LINEAR MULTISTEP METHODS FOR VOLTERRA INTEGRAL EQUATIONS OF THE SECOND KIND

P.J. van der Houwen* and H.J.J. te Riele*

* *Mathematical Centre, Amsterdam*

1. INTRODUCTION

The simplest linear multistep (LM) method for solving the Volterra integral equation of the second kind

$$y(t) = g(t) + \int_{t_0}^{t} K(t,\tau,y(\tau))d\tau, \quad t \in I:=[t_0,T], \tag{1.1}$$

is obtained by writing down this equation in a sequence of equidistant points

$$t_n := t_0 + nh, \; n = 0(1)N \; (h \text{ fixed and } t_N = T) \tag{1.2}$$

and by approximating the integral term by some suitably chosen quadrature formula. Such a method is called a *direct quadrature* (DQ) method for (1.1). Recently, several other LM methods for solving (1.1) have been proposed (cf. the *indirect backward differentiation* method in [5] and the *multilag* and *modified multilag* methods in [9](see also[12]).

In this paper a general class of linear multistep methods is presented which includes all these methods, and many others (Section 2). This enables us to give a uniform treatment of the problems of consistency (Section 3), of convergence (Section 4) and of stability (Section 6). Since the ordinary differential equation $dy/dt = f(t,y)$, $y(t_0)=y_0$, is a special case of the differentiated version of (1.1), the relation with linear multistep methods for ODEs is analyzed and fixed terms recurrence relations are derived for a class of convolution kernels(Section). Finally, two numerical experiments are reported (Section 7).

Space prevents us including the detailed proofs of the theorems presented here. These may be found in [6]. A number of additional numerical experiments which support and confirm the theory, may also be found in [6].

The work presented here can easily be extended to Volterra integral equations of the *first* kind, and to Volterra integro-differential equations (cf.[6]).

2. A GENERAL CLASS OF LM METHODS FOR SOLVING (1.1)

Let us associate with (1.1) the so-called *lag* term

$$Y(t,s) := g(t) + \int_{t_0}^{s} K(t,\tau,y(\tau))d\tau \tag{2.1}$$

for $(t,s) \in S := \{(t,s): t_0 \le s \le t \le T\}$. Note that $Y(t,t) \equiv y(t)$. Let y_n and $Y_n(t)$ denote numerical approximations to $y(t_n)$ and to $Y(t,t_n)$, respectively, and let

$$K_n(t) := K(t,t_n,y_n),\ n \ge 0. \tag{2.2}$$

Usually, $Y_n(t)$ will be computed by a quadrature formula of the form

$$Y_n(t) = g(t) + h \sum_{j=0}^{n} w_{n,j} K_j(t),\ n \ge n_0\ , \tag{2.3}$$

where the $w_{n,j}$ are given weights and n_0 is sufficiently large to ensure the required order of accuracy. We assume that this quadrature formula is of order r, i.e.,

$$E_n(h;t) := \int_{t_0}^{t_n} K(t,\tau,y(\tau))d\tau - h \sum_{j=0}^{n} w_{n,j} K(t,t_j,y(t_j)) = \mathcal{O}(h^r) \tag{2.4}$$

as $h \to 0$, $n \to \infty$, with $t_n = t_0 + nh$ fixed. Our general LM *method* for (1.1) consists of the quadrature *formula* (2.3) and the LM *formula*

$$\sum_{i=0}^{k} \alpha_i y_{n-i} + \sum_{i=0}^{k} \sum_{j=-k}^{k} \beta_{ij} Y_{n-i}(t_{n+j}) = h \sum_{i=0}^{k} \sum_{j=-k}^{k} \gamma_{ij} K_{n-i}(t_{n+j}),\ n = k(1)N,\ k \text{ fixed}, \tag{2.5}$$

where the parameters α_i, β_{ij} and γ_{ij}, $i=0(1)k$, $j=-k(1)k$, are to be prescribed. From this scheme the quantities $y_k, y_{k+1}, \ldots, y_N$ can be computed successively. The quantities $y_1, \ldots, y_{k-1}$ and $Y_1(t), \ldots, Y_{k-1}(t)$ are assumed to be precomputed by some starting method. Since the kernel $K(t,\tau,y)$ is not necessarily defined outside S, we usually require (cf. Figure 1) that $\beta_{ij} = \gamma_{ij} = 0$, for $j < -i$.

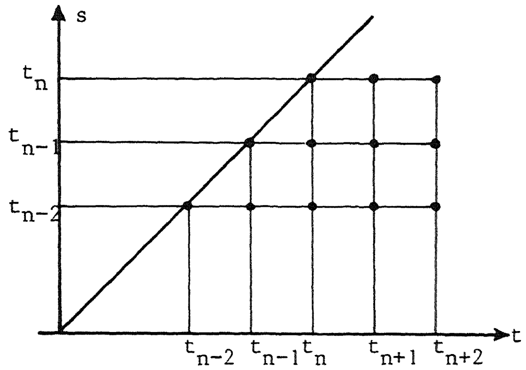

FIG. 1. Points in the (t,s)-plane needed in (2.5) for k = 2

Furthermore, it will be assumed that the points t_j are equally spaced (cf. (1.2)) although most of the analysis can be carried through for non-uniform spacing (compare a similar situation in the analysis of LM methods for ODEs). It is convenient to characterize the formula (2.5) by the matrices

$$A = (\alpha_i),\ B = (\beta_{ij}),\ C = (\gamma_{ij}) \tag{2.6}$$

where the row index i assumes the values 0(1)k and the column index j the values -k(1)k. We now describe four subclasses of (2.5) from which we will borrow several illustrating examples in this paper.

2.1 Direct quadrature methods

Consider the LM formula defined by the (1x1) matrices

$$A = 1,\ B = -1,\ C = 0, \text{ for which (2.5) reduces to} \quad y_n = Y_n(t_n). \tag{2.7}$$

Evidently, this is the *direct quadrature* (DQ) method, described in the introduction.

2.2 Indirect linear multistep methods

We formally derive this subclass by applying a linear multistep method for ODEs with coefficients α_i and γ_i, i=0(1)k, to the differentiated version of (1.1) (cf. [5]) :

$$y'(t) = K(t,t,y(t)) + Y_t(t,t), \tag{2.8}$$

where $Y_t(t,t)$ denotes the partial derivative of $Y(t,s)$ with respect to its first variable t, in the point (t,t). This yields the scheme

$$\sum_{i=0}^{k} \alpha_i y_{n-i} = h \sum_{i=0}^{k} \gamma_i K_{n-i}(t_{n-i}) + h \sum_{i=0}^{k} \gamma_i Y_t(t_{n-i}, t_{n-i}) , \quad n \geq k. \tag{2.9}$$

Now we approximate the derivative Y_t of Y by the k-step *forward differentiation* formula (cf. [1, Table 25.2])

$$Y_t(t_{n-i}, t_{n-i}) \approx - \frac{1}{h} \sum_{\ell=0}^{k} \delta_\ell Y(t_{n+\ell-i}, t_{n-i}). \tag{2.10}$$

Using (2.3) we obtain

$$\sum_{i=0}^{k} \alpha_i y_{n-i} + \sum_{i=0}^{k} \sum_{j=-i}^{k-i} \gamma_i \delta_{i+j} Y_{n-i}(t_{n+j}) = h \sum_{i=0}^{k} \gamma_i K_{n-i}(t_{n-i}), \quad n \geq k, \tag{2.11a}$$

or,equivalently, the generating matrices

$$A=\begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_k \end{pmatrix}, \quad B=\begin{pmatrix} 0 & \gamma_0\delta_0 & \gamma_0\delta_1 & \cdots & \gamma_0\delta_k \\ & \gamma_1\delta_0 & \gamma_1\delta_1 & \cdots & \gamma_1\delta_k \\ & \iddots & \iddots & & \iddots \\ \gamma_k\delta_0 & \gamma_k\delta_1 & \cdots & \gamma_k\delta_k & 0 \end{pmatrix}, \quad C=\left(\begin{array}{cc|c} 0 & \gamma_0 & \\ & \gamma_1 & \\ & \iddots & 0 \\ \gamma_k & 0 & \end{array}\right). \tag{2.11b}$$

These matrices generate an *indirect linear multistep* (ILM) method. When the α_i and γ_i are the coefficients of a backward differentiation method, (2.11) represents the IBD (indirect backward differentiation) method, analyzed in [5]. We notice that for this IBD method we have $\gamma_i=0$, $i=1(1)k$, and $\gamma_0\delta_j=\alpha_j$, $j=0(1)k$.

2.3 Multilag methods

In Wolkenfelt et.al. [12] we find methods which can be characterized by the matrices

$$A=\begin{pmatrix} \alpha_0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad B=\begin{pmatrix} & 0 & \\ & \alpha_1 & \\ 0 & \vdots & 0 \\ & \alpha_k & \end{pmatrix}, \quad C=\begin{pmatrix} & \gamma_0 & \\ & \gamma_1 & \\ 0 & \vdots & 0 \\ & \gamma_k & \end{pmatrix}. \tag{2.12}$$

Here, the α_i and γ_i, $i=0(1)k$, may be the coefficients of any LM method for ODEs. If the lag term $Y_n(t)$ is computed by using a quadrature rule which is (α,γ)-reducible (see Section 2.5), then the resulting method turns out to be equivalent to a DQ method based on the same (α,γ)-reducible quadrature rule (provided, of course, that the starting values are identical). Thus, a different implementation of the same method was used for the stability analysis of DQ methods. However, as was pointed out by Wolkenfelt [9], this implementation requires a lot of additional arithmetic operations and, although suitable for theorerical analysis, it is not recommendable in actual computations. In order to avoid this disadvantage, he proposed to compute the lag term simply by a quadrature rule of the form (2.3) to obtain the *multilag* (ML) methods.

2.4 *Modified multilag methods*

In [9] Wolkenfelt also introduced a modification of the ML methods, viz., the so-called *modified multilag* (MML) methods, characterized by the matrices

$$A=\begin{pmatrix}\alpha_0\\ \alpha_1\\ \vdots\\ \alpha_k\end{pmatrix},\quad B=\begin{pmatrix} & & & 0 & \\ 0 & & -\alpha_1 & \alpha_1 & \\ & \cdot^{\cdot^{\cdot}} & & \vdots & 0\\ \cdot^{\cdot^{\cdot}} & 0 & & \vdots & \\ -\alpha_k & & & \alpha_k & \end{pmatrix},\quad C=\begin{pmatrix} & \gamma_0 & \\ & \gamma_1 & \\ 0 & \vdots & 0\\ & \vdots & \\ & \gamma_k & \end{pmatrix}. \tag{2.13}$$

The α_i and γ_i are, again, the coefficients of any LM method for ODEs.

2.5 *The quadrature weights of the lag term*

In order to define a specific LM method for (1.1) we have not only to specify the generating matrices A, B and C, but also the quadrature weights $w_{n,j}$ in (2.3). An important family of quadrature formulas, including the well-known Gregory quadrature formulas, are the so-called *reducible quadrature formulas* [8]. The weights $w_{n,j}$ in such formulas are recursively defined by the equations

$$\sum_{i=0}^{k^*} a_i w_{n-i,j} = \begin{cases} 0 & \text{if } j = 0(1)n-k^*-1 \\ b_{n-j} & \text{if } j = n-k^*(1)n \end{cases}, \quad n=k^*,k^*+1,\ldots, \tag{2.14}$$

where the a_i and b_i, $i=0(1)k^*$, are the coefficients of some given LM method for ODEs. Here, we define $w_{n,j}=0$ for $j>\max(n,k^*-1)$, and the "starting weights" $w_{n,j}$, $0\le n,j \le k^*-1$ are assumed to be prescribed.

Defining the characteristic polynomials

$$\rho(z) := \sum_{i=0}^{k^*} a_i z^{k^*-i}, \quad \sigma(z) := \sum_{i=0}^{k^*} b_i z^{k^*-i}, \tag{2.15}$$

the quadrature formulas generated by (2.14) are said to be (ρ,σ)-*reducible*. We note that the characteristic polynomials of the Adams-Moulton methods generate the weights of the Gregory formulas. The backward differentiation methods generate rather unconventional quadrature rules, which were analysed in [11].

3. CONSISTENCY OF THE LM FORMULA (2.5)

Let us associate with the LM formula (2.5) the difference-differential operator L_n defined by

$$L_n(Y) := \sum_{i=0}^{k} \Big\{\alpha_i Y(t_{n-i},t_{n-i}) + \sum_{j=-k}^{k} [\beta_{ij} - \gamma_{ij} h \frac{\partial}{\partial s}]\, Y(t_{n+j},t_{n-i})\Big\}, \tag{3.1}$$

where $Y(t,s)$ is an arbitrary function, differentiable with respect to s on $t_0 \le s \le T$. As in the case of LM methods for ODEs, the operator L_n is introduced in order to operate on test functions Y of sufficient differentiability (cf. e.g. Lambert [7,p.23]). Unlike the ODE case, the relation of the operator L_n with the LM formula (2.5) is not immediate, and needs some explanation. Suppose that $Y(t,s)$ is defined by (2.1) with $y(t)$ the exact solution of (1.1). Observing that $Y(t,t)=y(t)$ and $\frac{\partial Y}{\partial s}(t,s)=K(t,s,y(s))$, and using (2.4) we find, on substitution of $y(t)$ into (2.5), the equation

$$\sum_{i=0}^{k} \Big\{\alpha_i y(t_{n-i}) + \sum_{j=-k}^{k} [\beta_{ij} Y_{n-i}(t_{n+j}) - h\gamma_{ij} K_{n-i}(t_{n+j})]\Big\} = L_n(Y) - \sum_{i=0}^{k}\sum_{j=-k}^{k} \beta_{ij} E_{n-i}(h;t_{n+j}). \tag{3.2}$$

Thus, the exact solution of (1.1) satisfies the method {(2.3)-(2.5)} apart from the residual terms in the right-hand side of (3.2). In this section, we concentrate on the first residual term.

Definition 3.1 The operator (3.1) and the associated LM formula (2.5) are said to be *consistent of order p* if for all $Y \in C^{p+1}[S]$, $L_n(Y)=O(h^{p+1})$ as $h \to 0$ with nonvanishing error constant. If Y corresponds to the theoretical solution of (1.1), then $L_n(Y)$ will be called the *local truncation error* of (2.5). ⊠

The following theorem provides the consistency conditions in terms of the parameters α_i, β_{ij} and γ_{ij}.

Theorem 3.1 The operator (3.1) and the associated LM formula (2.5) are consistent of order p if

$$\sum_{i=0}^{k} [(-i)^q \alpha_i - \sum_{j=-k}^{k} j^{q-\ell}(-i)^{\ell-1}(i\beta_{ij} + \ell\gamma_{ij})] =: C_{q\ell} = 0 \quad (3.3)$$

for $q = 0(1)p$ and $\ell = 0(1)q$ (with $(-i)^{\ell-1}\ell := 0$ if $\ell=i=0$). ⊠

Corollary 3.1 Let $\tilde{p}$ be the order of consistency of the LM method for ODEs defined by the coefficients $\{\alpha_i, \gamma_i\}$ employed in the sections 2.2, 2.3 and 2.4. Then the order of consistency p of the LM formula (2.5) for (1.1) is given by $p = \infty$ for the DQ method, $p = \min\{k, \tilde{p}\}$ for the ILM method, and $p = \tilde{p}$ for both the ML method and the MML method. ⊠

If the LM formula (2.5) is consistent of order p, then the local truncation error $L_n(Y)$ can be expressed in terms of the constants defined in (3.3) as follows:

$$L_n(Y) = h^{p+1} \sum_{\ell=0}^{p+1} C_{p+1,\ell} \binom{p+1}{\ell} (\tfrac{\partial}{\partial t})^{p+1-\ell} (\tfrac{\partial}{\partial s})^{\ell} Y(t,s)|_{t=s=t_n} + O(h^{p+2}) \text{ as } h \to 0. \quad (3.4)$$

It is of some interest now to compare the values of the error constants $C_{p+1,\ell}$, $\ell=0(1)p+1$, for the various subclasses given in Section 2. We have evaluated and simplified the expressions for these constants as much as possible:
For the ILM method Corollary 3.1 gives p=k, under the (reasonable) assumption that $\tilde{p} \ge k$. We then find

$$C_{p+1,\ell} = (-1)^{p-1} \sum_{i=0}^{k} [i^p\{i\alpha_i + (p+1)\gamma_i\} - R] , \quad (3.5)$$

where $R = k!\gamma_i$ if $\ell = 0$ and $R = 0$ if $\ell = 1(1)p+1$.
For both the ML and MML method we find

$$C_{p+1,\ell} = \begin{cases} 0, & \text{if } \ell = 0(1)p, \\ (-1)^{p-1} \sum_{i=0}^{k} i^p\{i\alpha_i + (p+1)\gamma_i\}, & \text{if } \ell = p+1. \end{cases} \quad (3.6)$$

We have computed the numerical values of the error constants for two usual choices of the coefficients $\{\alpha_i, \gamma_i\}_{i=0}^{k}$, viz., the *backward differentiation* (BD) method, for which $\tilde{p}$=k, and the *Adams-Moulton* (AM) method for which $\tilde{p}$=k+1. Table 1 gives the values of the relevant constants $C_{p+1,\ell}$ where p is prescribed by Corollary 3.1. Note that the (M)ML-AM methods have p=k+1, whereas the other methods have p=k.

TABLE 1

Error constants in (3.4) for various choices of $\{\alpha_i,\gamma_i\}$ *in (2.5)*

method	$\{\alpha_i,\gamma_i\}$		k=1	k=2	k=3	k=4	k=5
ILM	BD	$C_{k+1,0}$	-2	0	-72/11	0	-14400/1
		$C_{k+1,>0}$	-1	-4/3	-36/11	-288/25	-7200/1
	AM	$C_{k+1,0}$	-1	2	-6	24	-120
		$C_{k+1,>0}$	0	0	0	0	0
ML & MML	BD	$C_{k+1,<k+1}$	0	0	0	0	0
		$C_{k+1,k+1}$	-1	-4/3	-36/11	-288/25	-7200/1
	AM	$C_{k+2,<k+2}$	0	0	0	0	0
		$C_{k+2,k+2}$	-1/2	-1	-19/6	-27/2	-863/12

The *order of convergence* of the LM method is dictated not only by its order of consistency, but also, of course, by the quadrature error (2.4) and by the errors in the starting values $y_1,\ldots,y_{k-1}$. In the next Section we shall analyze the convergence of the LM method $\{(2.3)-(2.5)\}$.

4. CONVERGENCE

Similarly as with LM methods for ODEs, a *necessary condition* for convergence of the LM method $\{(2.3)-(2.5)\}$ is that the characteristic polynomial

$$\alpha(z) := \sum_{i=0}^{k} \alpha_i z^{k-i} \tag{4.1a}$$

satisfies the *root condition*, i.e., its roots are on the unit disk, those on the unit *circle* being simple.

In the *sufficient conditions* for convergence the parameters β_{ij} and γ_{ij} are also involved. We define

$$\beta(z) := \sum_{i=0}^{k} \beta_i z^{k-i}, \quad \beta_i := \sum_{j=-k}^{k} \beta_{ij}; \tag{4.1b}$$

$$\gamma(z) := \sum_{i=0}^{k} \gamma_i z^{k-i}, \quad \gamma_i := \sum_{j=-k}^{k} \gamma_{ij}. \tag{4.1c}$$

Furthermore, we will use the notation

$$\Delta K_n(t) = K(t,t_n,y(t_n)) - K(t,t_n,y_n),$$

$$\Delta E_n(h) = \max_{\substack{i\le j\le n\\ \ell\le k}} |E_i(h;t_{j+\ell}) - E_i(h;t_j)|,$$

$$E_n(h) = \max_{i\le j\le n} |E_i(h;t_j)|,$$

$$T_n(h) = \max_{i\le n} |L_i(Y)| \quad \text{and} \quad \delta(h) = \max_{j\le k-1} |y(t_j) - y_j|.$$

$E_n(h)$ is the maximal error arising in the approximation of the lag terms $Y(t,t_n)$ by $Y_n(t)$ (cf. (2.4)). $T_n(h)$ may be considered as the maximal local truncation error of the LM formula, and $\delta(h)$ is the maximal starting error. We now formulate a general convergence theorem which provides an estimate for the *global error*

$$\varepsilon_n = y(t_n) - y_n. \tag{4.2}$$

We assume that K satisfies the Lipschitz conditions

$$|\Delta K_\ell(t)| \le L_1|\varepsilon_\ell| \text{ and } |\Delta K_\ell(t) - \Delta K_\ell(t^*)| \le L_2|t-t^*||\varepsilon_\ell|,$$

where L_1 and L_2 are the Lipschitz constants.

Theorem 4.1 Let $\alpha(z)$ satisfy the root condition.

(i) If $\alpha(z) = \alpha_0 z^k$ then there exists a constant $C > 0$ such that

$$|\varepsilon_n| \le C\ [h\delta(h) + E_N(h) + \Delta E_N(h) + T_N(h)],\ n=k(1)N. \tag{4.3a}$$

(ii) If $\beta(z) \equiv 0$ then there exists a constant $C > 0$ such that

$$|\varepsilon_n| \le C\ [\delta(h) + h^{-1}\{\Delta E_N(h) + T_N(h)\}],\ n=k(1)N. \tag{4.3b}$$

☒

Now it is easy to derive the following

Corollary 4.1 Let $\delta(h) = O(h^q)$, $E_N(h) = O(h^r)$ (as in (2.4)), $\Delta E_N(h) = O(h^{r+1})$ as $h \to 0$ and let $\{\alpha_i,\gamma_i\}$ in (2.5) be the coefficients of a $\tilde{p}$ - th order consistent LM method for ODEs. Then the order of convergence p^* of the LM method for (1.1) is given by: $p^* = \min(q+1,r)$ for the DQ method, $p^* = \min(q,r,p)$ for the ILM method, $p^* = \min(q+1,r,p+1)$ for the ML method and $p^* = \min(q,r,p)$ for the MML method, where p is the order of consistency of (2.5), given by Corollary 3.1. ☒

5. RELATION WITH LM METHODS FOR ODEs

The Volterra equation (1.1) contains the classes of ordinary differential equations as special cases. For example, if in (1.1) $g(t) \equiv$ constant then

$$K(t,\tau,y) = f(\tau,y) \quad \to \frac{dy}{dt} = f(t,y) \tag{5.1a}$$

, etc.

$$K(t,\tau,y) = (t-\tau)f(\tau,y) \to \frac{d^2y}{dt^2} = f(t,y) \tag{5.1b}$$

Therefore, it is natural to ask to what method the LM formula (2.5) reduces when it is applied to the special cases (5.1). Furthermore, one may ask for the relationship with LM methods for ODEs of the form(5.1). In order to formulate this relationship we introduce, in addition to the polynomials $\alpha(z)$, $\beta(z)$ and $\gamma(z)$ (cf. (4.1)), the polynomials

$$\bar{\beta}(z) := \sum_{i=0}^{k} \bar{\beta}_i z^{k-i} \ , \ \bar{\beta}_i := \sum_{j=-k}^{k} j\beta_{ij} \ ; \tag{5.2a}$$

$$\bar{\gamma}(z) := \sum_{i=0}^{k} \bar{\gamma}_i z^{k-i} \ , \ \bar{\gamma}_i := \sum_{j=-k}^{k} j\gamma_{ij} \ . \tag{5.2b}$$

We shall also employ the shift operator E defined by $Ey_n = y_{n+1}$.

Theorem 5.1 Let $g(t) \equiv$ constant.
(i) If $K(t,\tau,y) \equiv f(\tau,y)$ then the formula (2.5) reduces to

$$\alpha(E)y_n + \beta(E)Y_n(t_n) = h\gamma(E)f(t_n,y_n), \ n \geq 0. \tag{5.3a}$$

(ii) If $K(t,\tau,y) \equiv (t-\tau)f(\tau,y)$, $\beta(z) \equiv 0$, and if the weights $w_{n,j}$ in(2.3) are (ρ,σ)-reducible, then the formula (2.5) reduces to

$$\alpha(E)\rho(E)y_n + h^2[\sigma(E)\bar{\beta}(E) - \rho(E)\bar{\gamma}(E) - k\rho(E)\gamma(E) + \rho(E)\gamma'(E)E]f(t_n,y_n) = 0 \ , \tag{5.3b}$$

where γ' denotes the derivative of γ. ⊠

From part (i) of this theorem it follows that the LM formula (2.5), when applied to the integrated form of the first order equation $dy/dt=f(t,y)$,reduces to a linear multistep method $\{\alpha,\gamma\}$ for this equation, provided that $\beta(z)\equiv 0$. This statement holds, *irrespective of the weights* $w_{n,j}$ used in the definition of the lag term $Y_n(t)$. In other words, if the matrix B is chosen such that the *row sums vanish* ($\beta_i=0$) then our linear method is in fact an LM method for ODEs whenever the Volterra equation (1.1) is a (first order) ODE. Such linear methods will be called (α,γ)-*reducible*. The recurrence relation (5.3a) plays an important rôle in the stability analysis of Volterra equations with (5.1a) as test kernel. In particular, for (α,γ)-reducible methods, the ODE-stability theory directly applies and may suggest suitable polynomials α and γ for the construction of stable numerical methods for solving equation (1.1).

Example 5.1 The ILM and the MML methods are (α,γ)-reducible, whereas the DQ and the ML methods are not. ⊠

Part (ii) of Theorem 5.1 provides us with further information about how we should choose the weights $w_{n,j}$ and the matrices A, B and C in order to construct a suitable integration method. Observe, that here the structure of the matrices B and C is such, that the same set of polynomials $\{\rho,\sigma,\alpha,\gamma\}$ may lead to *different* recurrence relations.

Example 5.2 Let both $\{\alpha,\gamma\}$ and $\{\rho,\sigma\}$ be the trapezoidal rule, i.e., $\alpha(z)=\rho(z)=z-1$ and $\gamma(z)=\sigma(z)=\frac{1}{2}(z+1)$. Now it is a simple calculation to find that (5.3b) reduces to a LM method $\{\rho^*,\sigma^*\}$ for second order ODEs with $\rho^*(z)=(z-1)^2$ both for the ILM and the MML method, but with $\sigma^*(z)=\frac{1}{4}(z+1)^2$ for the ILM and $\sigma^*(z)=z$ for the MML method, respectively. (Note that both methods have order 2 (cf. [7, p.253]), with error constant -1/6 for the ILM and 1/12 for the MML method, respectively.) ☒

For an extension of Theorem 5.1 to the case of a general convolution kernel $K(t,\tau,y) = \Sigma_{\ell=0}^{m} (t-\tau)^{\ell} f_{\ell}(\tau,y)$, the reader is referred to [6].

6. V_0-STABILITY

Definition 6.1 A discretization method for (1.1) is said to be *V_0-stable* if $y_n \to 0$ as $n \to \infty$ whenever it is applied, with fixed stepsize $h > 0$, to the test equation

$$y(t) = y_0 + \int_0^t \{\lambda + \mu(t-\tau)\}y(\tau)d\tau, \qquad (6.1)$$

with arbitrary $(\lambda,\mu) \in Q_{\lambda,\mu} := \{(\lambda,\mu): \lambda < 0,\ \mu \le 0\}$. ☒

Wolkenfelt[10] has shown that the DQ method (2.7) can *not* be V_0-stable when the quadrature weights in (2.3) are (ρ,σ)-reducible. This negative result raised the question of whether V_0-stable methods for (1.1) do exist at all. Brunner, Nørsett and Wolkenfelt[4] answered this question affirmatively for a certain class of so-called one-stage implicit *Runge-Kutta* methods. In the class of *LM* methods analysed in the present paper, V_0-stable methods do also exist. In particular, they occur in the subclass of ILM methods. To see this, we observe that for the ILM methods we have

$$\bar{\beta}(z) = -\gamma(z), \quad \bar{\gamma}(z) = -k\gamma(z) + z\gamma'(z), \qquad (6.2)$$

and from Theorem 5.1 we derive the following result.

Theorem 6.1 Let the conditions of Theorem 5.1 part (ii) be satisfied, then the LM method, when applied to the test equation (6.1), assumes the form

$$\{\rho(E)[\alpha(E) - h\lambda\gamma(E)] +$$
$$+ h^2\mu[\sigma(E)\bar{\beta}(E) - \rho(E)(\bar{\gamma}(E)+k\gamma(E)-E\gamma'(E))]\}y_n = 0. \qquad (6.3)$$

In the ILM case this equation reduces to

$$\{\rho(E)[\alpha(E) - h\lambda\gamma(E)] - h^2\mu\sigma(E)\gamma(E)\}y_n = 0. \qquad (6.3')$$

☒

Since equation (6.3') is identical to the one obtained by Brunner and Lambert ([3]) in their stability analysis of numerical methods for the test integro-differential equation

$$\frac{dy}{dt}(t) = \lambda y(t) + \mu\int_0^t y(\tau)d\tau, \qquad (6.4$$

we may find examples of V_0-stable ILM methods just by inspectin the stability regions given by Brunner and Lambert. In this way we immediately conclude from [3] that the four combinations, with $\{\alpha,\gamma\}$ and $\{\rho,\sigma\}$ defining either the trapezoidal rule or th backward Euler rule, are V_0-stable methods. It turns out that the MML versions of these methods are *not* V_0-stable. (In fact, as communicated to us by S. Amini, no MML methods can be V_0-stable.) In Figure 2 the stability regions of both the IML and the MML methods are given. Evidently, the ILM methods have considerably larger regions of stability.

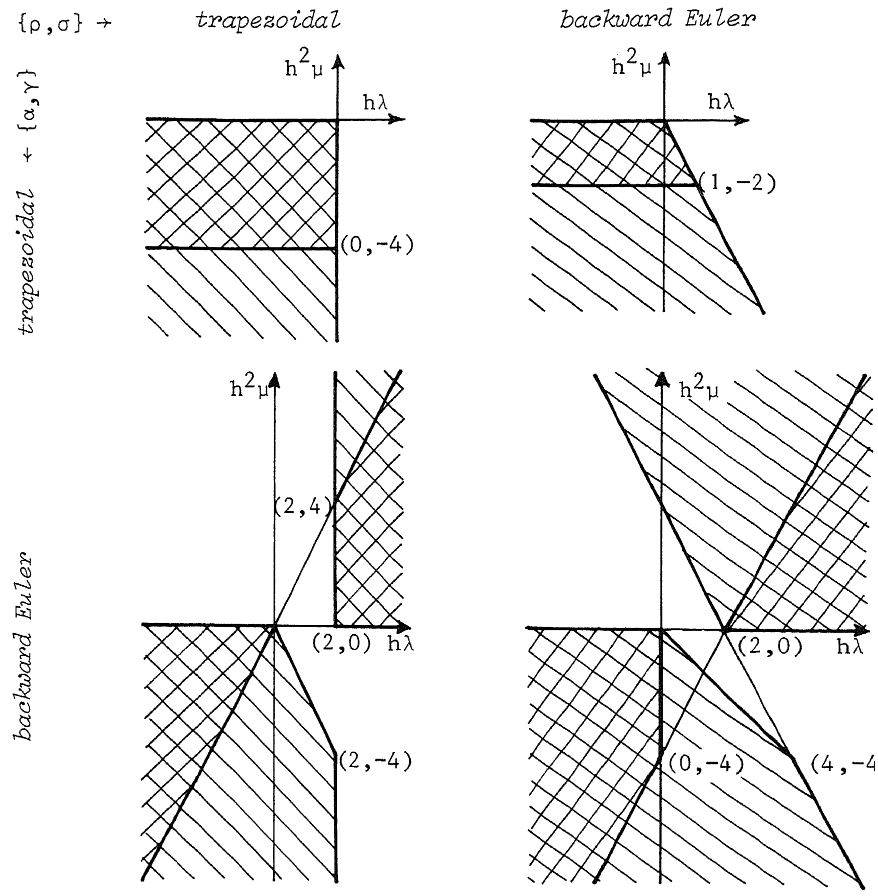

FIG. 2. Stability regions for MML(///) and ILM(\\\\\\) methods

7. NUMERICAL EXPERIMENTS

In this Section we illustrate by a few numerical experiments the convergence theorem 4.1 and the improved stability behaviour of the ILM and the MML methods. In the tables of results we list the accuracy obtained, by

$$A(h) := -\log_{10}(|\text{relative error at the end point}|), \qquad (7.1)$$

i.e., the number of correct digits in the numerical solution. The pair $\{\rho,\sigma\}$ used for computing the lag term will always define a Gregory formula of order r; the pair $\{\alpha,\gamma\}$ defines either an Adams-Moulton or a backward differentiation formula of order $\tilde{p}$. Methods are denoted by, e.g., $ILM(G_r\text{-}BD_{\tilde{p}})$.

7.1 Order of convergence

In the first experiment we integrated the equation

$$y(t) = 1 + \sin(t) - \cos(t) - \int_0^t y(\tau)d\tau, \; 0 \le t \le 2. \qquad (7.2)$$

The starting values were taken from the exact solution y(t) = sin(t). The generating characteristic polynomials $\{\rho,\sigma,\alpha,\gamma\}$ were chosen such that, according to Corollary 4.1, all methods listed in Table 2 are just of order $p^*= 5$. In this Table the values of A(h) and the corresponding *effective order* p^*_{eff} are presented, where $p^*_{eff} = [A(h) - A(2h)]/\log_{10}2$

TABLE 2

Tests of order of convergence

h^{-1}	$DQ(G_5)$	$ILM(G_5\text{-}AM_6)$	$ML(G_5\text{-}AM_4)$	$MML(G_5\text{-}AM_5)$
4	5.0	3.8	4.7	5.3
	)*5.7*	)*5.0*	)*5.5*	)*5.1*
8	6.7	5.3	6.3	6.8
	)*5.4*	)*5.2*	)*5.3*	)*5.1*
16	8.3	6.9	7.9	8.4
	)*5.2*	)*5.1*	)*5.2*	)*5.1*
32	9.9	8.5	9.5	9.9

From the results we see that the effective order tends to the asymptotic order as h decreases. We also see that the ILM method is less accurate than the other methods, which may be explained by its larger error constants (cf. Table 1).

7.2 Stability

In the second experiment we chose an example in which the kernel has a large Lipschitz constant (obtained by modifying an example given by Bownds[2]):

$$y(t) = 50(1-t^2)\ln(1+t) + 75t^2-51t+1 - 100\int_0^t \ln(1+t-\tau)y(\tau)d\tau, \qquad 0 \le t \le 4. \tag{7.3}$$

Again, the starting values were taken from the exact solution $y(t) = 1-t$. The results listed in Table 3 clearly show the better stability properties of the ILM method (a negative A(h)-value may be interpreted as an unstable behaviour). In particular, we observe the only marginally better performance of the MML methods when compared with the ML methods.

TABLE 3

Stability tests

h^{-1}	$DQ(G_5)$	ILM		ML		MML	
		G_5-BD_5	G_5-AM_6	G_5-BD_4	G_5-AM_4	G_5-BD_5	G_5-AM_5
4	-4.3	+2.5	+1.2	-2.8	-4.3	-2.6	-2.8
8	-6.5	+2.2	+2.2	-6.2	-5.6	-2.7	-4.9
16	+2.3	+2.6	+4.4	-6.9	+3.7	-2.4	+5.6

REFERENCES

1. ABRAMOWITZ, M. & I.A. Stegun (Eds.), *Handbook of mathematical functions*, Nat. Bureau of Standards, Washington D.C., (1964).
2. BOWNDS, J.M., Theory and performance of a subroutine for solving Volterra integral equations, to appear in *Computing*.
3. BRUNNER, H. & J.D. LAMBERT, Stability of numerical methods for Volterra integro-differential equations, *Computing* 12, 75-89 (1974).
4. BRUNNER, H., S.P. NØRSETT & P.H.M. WOLKENFELT, On V_0-stability of numerical methods for Volterra integral equations of the second kind, *Report NW 84/80*, Mathematical Centre, Amsterdam (1980).
5. HOUWEN, P.J. VAN DER & H.J.J. TE RIELE, Backward differentiation type formulas for Volterra integral equations of the second kind, *Numer. Math.* 37, 205-217 (1981).
6. HOUWEN, P.J. VAN DER & H.J.J. TE RIELE, Linear multistep methods for Volterra integral and integro-differential equations, *Report NW /82*, Mathematical Centre, Amsterdam (1982).
7. LAMBERT, J.D., *Computational methods in ordinary differential equations*, John Wiley & Sons, London etc. (1973).
8. MATTHYS, J., A-stable linear multistep methods for Volterra integro-differential equations, *Numer. Math.* 27, 85-94 (1976).

9. WOLKENFELT, P.H.M., Modified multilag methods for Volterra functional equations, to appear in *Math. Comp.*.
10. WOLKENFELT, P.H.M., Stability analysis of reducible quadrature methods for Volterra integral equations of the second kind, *ZAMM* 61, 399-401(1981).
11. WOLKENFELT, P.H.M., The construction of reducible quadrature rules for Volterra integral and integro-differential equations, to appear in *IMA J. Numer. Anal.*.
12. WOLKENFELT, P.H.M., P.J. VAN DER HOUWEN & C.T.H. BAKER, Analysis of numerical methods for second kind Volterra equations by imbedding techniques, *J. Integral Eq.* 3, 61-82 (1981).