IMPLEMENTATION OF CHEBYSHEVIAN LINEAR MULTISTEP FORMULAS

STIG SKELBOE

Abstract.
A Chebyshevian linear multistep formula is a formula fitted to a Chebyshev set of basis functions. This paper presents a unified approach for the implementation of Chebyshevian backward differentiation and Adams formulas for solving ordinary differential equations. The approach is based on generalized scaled differences, derived from generalized divided differences, and it includes the generalized Newton interpolation formula as predictor for the Chebyshevian implicit backward differentiation formula and Chebyshevian Adams-Bashforth-Moulton formulas. The local truncation errors are estimated by means of the scaled differences providing information for the control of order and steplength.

1. Introduction.
The classical linear multistep formulas, backward differentiation formulas, Adams formulas etc., are based on a set of polynomial basis functions. This fact is exploited in the implementation of these methods by the Nordsieck vector [1] and by divided differences [2, 3].

Let a system of ordinary differential equations be given by

\[ y' = f(x, y), \quad y(x_0) = y_0 \]

and let \( z_n = z(x_n) \) denote a numerical approximation of \( y(x_n) \). The implicit backward differentiation formula

\[ \sum_{r=0}^{k} a_r z_{n-r} = hf(x_n, z_n) \]

with the predictor formula

\[ z_n = \sum_{r=1}^{k+1} \hat{a}_r z_{n-r} \]

and the Adams-Bashforth and -Moulton formulas

\[ z_n = z_{n-1} + h \sum_{r=1}^{k+1} \beta_r f(x_{n-r}, z_{n-r}) \]

\[ z_n = z_{n-1} + h \sum_{r=0}^{k} \beta_r f(x_{n-r}, z_{n-r}) \]

Received June 20, 1979. Revised May 5, 1980.
have been used extensively for the numerical solution of stiff and non-stiff systems of ordinary differential equations (1.1).

There has been some interest in basing linear multistep methods on other basis systems than the polynomial (see [4, 5] with references). Let a linear multistep formula be defined by the operator

\[ L[y(x) ; h] = \sum_{r=0}^{k} \alpha_r y(x_{n-r}) + h\beta_r y'(x_{n-r}). \]

This linear multistep formula is fitted to the basis

\[ B_p = \{g_0, g_1, \ldots, g_p\}, \quad g_r \in C^1[a, b], \quad r = 0, 1, \ldots, p \]

if it fulfils the conditions

\[ L[g_r(x_n) ; h] = 0, \quad r = 0, 1, \ldots, p. \]

This implies that the formula will integrate a problem exactly if the solution of the problem is a linear combination of the basis functions \( g_p \).

Let a determinant of continuous real-valued basis functions be defined by

\[ V\left( \begin{array}{c} g_0, g_1, \\ x_{n-p}, x_{n-p+1}, \ldots, x_n \end{array} \right) \equiv \det [g_i(x_{n-p+k})], \quad i, k = 0(1)p. \]

The basis \( B_p \) is called a Chebyshev system on the real interval \([a, b]\) if

\[ V\left( \begin{array}{c} g_0, g_1, \\ x_{n-p}, x_{n-p+1}, \ldots, x_n \end{array} \right) > 0, \quad a \leq x_{n-p} < x_{n-p+1} < \ldots < x_n \leq b \]

and \( B_p \) is called a complete Chebyshev system if \( B_p, r = 0, 1, \ldots, p \) are Chebyshev systems.

**Definition.** A Chebyshevian backward differentiation or Adams formula is a formula fitted to a set of basis functions \( B_p \) where \( B_p \) fulfils (1.8) for the backward differentiation formula and where the derivatives of \( B_p \) fulfil a relation similar to (1.8) for the Adams formula.

The conditions in the definition of Chebyshevian backward differentiation and Adams formulas guarantee that the coefficients of the formulas (1.2–1.5) can be calculated from the fitting conditions (1.7).

Variable order variable steplength implementations of Chebyshevian linear multistep formulas can be constructed by computing the coefficients from (1.7) for the actual basis functions and steplength distribution [5]. This approach is not very elegant and if frequent error estimation for various orders is requested it is not too efficient either. In this paper a general method for implementing Chebyshevian backward differentiation and Adams formulas will be described. The approach is based on generalized scaled differences which is a new concept.
defined in this paper. During change of steepness the formulas of the new
implementation are equivalent to (1.2–1.5) with the coefficients computed for
fitting for the actual steepness distribution. In a practical computer
implementation the basis functions and their derivatives or indefinite integrals can
be supplied to the integration program as subroutines, and an exchange of these
subroutines will then give a new Chebychevian method. That is, the bulk of the
integration program is independent of the basis functions.

The preliminary report [7], dealing only with backward differentiation
formulas, gives alternative derivations and more programming details for practical
implementations.

2. Generalized divided and scaled differences.

For a Chebshev basis system $B_r$ (1.6), the generalized divided difference of
order $r$ of a function $g$ is defined by [6]

\begin{equation}
D_{r,n}g(x) \equiv \frac{V\left( g_0, \ldots, g_{r-1}, g \right)}{V\left( x_0, \ldots, x_{n-1}, x \right)} \cdot \frac{V\left( g_0, \ldots, g_{r-1}, g_r \right)}{V\left( x_0, \ldots, x_{n-1}, x \right)}
\end{equation}

for $x_{n-r} < x_{n-r+1} < \ldots < x_{n-1} < x$.

In the notation $D_{r,n}g(x)$, the nodes $x_{n-r}, \ldots, x_{n-1}$ and the basis functions
$g_0, \ldots, g_{r-1}, g_r$ are implied. This should not cause any problems in connection
with linear multistep formulas. The usual notation for the divided difference
declared in (2.1) is [6]:

\[
\begin{bmatrix}
g_0 & \ldots & g_{r-1} & g_r \\
x_0 & \ldots & x_{n-1} & x
\end{bmatrix}
\]

but it is considered too unwieldy for the following derivations.

In [6] the following recurrence relation is proved for $B_r$ being a complete
Chebyshev system:

\begin{equation}
D_0,n g(x) = g(x)/g_0(x)
\end{equation}

\begin{equation}
D_{r,n}g(x) = [D_{r-1,n}g(x) - D_{r-1,n-1}g]/[D_{r-1,n}g_0(x) - D_{r-1,n-1}g_r]
\end{equation}

where the shorter form $D_{r,n}g \equiv D_{r,n}g(x_n)$ has been used whenever possible.

**Definition.** The generalized scaled difference $H_{r,n}g(x)$ of order $r$ of the function
$g$ over the complete Chebyshev system $B_r$ is defined by

\begin{equation}
H_{0,n}g(x) = g(x) \quad \text{and for } r > 0:
\end{equation}

\begin{equation}
H_{r,n}g(x) = g_0(x) \prod_{s=1}^{r} [D_{s-1,n}g_0(x) - D_{s-1,n-1}g_0]D_{r,n}g(x).
\end{equation}

From (2.1) follows $D_{r,n}g(x) = 1$ and therefore
(2.4) \[ H_{r,n}g(x) = H_{r,n} g_r(x)D_{r,n}g(x) \]

From (2.2) and (2.3) it is clear that \( H_{s,n}g_s(x) + 0 \) for \( g_s \in B_s \), \( s = 0, 1, \ldots, r \) where \( B_r \) is a complete Chebyshev system over the interval \([a, b]\) and \( a \leq x_{n-r-1} < \ldots < x_{n-1} < x \leq b \).

The following simple but very important relation for generalized scaled differences is immediately derived from (2.2) and (2.3):

(2.5) \[ H_{r+1,n}g(x) = H_{r,n}g(x) - C_{r,n}(x)H_{r,n-1}g \]

where

(2.6) \[ C_{r,n}(x) = H_{r,n}g_r(x)/H_{r,n-1}g_r \]

and the short form \( H_{r,n}g = H_{r,n}g(x_n) \) has been used. The analogous short form for the \( C \)-coefficients is \( C_{r,n} = C_{r,n}(x_n) \).

By using (2.5) repeatedly, \( g(x) \) can be separated from the scaled difference,

(2.7) \[ H_{r,n}g(x) = g(x) - \sum_{s=0}^{r-1} C_{n,s}(x)H_{s,n-1}g \]

The formulas (2.4), (2.5) and (2.7) will form the basic tools of the following derivations.

From (2.7) and [6] follows the determinant form

\[ H_{r,n}g(x) = \begin{vmatrix} g_0, & \ldots, & g_{r-1}, g_r \end{vmatrix} \begin{vmatrix} g_0, & \ldots, & g_{r-1}, x \end{vmatrix} \begin{vmatrix} x_{n-r}, & \ldots, & x_{n-1} \end{vmatrix} \begin{vmatrix} x_{n-r}, & \ldots, & x_{n-1} \end{vmatrix} \]


The interpolation formula (1.2) is common as a predictor for the backward differentiation formulas. Newton's interpolation formula based on divided differences can be generalized as follows

**Theorem 3.1.** The generalized Newton interpolation formula

(3.1) \[ z_n = \sum_{r=0}^{k} C_{r,n}H_{r,n-1}z \]

over the complete Chebyshev system \( B_k \) and the nodes \( x_{n-k-1}, x_{n-k}, \ldots, x_n \) is equivalent to (1.3) fitted to \( B_k \) at the same nodes. The local truncation error of (3.1) is

\[ y(x_n) - \sum_{r=0}^{k} C_{r,n}H_{r,n-1}y = H_{k+1,n}y \]

**Proof.** The expression for the local truncation error follows immediately from (2.7). Since
the formula (3.1) fulfils the same fitting conditions (1.7) as (1.3) does. Since the coefficients of (1.3) are uniquely determined from this fitting condition, formulas (3.1) and (1.3) are equivalent. This completes the proof. ■

Comment.

The formula (3.1) is the classical Newton interpolation formula with divided differences when \( B_k \) is the classical polynomial basis. The generalized Newton interpolation formula is given in [6] by generalized divided differences.

**Theorem 3.2.** The backward differentiation formula

\[
 f(x_n \Delta_n) = \sum_{r=0}^{k} C_{r,n+1}(x_n) H_{r,n} \Delta_n^n
\]

where

\[
 C_{r,n+1}(x) = \frac{d C_{r,n+1}}{dx}(x)
\]

taken over the complete Chebyshev system \( B_k \) and the nodes \( x_{n-k}, \ldots, x_n \) is equivalent to (1.2) fitted to \( B_k \) at the same nodes. The local truncation error of the derivative is

\[
 y'(x_n) - \sum_{r=0}^{k} C_{r,n+1}(x_n) H_{r,n} y = H_{k+1,n+1} y(x_n)
\]

where

\[
 H_{k+1,n+1} y(x) = \frac{d H_{k+1,n+1}}{d x} y(x)
\]

**Proof.** The local truncation error (3.4) follows immediately by differentiation of (2.7). Since

\[
 H_{k+1,n+1} g_r(x_n) = 0, \quad g_r \in B_k, \quad r = 0, 1, \ldots, k
\]

the equivalence to (1.2) follows by similar arguments as in the proof of Theorem 3.1, thus completing the proof. ■

The approach used for deriving the formula (3.2) is the classical one for deriving backward differentiation formulas [9].

Formula (3.2) can not be applied directly since all the differences \( H_{r,n} \Delta_n \) include \( x_n \). By applying (2.7) to \( H_{r,n} \Delta_n \), substituting into (3.2) and collecting \( H_{r,n-1} \Delta_n \) terms, the final form of the corrector formula is obtained:

\[
 f(x_n \Delta_n) = A_{0,n+1} \Delta_n^n - \sum_{r=0}^{k-1} A_{r+1,n} C_{r,n} H_{r,n-1} \Delta_n^n
\]

where
\begin{equation}
A_{r,n}^{k} = \sum_{s=r}^{k} C_{s,n+1}(x_{n}).
\end{equation}

The principal local truncation error is obtained by applying (2.5) once to (3.4) and discarding the difference of order \(k+2\)

\[H_{k+1,n+1}(x_{n}) \approx C_{k+1,n+1}(x_{n})H_{k+1,n}y.\]

For \(A_{0,n}^{k} + 0\) the estimate of the principal local truncation error is therefore

\begin{equation}
T_{k,n}^{E} = \left(C_{k+1,n+1}(x_{n})/A_{0,n}^{k}\right) H_{k+1,n}y.
\end{equation}

The difference \(H_{k+1,n}y\) can be expressed by (2.7) as

\[H_{k+1,n}y = z_{n} - \sum_{r=0}^{k} C_{r,n}H_{r,n}z_{n}.
\]

For \(z\) being the numerical approximation calculated from (3.5), the summation term is recognized as the result of the predictor (3.1). Thus, (3.7) can be written in the form:

\[T_{k,n}^{E} = \left(C_{k+1,n+1}(x_{n})/A_{0,n}^{k}\right)(z_{n} - z^{p}_{n})\]

where \(z^{p}_{n}\) and \(z_{n}\) denote predicted and corrected solutions respectively.

4. Adams formulas.

Like the derivation of the classical Adams formulas [9], the present approach is based on interpolation of the derivative. The Adams formulas are therefore fitted to a set of basis functions \(\mathcal{B}_{k}\) being indefinite integrals of the basis functions in \(B_{k}\) used for the interpolation of \(f(x_{n-k}, z_{n-k}), \ldots, f(x_{n}, z_{n})\). The basis \(\mathcal{B}_{k+1}\) derived from \(B_{k}\) (1.6) is

\begin{equation}
\mathcal{B}_{k+1} = \left\{1, \int g_{0}(x)dx, \int g_{1}(x)dx, \ldots, \int g_{k}(x)dx\right\}
\end{equation}

where the constant 1 accounts for the arbitrary constants of the indefinite integrals.

**Theorem 4.1.** The generalized explicit Adams formula

\begin{equation}
z_{n} = z_{n-1} + \sum_{r=0}^{k} \mathcal{G}_{r,n}H_{r,n-1}f
\end{equation}

where

\begin{equation}
\mathcal{G}_{r,n} = \int_{x_{n-1}}^{x_{n}} C_{r,n}(x)dx
\end{equation}

and
\[ H_{r,n-1}f = H_{r,n-1}f(x_{n-1}, z_{n-1}) \]

is equivalent to (1.4) when both formulas are taken over the nodes \( x_{n-k}, \ldots, x_n \) and the basis \( \mathcal{B}_{k+1} \) (4.1). The scaled differences are based on the complete Chebyshev system \( B_k \) (1.6). The local truncation error is

\[
y(x_n) - y(x_{n-1}) - \sum_{s=0}^{k} C_{s,n} H_{s+1,n} y' = \int_{x_{n-1}}^{x_n} H_{k+1,n} y'(x) dx.
\]

**Proof.** The local truncation error (4.4) follows immediately from (2.7). Since \( g(x) \in \mathcal{B}_k \Rightarrow g'(x) \in B_k \),
\[
\int_{x_{n-1}}^{x_n} H_{k+1,n} g_r(x) dx = 0, \quad g_r \in B_k, \quad r = 0, 1, \ldots, k.
\]

Both (4.2) and (1.4) are therefore fitted to \( \mathcal{B}_k \) and the equivalence follows by similar arguments as in the proof of Theorem 3.1 thus completing the proof. \( \blacksquare \)

**Theorem 4.2.** The generalized implicit Adams formula

\[
z_n = z_{n-1} + \sum_{r=0}^{k} C_{r,n+1} H_{r,n} f
\]

where
\[
C_{r,n+1}(x_n) = \int_{x_{n-1}}^{x_n} C_{r,n+1}(x) dx
\]

is equivalent to (1.5) when both formulas are taken over the nodes \( x_{n-k}, \ldots, x_n \) and the basis \( \mathcal{B}_{k+1} \) (4.1). The scaled differences are based on the complete Chebyshev system \( B_k \) (1.6). The local truncation error is

\[
y(x_n) - y(x_{n-1}) - \sum_{s=0}^{k} C_{s,n+1}(x_n) H_{s,n} y' = \int_{x_{n-1}}^{x_n} H_{k+1,n+1} y'(x) dx.
\]

**Proof.** The proof is similar to the proof of the previous theorem. \( \blacksquare \)

The corrector formula (4.5) is not useful in its present form since \( z_n \) appears in all of the scaled differences \( H_{r,n} f(t_n, z_n) \). By using formula (2.7), \( f(t_n, z_n) \) can be extracted from the differences and the final form of the implicit Adams formula is obtained

\[
z_n = z_{n-1} + E_{0,n}^k f(x_n, z_n) - \sum_{r=0}^{k-1} E_{r+1,n}^k C_{r,n} H_{r,n-1} f
\]

where

\[
E_{r,n}^k = \sum_{s=r}^{k} C_{s,n+1} (x_n).
\]
The principal local truncation error $T_{n,k}^C$ of (4.7) can be obtained from (4.6) by expanding $H_{k+1,n+1}y'(x)$ after (2.5) and discarding the term of order $k+2$:

$$T_{n,k}^C = \mathcal{C}_{k+1,n+1}(x_n) H_{k+1,n}f.$$ 

The principal local truncation error of the predictor (4.2) is obtained by applying (2.5) twice to (4.4) and discarding the terms of order $k+2$:

$$T_{n,k}^P = (\mathcal{C}_{k+1,n}/C_{k+1,n}) H_{k+1,n}f.$$ 

The error of the corrector formula of order $k$ can be estimated by the predictor formula of the same order in a generalization of Milne's device:

$$T_{n,k}^C = \frac{\mathcal{C}_{k+1,n+1}(x_n)}{\mathcal{C}_{k+1,n}/C_{k+1,n} - \mathcal{C}_{k+1,n+1}(x_n)} (z^P_n - z_n)$$

where $z^P_n$ and $z_n$ denote predicted and corrected solutions respectively.

5. Translation invariance.

Translation invariance of a basis is a property which has not received much attention in the literature [4,5], perhaps because the classical polynomial basis is translation invariant. However, it is an important property for the implementation of linear multistep methods since it implies that the coefficients only depend on the steplength distribution and not on the nodes $x_{n-k}, x_{n-k+1}, \ldots, x_{n}$.

**Definition.** A basis $B_s(x) = \{g_0(x), g_1(x), \ldots, g_s(x)\}$ is called translation invariant if $B_s(x)$ and $B_s(x + h)$ span the same space for all real $h$. The basis $B_s$ is called completely translation invariant if $B_s$, $s=0,1,\ldots,r$, are all translation invariant. Generalized scaled differences and coefficients depending on the basis $B_s(x)$ are called translation invariant if they remain constant during translation of the basis to $B_s(x + h)$, $h \neq 0$.

**Theorem 5.1.** Generalized scaled differences (2.3) and C-coefficients (2.6) based on a completely translation invariant complete Chebyshev system $B_s$ are translation invariant.

**Proof.** Let $\tilde{g}_s(x) \in B_s(x + h)$, $r=0,1,\ldots,k$, such that $\tilde{g}_s(x) = g_s(x + h)$. Since $B_s$ is completely translation invariant, there exist non-singular matrices $M_s(h)$ independent of $x$ such that

$$g_s(x) = M_s(h)g_s(x), \quad s=0,1,\ldots$$

where $g_s(x)^T = \{g_0(x), \ldots, g_s(x)\}$ and $\tilde{g}_s(x)^T = \{\tilde{g}_0(x), \ldots, \tilde{g}_s(x)\}$. 

The proof is carried out by induction:

a) \( \tilde{H}_{0,n}g = H_{0,n}g = g \) by (2.3)
\[
\tilde{C}_{0,n} = \tilde{g}_0(x_n)/\tilde{g}_0(x_{n-1}) = C_{0,n} \text{ by (2.4) and (5.1)}
\]

therefore \( H_{0,n}g \) and \( C_{0,n} \) are translation invariant.

b) Assume that \( H_{s,n}g \) and \( C_{s,n} \), \( s = 0, 1, \ldots, r \) are translation invariant. Prove that \( H_{r+1,n}g \) and \( C_{r+1,n} \) are translation invariant.

\( H_{r+1,n}g \) is translation invariant by (2.5) and the previous assumption so that
\[
H_{r+1,n}g = \tilde{H}_{r+1,n}g.
\]

From (5.1) follows \( \tilde{g}_{r+1} = (M_{r+1}(h))_{r+2}g_{r+1} \), where \(( \cdot )_{r+2}\) denotes row \( r+2 \) of the matrix. This relation and (2.1) lead to
\[
D_{r+1,n} \tilde{g}_{r+1} = D_{r+1,n} \tilde{g}_{r+1} = (M_{r+1}(h))_{r+2,r+2}
\]

so that (2.3) and (5.2) give \( C_{r+1,n} = \tilde{C}_{r+1,n} \).

By the induction axiom, this completes the proof of the theorem. \( \blacksquare \)

Note that in general neither \( H_{s,n}g \), nor \( D_{s,n}g \) are translation invariant while \( C_{r,n} = H_{r,n}g, D_{r,n}g = H_{r,n}g, D_{r,n}g \) both are. The translation invariance of the latter is one of the reasons for using scaled differences instead of divided differences for the linear multistep formulas.

**Theorem 4.2.** The \( A \)-coefficients (3.6) based on a completely translation invariant, complete Chebyshev system are translation invariant.

**Proof.** For a \( k \)-step formula (1.2) the coefficients \( a_0, a_1, \ldots, a_k \) are translation invariant. Therefore \( A_{k,n}^0 = a_0 \) is translation invariant. In formula (3.7) all the scaled differences and \( C \)-coefficients are translation invariant according to Theorem 4.1. The value \( z_{n-k} \) is only involved in \( H_{k-1,n-1}z \) and \( A_{k-1,n}^0 \) is therefore translation invariant since \( z_k \) is. Since \( y_{n-k+1} \) is only involved in \( H_{k-1,n-1}z \) and \( H_{k-2,n-1}z \), where \( A_{k-1,n}^0 \) and \( z_{k-1} \) are translation invariant, so is \( A_{k-2,n}^0 \). The argument can be continued to complete the proof. \( \blacksquare \)

**Theorem 4.3.** The \( E \)-coefficients (4.8) based on a completely translation invariant, complete Chebyshev system are translation invariant.

**Proof.** The proof is similar to the proof of Theorem 4.2.

A computer program implementing Chebyshevian linear multistep methods can be based on two fundamental data structures and some primitive operations to operate on these data structures.

For the backward differentiation formulas the solution points $z_{n-k+1}, z_{n-k}, \ldots, z_n$ are stored as scaled differences

$$H_{k+1,n} z_{k+1}, \ldots, H_{1,n} z_n$$

and for the Adams formulas the derivatives $f(x_{n-k+1}, z_{n-k+1}), \ldots, f(x_n, z_n)$ are stored analogously as

$$H_{k+1,n} f_{k+1}, H_{k,n} f_k, \ldots, H_{1,n} f_{x_n, z_n}.$$

The two data structures support variable order variable steplength implementations of maximally $k$-step formulas since the difference of order $k+1$ is used to estimate the error of a $k$-step formula while integrating with a $(k-1)$-step formula. When $z_{n+1}$ or $f(x_{n+1}, z_{n+1})$ have been computed the data structures are updated with (2.5).

The vital coefficients $C_{r,n}(x)$ are derived from the data structure (6.1). This structure also supports maximally $k$-step formulas.

$$
\begin{aligned}
g_0(x_{n-k-1}) & \quad g_0(x_{n-k}) & \quad \ldots & \quad g_0(x_{n-2}) & \quad g_0(x_{n-1}) & \quad g_0(x_n) \\
H_{1,n-k} & \quad H_{1,n-k+1} & \quad \ldots & \quad H_{1,n-1} & \quad H_{1,n} & \\
H_{2,n-k+1} & \quad H_{2,n-k+2} & \quad \ldots & \quad H_{2,n} & \\
\vdots & \quad \vdots & \quad \ldots & \quad \vdots & \\
H_{k,n-1} & \quad H_{k,n} & \quad \ldots & \quad H_{k,n-k} & \quad H_{k,n} & \quad g_k(x_n)
\end{aligned}
$$

(6.1)

$C_{r,n}$ can be calculated from (2.6) and the data structure (6.1). From (2.7) follows

$$H_{r,n} g_r(x) = g_r(x) - \sum_{s=0}^{r-1} C_{s,n}(x) H_{s,n-1} g_s$$

which gives (3.3):

$$H_{r,n} g_r(x) = g_r(x) - \sum_{s=0}^{r-1} C_{s,n}(x) H_{s,n-1} g_s$$

and $C_{r,n}(x) = H_{r,n} g_r(x) / H_{s,n-1} g_s$

$$\mathcal{H}_{r,n} g_r(x) = \int_{x_{n-1}}^{x_n} g_r(t) dt - \sum_{s=0}^{r-1} C_{s,n}(x) H_{s,n-1} g_s$$

and

$$C_{s,n}(x) = \mathcal{H}_{s,n} g_s(x) / H_{s,n-1} g_s.$$

The coefficient $C_{r,n+1}(x_n)$ for the implicit Adams formula is calculated in a similar way.
The data structure (6.1) is updated with (2.5) for both new steps $x_{n+1}, x_{n+2}, \ldots$ and extra basis functions $g_{k+2}, g_{k+3}, \ldots$.

While the classical polynomial basis $\{1, x, \ldots, x^r\}$ is a complete Chebyshev system, an important basis like the trigonometric $\{1, \sin \omega x, \cos \omega x, \ldots, \sin r\omega x, \cos r\omega x\}$ is only quasicomplete [8] since it is only Chebysheuvian when the sine and cosine terms appear in pairs.

Scaled differences over a quasicomplete Chebyshev basis may be undefined since $H_{m,n} \delta_t$ is no longer guaranteed to be nonzero. In an implementation with automatic steplength control this is of minor importance since the steplength can be adjusted if rounding errors threaten to destroy the accuracy of the table (6.1). As another precaution, (6.1) may be rebuilt from scratch at regular intervals.

REFERENCES


DENMARK RESEARCH CENTRE
FOR APPLIED ELECTRONICS
HORSHOLM
DENMARK