c_s h, Y_{n,s})) \in \mathbb{R}^{ms}$, $e = (1, \ldots, 1)^T \in \mathbb{R}^s$, $\otimes$ standing for the standard Kronecker product of matrices ($A \otimes B = (a_{ij}B)$) and I for the

identity matrix of appropriate dimension. The matrix $\bar{A} = (\bar{a}_{i,j})_{i,j=1,s}$ and the vectors $c^T = (c_j)_{j=1,s}$, $b^T = (b_j)_{j=1,s}$ and $\bar{b}^T = (\bar{b}_j)_{j=1,s}$ represent the coefficients of the underlying RKN method. Throughout the remainder of the paper it will be assumed that

$$\bar{A} = A^2, \qquad \bar{b}^T = b^T A, \quad c = Ae, \quad \det A \neq 0,$$

where the matrix-vector coefficients (A, b) are those ones of some Runge-Kutta method aimed to integrate first order differential systems. That is the case for Gauss methods.

The usual Simplified Newton Iteration computes the internal stages by the formula

$$(I - h^2(A^2 \otimes J_n))(Y_n^{(\nu)} - Y_n^{(\nu-1)}) = D(Y_n^{(\nu-1)}, y_n, hy'_n), \quad \nu = 1, 2, \ldots, \tag{2.3}$$

for a given initial guess $Y_n^{(0)}$, where $J_n \simeq \dfrac{\partial f}{\partial y}(t_n, y_n)$ and the residual mapping $D(\cdot)$ is defined by,

$$D(Z, u, v) := -Z + e \otimes u + (c \otimes v) + h^2(A^2 \otimes I)F(Z), \quad u, v \in \mathbb{R}^m, \ Z \in \mathbb{R}^{ms}. \tag{2.4}$$

For the s-stage Gauss method the algebraic cost of the iteration (2.3) can be reduced, by decoupling the ms real linear systems into $[s/2]$ complex linear systems (here $[x]$ denotes the integer part of the real number x) of dimension m plus one real linear system of dimension m when s is odd. These linear systems are usually solved by making LU factorizations and reducing each linear system to two triangular systems. In this way the same LU can be used over all the iterates at every integration step and also on several consecutive integration steps when the convergence is fast and the step-size is unchanged. For more details on this technique when applied to the two-stage Gauss method, see Gladwell and Thomas [7, p.187-190].

Once the convergence in (2.3) is reached, namely after μ iterations, then we might replace $Y_n \simeq Y_n^{(\mu)}$, and the advancing solution can be computed by using (2.1). This way presents two major inconveniences, firstly s extra derivative evaluations are required, i.e. the updating of the super-vector $F(Y_n^{(\mu)})$, and secondly and most important the difference $F(Y_n) - F(Y_n^{(\mu)})$, can be large in the presence of stiffness. To avoid both drawbacks we use the following alternative formula

$$\begin{aligned} y_{n+1} &= r^* y_n + (b^T A^{-1} \otimes I)Y_n, \quad r^* = 1 - b^T A^{-1} e, \ r' = -b^T A^{-2} e, \\ v_{n+1} &= r' y_n + r^* v_n + (b^T A^{-2} \otimes I)Y_n, \quad v_n := hy'_n, \ v_{n+1} := hy'_{n+1}. \end{aligned} \tag{2.5}$$

It is interesting to notice that for nonlinear differential systems in general, the error $Y_n - Y_n^{(\nu)}$ behaves as

$$Y_n - Y_n^{(\nu)} = \mathcal{O}(h^3)(Y_n - Y_n^{(\nu-1)}), \quad h \to 0, \quad \nu = 1, 2, \ldots, \tag{2.6}$$

provided that

$$J_n - \frac{\partial f}{\partial y}(t_n, y_n) = \mathcal{O}(h) \quad \text{and} \quad Y_n - Y_n^{(0)} = \mathcal{O}(h), \quad \text{when } h \to 0. \tag{2.7}$$

This follows from (2.3) and (2.4), by taking into account that
$D(Y_n^{(\nu-1)}, y_n, v_n) - D(Y_n, y_n, v_n) = (-I + h^2(A^2 \otimes J_n) + \mathcal{O}(h^3))(Y_n^{(\nu-1)} - Y_n), \quad \nu = 1, 2, \ldots.$
Now, from (2.6) we get that

$$Y_n - Y_n^{(\nu)} = h^{3\nu}(C_\nu + \mathcal{O}(h))(Y_n - Y_n^{(0)}), \quad \nu = 0, 1, \ldots, \quad (h \to 0), \tag{2.8}$$

where C_ν is a moderately sized constant matrix which may depend on ν.
As remarked by Gladwell and Thomas [7] and the references therein, the Simplified Newton Iteration presents a few disadvantages which had led several authors to look for better alternatives [3,7,18]. The main drawback of (2.3) comes from the fact that it requires the usage of complex arithmetic in order to get an efficient implementation for highly implicit methods such as those of the Gauss family ($s \geq 2$ stages). The usage of complex arithmetic implies that the operational costs (LU-factorizations, solution of triangular systems, multiplication of matrix by vector) are four times more expensive than their real arithmetic counterparts. On the other hand, with regard to the storage and the matrix updating, it must be observed that the Simplified Newton Iteration involves $[s/2]$ complex LU-factorizations and $2(s/2 - [s/2])$ real LU-factorizations per integration step. It is also worthwhile to mention the conclusions of Gladwell and Thomas [7, p.205], where for the two stage Gauss method, they "tentatively" recommend the scheme proposed by Cooper and Butcher [3] rather than the Simplified Newton Iteration. It is also recognized in [8] that the Cooper and Butcher iteration is a good alternative for first order stiff problems. However, this iteration [3] when applied to (1.1) presents a few inconveniences. Firstly, it was designed for first order problems, thus it requires a lot of intermediate matrix by vector transformations when it is applied to second order problems, see [7, p.192-193] for the case of the two-stage Gauss method. Secondly, and this is even more important, the order in h is not increased during the iteration process, although the error coefficients might decrease. Hence, independently of the number of iterations per step, we will end up with a low order method caused by the order of the predictor, which is usually rather low.
To overcome the drawbacks of the iteration [3] on second order problems, we propose a direct iteration of Newton-type for solving the stage equations (2.2), which reminds of the perfect square iteration [18] and it can also be considered as the second order version of the iteration in [8, see (1.8)], which only applies to first order problems. The new iteration read as,

$$(I - h^2(T \otimes J_n))(Y_n^{(\nu)} - Y_n^{(\nu-1)}) = D(Y_n^{(\nu-1)}, y_n, hy_n'), \quad \nu = 1, 2, \ldots, \tag{2.9}$$

where $J_n \simeq \partial f/\partial y(t_n, y_n)$, the residual $D(\cdot)$ is defined in (2.4) and $T \in \mathbb{R}^{s\times s}$ is a constant matrix with an one-point spectrum $\sigma(T) = \{\gamma\}$, $\gamma > 0$. This matrix will be properly optimized

for some particular methods, taking into account some considerations given later. Since the matrix T has a single spectrum we can set

$$T = \gamma S(I - L)^{-1}S^{-1}, \tag{2.10}$$

where L is a strictly lower triangular matrix and S is a nonsingular matrix. In this way, it is not very difficult to see that the iteration (2.9) can be rewritten as

$$\begin{array}{c}(\xi I - (I \otimes J_n))\Delta^{(\nu)} = (\xi P \otimes I)D(Y_n^{(\nu-1)}, y_n, hy_n') + (\xi L \otimes I)\Delta^{(\nu)}, \quad \nu = 1, 2, \ldots, \\ Y^{(\nu)} = Y^{(\nu-1)} + (S \otimes I)\Delta^{(\nu)}, \quad \text{with } \xi = (\gamma h^2)^{-1} \text{ and } P = (I - L)S^{-1}.\end{array} \tag{2.11}$$

For computational purposes the formulation (2.11) is preferred to (2.9) since the former allows to decouple each ms-dimensional linear system into s systems of dimension m. Moreover, only one real LU factorization (of dimension m) is needed in the case of (2.11). However, for analysis purposes we will work with the formulation (2.9). It is also important to remark that when numerical convergence is achieved in (2.11), namely after μ iterations, we replace $Y_n \simeq Y_n^{(\mu)}$ and compute the advancing solution by formula (2.5), which is more stable as said above.

The error of the iterates in (2.9) for differential systems satisfies the recursion

$$Y_n - Y_n^{(\nu)} = \mathcal{O}(h^2)(Y_n - Y_n^{(\nu-1)}), \quad h \to 0, \quad \nu = 1, 2, \ldots, \tag{2.12}$$

provided that (2.7) is fulfilled. This implies that

$$Y_n - Y_n^{(\nu)} = h^{2\nu}(C_\nu^* + \mathcal{O}(h))(Y_n - Y_n^{(0)}), \quad h \to 0, \quad \nu = 0, 1, \ldots, \tag{2.13}$$

where C_ν^* is a constant matrix, possibly depending on ν. The $\mathcal{O}(h)$-term in (2.13) can be large for stiff problems.

In order to get the same accuracy for the stages when comparing (2.3)-complex-version and (2.11) we get, by virtue of (2.8) and (2.13), that $\nu_T \simeq \frac{3}{2}\nu_{\bar{A}}$. Here, ν_T and $\nu_{\bar{A}}$ denote the number of iterations with (2.11) and (2.3), respectively. For the s-stage Gauss method ($s = 2, 4, 6, \ldots$) we compare in Table 2.1 the computational cost of both iterations for achieving the same accuracy. There, a derivative evaluation accounts $2m^2$ flops in the same way as accounts the solution of two real triangular systems in LU form. *Iter* denotes the number of iterations given, LS the number of m-dimensional linear systems, fn the number of derivatives and it is assumed that $Y_n - Y_n^{(0)} = \mathcal{O}(h^q)$. From this table it is clear that both iteration processes reach the same accuracy (in terms of power of h) at the same costs. However, it should be mentioned that the costs for the LU-factorizations have not been taken into account in Table 2.1. Since the factorizations substantially contribute to the total costs of the methods (proportional to the third power of the ODE dimension) we remark that (2.3) requires $s/2$ *complex* LU-decompositions (resulting in $4sm^3/3$ flops), whereas (2.11) only needs the LU-factorization

Table 2.1
Costs with Simplified Newton Iteration and (2.11) for the s-stage Gauss method (s even).

Iterative scheme	$Iter$	LS	fn	flops	Accuracy
Simplified Newton Iter.	ν	$s\nu/2$ (Complex)	$s\nu$	$6s\nu m^2$	$\mathcal{O}(h^{3\nu+q})$
(2.11)	$3\nu/2$	$3s\nu/2$ (Real)	$3s\nu/2$	$6s\nu m^2$	$\mathcal{O}(h^{3\nu+q})$

of one real matrix ($2m^3/3$ flops). It is to be expected that the iteration process (2.11) is more efficient than the Simplified Newton Iteration, especially for medium to large dimensional problems.

2.1 Global error analysis when μ iterations per integration step are given

An interesting issue is the analysis of the global errors when comparing both iterations ((2.3) and (2.9)) for a fixed number of iterations μ per integration step. To be more precise, we will be interested in the size of

$$\varepsilon_n := y_n - y_n{}^{(\mu)},\ \tau_n := h(y'_n - y'_n{}^{(\mu)}), \quad n = 1, 2, \ldots, N, \tag{2.14}$$

where (y_n, y'_n) denotes the exact Runge-Kutta solution after n consecutive steps of size $h = (t_{end} - t_0)/N$ and $(y_n^{(\mu)}, y'_n{}^{(\mu)})$ stands for the numerical solution after n consecutive steps and μ iterations per integration step. It will also be assumed that the same predictor is used on each integration step to start the iterates, except possibly at the first step.

We need some extra notations to make the analysis. Thus, by considering we have selected the iteration (2.9), $Y_n^{(0)}$ represents the initial guess for the stages in order to advance from $(t_n, y_n^{(\mu)}, y'_n{}^{(\mu)})$ to $(t_{n+1}, y_{n+1}^{(\mu)}, y'_{n+1}{}^{(\mu)})$. $Y_n^{(\nu)}$ $(\nu = 1, \ldots, \mu)$ denotes the ν-iterate, $Y_n^{(\infty)}$ stands for the exact solution of the stage equations $D(Y, y_n^{(\mu)}, hy'_n{}^{(\mu)}) = 0$ and Y_n represents the exact Runge-Kutta stages after n integration steps. It will be assumed that the initial guess is of order q, i.e.

$$Y_n^{(\infty)} - Y_n^{(0)} = (K(t_n) + \mathcal{O}(h))h^q. \tag{2.15}$$

Here $Y_n^{(0)}$ denotes an initial guess based in the information obtained at the previous step $(y_{n-1}^{(\mu)}, y'_{n-1}{}^{(\mu)}, Y_{n-1}^{(\mu)})$ and $K(t) = \mathcal{O}(1)$ is supposed to be a "smooth" t-depending mapping (ms vector). For instance, the natural starting algorithm, $Y_n^{(0)} := e \otimes y_n^{(\mu)} + h(c \otimes y'_n{}^{(\mu)})$, is a second order initial guess, since we have that, $Y_n^{(\infty)} - Y_n^{(0)} = ((A^2 \otimes I)F(e \otimes y_n^{(\mu)}) + \mathcal{O}(h))h^2$, where $y_n^{(\mu)} = r^* y_{n-1}^{(\mu)} + (b^T A^{-1} \otimes I)Y_{n-1}^{(\mu)}$, according to (2.5).

With the notations above, from (2.14) and (2.5) we deduce that,

$$\begin{aligned}\varepsilon_{n+1} &= r^*\varepsilon_n + (b^TA^{-1}\otimes I)(Y_n - Y_n^{(\mu)})\\ \tau_{n+1} &= r'\varepsilon_n + r^*\tau_n + (b^TA^{-2}\otimes I)(Y_n - Y_n^{(\mu)}).\end{aligned}\qquad n=0,1,\ldots \tag{2.16}$$

Now, by splitting, $Y_n - Y_n^{(\mu)} = (Y_n - Y_n^{(\infty)}) + (Y_n^{(\infty)} - Y_n^{(\mu)})$, by virtue of $D(Y_n, y_n, hy'_n) = D(Y_n^{(\infty)}, y_n^{(\mu)}, hy_n'^{(\mu)}) = 0$, it follows after a direct calculation, assuming f suitably smooth, that

$$Y_n - Y_n^{(\infty)} = (I + h^2\mathcal{O}(1))(e\otimes\varepsilon_n + c\otimes\tau_n). \tag{2.17}$$

By setting $E^{(\nu)} := Y_n^{(\infty)} - Y_n^{(\nu)}$, from (2.9) we get that
$(I - h^2(T\otimes J_n))(-E^{(\nu)} + E^{(\nu-1)}) = D(Y_n^{(\nu-1)}, y_n^{(\mu)}, hy_n'^{(\mu)}) - D(Y_n^{(\infty)}, y_n^{(\mu)}, hy_n'^{(\mu)}) =$
$$(-I + h^2(A^2\otimes J_n) + \mathcal{O}(h^3))(-E^{(\nu-1)}),\quad \nu = 1,2,\ldots,$$
provided that

$$J_n = \frac{\partial f}{\partial y}(t_n, y_n^{(\mu)}) + \mathcal{O}(h).$$

From here and taking into account (2.15) we arrive at

$$\begin{aligned}E^{(\mu)} &= h^{2\mu}\left((A^2-T)^\mu\otimes(J_n)^\mu + \mathcal{O}(h)\right)E^{(0)} =\\ &h^{2\mu+q}\Big(((A^2-T)^\mu\otimes(J_n)^\mu)K(t_n) + \mathcal{O}(h)\Big).\end{aligned} \tag{2.18}$$

From (2.16) and bearing in mind (2.5), (2.17) and (2.18), we arrive at

$$\begin{aligned}\varepsilon_{n+1} &= (1+\mathcal{O}(h^2))\varepsilon_n + (1+\mathcal{O}(h^2))\tau_n + h^{2\mu+q}\left((b^TA^{-1}(A^2-T)^\mu\otimes(J_n)^\mu)K(t_n) + \mathcal{O}(h)\right),\\ \tau_{n+1} &= (1+\mathcal{O}(h^2))\tau_n + \mathcal{O}(h^2)\varepsilon_n + h^{2\mu+q}\left((b^TA^{-2}(A^2-T)^\mu\otimes(J_n)^\mu)K(t_n) + \mathcal{O}(h)\right),\\ &n = 0,1,2\ldots,\qquad \text{for each } \mu = 1,2,\ldots\end{aligned} \tag{2.19}$$

From (2.19) it can be seen that

$$\tau_n = h^{2\mu+q-1}\left((b^TA^{-2}(A^2-T)^\mu\otimes I)M(t_n) + \mathcal{O}(h)\right),\qquad n = 0,1\ldots, \tag{2.20}$$

where $M(t_n) = h\sum_{j=0}^{n-1}(I\otimes(J_j)^\mu)K(t_j) = \mathcal{O}(1)$. Then, by requiring

$$0^T = b^TA^{-2}(A^2-T) = b^T(I - A^{-2}T), \tag{2.21}$$

we achieve, for $\mu\geq 1$, that

$$\varepsilon_n = y_n - y_n^{(\mu)} = \mathcal{O}(h^{2\mu+q-1}),\quad h^{-1}\tau_n = y'_n - y_n'^{(\mu)} = \mathcal{O}(h^{2\mu+q-1}),\ n = 0,1,\ldots,N. \tag{2.22}$$

Theorem 1 *The global errors associated with (2.5) at the end point after N integration steps, where μ iterates per step are carried out with (2.9) satisfying (2.21) and a fixed initial guess of order q (see (2.15)), satisfy the relation (2.22) for $h = (t_{end} - t_0)/N \to 0^+$.*

If (2.21) is violated, then only order $2\mu + q - 2$ can be guaranteed for the advancing solution. On the other hand, a similar analysis to the previous one about the global convergence order after μ iterations (per integration step) for the Simplified Newton Iteration (2.3) shows that for nonlinear problems in general we get that,

Theorem 2 *Under the assumptions of theorem 1, for the scheme (2.3)-(2.5) we have that,*

$$\begin{aligned} y_n - y_n^{(\mu)} &= \mathcal{O}(h^{3\mu+q-2}) \\ y'_n - y'^{(\mu)}_n &= \mathcal{O}(h^{3\mu+q-2}) \end{aligned} \qquad n = 0, 1, \ldots, N, \quad (h \to 0^+). \tag{2.23}$$

This implies that for $\mu = 1$, both iteration processes ((2.3) and (2.11)) give the same accuracy for the advancing solution (order $q+1$), but our iteration is quite cheaper. However, in practice when integrating with RK-Nyström based codes, more than one iteration is currently performed in most of the integration steps. In that case, theoretical comparisons among both iterative schemes are shown in Table 2.1 for the s-stage stage Gauss methods (s even). From the results in Table 2.1 and from the Theorems 1 and 2, it is not difficult to deduce that at the same computational effort (without considering LU-factorizations), our iteration gains one order more on the advancing solution than the simplified Newton Iteration. Observe that our iteration has the additional advantage of requiring only one real LU-decomposition independently of the stage's number of the method. In section 5, the statements of the theorems 1 and 2 will be numerically illustrated.

3 By selecting the iteration for the Gauss methods

We are interested in iterations of type (2.9) that can cope satisfactorily with general stiff nonlinear differential systems. Hence, these iterations should be also convergent for stiff linear problems of the form, $y''(t) = Jy, \quad J \in \mathbb{R}^{m,m}$, independently of the size of the eigenvalues of J whenever they are non-positive. Thus, the iteration must be convergent on the linear test equation

$$y''(t) = -\lambda^2 y, \qquad \lambda \in \mathbb{R}. \tag{3.1}$$

Usually, the frequencies $\{\lambda\}$ involved with the original linear system, have small amplitudes when they are large and medium amplitudes when they are short (dominant frequencies). Thus, it is vital for advancing the integration with no too small step-sizes, that the iteration damps those large frequencies and that it possesses a short rate of convergence on the smaller

(dominant) frequencies. In this way, a low number of iterations μ with (2.9) will allow to get a very stable and approximate advancing solution $(y_n^{(\mu)}, y_n'^{(\mu)})$. By setting $z = h\lambda$ and applying the iteration (2.9) with $J_n = -\lambda^2$, to the test (3.1), we get for the error in the iterates that

$$Y_n - Y_n^{(\nu)} = N(z)(Y_n - Y_n^{(\nu-1)}) = N(z)^\nu (Y_n - Y_n^{(0)}), \quad \nu = 1, 2 \ldots, \tag{3.2}$$

where $N(z) := z^2(I + z^2T)^{-1}(T - A^2)$. It is clear that the convergence of (2.9) for all z is equivalent to require that the spectral radius of $N(z)$ satisfies $\rho(N(z)) < 1$, $\forall z \in \mathbb{R}$.

In the construction of our iterations for the Gauss methods the condition (2.21) will be always demanded. This implies that one eigenvalue of $N(z)$ vanishes. Moreover, as said before, the matrix T must have an one-point spectrum, $\sigma(T) = \{\gamma\}$, $\gamma > 0$.

The two stage Gauss method ($s = 2$).

From the requirements above, the other eigenvalue of $N(z)$ is given by

$$\phi(z) = 1 - \det\left(I - N(z)\right) = 1 - \left(\det(I + z^2T)\right)^{-1} \det(I + z^2A^2). \tag{3.3}$$

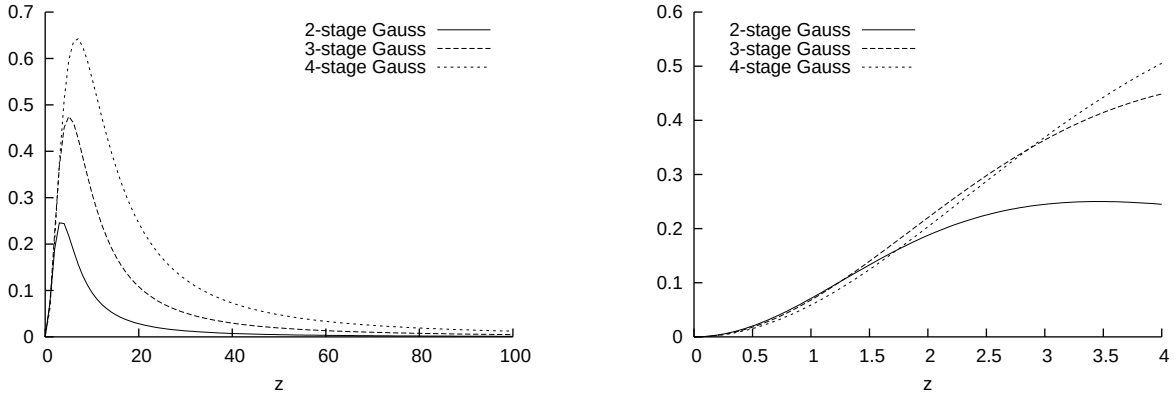
A straightforward calculation shows that (below, $\mathrm{tr}\, A^2$ denotes the trace of the matrix A^2),

$$\phi(z) = \frac{z^2(2\gamma - \mathrm{tr}A^2) + z^4(\gamma^2 - (\det A)^2)}{(1 + \gamma z^2)^2}.$$

To increase the order at the origin $z \to 0$, we must demand $\mathrm{tr}A^2 - 2\gamma = 0$, but in that case we have that $\phi(\infty) = -3$ and the stiff frequencies are not damped. In this way a compromise between some damping on the whole real line (and specially at $z = \infty$, because of the high frequencies) and a small value for $\delta := \mathrm{tr}A^2 - 2\gamma$ should be searched. One option might be asking for $\phi(\infty) = 0$. This implies that $\gamma = \det A = 1/12$, $\delta = -1/12$ and that $\phi_1 := \max_{z\in\mathbb{R}} |\phi(z)| = 0.25$. Another option could be choosing γ so that $g(\gamma) := \max_{z\in\mathbb{R}} |\phi(z)|$ is minimum. In that case, we have that $\gamma^* = 0.07644...$ and $\phi_2 := g(\gamma^*) = 0.18...$ The maximum spectral radius argument is favorable to the second option, but in that case we have that $\phi(\infty) = -\phi_2$, and in most part of the real line the spectral radius for the second case is larger than for the first case, also the difference $\phi_1 - \phi_2$ is relatively small. Moreover, for a fixed number of iterations, the global errors (associated with high frequencies on linear problems) accumulated at the previous step t_n are more amplified when considering the second option. This fact has some importance in the choice of high order predictors as it will be seen in the next section. These arguments and practical reasons comparing both iterations on many test problems seem to indicate that the first option performs at least as well as the second one in most problems and the first one has the additional advantage of being more robust. In conclusion, we take $\gamma = 1/12$, this gives a unique matrix T whose inverse matrix T^{-1} satisfies

$$b^T(A^{-2} - T^{-1}) = 0^T, \quad \mathrm{tr}(T^{-1}) = 2\gamma^{-1} = 24, \quad \det(T^{-1}) = \det A^{-2} = 12^2.$$

Fig. 3.1. Spectral radius of $N(z)$ for the s-stage Gauss methods ($s = 2, 3, 4$).



The matrix T can be decomposed according to (2.10), and although such a decomposition is not unique, we have taken that one requiring that S is an upper triangular matrix with "1" on its diagonal (for computational savings). Thus, L and S are respectively, the strictly lower triangular part and the upper triangular part of the next matrix denoted by $L \boxplus S$,

$$L \boxplus S = \begin{pmatrix} 1 & -7 + 4\sqrt{3} \\ (12 + 7\sqrt{3})/6 & 1 \end{pmatrix}, \quad \gamma = 1/12. \tag{3.4}$$

The three stage Gauss method ($s = 3$).

In this case, the matrix T provides nine unknowns. The condition (2.21) involves three linear equations and also implies a null eigenvalue for $N(z)$. On the other hand, the requirement $\sigma(T) = \{\gamma\}$ imposes three equations (one linear one and two nonlinear ones) and supplies a new parameter γ. The approach we have followed consists of demanding that the matrix $N(z)$ has just one non-vanishing eigenvalue, $\phi(z)$. In that case $\phi(z)$ is given by (3.3).

As in the two-stage case, by requiring $\phi(\infty) = 0$, we get that $\gamma = (\det A)^{2/3} = (1/120)^{2/3} = 0.041103534...$ Moreover, the condition (2.21) is equivalent to $0^T = b^T N(\infty)$, and the latter is equivalent to $0^T = b^T(A^{-2} - T^{-1})$. Thus, by requiring the new condition $0^T = e_3^T N(\infty)$, $e_3^T = (0, 0, 1)$, which is equivalent to $0^T = e_3^T(A^{-2} - T^{-1})$, it follows that $N(z)$ has a null eigenvalue of multiplicity two (by virtue of that $A^2 - T$ has two left eigenvectors corresponding to the null eigenvalue) and the other eigenvalue is given by $\phi(z)$ in (3.3). This choice for T^{-1} has the advantage that its last row coincides with that one of A^{-2}, and this fact implies small errors for the iterates for the third component $Y_{n,3}^{(\nu)}$ ($\nu = 1, 2, \ldots$), mainly when high frequencies are considered ($z \to \infty$). It must be also observed that $Y_{n,3}$ is the farthest stage to be approached with initial guesses from the previous step, thus a null damping for the iterate errors on this stage is important. With these requirements (nine equations and nine unknowns) the matrix T is univocally determined. Again, we omit writing T and we give its $L \boxplus S$-decomposition with

16 significant figures,

$$L \boxplus S = \begin{pmatrix} 1 & -0.34134805819933375 & 0.08060287745941966 \\ 3.0972763877611617 & 1 & 0.09100037186032114 \\ -6.381775393482425 & 4.3129085842580614 & 1 \end{pmatrix} \quad (3.5)$$

The four stage Gauss method ($s = 4$).

In this case and by giving similar arguments as those ones for the cases $s = 2, 3$, we require the following 16 equations for the matrix T^{-1} (16 unknowns).

$$b^T(A^{-2} - T^{-1}) = 0^T,\ \sigma(T^{-1}) = \{\gamma^{-1}\},\ e_3^T(A^{-2} - T^{-1}) = 0^T,\ e_4^T(A^{-2} - T^{-1}) = 0^T. \quad (3.6)$$

Here, e_3 and e_4 denote the third and fourth vectors of the canonical basis of $\mathbb{R}^4$ respectively and the value for γ is obtained from demanding that the non-vanishing eigenvalue of $N(z)$ satisfies, $\phi(\infty) = 0$. This implies that $\gamma = (\det A)^{1/2} = \dfrac{\sqrt{105}}{420} = 0.0243975018....$ From the requirements in (3.6) a unique real matrix T is obtained. This matrix can be also decomposed in the $L \boxplus S$-form mentioned before as

$$\begin{pmatrix} 1 & -0.6259618003648055 & 0.28004508015579187 & -0.026021682011884171 \\ 2.8903693478178977 & 1 & 0.04262904477976469 & 0.04746745319522649 \\ -3.880435892734769 & 3.1457588737925907 & 1 & 0.16883422865076189 \\ 10.056003837018828 & -10.386602189612067 & 5.349223474607041 & 1 \end{pmatrix}$$

In Figure 3.1, the plot of the non-vanishing eigenvalue of $N(z)$ ($\phi(z)$ in (3.3)) is displayed for each s-stage Gauss method with $s = 2, 3, 4$. It can be appreciated that the iteration is always convergent and that the speed of convergence slightly decreases when the number of stages s is increased. It is also interesting to consider the averaged rate of convergence for a given iteration defined by van der Houwen and de Swart [16, p.45] as, $\rho_j(z) := \sqrt[j]{\| N(z)^j \|_\infty},\ j = 1, 2, \ldots$. For our iteration (2.11), we have that $\lim_{j\to\infty} \rho_j(z) = \rho(N(z)) = |\phi(z)|$. In [16] it is remarked the importance of having $\rho_j(z) \simeq \rho(N(z))$ for small values of j, since this would imply an optimal convergence rate (closed to the spectral radius of $N(z)$) from the first iterates. In Table 3.1 we have picked up the numbers $\rho_j^* = \max_{z\in\mathbb{R}} \rho_j(z)$ and $\rho_j' \equiv \rho_j(\infty)$ for the cases $j = 1, 2, 3, 4$, and $s = 2, 3, 4$.

Table 3.1
Averaged rates of convergence for the Gauss methods.

j	2-stage Gauss	3-stage Gauss	4-stage Gauss
1	$\rho_j^* = 0.577;\ \rho_j' = 0.577$	$\rho_j^* = 1.72;\ \rho_j' = 1.72$	$\rho_j^* = 3.72;\ \rho_j' = 3.72$
2	$\rho_j^* = 0.280;\ \rho_j' = 0$	$\rho_j^* = 0.830;\ \rho_j' = 0$	$\rho_j^* = 1.17;\ \rho_j' = 0$
3	$\rho_j^* = 0.261;\ \rho_j' = 0$	$\rho_j^* = 0.675;\ \rho_j' = 0$	$\rho_j^* = 0.881;\ \rho_j' = 0$
4	$\rho_j^* = 0.256;\ \rho_j' = 0$	$\rho_j^* = 0.613;\ \rho_j' = 0$	$\rho_j^* = 0.803;\ \rho_j' = 0$

4 Initial guesses for the s-stage Gauss methods ($s \geq 2$)

The order conditions to achieve order q for predictors (denoted by $(Y_{n,i}^{(0)})$),

$$Y_{n,i} - Y_{n,i}^{(0)} = \mathcal{O}(h^q), \quad i = 1, \ldots, s, \qquad h \to 0,$$

where $Y_{n,i}^{(0)}$ makes use of the information from the previous step $t_{n-1} \xrightarrow{h} t_n$ (the next step is $t_n \to t_{n+1} = t_n + \tau h$), and having the form

$$Y_{n,i}^{(0)} = a_i y_{n-1} + h d_i y'_{n-1} + \sum_{j=1}^{s} b_{ij} Y_{n-1,j}, \quad i = 1, \ldots, s \tag{4.1}$$

can be deduced by using the *special Nyström tree theory* (see e.g. [12, Ch. II.14]). In this case, the order conditions are given by the following linear equations:
Order 1 iff $a_i + \sum_{j=1}^{s} b_{ij} = 1, \quad i = 1, \ldots, s.$
Order 2 iff (Order 1 and $d_i + \sum_{j=1}^{s} b_{ij} c_j = 1 + \tau c_i, \quad i = 1, \ldots, s$).
Next, we use the $C(s)$-simplifying order condition, which is fulfilled for the s-stage Gauss method (below, power on a vector means power on each component),

$$C(s): \qquad A c^{j-1} = \frac{1}{j} c^j, \quad j = 1, \ldots, s.$$

Order 3 (for $s \geq 2$) iff (Order 2 and $\sum_{j=1}^{s} b_{ij} (c_j)^2 = (1 + \tau c_i)^2, \quad i = 1, \ldots, s$).
Order 4 (for $s \geq 2$) iff (Order 3 and

$$\sum_{j=1}^{s} b_{ij} \kappa_j = \frac{1}{6} \left((1 + \tau c_i)^3 - (\tau c_i)^3 \right) + \kappa_i \tau^3,\ i = 1, \ldots, s, \quad \text{where } \kappa = (\kappa_i) = A^2 c).$$

For $s \geq 3$ stages, the latter condition is equivalent to

$$\sum_{j=1}^{s} b_{ij} (c_j)^3 = (1 + \tau c_i)^3, \quad i = 1, \ldots, s.$$

Order 5 (for $s \geq 3$ stages) iff (Order 4 and

$$\sum_{j=1}^{s} b_{ij}\zeta_j = \frac{1}{12}\left((1+\tau c_i)^4 - (\tau c_i)^4\right) + \zeta_i \tau^4,\ i = 1, \ldots, s, \quad \text{where } \zeta = (\zeta_i) = A^2c^2).$$

It is very interesting to analyze the effect of amplification of errors through the starting algorithms when high frequencies are considered. Thus, for linear problems $y''(t) = Jy + g(t)$, by setting $\epsilon_{n-1} := y_{n-1} - \hat{y}_{n-1}$, $\quad \omega_{n-1} := hy'_{n-1} - h\hat{y}'_{n-1}$, and denoting respectively by $y(t)$ and $\hat{y}(t)$ two nearby local solutions passing through $(t_{n-1}, y_{n-1}, y'_{n-1})$ and $(t_{n-1}, \hat{y}_{n-1}, \hat{y}'_{n-1})$, it follows that its difference $\xi(t) = y(t) - \hat{y}(t)$ satisfies the linear system

$$\xi''(t) = J\xi(t), \quad \xi(t_{n-1}) = \epsilon_{n-1}, \quad \xi'(t_{n-1}) = h^{-1}\omega_{n-1}.$$

From here, it is clear that the behavior of an initial guess on the simple test

$$y''(t) = -\lambda^2 y, \quad y(t_{n-1}) = \epsilon_{n-1}, \quad y'(t_{n-1}) = h^{-1}\omega_{n-1}, \tag{4.2}$$

is significant in order to know how the accumulated errors $(\epsilon_{n-1}, \omega_{n-1})$ at the step t_{n-1} are propagated, specially in the case when high frequencies ($\lambda >> 1$) are present.
By applying the s-stage Gauss method to the test (4.2) by putting $z = \lambda h$ and making $z \to \infty$, from (2.2) and (2.5) after a straightforward calculation we get that

$$Y_{n-1} = Y_n = 0 \in \mathbb{R}^s, \quad y_n = r^*\epsilon_{n-1} \quad \text{and} \quad hy'_n = r'\epsilon_{n-1} + r^*\omega_{n-1}. \tag{4.3}$$

Then, for the starting algorithm (4.1), by letting $z \to \infty$, it follows that

$$Y^{(0)}_{n,i} = a_i\epsilon_{n-1} + d_i\omega_{n-1}, \quad i = 1, \ldots, s. \tag{4.4}$$

Since the error is given by $Y^{(0)}_n - Y_n = Y^{(0)}_n$, then for each order $q = 1, 2, \ldots$, we will select a predictor with amplification factors (vectors $a = (a_i)$ and $d = (d_i)$) of size as small as possible.

Initial guesses of orders $q = 1, 2, \ldots, s$, with 0-amplification factor at $z = \infty$ are respectively given by (interpolating polynomials on nearby stages of the previous integration step),

$$Y^{(0),q}_{n,i} = P_{s,q}(1 + \tau c_i), \quad i = 1, \ldots, s, \quad q = 1, \ldots, s, \tag{4.5}$$

where $P_{s,q}(t)$ is a polynomial of degree $q - 1$ at most, satisfying

$$P_{s,q}(c_{s+1-i}) = Y_{n-1,s+1-i}, \quad i = 1, \ldots, q. \tag{4.6}$$

The unique predictor of order $s + 1$ satisfying $d^T = (d_i) = 0^T$, is given by

$$Y^{(0),s+1}_{n,i} = P_{s,s+1}(1 + \tau c_i), \quad P_{s,s+1}(0) = y_{n-1}, \quad P_{s,s+1}(c_i) = Y_{n-1,i}, (i = 1, \ldots, s). \tag{4.7}$$

Table 4.1
y-component at $t_{end} = 4$ after $N = 40$ steps of size $h = 0.1$, with our scheme y_{our} and also with Simplified Newton Iteration y_{sni} by making μ iterations per step and starting algorithm $Y_{n,i}^{(0),q}$.

2-stage Gauss	$Y_{n,i}^{(0),1}$	$Y_{n,i}^{(0),2}$	$Y_{n,i}^{(0),3}$	$Y_{n,i}^{(0),4}$
$\mu = 1$	$y_{our} = -2.27e\text{-}9$	$y_{our} = 5.12e{+}2$	$y_{our} = -4.61e{+}22$	$y_{our} = -1.16e{+}33$
	$y_{sni} = 1.09e\text{-}8$	$y_{sni} = 1.10e\text{-}8$	$y_{sni} = 1.76e\text{-}3$	$y_{sni} = 1.88e{+}7$
$\mu = 2$	$y_{our} = 8.11e\text{-}9$	$y_{our} = 8.33e\text{-}9$	$y_{our} = -2.03e\text{-}13$	$y_{our} = 1.09e{+}0$
	$y_{sni} = 1.00e\text{-}8$	$y_{sni} = 1.00e\text{-}8$	$y_{sni} = 1.94e\text{-}8$	$y_{sni} = 2.91e\text{-}5$
$\mu = 3$	$y_{our} = 1.10e\text{-}8$	$y_{our} = 1.00e\text{-}8$	$y_{our} = 5.02e\text{-}8$	$y_{our} = 6.52e\text{-}3$
	$y_{sni} = 9.96e\text{-}9$	$y_{sni} = 9.96e\text{-}9$	$y_{sni} = 1.03e\text{-}8$	$y_{sni} = 2.30e\text{-}8$

Here, $P_{s,s+1}(t)$ is a polynomial of degree s at most.

For the cases $s = 2$ and $s = 3$, predictors $(Y_{n,i}^{(0),s+2})$ of order $s + 2$ can be easily computed from the linear equations indicated at the beginning of this section. Moreover, predictors of higher orders can be calculated by using the *special Nyström tree theory* and the $C(s)$-condition.

In order to illustrate a potential danger associated with high order predictors, we have computed the amplification factors for the 2-stage Gauss method and its natural fourth order starting algorithm (mentioned before). The values for $a_2(\tau)$ and $d_2(\tau)$ for several step-size ratios $\tau = 1, 2, 3$, are given next, these τ-values can be representative for cases in which the step-size increases in the integration,

$$a_2(1) = 31.86,\ d_2(1) = 3.732, \quad a_2(2) = 131.4,\ d_2(2) = 17.66, \quad a_2(3) = 338.3,\ d_2(3) = 48.25.$$

From this circumstance, we can say that the amplitudes associated with the high frequencies will be not damped in very few iterations unless the spectral radius of the matrix $N(\infty)$ (see section 3) is fairly small.

Table 4.1 illustrates, by considering the simple problem

$$y''(t) = -\eta(1 + t)^{-1}y, \quad y(0) = 10^{-8},\ y'(0) = 0, \quad \eta = 10^{10},\ t \in [0, 4],$$

the negative influence of large amplification factors (for the predictors) on the accuracy achieved at the end point when giving a small number of iterations per integration step with a prefixed iterative scheme. There, the integrations were carried out by using the two stage Gauss method and the iterations employed were the Simplified Newton Iteration and our iteration (see (2.11) and (3.4)) respectively. Moreover, the Jacobian matrix and the LU-factorization were updated at each integration step. For the first step, the predictor taken was $Y_{0,i}^{(0)} = y_0 + c_i h y_0'$ $(i = 1, 2)$, and $\mu + 1$ iterations were performed (and μ iterations for subsequent steps).

4.1 VOS: Variable Order Strategy to select initial guesses

From the previous study we cannot infer that the starting algorithms of higher order are the best. Thus, to achieve a balance between high order and small amplification factors for high frequencies, the following strategy can be adopted (we refer to it as VOS) to select the best initial guess at a current step. This strategy has proved to be very efficient for first order stiff differential systems [9, p. 89-93]. The idea is to choose the predictor having the smallest error (in some norm). Since the starting algorithms above ($Y_{n,i}^{(0),q}$), have consecutive orders $q = 1, 2, \ldots, q_{max}$, the error of $Y_{n,i}^{(0),q}$ is computed from

$$E_{n,i}^{(0),q} = \| \, Y_{n,i}^{(0),q} - Y_{n,i}^{(0),q+1} \, \|, \ (i = 1, .., s), \quad (q = 1, .., q_{max} - 1).$$

The selection can be done as follows: for $q = 1, 2, \ldots, q_{max} - 1$, take the first q (and the predictor $Y_{n,i}^{(0),q}$; $i = 1, \ldots, s$) such that

$$E_{n,s}^{(0),q+1} \geq \kappa E_{n,s}^{(0),q}, \quad \text{typically } \kappa = 0.5. \tag{4.8}$$

If (4.8) is not satisfied for any $q = 1, 2, \ldots, q_{max} - 1$, then $q = q_{max}$ is chosen (the highest order predictor) only when $E_{n,s}^{(0),q_{max}} \leq \mu\kappa E_{n,s}^{(0),q_{max}-1}$ ($\mu = 0.2$ gives good results). In other case, $q = q_{max} - 1$ is taken.

Observe that the error in (4.8) is only measured on the s-stage, since this stage possibly has the largest error (it is the farthest away stage to be interpolated from the previous step). Moreover, although there is not an error estimator for the highest order starting algorithm, this one will be selected when its difference with the preceding predictor is not significant. It must be observed that for two given guesses having similar errors the lower order one will be preferred (as indicated in (4.8)) since it has smaller amplification factors. For the first step we can adopt a similar strategy, but taking as starting algorithms:
$Y_{0,i}^{(0),1} = y_0$ (order 1), $\quad Y_{0,i}^{(0),2} = y_0 + h_0 c_i y_0'$ (order 2), $\quad$ and
$Y_{0,i}^{(0),3} = y_0 + h_0 c_i y_0' + 2^{-1}(h_0 c_i)^2 f(t_0, y_0)$ (order 3), $\quad$ for $\quad i = 1, \ldots, s$.

5 A few more numerical experiments

This section is devoted to illustrate the theory on previous sections and also to make some comparisons among the Simplified Newton Iteration and our iterative scheme, when both iterations are implemented on the Gauss methods by using a fixed step-size strategy and by evaluating the Jacobian matrix (and updating the corresponding LU-factorization) at each integration step. We have considered three test problems often appearing in the related literature,
Problem 1[7, p.201], $y''(t) + \sinh y(t) = 0, \quad y(0) = 1, \quad y'(0) = 0, \quad t \in [0, 4]$.

Problem 2 [11, p.10-12] is *the outer solar system* (of dimension $m = 18$), $t \in [0, 500000]$. In this case the smallest y-component at the end point is $y_4 = -5.56..$ and the biggest one is $y_{14} = 38.6...$. This is a non-stiff problem on the whole integration interval and, for instance, the eigenvalues (λ) of the Jacobian matrix $J_0 = \partial f/\partial y(t_0, y_0)$ at the initial point are very small. A numerical calculation shows that all of them are real and satisfy $|\lambda| \leq 10^{-5}$.

Problem 3 [14, p.610] is the one-dimensional wave equation with friction given by,

$$u_{tt} = gd\, u_{xx} + g^2u^3/(C^4d^2), \quad 0 < x < l,\ t \in [0, 10],$$
$$u_x(t, 0) = u_x(t, l) = 0, \quad u(0, x) = \sin(l^{-1}\pi x),\ u_t(0, x) = -l^{-1}\pi\sqrt{gd}\cos(l^{-1}\pi x),$$

where $d = d(x) = d_0(2 + \cos(2l^{-1}\pi x))$, g is the acceleration of gravity, and C is the Chezy coefficient. Fourth order symmetric differences have been used to discretize $u_{xx}(x, t)$ on the x-variable at $x_j = j\Delta x,\ j = 2, 3, \ldots, m-1,\ (\Delta x = l/(m+1))$ and also fourth order differences to discretize on the x_1-line (involving u-values from x_0 up to x_5) and on x_m-line. The boundary conditions, $u_x(t, 0)$ and $u_x(t, l)$, are respectively discretized by using fifth order forward-backward differences in order to keep the disctretization errors of the same order, when passing from the partial differential problem to the second order ordinary differential system. The solution components u_0 and u_{m+1} are eliminated from the boundary conditions ($u_x(t, 0) = u_x(t, l) = 0$), hence this yields a differential system of dimension m. The parameters we have taken are,

$$m = 41, \quad l = 100, \quad d_0 = 10, \quad C = 50, \quad g = 9.81.$$

This problem is medium-stiff, since the 41 eigenvalues of $J_0 = \partial f/\partial y(t_0, y_0)$ are real and they are almost uniformly distributed on the interval $[-276.3..., 0]$. There is not much change when the Jacobian matrix is evaluated on other $(t, y(t))$-points with $t > 0$. At $t = 10$, the smallest solution component is $y_{39} = 0.0550...$ and the biggest one is $y_1 = 1.956...$. In all problems the errors (absolute errors are displayed in each table) were measured by using the weighted Euclidean norm $\| x \| := m^{-1/2} \| x \|_2,\ x \in \mathbb{R}^m$.

The numbers exposed in Table 5.1 and in Table 5.2, denoted by $p(h)$ and $p'(h)$ respectively, are computed by global extrapolation as follows,

$$p(h) = \frac{\ln(e(h/2)) - \ln(e(h))}{\ln 2}, \quad p'(h) = \frac{\ln(e'(h/2)) - \ln(e'(h))}{\ln 2},$$

where $e(h) = \| y_{RK}(t_{end}, h) - y_N^{(\mu)} \|$, $e'(h) = \| y'_{RK}(t_{end}, h) - y'^{(\mu)}_N \|$, and $y_{RK}(t_{end}, h)$ and $y'_{RK}(t_{end}, h)$ stand for the exact two-stage Gauss solution at the end point after N steps of size $h = t_{end}/N$ and $\{y_n^{(\mu)},\ y'^{(\mu)}_n\}$ $(n = 0, 1, \ldots, N)$ denote the advancing solutions after n steps and μ iterations per integration step. The predictors used were those ones designated by $(Y_{n,i}^{(0),q})$, which have order $q = 1, 2, 3, 4$, as it was indicated in section 4. In brackets we have included the errors $e(h)$ and $e'(h)$. For the first step the predictor used in all cases was $Y_0^{(0)} = e \otimes y_0$ (order one) and two iterations more than for each coming step were performed. The results in Table 5.1 nicely confirm the statement of Theorem 1. On the other hand, the results in Table 5.2 also fit well with the statement of Theorem 2, which applies to the Simplified Newton Iteration.

In order to gain more insight about the numerical behavior of several iterations on practical problems, we have integrated problem 2 and problem 3, by using the s-stage Gauss methods ($s = 2, 3, 4$) and giving a fixed number of iterations μ per integration step, except at the first step where $\mu + 2$ iterations were carried and the predictor $Y_0 = e \otimes y_0$ was taken. We have used the variable order strategy (VOS) for predictors when either the Simplified Newton Iteration or our iteration was implemented. In the case of the two stage Gauss method we have also included the results obtained by using the Cooper-Butcher iteration as it was customized by Gladwell and Thomas in [7, p.191-194] with the predictors indicated therein [7, p.198-199] (and also giving two iterations more at the first step).

As for the two stage Gauss method, it can be appreciated in Table 5.3 and in Table 5.5 that our iteration (2.11)-(3.4) behaves as well as the Simplified Newton Iteration even at the same number of iterations, i.e. they practically give similar global errors (denoted by ϵ) at the end point. Also both iterations practically take the same predictors, in brackets we have displayed the number of times that each predictor was chosen, thus $[0, 0, 187, 12]$ means that the predictors of order 1 and 2 were never chosen that the predictor of order three was taken 187 times and the predictor of order four was employed 12 times. It is also clearly appreciated in Table 5.3 (problem 2 is non-stiff) that the Cooper-Butcher iteration needs more iterations than our iterative scheme to achieve the same accuracy, this can be explained from the order theory in section 2, because the Cooper-Butcher iteration does not gain any order on h with the iterates. On the problem 3 (stiff case) the Cooper-Butcher iteration behaves in a similar way as ours. We can also see that the VOS strategy works nicely well on both problems and that the best predictors for non-stiff problems (problem 2) are currently the highest order ones, however when the problem becomes more stiff (problem 3) medium order predictors are preferred.

As for the three stage Gauss method and the four stage Gauss method, when comparing our iteration (2.11) (see also section 3) with the Simplified Newton Iteration it can be appreciated in Table 5.4 and in Table 5.6 that both iterations practically achieve the same accuracy at the same number of iterations. In the worst case, our iteration needs one iteration more than the Simplified Newton Iteration to get similar global errors. This gives a great advantage to our iteration when comparing the computational efforts. It is also shown that when both iterations are implemented with the VOS strategy, they practically take the same predictors and in the case of stiff problems (problem 3) the medium order predictors are preferred, observe that the highest order predictor was practically never chosen. Conversely, for the non-stiff case (problem 2) the higher order predictors are taken in most of times. From these tables it can be seen, as it is expected, the increasing accuracy of the s-stage method when s grows and the iteration process converges ($\mu = 4, 5$ iterations). However, for $\mu = 2, 3$ iterations per integration step, the accuracy is practically independent of the stage number $s = 2, 3, 4$.

From these results and from other numerical experiments not presented here, we conclude that the iteration given by (2.11) with the matrices L, S and the parameter $\gamma = (\det A)^{2/s}$ chosen as indicated in section 3, seems to be a very competitive alternative to the Simplified Newton Iteration at least for $s = 2, 3, 4$ stages, when it is implemented on Gauss methods and possibly

Table 5.1
Orders and errors for the advancing solution: $p(h)[e(h)], p'(h)[e'(h)]$, in Problem 1 ($t_{end} = 4$, $h = 0.4$), with Our iterative scheme (μ iterations per integration step), 2-stage Gauss method and starting algorithm $Y_{n,i}^{(0),q}$.

2-stage Gauss	$Y_{n,i}^{(0),1}$	$Y_{n,i}^{(0),2}$	$Y_{n,i}^{(0),3}$	$Y_{n,i}^{(0),4}$
$\mu = 1$	$p = 1.9[4.6e\text{-}2]$	$p = 2.7[5.9e\text{-}3]$	$p = 3.8[3.1e\text{-}3]$	$p = 5.1[2.3e\text{-}4]$
	$p' = 2.2[2.7e\text{-}2]$	$p' = 3.6[3.1e\text{-}3]$	$p' = 4.0[2.4e\text{-}3]$	$p' = 6.6[1.4e\text{-}4]$
$\mu = 2$	$p = 3.9[1.1e\text{-}3]$	$p = 5.1[2.3e\text{-}4]$	$p = 5.7[4.3e\text{-}5]$	$p = 7.1[4.3e\text{-}6]$
	$p' = 4.0[7.5e\text{-}4]$	$p' = 8.6[6.1e\text{-}5]$	$p' = 6.0[3.7e\text{-}5]$	$p' = 9.2[1.6e\text{-}6]$
$\mu = 3$	$p = 5.8[1.8e\text{-}5]$	$p = 7.1[4.7e\text{-}6]$	$p = 7.6[6.9e\text{-}7]$	$p = 9.1[8.3e\text{-}8]$
	$p' = 6.1[1.3e\text{-}5]$	$p' = 9.4[1.1e\text{-}6]$	$p' = 8.0[6.4e\text{-}7]$	$p' = 10.5[2.9e\text{-}8]$

Table 5.2
Orders and errors for the advancing solution: $p(h)[e(h)], p'(h)[e'(h)]$, in Problem 1 ($t_{end} = 4$, $h = 0.4$), with the Simplified Newton Iteration (μ iterations per integration step), the two-stage Gauss method and starting algorithm $Y_{n,i}^{(0),q}$.

2-stage Gauss	$Y_{n,i}^{(0),1}$	$Y_{n,i}^{(0),2}$	$Y_{n,i}^{(0),3}$	$Y_{n,i}^{(0),4}$
$\mu = 1$	$p = 1.8[9.2e\text{-}3]$	$p = 4.7[4.1e\text{-}3]$	$p = 3.9[1.6e\text{-}3]$	$p = 5.0[8.8e\text{-}5]$
	$p' = 2.2[1.4e\text{-}2]$	$p' = 2.6[1.1e\text{-}2]$	$p' = 3.8[1.4e\text{-}3]$	$p' = 5.2[3.4e\text{-}4]$
$\mu = 2$	$p = 8.2[6.6e\text{-}7]$	$p = 5.6[1.2e\text{-}5]$	$p = 8.2[8.5e\text{-}7]$	$p = 7.6[1.3e\text{-}7]$
	$p' = 5.0[1.8e\text{-}5]$	$p' = 5.9[1.6e\text{-}5]$	$p' = 6.7[2.3e\text{-}6]$	$p' = 8.4[3.1e\text{-}7]$
$\mu = 3$	$p = 8.7[1.2e\text{-}9]$	$p = 14.6[1.1e\text{-}8]$	$p = 10.1[3.6e\text{-}10]$	$p = 10.5[2.1e\text{-}11]$
	$p' = 9.1[2.5e\text{-}9]$	$p' = 8.7[8.2e\text{-}9]$	$p' = 10.0[2.7e\text{-}10]$	$p' = 10.9[2.6e\text{-}11]$

Table 5.3
Two stage Gauss method on problem 2. Global errors on the y-component at $t_{end} = 5 \cdot 10^5$ ($h = 125$), by using our iteration, the Simplified Newton Iteration and the Cooper-Butcher iteration. μ iterations per step were carried out and the variable order strategy VOS for predictors was used (in brackets the number of times that each starting algorithm was chosen)

2-stage Gauss	Our iteration	Simp. Newt. Iter.	Cooper-Butcher Iter.
$\mu = 1$	$\epsilon = 2.26e{+}1[0, 0, 0, 3999]$	$\epsilon = 2.23e{+}1[0, 0, 0, 3999]$	$\epsilon = 2.85e{+}1$
$\mu = 2$	$\epsilon = 1.89\text{-}1[0, 0, 0, 3999]$	$\epsilon = 1.78e\text{-}2[0, 0, 0, 3999]$	$\epsilon = 2.92e{+}1$
$\mu = 3$	$\epsilon = 1.60e\text{-}2[0, 0, 0, 3999]$	$\epsilon = 1.62e\text{-}2[0, 0, 0, 3999]$	$\epsilon = 2.37e{+}0$
$\mu = 4$	$\epsilon = 1.50e\text{-}2[0, 0, 0, 3999]$	$\epsilon = 1.50e\text{-}2[0, 0, 0, 3999]$	$\epsilon = 4.69e\text{-}1$

on any (high order) highly implicit Runge-Kutta-Nyström method. It is also remarkable that the variable order strategy for predictors (VOS) seems to be more attractive than any prefixed-predictor option, since the former alternative is more flexible and take the most convenient guess on the current step. For the Lobatto IIIA methods an iteration of type (2.9) can be adapted, by making some minor modifications. Observe that for those methods the first stage is explicit, hence the matrix T in (2.9) must be adjusted for the implicit stages only.

Table 5.4
Three-stage Gauss method and four-stage Gauss method on problem 2. Global errors on the y-component at $t_{end} = 5 \cdot 10^5$ ($h = 125$), with our iteration and with the Simplified Newton Iteration, by making μ iterations per step and the variable order strategy for predictors (in brackets the number of times that each starting algorithm was chosen)

3-stage Gauss	Our iteration	Simp. Newt. Iter.
$\mu = 1$	$\epsilon = 9.90e{+}0[0, 0, 0, 3527, 472]$	$\epsilon = 3.48e{+}0[0, 0, 0, 3316, 683]$
$\mu = 2$	$\epsilon = 8.29e\text{-}2[0, 0, 0, 2635, 1364]$	$\epsilon = 5.69e\text{-}3[0, 0, 0, 2632, 1367]$
$\mu = 3$	$\epsilon = 5.78e\text{-}4[0, 0, 0, 2630, 1369]$	$\epsilon = 4.11e\text{-}4[0, 0, 0, 2630, 1369]$
$\mu = 4$	$\epsilon = 4.87e\text{-}6[0, 0, 0, 2630, 1369]$	$\epsilon = 3.10e\text{-}6[0, 0, 0, 2630, 1369]$
$\mu = 5$	$\epsilon = 2.93e\text{-}6[0, 0, 0, 2630, 1369]$	$\epsilon = 2.94e\text{-}6[0, 0, 0, 2630, 1369]$
4-stage Gauss	Our iteration	Simp. Newt. Iter.
$\mu = 1$	$\epsilon = 1.96e{+}1[0, 0, 0, 498, 3501]$	$\epsilon = 1.74e{+}1[0, 0, 0, 436, 3563]$
$\mu = 2$	$\epsilon = 9.23e\text{-}3[0, 0, 0, 535, 3464]$	$\epsilon = 4.65e\text{-}3[0, 0, 0, 535, 3464]$
$\mu = 3$	$\epsilon = 1.75e\text{-}4[0, 0, 0, 535, 3464]$	$\epsilon = 1.46e\text{-}4[0, 0, 0, 535, 3464]$
$\mu = 4$	$\epsilon = 2.69e\text{-}7[0, 0, 0, 535, 3464]$	$\epsilon = 4.84e\text{-}8[0, 0, 0, 535, 3464]$
$\mu = 5$	$\epsilon = 2.07e\text{-}8[0, 0, 0, 535, 3464]$	$\epsilon = 4.01e\text{-}8[0, 0, 0, 535, 3464]$

Table 5.5
Two-stage Gauss method on problem 3. Global errors on the y-component at $t_{end} = 10$ ($h = 0.05$), with our iteration, with the Simplified Newton Iteration and with the Cooper-Butcher iteration, by making μ iterations per step and the variable order strategy for predictors (in brackets the number of times that each starting algorithm was chosen)

2-stage Gauss	Our iteration	Simp. Newt. Iter.	Cooper-Butcher Iter.
$\mu = 1$	$\epsilon = 3.65e\text{-}3[0, 0, 187, 12]$	$\epsilon = 3.40e\text{-}3[0, 0, 176, 23]$	$\epsilon = 4.92e\text{-}3$
$\mu = 2$	$\epsilon = 1.22e\text{-}4[0, 0, 161, 38]$	$\epsilon = 4.66e\text{-}5[0, 0, 158, 41]$	$\epsilon = 2.22e\text{-}4$
$\mu = 3$	$\epsilon = 2.20e\text{-}5[0, 0, 158, 41]$	$\epsilon = 1.81e\text{-}5[0, 0, 158, 41]$	$\epsilon = 2.18e\text{-}5$
$\mu = 4$	$\epsilon = 1.85e\text{-}5[0, 0, 158, 41]$	$\epsilon = 1.83e\text{-}5[0, 0, 158, 41]$	$\epsilon = 1.96e\text{-}5$

References

[1] J. R. Cash, *High order, P-stable formulae for the numerical integration of periodic initial values problems*, Numer. Math., 37 (1981), pp. 355-370.

[2] J. R. Cash, *Efficient P-stable methods for periodic initial values problems*, BIT, 24 (1984), pp. 248-252.

[3] G. J. Cooper and J. C. Butcher, *An iteration scheme for implicit Runge-Kutta methods*, IMA J. Numer. Anal., 3 (1983), pp. 127-140.

[4] M. M. Chawla, *Unconditionally stable Numerov-type methods for second order differential*

Table 5.6
Three-stage Gauss method and four-stage Gauss method on problem 3. Global errors on the y-component at $t_{end} = 10$ ($h = 0.05$) with our iteration and with the Simplified Newton Iteration, by making μ iterations per step and the variable order strategy for predictors (in brackets the number of times that each starting algorithm was chosen)

3-stage Gauss	Our iteration	Simp. Newt. Iter.
$\mu = 1$	$\epsilon = 3.16e\text{-}4[0, 0, 1, 198, 0]$	$\epsilon = 3.27e\text{-}4[0, 0, 1, 198, 0]$
$\mu = 2$	$\epsilon = 5.42\text{-}6[0, 0, 1, 198, 0]$	$\epsilon = 3.68e\text{-}6[0, 0, 1, 198, 0]$
$\mu = 3$	$\epsilon = 1.20e\text{-}7[0, 0, 1, 198, 0]$	$\epsilon = 3.52e\text{-}8[0, 0, 1, 198, 0]$
$\mu = 4$	$\epsilon = 3.03e\text{-}8[0, 0, 1, 198, 0]$	$\epsilon = 3.16e\text{-}8[0, 0, 1, 198, 0]$
4-stage Gauss	**Our iteration**	**Simp. Newt. Iter.**
$\mu = 1$	$\epsilon = 3.40e\text{-}4[0, 0, 1, 197, 1]$	$\epsilon = 3.43e\text{-}4[0, 0, 1, 197, 1]$
$\mu = 2$	$\epsilon = 4.83\text{-}6[0, 0, 1, 198, 0]$	$\epsilon = 4.25e\text{-}6[0, 0, 1, 198, 0]$
$\mu = 3$	$\epsilon = 7.77e\text{-}8[0, 0, 1, 198, 0]$	$\epsilon = 4.01e\text{-}8[0, 0, 1, 198, 0]$
$\mu = 4$	$\epsilon = 1.48e\text{-}9[0, 0, 1, 198, 0]$	$\epsilon = 2.43e\text{-}10[0, 0, 1, 198, 0]$
$\mu = 5$	$\epsilon = 5.25e\text{-}11[0, 0, 1, 198, 0]$	$\epsilon = 4.22e\text{-}11[0, 0, 1, 198, 0]$

equations, BIT, 23 (1983), pp. 541-542.

[5] M. M. Chawla, P. S. Rao and Beny Neta, *Two-step fourth order P-stable methods with phase-lag of order six for $y'' = f(t, y)$*, J. Comput. Appl. Math., 16 (1986), pp. 233-236.

[6] J. M. Franco, J. M. Gómez and L. Rández, *SDIRK methods for ODEs with oscillating solutions*, J. Comput. Appl. Math., 81 (1997), pp. 197-209.

[7] I. Gladwell and R. M. Thomas, *Efficiency of Methods for Second–Order Problems*, IMA J. Numer. Anal., 10 (1990), pp. 181–207.

[8] S. González–Pinto, J. I. Montijano and L. Rández, *Iterative schemes for three-stage implicit Runge-Kutta methods*, Appl. Numer. Math., 17 (1995), pp. 363–382.

[9] S. González–Pinto, J. I. Montijano and S. Peréz–Rodríguez, *Variable-order starting algorithms for implicit Runge-Kutta methods on stiff problems*, Appl. Numer. Math., 44 (2003), pp. 77–94.

[10] E. Hairer, *Unconditionally stable methods for second order differential equations*, Numer. Math., 32 (1979), pp. 373–379.

[11] E. Hairer, C. Lubich and G. Wanner, *Geometric Numerical Integration*, Springer, Berlin, 2002.

[12] E. Hairer, S. P. Nørsett and G. Wanner, *Solving ordinary differential equations I*, Springer Verlag, 2^{nd} ed., Berlin, 1993.

[13] E. Hairer and G. Wanner, *Solving ordinary differential equations II*, Springer Verlag, 2^{nd} ed., Berlin, 1996.

[14] P. J. van der Houwen and B. P. Sommeijer, *Explicit Runge-Kutta(-Nyström) methods with reduced phase errors for computing oscillating solutions*, SIAM J. Numer. Anal., 24 No. 3 (1987), pp. 595–617.

[15] P. J. van der Houwen and B. P. Sommeijer, *Diagonally implicit Runge-Kutta-Nyström methods for oscillating problems*, SIAM J. Numer. Anal., 26 No. 2 (1989), pp. 414–429.

[16] P. J. van der Houwen and J. J. B. de Swart, *Triangularly implicit iteration methods for ODE-IVP solvers*, SIAM J. Sci. Comput., 18 No. 1 (1997), pp. 41–55.

[17] J. D. Lambert and I. A. Watson, *Symmetric multistep methods for periodic initial value problems*, J. Inst. Math. Appl., 18 (1976), pp. 189-202.

[18] S. P. Nørsett, *One step Hermite type methods for numerical integration of stiff system*, BIT, 14 (1974), pp. 63–67.

[19] P. W. Sharp, J. M. Fine and K. Burrage, *Two stage and Three stage Diagonally Implicit Runge-Kutta Nyström methods of orders three and four*, IMA J. Numer. Anal., 10 (1990), pp. 489–504.

[20] R. M. Thomas, *Phase properties of high order, almost P-stable formulae*, BIT, 24 (1984), pp. 225–238.