ma
the
ma
tisch

cen
trum

MC

AFDELING NUMERIEKE WISKUNDE NW 57/78 MEI
(DEPARTMENT OF NUMERICAL MATHEMATICS)

H.J.J. TE RIELE & P.J. VAN DER HOUWEN

BACKWARD DIFFERENTIATION FORMULAS FOR VOLTERRA INTEGRAL EQUATIONS OF THE SECOND KIND

II NUMERICAL EXPERIMENTS

amsterdam 1978

stichting
mathematisch
centrum

MC

AFDELING NUMERIEKE WISKUNDE NW 57/78 MEI
(DEPARTMENT OF NUMERICAL MATHEMATICS)

H.J.J. TE RIELE & P.J. VAN DER HOUWEN

BACKWARD DIFFERENTIATION FORMULAS FOR VOLTERRA INTEGRAL EQUATIONS OF THE SECOND KIND

II NUMERICAL EXPERIMENTS

2e boerhaavestraat 49 amsterdam

Printed at the Mathematical Centre, 49, 2e Boerhaavestraat, Amsterdam.

The Mathematical Centre, founded the 11-th of February 1946, is a non-profit institution aiming at the promotion of pure mathematics and its applications. It is sponsored by the Netherlands Government through the Netherlands Organization for the Advancement of Pure Research (Z.W.O).

AMS(MOS) subject classification scheme (1970): 65R05

Backward differentiation formulas for Volterra integral equations of the second kind

II Numerical experiments

by

H.J.J. TE RIELE & P.J. VAN DER HOUWEN

ABSTRACT

In this report the results are given of numerical experiments with backward differentiation formulas for linear and nonlinear Volterra integral equations of the second kind. In order to start these (multistep) formulas, a starting scheme is supplied, based on extrapolation of the trapezoidal rule, combined with a block-implicit Runge-Kutta scheme.

Convergence and stability tests are carried out and the efficiency is compared with that of a block-implicit Runge-Kutta scheme of de Hoog and Weiss.

It turns out that the backward differentiation formulas are especially suitable for problems where the kernel has a large Lipschitz constant with respect to its last argument.

KEY WORDS & PHRASES: *Volterra integral equations, backward differentiation formulas, block-implicit Runge-Kutta method.*

CONTENTS

1. INTRODUCTION

In reference [5] backward differentiation formulas (abbreviated: BDFs) are studied for the numerical solution of linear and nonlinear integral equations of Volterra of the second kind

$$(1.1) \qquad \vec{f}(x) = \vec{g}(x) + \int_{x_0}^{x} \vec{K}(x,y,\vec{f}(y))dy, \qquad x_0 \le x \le x_e,$$

where $\vec{g}$ and $\vec{K}$ are prescribed vector functions and $\vec{f}$ is the unknown vector function. The BDFs are based on the well-known Curtiss-Hirschfelder formulas for ordinary differential equations. When one uses a k-th order Curtiss-Hirschfelder formula ($k \ge 2$), a BDF is proved in [5] to have global error of order k, provided that for the quadrature formula to be used one selects a Gregory formula with k-2 correction terms. The BDFs are constructed in such a way that the excellent stability properties of the Curtiss-Hirschfelder formulas are preserved. Moreover, in [5] stability properties of the BDFs are investigated for a more general class of model equations (viz., $\vec{K} = \vec{A}(y,\vec{f}) + xH\vec{f}$, $\vec{A}$ an arbitrary vector function, $\vec{H}$ an arbitrary constant matrix) than the usual one ($\vec{K} = J\vec{f}$, J an arbitrary constant matrix).

The purpose of this report is (i) to give a starting scheme for the BDFs, which has a stability behaviour comparable with that of the BDFs themselves, (ii) to give the results of numerical experiments in order to test the convergence and stability properties, and (iii) to compare the computational performance of the BDFs with a scheme with comparable properties, viz., a block-implicit Runge-Kutta scheme of de Hoog and Weiss [4]. It turns out that for equations of the type (1.1) for which $\partial K/\partial f$ assumes large negative values, the sixth order BDF is more efficient than a block-implicit Runge-Kutta scheme of de Hoog and Weiss, which has a comparable stability region, and (even) seventh order of convergence.

2. THE COMPUTATIONAL SCHEME

2.1. The backward differentiation formulas

We give a short résumé of these formulas as given in [5]. Let $x_n = x_0 + nh$, $n = 0,1,2,\ldots$, denote reference points on the x-axis, h being the (fixed) integration step. Define the auxiliary function $\vec{F}_n$ by

$$\vec{F}_n(x) = \vec{g}(x) + \int_{x_0}^{x_n} \vec{K}(x,y,\vec{f}(y))dy, \qquad n = 0,1,2,\ldots .$$

Now for $x_n \le x \le x_{n+1}$ we write (1.1) as

$$\vec{f}(x) = \vec{F}_n(x) + \int_{x_n}^{x} \vec{K}(x,y,\vec{f}(y))dy, \tag{2.1}$$

so $\vec{F}_n$ may be considered as the "past" of the integral equation (1.1), since it only depends on $\vec{f}$ with arguments $x \in [x_0,x_n]$, whereas the integral term in (2.1) depends on $\vec{f}$ with arguments $x \in [x_n,x_{n+1}]$.

Let k (= 2,3,4,5 or 6) be fixed and let b_0 and $a_1,a_2,\ldots,a_k$ be the coefficients of the Curtiss-Hirschfelder formulas of order k for ordinary differential equations. Then *the backward differentation formula of order* k is given by

$$\begin{aligned}\vec{f}_{n+1} &= \vec{\tilde{F}}_{n+1}(x_{n+1}) + b_0 h\vec{K}(x_{n+1},x_{n+1},\vec{f}_{n+1}) \\ &\quad + \sum_{\ell=1}^{k} a_\ell[\vec{f}_{n+1-\ell} - \vec{\tilde{F}}_{n+1}(x_{n+1-\ell})], \qquad n = k-1,k,k+1,\ldots,\end{aligned} \tag{2.2a}$$

where $\vec{f}_n$ is a numerical approximation to the solution $\vec{f}$ of (1.1) in $x = x_n$, and $\vec{\tilde{F}}_n(x)$ is a numerical approximation to $\vec{F}_n(x)$ given by

$$\vec{\tilde{F}}_n(x) = \vec{g}(x) + \sum_{j=0}^{n} w_{nj}\vec{K}(x,x_j,\vec{f}_j), \qquad n = 0,1,\ldots . \tag{2.3}$$

The w_{nj} are the weights of a Gregory quadrature formula with k-2 correction terms. They satisfy

$$w_{n+1j} = w_{nj}, \qquad j = 0,1,\ldots,n-k.$$

Denoting the non-zero differences $w_{n+1j} - w_{nj}$ by ∇w_{n+1j}, $j = n+1-k,\dots,n$, the quantities $\vec{\tilde{F}}_{n+1}(x_{n+1-\ell})$, $\ell = 0,1,\dots,k$, in (2.2a) are computed by

$$\vec{\tilde{F}}_{n+1}(x_{n+1-\ell}) = \vec{\tilde{F}}_n(x_{n+1-\ell}) + \sum_{j=n+1-k}^{n} \nabla w_{n+1j}\vec{K}(x_{n+1-j},x_j,\vec{f}_j),$$

(2.2.b) $\qquad \ell = 1,2,\dots,k,$

$$\vec{\tilde{F}}_{n+1}(x_{n+1}) = \vec{g}(x_{n+1}) + \sum_{j=0}^{n+1} w_{n+1j}\vec{K}(x_{n+1},x_j,\vec{f}_j).$$

2.2. A starting scheme based on extrapolation of the trapezoidal rule

In order to start scheme (2.2) we need, in addition to the initial vector $\vec{f}_0$, the vectors $\vec{f}_1,\vec{f}_2,\dots,\vec{f}_{k-1}$. These starting vectors should be computed with an error of magnitude $O(h^k)$. We consider a starting formula based on extrapolation of the trapezoidal formula

(2.3) $$\vec{f}_n = \vec{g}(x_n) + h \sum_{j=0}^{n}{}'' \vec{K}(x_n,x_j,\vec{f}_j).$$

As is well-known, extrapolation methods are particularly profitable when the Taylor expansion of the error of the formula to be extrapolated, is a series of even or odd powers of h. Therefore, we first investigate the error expansion of formula (2.3).

THEOREM 2.1. *Let* $\vec{f}_h(x)$ *denote a sufficiently differentiable interpolating function through the vectors* $\vec{f}_n$, $n = 0,1,\dots,k-1$ *obtained by applying* (2.3) *with step length* h. *Then, at each point* $x = nh$ *and as* $h \to 0$, $x \to x_0$ *we have*

$$\vec{f}_h(x) - \vec{f}(x) = \int_{x_0}^{x} \frac{\partial \vec{K}}{\partial \vec{f}}(x,y,\vec{f}(y))(\vec{f}_h(y) - \vec{f}(y))dy + O(h^4(x-x_0)^3)$$

$$+ \vec{\alpha}_2 h^2 + \vec{\alpha}_4 h^4 + \vec{\alpha}_6 h^6 + \vec{\alpha}_8 h^8 + \dots$$

where the $\vec{\alpha}_{2m}$ *are of order* $x - x_0$ *as* $x \to x_0$, *and bounded as* $h \to 0$.

PROOF. From the Taylor expansion of the trapezoidal quadrature rule it follows that at the points x = nh (see e.g. [7, p.153])

$$(2.4)\qquad \vec{f}_h(x) = \vec{g}(x) + \int_{x_0}^{x} \vec{K}(x,y,\vec{f}_h(y))dy + \vec{\alpha}_2(\vec{f}_h')h^2 + \vec{\alpha}_4(\vec{f}_h''')h^4 + \vec{\alpha}_6(\vec{f}_h^{(5)})h^6 + \dots,$$

where

$$(2.5)\qquad \vec{\alpha}_{2m}(\vec{f}_h^{(2m-1)}) = -\frac{B_{2m}}{(2m)!}\left[\vec{f}_h^{(2m-1)}(x) - \vec{f}_h^{(2m-1)}(x_0)\right], \qquad m = 1,2,3,\dots,$$

B_{2m} being the Bernoulli numbers ($B_2 = 1/6$, $B_4 = -1/30, \dots$). Here, it is assumed that the interpolating function $\vec{f}_h$ is sufficiently differentiable. Notice that the functions $\vec{\alpha}_{2m}(\vec{f}_h(x))$ are of order $x - x_0$ as $x \to x_0$. Relation (2.4) enables us to get insight into the Taylor expansion of the global error of the trapezoidal rule (2.3). From relation (2.4) it follows that at the reference points x = nh

$$(2.6)\qquad \vec{f}_h(x) - \vec{f}(x) = \int_{x_0}^{x} [\vec{K}(x,y,\vec{f}_h(y)) - \vec{K}(x,y,\vec{f}(y))]dy + \vec{\alpha}_2(\vec{f}_h')h^2 + \vec{\alpha}_4(\vec{f}_h''')h^4 + \vec{\alpha}_6(\vec{f}_h^{(5)})h^6 + O(h^8),$$

hence,

$$(2.6')\qquad \vec{f}_h(x) - f(x) = \int_{x_0}^{x} \frac{\partial\vec{K}}{\partial\vec{f}}(x,y,\vec{f}(y))(\vec{f}_h(y) - \vec{f}(y))dy + (x - x_0)O([\vec{f}_h - \vec{f}]^2) + \vec{\alpha}_2(\vec{f}_h')h^2 + \vec{\alpha}_4(\vec{f}_h''')h^4 + \vec{\alpha}_6(\vec{f}_h^{(5)})h^6 + \dots .$$

In order to find the order of magnitude of the term $(x - x_0)[\vec{f}_h - \vec{f}]^2$ we observe that expansion (2.6) yields the inequality

$$\|\vec{f}_h(x) - \vec{f}(x)\| \le \int_{x_0}^{x} L(x,y)\|\vec{f}_h(y) - \vec{f}(y)\|dy + \|\vec{\alpha}_2(\vec{f}_h')\|h^2 + \ldots$$

$$\le (x-x_0)L(x,\bar{x})\|\vec{f}_h(\bar{x}) - \vec{f}(\bar{x})\| + \|\vec{\alpha}_2(\vec{f}_h')\|h^2 + \ldots$$

$$= (x-x_0)L(x,\bar{x})[1 + 0(x-x_0)]\|\vec{f}_h(x) - \vec{f}(x)\| +$$

$$+ \|\vec{\alpha}_2(\vec{f}_h')\|h^2 + \ldots, \qquad x_0 \le \bar{x} \le x,$$

where $L(x,y)$ is a Lipschitz constant for the function $\vec{K}(x,y,\vec{f})$ with respect to the variable $\vec{f}$. This inequality yields for sufficiently small values of $x - x_0$

$$\|\vec{f}_h(x) - \vec{f}(x)\| \le [1 - (x-x_0)L(x,\bar{x})(1 + 0(x-x_0))]^{-1}\cdot$$

$$\cdot[\|\vec{\alpha}_2(\vec{f}_h')\|h^2 + \ldots].$$

Thus, by using the relation $\|\vec{\alpha}_2(\vec{f}_h)\| = 0(x-x_0)$ as $x \to x_0$, we find

$$(2.7) \qquad \|\vec{f}_h(x) - \vec{f}(x)\| = 0(h^2(x-x_0)) \qquad \text{as } h \to 0 \text{ and } x \to x_0.$$

Substitution into (2.6') results in the following expansion of the error of the trapezoidal formula (2.3):

$$(2.8) \qquad \vec{f}_h(x) - \vec{f}(x) = \int_{x_0}^{x} \frac{\partial\vec{K}}{\partial\vec{f}}(x,y,\vec{f}(y))(\vec{f}_h(y) - \vec{f}(y))dy +$$

$$+ \vec{\alpha}_2(\vec{f}_h')h^2 + \vec{\alpha}_4(\vec{f}_h''')h^4 + h^4 0((x-x_0)^3) +$$

$$+ \vec{\alpha}_6(\vec{f}_h^{(5)})h^6 + \ldots . \qquad \square$$

Having derived the error expansion of formula (2.3) we are able to determine the order of accuracy of the extrapolated formulas. Suppose that we also apply formula (2.3) with step length $h/2$ and denote the corresponding interpolating function by $\vec{f}_{h/2}(x)$. From the theorem it then follows that

$$\vec{f}_{h/2}(x) = \vec{f}(x) + \int_{x_0}^{x} \frac{\partial K}{\partial \vec{f}}(x,y,\vec{f}(y))(\vec{f}_{h/2}(y) - \vec{f}(y))dy +$$

$$+ \frac{1}{4}\vec{\alpha}_2(\vec{f}'_{h/2})h^2 + \frac{1}{16}\vec{\alpha}_4(\vec{f}'''_{h/2})h^4 + O(h^7),$$

where we have used that in our case $x - x_0 = O(h)$. Thus, we obtain

(2.9)
$$\frac{4}{3}\vec{f}_{h/2}(x) - \frac{1}{3}\vec{f}_h(x) =$$

$$\vec{f}(x) + \int_{x_0}^{x} \frac{\partial K}{\partial \vec{f}}(x,y,\vec{f}(y))[\tfrac{4}{3}\vec{f}_{h/2}(y) - \tfrac{1}{3}\vec{f}_h(y) - \vec{f}(y)]dy +$$

$$+ \frac{1}{3}[\vec{\alpha}_2(\vec{f}'_{h/2}) - \vec{\alpha}_2(\vec{f}'_h)]h^2 +$$

$$+ \frac{1}{3}[\frac{1}{4}\vec{\alpha}_4(\vec{f}'''_{h/2}) - \vec{\alpha}_4(\vec{f}'''_h)]h^4 + O(h^7).$$

From (2.7) we conclude that

$$\vec{f}_h(x) = \vec{f}(x) + (x - x_0)O(h^2),$$

$$\vec{f}'_h(x) = \vec{f}'(x) + O(h^2)$$

and similar relations for $\vec{f}_{h/2}(x)$.
Hence, a first result deduced from (2.7) is that we have at least

$$\frac{4}{3}\vec{f}_{h/2}(x) - \frac{1}{3}\vec{f}_h(x) = \vec{f}(x) + O(h^4).$$

Substitution of this result into (2.9) reveals that we even have

(2.10)
$$\frac{4}{3}\vec{f}_{h/2}(x) - \frac{1}{3}\vec{f}_h(x) = \vec{f}(x) + O(h^5).$$

Thus, we have derived a *fifth order starting scheme* which can be written as follows ($x_0 = 0$):

$$\vec{f}_h(nh) = \vec{g}(nh) + h \sum_{j=0}^{n}{}'' \vec{K}(nh,jh,\vec{f}_h(jh)), \qquad n = 1,2,\dots,k-1,$$

$$(2.11) \qquad \vec{f}_{h/2}(\tfrac{nh}{2}) = \vec{g}(\tfrac{nh}{2}) + \tfrac{1}{2} h \sum_{j=0}^{n}{}'' \vec{K}(\tfrac{nh}{2}, \tfrac{jh}{2}, \vec{f}_{h/2}(\tfrac{jh}{2})), \quad n = 1,\dots,2(k-1),$$

$$\vec{f}_n = \tfrac{4}{3} \vec{f}_{h/2}(nh) - \tfrac{1}{3} \vec{f}_h(nh), \qquad n = 1,2,\dots,k-1.$$

This scheme yields the starting vectors within the required order of accuracy provided that $k \le 5$. Moreover, there is no danger for the development of instabilities in the starting scheme since the trapezoidal rule is known to be stable, under the condition, of course, that the implicit equations are solved by a Newton type process.

For $k = 6$ we may either continue the extrapolation process or try to improve the starting vectors by other methods. When the extrapolation method is used we may compute the vectors $\vec{f}_{h/4}(nh/4)$, $n = 1,2,\dots,4(k-1)$, and form the combinations

$$(2.12) \qquad \tfrac{4}{3}[\tfrac{4}{3} \vec{f}_{h/4}(x) - \tfrac{1}{3} \vec{f}_{h/2}(x)] - \tfrac{1}{3}[\tfrac{4}{3} \vec{f}_{h/2}(x) - \tfrac{1}{3} \vec{f}_h(x)], \quad x = nh,$$

$$n = 1,2,\dots,k-1.$$

By (2.7) and (2.10) it is easily proved that this combination approximates $f(x)$ within $O(h^6)$. Just like scheme (2.11) there is no danger for instabilities. The computational effort, however, is considerable. An alternative is the application of the corrector equations (cf. [1, p.915])

$$\vec{f}_1 = \vec{g}(h) + \frac{h}{1440} [475\vec{K}_{10} + 1427\vec{K}_{11} - 798\vec{K}_{12} + 482\vec{K}_{13} - 173\vec{K}_{14} + 27\vec{K}_{15}],$$

$$\vec{f}_2 = \vec{g}(2h) + \frac{h}{1440} [448\vec{K}_{20} + 2064\vec{K}_{21} + 224\vec{K}_{22} + 224\vec{K}_{23} - 96\vec{K}_{24} + 16\vec{K}_{25}],$$

$$(2.13) \qquad \vec{f}_3 = \vec{g}(3h) + \frac{h}{1440} [459\vec{K}_{30} + 1971\vec{K}_{31} + 1026\vec{K}_{32} + 1026\vec{K}_{33} - 189\vec{K}_{34} + 27\vec{K}_{35}],$$

$$\vec{f}_4 = \vec{g}(4h) + \frac{h}{1440}[448\vec{K}_{40} + 2048\vec{K}_{41} + 768\vec{K}_{42} + 2048\vec{K}_{43} + 448\vec{K}_{44}],$$

$$\vec{f}_5 = \vec{g}(5h) + \frac{h}{1440}[475\vec{K}_{50} + 1875\vec{K}_{51} + 1250\vec{K}_{52} + 1250\vec{K}_{53} + 1875\vec{K}_{54} + 475\vec{K}_{55}],$$

where

$$\vec{K}_{nj} = \vec{K}(nh, jh, \vec{f}_j).$$

The solutions of these equations approximate $\vec{f}(x_j)$ within $O(h^7)$. Hence, by using a fifth order accurate initial guess such as provided by (2.11), and by applying some iteration process (see section 2.3), we obtain sixth order accuracy after one iteration and seventh order accuracy after two or more iterations.

2.3. Implementation of the numerical schemes

We have programmed the BDFs (2.2) together with the starting schemes (2.11) and (2.13) in ALGOL 68. For the present, we have restricted ourselves to the scalar case of (1.1). Derivatives of the kernel K (when solving the nonlinear equations with Newton-type methods) are always computed numerically with the formula

$$\frac{\partial K}{\partial f}(x,y,f) \approx 1000\{K(x,y,f+0.001) - K(x,y,f)\}.$$

In the schemes (2.2a) and (2.2b) the nonlinear equation in f_{n+1} is solved by the modified Newton-Raphson process with re-evaluation of $\frac{\partial K}{\partial f}(x_{n+1}, x_{n+1}, f_{n+1}^{(j)})$ after every 10 iteration steps. Iteration is stopped when $|\text{correction term}| < 10^{-12}$ is reached. In the scheme (2.11) the nonlinear equation is solved by the (unmodified) Newton-Raphson process, and the iteration is stopped when $|\text{correction term}| < 10^{-12}$ is reached.

The system of 5 nonlinear equations occurring in the scheme (2.13) is solved by a so-called nonlinear Gauss-Seidel method [6]. We have avoided Jacobi iteration, because it produces instabilities in the case of kernels K with $|\partial K/\partial f|$ large. We have avoided Newton iteration because it may be very expensive on the case of *systems* of integral equations (1.1). Defining $B = (b_{ij})$, $i,j = 1,2,3,4,5$, by

$$B = \frac{1}{1440}\begin{bmatrix} 1427 & -798 & 482 & -173 & 27 \\ 2064 & 224 & 224 & -96 & 16 \\ 1971 & 1026 & 1026 & -189 & 27 \\ 2048 & 768 & 2048 & 448 & 0 \\ 1875 & 1250 & 1250 & 1875 & 475 \end{bmatrix},$$

and an additional vector $(b_{10},b_{20},b_{30},b_{40},b_{50}) = 1440^{-1}(475,448,459,448,475)$, *one* iteration step of the nonlinear Gauss-Siedel method is given by

$$(2.14) \qquad f_i^{(c)} = f_i^{(p)} - \omega\Big\{f_i^{(p)} - g(x_i) - h[b_{i0}K(x_i,x_0,f_0) + \sum_{r=1}^{i-1} b_{ir}K(x_i,x_r,f_r^{(c)}) + \sum_{r=i}^{5} b_{ir}K(x_i,x_r,f_r^{(p)})]\Big\} \cdot \left(1 - b_{ii}h\frac{\partial K}{\partial f}(x_i,x_i,f_i^{(p)})\right)^{-1}, \qquad i = 1,2,3,4,5.$$

For the relaxation factor ω we have selected the value $\omega = 0.578$ (see below for explanation). The iteration is started with the $O(h^5)$ solution of the scheme (2.11) as predictor vector $(f_1^{(p)},f_2^{(p)},f_3^{(p)},f_4^{(p)},f_5^{(p)})$. Every subsequent iteration step starts with a predictor which equals the corrector from the previous iteration step. The iteration is stopped when $|\text{maximum correction}| < 10^{-12}$.

The value of the relaxation factor ω is determined as follows. Let (f_1,f_2,f_3,f_4,f_5) be the exact solution of the (scalar version of) scheme (2.13), and let $\varepsilon_i^{(c)} = f_i^{(c)} - f_i$ and $\varepsilon_i^{(p)} = f_i^{(p)} - f_i$. We assume that $\partial K/\partial f$ is slowly varying, so $\partial K/\partial f \approx J$, J a constant. Since the predictor of the first iteration of (2.14) is the $O(h^5)$ solution of the scheme (2.11) for k = 6, we may write

$$\text{(2.15)}\qquad \begin{aligned} K(x_i,x_r,f_r^{(c)}) &\approx K(x_i,x_r,f_r) + \varepsilon_r^{(c)}J, \quad \text{and} \\ K(x_i,x_r,f_r^{(p)}) &\approx K(x_i,x_r,f_r) + \varepsilon_r^{(p)}J. \end{aligned}$$

Subtracting f_i from the left and the right hand side of (2.14) and multiplying with $1 - b_{ii}h\frac{\partial K}{\partial f}$ yields the error equation (writing $z = hJ$)

$$(1-b_{ii}z)\varepsilon_i^{(c)} = (1-b_{ii}z)\varepsilon_i^{(p)} - \omega[\varepsilon_i^{(p)} - z\sum_{r=1}^{i-1} b_{ir}\varepsilon_i^{(c)} - z\sum_{r=i}^{5} b_{ir}\,{}_i^{(p)}]$$

$$i = 1,2,3,4,5.$$

Writing $B = L + D + U$, where L is a lower triangular matrix, D is a diagonal matrix and U is an upper triangular matrix, this error equation reads in matrix notation

$$(I - zD - \omega zL)\varepsilon^{(c)} = [(1-\omega)(I-zD) + \omega zU]\varepsilon^{(p)},$$

where $\varepsilon^{(c)} = (\varepsilon_1^{(c)},\ldots,\varepsilon_5^{(c)})^T$ and $\varepsilon^{(p)} = (\varepsilon_1^{(p)},\ldots,\varepsilon_5^{(p)})$. In order to increase the rate of convergence of the nonlinear Gauss-Seidel iteration process, we have tried to find a value of ω, such that the spectral radius $\rho(\cdot)$ of the matrix

$$H(\omega,z) = (I - zD - \omega zL)^{-1}[(1-\omega)(I-zD) + \omega zU]$$

is minimal. We have $H(\omega,0) = (1-\omega)I$, hence $\rho(H(\omega,0)) = 1-\omega$. Furthermore,

$$\lim_{z\to-\infty} H(\omega,z) = (-D-\omega L)^{-1}[-(1-\omega)D + \omega H].$$

The following table gives $\rho(H(\omega,-\infty))$ for $\omega = 0(0.1)1.0$.

ω	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
$\rho(H)$	1	0.944	0.872	0.778	0.666	0.540	0.955	2.412	4.216	6.598	9.741

This suggests that $\rho(H(\omega,-\infty))$ is minimal for some value of $\omega \in [0.5,0.6]$. A more refined search shows that indeed $\rho(H(\omega,-\infty))$ is minimal for $\omega \approx 0.5782$, with value 0.4694. The following table gives $\rho(H(0.5782,z))$ for various values of $z \in (-\infty,0]$.

z	-10^6	-10^5	-10^4	-10^3	-10^2	-10	-1	0
$\rho(H)$	0.4694	0.4694	0.4693	0.4689	0.4645	0.4593	0.4707	0.4218

A more detailed search showed that certainly $\rho(H(0.5782,z)) < \frac{1}{2}$ for *all* $z \in (-\infty,0]$. Therefore, we have chosen $\omega = 0.578$ in (2.14), which guarantees convergence of the nonlinear Gauss-Seidel process for *all* values of $z = hJ \in (-\infty,0]$, provided, of course, that h is so small, that the extrapolation scheme (2.11) yields a sufficiently close initial approximation to justify the linearization (2.15).

3. THE IMPLICIT RUNGE-KUTTA METHOD OF DE HOOG AND WEISS

In section (4.4) we shall compare the efficiency of our sixth order BDF with that of a block-implicit Runge-Kutta method of de Hoog and Weiss ([4]) (which has comparable stability properties). The numerical scheme of this method reads as follows:

$$(3.1)\qquad f_{nj} = g(x_{nj}) + h \sum_{\ell=0}^{n-1} \sum_{i=1}^{s} c_{si} K(x_{nj}, x_{\ell i}, f_{\ell i}) + $$

$$+ h \sum_{i=1}^{s} c_{ji} K(x_{nj}, x_{ni}, f_{ni}),$$

$$j = 1,2,\ldots,s; \quad n = 0,1,2,\ldots .$$

Here $x_{nj} = x_n + u_j h$, where u_j, $j = 1,2,\ldots,s$, are s fixed numbers satisfying $0 \le u_1 < u_2 < \ldots < u_s = 1$; f_{nj} is the numerical approximation to $f(x_{nj})$ and

$$c_{ji} = \int_0^{u_j} L_i(t)dt, \qquad j,i = 1,2,\ldots,s,$$

where $L_i(t)$ is the Lagrange polynoom $\Pi_{j=1,j\neq i}^{s}$ $(t - u_j)/(u_i - u_j)$. For $n = 0,1,2,\ldots$ (3.1) is a system of s (nonlinear) equations in $f_{n1},\ldots,f_{ns}$. De Hoog and Weiss solve these with Newton iteration.

For our comparison scheme we have chosen $s = 4$, where u_1,u_2,u_3 and u_4 are the so-called *Radau points*, viz., $u_1 = 0.08858795951270$, $u_2 = 0.40946686444074$, $u_3 = 0.78765946176085$ and $u_4 = 1.0$. According to de Hoog and Weiss this scheme has global order of convergence $O(h^7)$, and it is stiffly A-stable.

4. NUMERICAL EXPERIMENTS

In this section we shall describe the results of numerical experiments, in order to test the convergence and stability results of the BDFs proved in [5] and in order to compare the performance of the sixth order BDF with that of the scheme of de Hoog and Weiss, described in section 3. We shall measure the accuracy (ACC) of a given method for a given test problem for a given stepsize h by the minimum number of correct significant digits (in absolute sense) in the computed solution, i.e.,

$$\mathrm{ACC} = \min_i\{-{}^{10}\log|f(x_i) - f_i|\}.$$

When we compare two methods (cf. section 4.3) we shall also measure the computational effort (CEF) by the ${}^{10}\log$ of the total number of K-evaluations needed to complete the computation of $f(x_e)$, including those for the numerical computation of $\partial K/\partial f$, i.e.,

$$\mathrm{CEF} = {}^{10}\log(\#\text{K-evaluations}).$$

All calculations have been carried out on a CDC Cyber 73/173-28 computer, using 14 significant digits.

4.1. Convergence tests

In [5] we proved that the BDF given in the scheme (2.2) has global order of convergence $O(h^k)$, as $h \to 0$, provided that for the quadrature weights in (2.3) one chooses a Gregory formula with k-2 correction terms. We shall test this result with two linear problems and one nonlinear problem.

Problem 4.1.

$$f(x) = 1 + x - \cos(x) - \int_0^x \cos(x-y)f(y)dy, \qquad 0 \le x \le 2,$$

with exact solution $f(x) = x$. The results are given in table 4.1.

h	k = 2	k = 3	k = 4	k = 5	k = 6
1/4	2.0	2.5	3.5	4.0	5.3
1/8	2.5	3.5	4.5	5.5	6.6
1/16	3.1	4.4	5.7	7.0	8.3
1/32	3.6	5.3	6.9	8.6	10.1
1/64	4.2	6.2	8.1	10.1	11.9

Table 4.1. ACC for problem 4.1.

Problem 4.2. (Renewal equation from FELLER [2])

$$f(x) = \frac{1}{2}x^2e^{-x} + \frac{1}{2}\int_0^x (x-y)^2e^{-(x-y)}f(y)dy, \qquad 0 \le x \le 2,$$

with exact solution $f(x) = \frac{1}{3} - \frac{1}{3}e^{-3x/2}\{\cos(\frac{1}{2}x\sqrt{3}) + \sqrt{3}\sin(\frac{1}{2}x\sqrt{3})\}$. The results are given in table 4.2.

h	k = 2	k = 3	k = 4	k = 5	k = 6
1/4	2.0	2.3	2.9	3.2	3.8
1/8	2.7	3.3	4.3	4.7	5.5
1/16	3.4	4.2	5.6	6.2	7.2
1/32	4.0	5.1	6.9	7.7	9.0
1/64	4.6	6.0	8.1	9.2	10.7

Table 4.2. ACC for problem 4.2.

Problem 4.3.

$$f(x) = (\sin\pi x + e^{-x})^{1/5} + \frac{1-\cos(\pi x)}{\pi} + 1 - e^{-x} + \int_0^x (f(y))^5 dy,$$

$$0 \le x \le 2,$$

with exact solution $f(x) = (\sin \pi x + e^{-x})^{1/5}$. The results are given in table 4.3.

h	k = 2	k = 3	k = 4	k = 5	k = 6
1/4	1.3	1.6	1.8	2.0	2.6
1/8	1.9	2.4	2.9	3.4	3.9
1/16	2.5	3.3	4.1	4.8	5.6
1/32	3.1	4.1	5.3	6.3	7.5
1/64	3.7	5.0	6.6	7.8	9.3

Table 4.3. ACC for problem 4.3.

Remark. Both for the linear problems and for the nonlinear problem, the numerical results confirm that the global order of convergence of the BDFs is $O(h^k)$, as $h \to 0$.

4.2. Stability tests

In [5] we have given a local stability theory for kernels K of the form

$$K(x,y,f) = A(y,f) + xHf,$$

where A is an arbitrary function of y and f, and H is an arbitrary constant. We have displayed there the regions in the (z,u)-plane, $z = h \frac{\partial K}{\partial f}(x_{n+1}, x_{n+1}, f_{n+1})$, $u = h^2 \frac{\partial^2 K}{\partial x \partial f}(x_{n+1}, x_{n+1}, f_{n+1})$, where the amplification matrix of the variational equation has spectral radius ≤ 1. We shall test these results with one linear and one nonlinear problem.

Problem 4.4. (cf. WOLKENFELT [8]).

$$f(x) = 50x + \frac{1}{4} - 49\sin(x) - \frac{1}{4}\cos(x) + \int_0^x \{50(y-x) - \frac{1}{4}\}f(y)dy,$$

$$0 \le x \le 10,$$

with exact solution $f(x) = \sin(x)$. For this problem we have $z = -\frac{1}{4}h$ and $u = -50h^2$. In table 4.4a we indicate by S and I, respectively, whether or not the point (z,u) belongs to the stability region, as given in [5].

h	k = 2	k = 3	k = 4	k = 5	k = 6
1/2	S	S	I	I	I
1/4	S	I	I	I	I
1/8	S	I	I	S	S
1/16	S	S	S	S	S
1/32	S	S	S	S	S

Table 4.4a. Theoretical stability behaviour of the BDFs with respect to problem 4.4.

In table 4.4b we give the numerical results.

h	k = 2	k = 3	k = 4	k = 5	k = 6
1/2	0.3	-0.6	-2.2	*	*
1/4	1.2	1.1	-1.1	-2.7	*
1/8	1.6	2.2	1.8	3.1	4.7
1/16	2.1	3.7	4.2	4.3	7.0
1/32	2.6	4.7	5.6	5.7	8.9

Table 4.4b. ACC for problem 4.4

Remark. In the cases k = 5 and k = 6 the step $h = \frac{1}{2}$ was too large, so that the Newton process for the computation of f_4 in (2.11) did not converge. In the case k = 6, $h = \frac{1}{4}$, the Newton process for the solution of (2.11) did converge, but now the nonlinear Gauss-Seidel process for the solution of (2.13) did not. When switching off the starting procedure in these three cases, and using *exact* starting values, the numerical results clearly showed an unstable behaviour. Therefore, we conclude that the linear stability theory is confirmed by the numerical results.

Problem 4.5.

$$f(x) = -15x + 17(e^{x} - 1) + \int_{0}^{x} [16(y - x) - 1]e^{f(y)}dy, \qquad 0 \le x \le 10,$$

with exact solution $f(x) = x$. For this nonlinear problem we have $u \approx 16hz$. Using the stability regions as given in [5] we have constructed the following tentative S/I-table (assuming that the linear stability theory remains valid for this nonlinear problem):

h	k = 2	k = 3	k = 4	k = 5	k = 6
1/2	S	I	I	I	I
1/4	S	S	I	I	I
1/8	S	S	S	S/I	I
1/16	S	S	S	S	S
1/32	S	S	S	S	S

Table 4.5. Tentative stability behaviour of the BDFs with respect to problem 4.5.

The numerical results are given in table 4.5b

h	k = 2	k = 3	k = 4	k = 5	k = 6
1/2	0.7	0.5	0.3	*	*
1/4	1.2	1.9	2.4	2.5	*
1/8	1.7	2.6	3.9	4.1	4.4
1/16	2.3	3.5	5.0	5.3	7.4
1/32	2.9	4.4	6.1	6.7	9.2

Table 4.5b. ACC for problem 4.5.

Remark. Again, in the three cases $k = 5,6$, $h = \frac{1}{2}$ and $k = 6$, $h = \frac{1}{4}$, h was too large, so that the starting procedure did not converge. When leaving out the starting procedure, again an unstable behaviour of the numerical results was noticed. Hence, we conclude that the linear stability theory correctly predicts the stability behaviour of the BDFs, when applied to the nonlinear problem 4.5.

4.3. Comparison with the method of de Hoog and Weiss.

In this section we compare the highest order BDF (i.e. the one with $k = 6$) with a block-implicit Runge-Kutta method of de Hoog and Weiss with four Radau points, as given in section 3. In order to make this comparison as fair as possible, we give accuracy/efficiency plots of both numerical schemes. For a number of different values of h the quantities ACC and CEF (defined in the introduction of this section) are set out graphically and connected by a drawn line for the BDF and by a dotted line for the scheme of de Hoog and Weiss. When measuring CEF for the scheme of de Hoog and Weiss, we have *neglected* the amount of work necessary to solve the four linear equations in every Newton iteration step for the solution of (3.1).

Problem 4.6.

$$f(x) = \sin(x) + \lambda(1 - \cos(x)) - \lambda \int_0^x f(y)dy, \qquad 0 \le x \le 10,$$

$$\lambda = 1, 10, 100, 1000,$$

with exact solution $f(x) = \sin(x)$. For the four different values of λ the results of the sixth order BDF are almost the same, i.e., the accuracy/efficiency plot of this scheme, when applied to problem 4.6, is hardly affected by the value of λ. The results are given in table 4.6a.

h	CEF	ACC(all λ)
1/4	3.47	4.5
1/8	3.83	6.3
1/16	4.27	8.1
1/32	4.80	9.9

Table 4.6a. CEF - ACC values for the sixth order BDF, applied to problem 4.6.

The results of the scheme of de Hoog and Weiss, however, depend very strongly on the value of λ, as can be seen from table 4.6b.

h	CEF	ACC			
		$\lambda = 1$	$\lambda = 10$	$\lambda = 100$	$\lambda = 1000$
1	3.08	6.1	3.7	3.1	3.0
1/2	3.60	8.2	5.4	4.4	4.2
1/4	4.16	10.3	7.3	5.7	5.5
1/8	4.74	12.4	9.3	7.2	6.7

Table 4.6b. CEF - ACC values for the scheme of de Hoog and Weiss with four Radau points, applied to problem 4.6.

In figure 4.6 we have combined the results of tables 4.6a and 4.6b in an accuracy/efficiency plot.

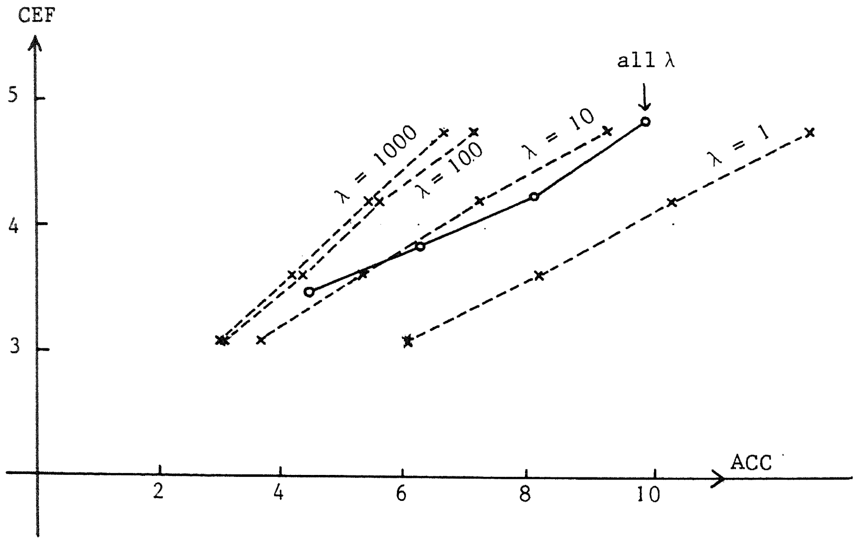

Figure 4.6. Accuracy/efficiency plots of the sixth order BDF (drawn lines) and the scheme of de Hoog and Weiss with four Radau points (dotted lines), both applied to problem 4.6.

Remarks. Table 4.6b and figure 4.6 indicate that the order of convergence of the scheme of de Hoog and Weiss decreases from the expected value $O(h^7)$ in the case $\lambda = 1$ to about $O(h^4)$ in the case $\lambda = 1000$, and that, consequently, at least for $\lambda > 10$ the sixth order BDF is more efficient than the scheme of de Hoog and Weiss. The decrease in order of the scheme of de Hoog and Weiss may be explained by the fact that the error "constant" in the expansion of the error satisfies a linear Volterra integral equation of the second kind, *with kernel* $\partial K/\partial f$ (cf. [4], theorem 4.1). So when $\lambda = -\partial K/\partial f$ increases, one has to decrease the step h in order to preserve the order of convergence.

Problem 4.7.

$$f(x) = \{1 + (1+x)e^{-10x}\}^{\frac{1}{2}} + \frac{\lambda}{10}(1+x)\{10 \log(1+x) + 1 - e^{-10x}\}$$
$$- \lambda(1+x) \int_0^x \frac{f^2(y)}{1+y}\, dy, \qquad 0 \le x \le 10, \quad \lambda = 1,10,100,$$

with exact solution $f(x) = \{1 + (1+x)\exp(-10x)\}^{1/2}$. The case $\lambda = 10$ is treated by de Hoog and Weiss in [4], as an example for which *variable* step size is suitable (because of the relatively large derivative of the solution f(x) near x = 0). Since our BDF only uses *fixed* step size, we shall only compare the two schemes with fixed step size h.

The results are displayed in figure 4.7. They were obtained for the BDF with step sizes h = 1/4,1/8,1/16,1/32, and for the scheme of de Hoog and Weiss with step sizes h = 1,1/2,1/4 and 1/8.

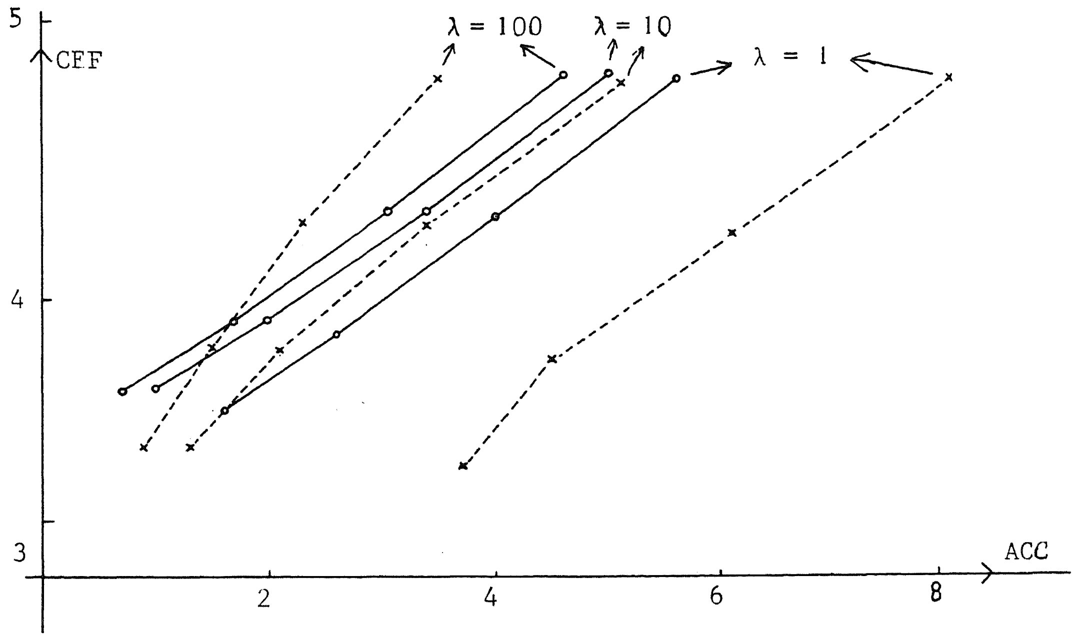

Figure 4.7. Accurracy/efficiency plots of the sixth order BDF (drawn lines) and the scheme of de Hoog and Weiss with four Radau points (dotted lines), both applied to problem 4.7.

Remarks. Again, the strong dependency of the efficiency of the scheme of de Hoog and Weiss on λ is obvious. For $\lambda = 1$ the scheme of de Hoog and Weiss is more efficient then the BDF, whereas for $\lambda = 100$ we observe the opposite behaviour.

5. CONCLUDING REMARKS

The results of the numerical experiments with backward differentiation formulas for Volterra integral equations of the second kind (including the starting schemes (2.11) and (2.13)) support the convergence and stability theory, as given in [5]. Moreover, comparison of our formulas with a scheme of de Hoog and Weiss (which has comparable stability properties) shows that for *scalar* equations with a kernel K for which $|\partial K/\partial f|$ is large, the sixth order BDF is more efficient than the scheme of de Hoog and Weiss. Although we have not yet implemented and tested our formulas for *vector* equations, it is to be expected that the sixth order BDF is also more efficient for this type of equations, since the BDF-scheme (2.2) and the starting schemes (2.11) and (2.13) are only implicit in the unknown vector value $\vec{f}_{n+1}$, whereas the scheme (3.1) of de Hoog and Weiss is implicit in a block of s unknown vector values $\vec{f}_{n1}, \vec{f}_{n2}, \ldots, \vec{f}_{ns}$.

REFERENCES

[1] ABRAMOWITZ, M. & I.A. STEGUN (eds.), *Handbook of mathematical functions*, Nat. Bureau of Standards, Washington, D.C., 1964.

[2] FELLER, W., *On the integral equation of renewal theory*, Ann. Math. Stat., 12 (1941) 243-267.

[3] HENRICI, P., *Discrete variable methods in ordinary differential equations*, John Wiley and Sons, New York, 1962.

[4] HOOG, F. DE & R. WEISS, *Implicit Runge-Kutta methods for second kind Volterra integral equations*, Numer. Math., 23 (1975) 199-213.

[5] HOUWEN, P.J. VAN DER & H.J.J. TE RIELE, *Backward differentiation formulas for Volterra integral equations of the second kind, I Convergence and stability*, Report NW 48/77, September 1977, Mathematisch Centrum, Amsterdam.

[6] ORTEGA, J. & W. RHEINBOLDT, *Nonlinear equations in several variables*, Academic Press, 1970.

[7] STROUD, A.H., *Numerical quadrature and solution of ordinary differential equations*, Springer-Verlag, Berlin, 1974.

[8] WOLKENFELT, P., *Backward differentiation formulas for Volterra integro-differential equations*, Report NW 53/77, November 1977, Mathematisch Centrum, Amsterdam.